तत् त्वं पूषन् अपावृणु
केन्द्रीय विद्यालय संगठन

केन्द्रीय विद्यालय संगठन

KENDRIYA VIDYALAYA SANGATHAN

एर्नाकुलम क्षेत्र ERNAKULAM REGION

COMPUTER SCIENCE (083)

CLASS - XII

PART B

COMPREHENSIVE

STUDENT SUPPORT MATERIAL

Based on the latest CBSE Exam Pattern
for the Session 2023-24

CHIEF PATRON

Mr.Santhosh Kumar N

Deputy Commissioner, KVS RO Ernakulam

PATRONS

श्री गनेश स. ईन्द्राले ,Mrs Deepti Nair, पाटिल रमेश अभिमन्यू

Assistant Commissioners, KVS RO Ernakulam

PRINCIPAL IN-CHARGE

Mr Alex Jose

Principal I/C, K V Konni

MESSAGE BY THE DEPUTY COMMISSIONER

I am delighted to announce the release of Support Material/Study material for the students of class X and XII. In our relentless pursuit of academic excellence, we have been constantly revising and upgrading our teaching methodologies and resources. And one of the important resources is the support material which is prepared by Kendriya Vidyalayas for the students of class X and XII. This Support material is in two parts: Part A is the core concepts/topics of all subjects which aims minimum level of learning for each student and Part B is a detailed topics/lessons with practice questions which aims to foster a deeper understanding of subjects, stimulate critical thinking and helps in achieving better score in CBSE exam. Whereas Part A is planned to be printed and distributed amongst all students whereas Part B which is the larger one is planned to disseminate through electronic media/blogs.

I hope this support material in two parts will greatly benefit the academic journey of class X and XII not only in pursuit of good result in CBSE exams but also helpful for various entrance examinations.

Let's march ahead with dedicated minds and relentless endeavours for better future through better education.

With warm regards,

Shri. Santosh Kumar N

Deputy Commissioner

KVS Ernakulam Region

CONTENT AND REVIEW

BY PGT COMPUTER SCIENCE OF ERNAKULAM REGION

Teacher Coordinator

- Mr.SREEJITH K V,KV ARMY CANTT.PANGODE

Content preparation & compilation

- Mrs. LIJINA T,K V AFS AKKULAM(Revision of Python)
- Mrs. ANJANA K K,K V NO.2CALICUT (Revision of Python)
- Mrs. DEEPA P R,K V SAP PEROORKADA(Functions)
- Mrs. SINDHU S,K V KOLLAM(Exception Handling)
- Mrs. RAGENDHU S P,KV CHENNEERKARA(Introduction to files &Text File)
- Mr. VINOD B,K V NTPC KAYAMKULAM(Binary file)
- Mr. SHABIN MUHAMMED,K V PAINAVU ,IDUKKI(CSV file)
- Mrs. K VIMALA THOMAS,K V INS DRONACHARYA(Data Structure)
- Mrs. HEMA C N,K V No.1 KOCHI(Evolution of networking)
- Mr. ABHILASH V JOSE,K V CRPF PERINGOME(Data communication)
- Mr. ARUN PRASANTH P V,K V KANJIKODE(Network devices)
- Mr. P C SAJESH,K V MALAPPURAM(Network topologies, Protocol)
- Mrs. SOUMYAMOL D,K V KANHANGAD(Database concepts)
- Mr. PRAJUN P P,KV NO.1 KASARAGOD(SQL & JOINS)
- Mr. SHAJI M,KV EZHIMALA(Interface of python with an SQL database)
- Mr. JIMMY JOHN K V OTTAPALAM (PART A – Compilation)
- Mrs.REENA P V,K V PAYYANUR (PART B – Compilation)
- Mr. AJITH N K,K V KALPETTA (Sample Paper)
- Mrs.ANU MATHEW,KV RB KOTTAYAM (Sample Paper)

Final content review

- Mrs. AMBILY KRISHNAN, K V CRPF PALLIPURAM
- Mrs. VIDYA R PRABHU, K V No.1 PALAKKAD
- Mrs. ANEESA N ALI, K V NO.I CALICUT
- Mrs. RESHMA SURENDRAN, K V No.2 KOCHI
- Mrs. MANJU N, K V THRISSUR

INDEX

Computer Science

CLASS-XII

Code No. 083

2023-24

1. Prerequisites

Computer Science- Class XI

2. Learning Outcomes

Student should be able to

a) apply the concept of function.

b) explain and use the concept of file handling.

c) use basic data structure: Stacks

d) explain basics of computer networks.

e) use Database concepts, SQL along with connectivity between Python and SQL.

3. Distribution of Marks:

Unit No.	Unit Name	Marks	Periods	
			Theory	**Practical**
I	Computational Thinking and Programming – 2	40	70	50
II	Computer Networks	10	15	…
III	Database Management	20	25	20
	Total	70	110	70

4. Unit wise Syllabus

Unit I: Computational Thinking and Programming – 2

- Revision of Python topics covered in Class XI.
- Functions: types of function (built-in functions, functions defined in module, user defined functions), creating user defined function, arguments and parameters, default

parameters, positional parameters, function returning value(s), flow of execution, scope of a variable (global scope, local scope)

- Exception Handling: Introduction, handling exceptions using try-except-finally blocks
- Introduction to files, types of files (Text file, Binary file, CSV file), relative and absolute paths
- Text file: opening a text file, text file open modes (r, r+, w, w+, a, a+), closing a text file, opening a file using with clause, writing/appending data to a text file using write() and writelines(), reading from a text file using read(), readline() and readlines(), seek and tell methods, manipulation of data in a text file
- Binary file: basic operations on a binary file: open using file open modes (rb, rb+, wb, wb+, ab, ab+), close a binary file, import pickle module, dump() and load() method, read, write/create, search, append and update operations in a binary file
- CSV file: import csv module, open / close csv file, write into a csv file using writer(),writerow(),writerows() and read from a csv file using reader()
- Data Structure: Stack, operations on stack (push & pop), implementation of stack using list.

Unit II: Computer Networks

- Evolution of networking: introduction to computer networks, evolution of networking (ARPANET, NSFNET, INTERNET)
- Data communication terminologies: concept of communication, components of data communication (sender,receiver, message, communication media, protocols), measuring capacity of communication media (bandwidth, data transfer rate), IP address, switching techniques (Circuit switching, Packet switching)
- Transmission media: Wired communication media (Twisted pair cable, Co-axial cable, Fiber-optic cable), Wireless media (Radio waves, Micro waves, Infrared waves)
- Network devices (Modem, Ethernet card, RJ45, Repeater, Hub, Switch, Router, Gateway, WIFI card)
- Network topologies and Network types: types of networks (PAN, LAN, MAN, WAN), networking topologies (Bus, Star, Tree)
- Network protocol: HTTP, FTP, PPP, SMTP, TCP/IP, POP3, HTTPS, TELNET, VoIP
- Introduction to web services: WWW, Hyper Text Markup Language (HTML), Extensible Markup Language (XML), domain names, URL, website, web browser, web servers, web hosting

Unit III: Database Management

- Database concepts: introduction to database concepts and its need
- Relational data model: relation, attribute, tuple, domain, degree, cardinality, keys (candidate key, primary key, alternate key, foreign key)
- Structured Query Language: introduction, Data Definition Language and Data Manipulation Language, data type (char(n), varchar(n), int, float, date), constraints (not null, unique, primary key), create database, use database, show databases, drop database, show tables, create table, describe table, alter table (add and remove an attribute, add and remove primary key), drop table, insert, delete, select, operators (mathematical, relational and logical), aliasing, distinct clause, where clause, in, between, order by, meaning of null, is null, is not null, like, update command, delete command, aggregate functions (max, min, avg, sum, count), group by, having clause,

joins: cartesian product on two tables, equi-join and natural join

- Interface of python with an SQL database: connecting SQL with Python, performing insert, update, delete queries using cursor, display data by using connect(), cursor(), execute(), commit(), fetchone(), fetchall(), rowcount, creating database connectivity applications, use of %s format specifier or format() to perform queries

5. Practical

S.No	Unit Name	Marks (Total=30)
1	Lab Test: 1. Python program (60% logic + 20% documentation + 20% code quality)	8
	2. SQL queries (4 queries based on one or two tables)	4
2	Report file: ● Minimum 15 Python programs. ● SQL Queries – Minimum 5 sets using one table / two tables. ● Minimum 4 programs based on Python - SQL connectivity	7
3	Project (using concepts learnt in Classes 11 and 12)	8
4	Viva voce	3

6. Suggested Practical List:

Python Programming

- Read a text file line by line and display each word separated by a #.
- Read a text file and display the number of vowels/consonants/uppercase/lowercase characters in the file.
- Remove all the lines that contain the character 'a' in a file and write it to another file.
- Create a binary file with name and roll number. Search for a given roll number and display the name, if not found display appropriate message.
- Create a binary file with roll number, name and marks. Input a roll number and update the marks.
- Write a random number generator that generates random numbers between 1 and 6 (simulates a dice).

- Write a Python program to implement a stack using list.
- Create a CSV file by entering user-id and password, read and search the password for given userid.

Database Management

- Create a student table and insert data. Implement the following SQL commands on the student table:
    - o ALTER table to add new attributes / modify data type / drop attribute
    - o UPDATE table to modify data
    - o ORDER By to display data in ascending / descending order
    - o DELETE to remove tuple(s)
    - o GROUP BY and find the min, max, sum, count and average
- Similar exercise may be framed for other cases.
- Integrate SQL with Python by importing suitable module.

7. Suggested Reading Material

- NCERT Textbook for COMPUTER SCIENCE (Class XII)
- Support Materials on the CBSE website.

8. Project

The aim of the class project is to create something that is tangible and useful using Python file handling/ Python-SQL connectivity. This should be done in groups of two to three students and should be started by students at least 6 months before the submission deadline. The aim here is to find a real world problem that is worthwhile to solve.

Students are encouraged to visit local businesses and ask them about the problems that they are facing. For example, if a business is finding it hard to create invoices for filing GST claims, then students can do a project that takes the raw data (list of transactions), groups the transactions by category, accounts for the GST tax rates, and creates invoices in the appropriate format. Students can be extremely creative here. They can use a wide variety of Python libraries to create user friendly applications such as games, software for their school, software for their disabled fellow students, and mobile applications, of course to do some of these projects, some additional learning is required; this should be encouraged. Students should know how to teach themselves.

The students should be sensitized to avoid plagiarism and violations of copyright issues while working on projects. Teachers should take necessary measures for this.

REVIEW OF PYTHON

Characteristics of Python:

1. Python is Object-Oriented

   Python supports concepts such as polymorphism, operator overloading and multiple inheritance.
2. Indentation

   Indentation is one of the greatest features in python – indentation identifies the blocks of code.
3. Open source Programming Language

   Python is an open source programming language.

   It can be freely downloaded. Installing python is free and easy
4. It's Powerful
   - Dynamic typing
   - Built-in types and tools
   - Library utilities
   - Third party utilities (e.g. NumPy, SciPy)
   - Automatic memory management
5. It's Portable and Platform independent
   - Python runs virtually in every major platform used today
   - As long as you have a compatible python interpreter installed, python programs will run in exactly the same manner, irrespective of platform.
6. It's easy to use and learn
   - No intermediate compiler
   - Python Programs are compiled automatically to an intermediate form called byte code, which the interpreter then reads.
   - This gives python the development speed of an interpreter without the performance loss inherent in purely interpreted languages.
   - Structure and syntax are pretty intuitive and easy to grasp.
7. Interpreted Language

   Python is processed at runtime by python Interpreter
8. Interactive Programming Language

   Users can interact with the python interpreter directly for writing the programs
9. Straight forward syntax

   The formation of python syntax is simple and straight forward which also makes it popular.

Character Set of Python:

Set of all characters recognised by python. It can be any alphabet, digit or any symbol.
Python uses ASCII character set.
Python Supports the following character set:
Letters: A- Z and a -z
Digits: 0-9
Special Symbols: + , - , /, % , ** , * , [ ], { }, #, $ etc.
White spaces: Blank space, tabs, carriage return, newline, form feed
Other Characters: All ASCII and UNICODE characters.

Tokens:

Tokens are smallest identifiable units.
Different tokens are:

- Keywords: reserved words which are having special meaning.
- Examples: int, print, input, for, while
- Identifiers: name of given by the user to identify variables, functions, constants, etc.
- Literals: literals are constant values. It can be numeric or non-numeric
- Operators: triggers an operation. Parameters provided to the operators are called
    - operands
    - Examples: + - * % ** / //
- Delimiters: Symbols used as separators.
    - Examples: {} , [ ] ; :

Rules for identifier names:

- Keywords and operators are not to be used
- Start with an alphabet or an underscore.
- Special characters other than underscores are allowed.
- Space is not allowed.
- Cannot start with a number.
- Note: Python is case sensitive and hence uppercase and lowercase are treated differently.
- Example:
    - engmark, _abc , mark1, mark2

Variables and data types:

Variables are used to store data. In Python a variable is an object. Variables are nothing but reserved memory locations to store values. This means that when you create a variable you reserve some space in memory. Memory locations are addressable.
The data stored in memory can be of many types. For example, a student roll number is stored as a numeric value and his or her address is stored as alphanumeric characters. Python has various standard data types that are used to define the operations possible on them and the storage method for each of them. Some are mutable and some are immutable.

- Numeric Types: int, float, complex
- Boolean – True or False values. Used when comparisons are made and the result can be expressed as True or False
- None – a special type with an unidentified value or absence of value.
- Sequence: an ordered collection of elements. String, List and Tuples are sequences. Items/elements are Identified by its index.
- Sets – an unordered collection of any type without duplicate entry.
- Mappings – Dictionaries are mappings. Elements of dictionaries are key-value pairs. Keys are used to access values. Keys are immutable.

Based on the data type of a variable, the interpreter allocates memory and decides what can be stored in the reserved memory. Therefore, by assigning different data types to variables, you can store integers, decimals or characters in these variables.

Mutable and Immutable types:

If the values can be changed after creating the variable and assigning values, then it is called **mutable**. If the value assigned cannot be changed after creating and initializing it, then it is known as **Immutable**.

When an attempt is made to update the immutable type, a new memory location is used with name remaining the same.

Assigning Values to Variables:

Python variables do not need explicit declaration to reserve memory space. The declaration happens automatically when you assign a value to a variable. The assignment operator (=) is used to assign values to variables.

The operand to the left of the = operator is the name of the variable and the operand to the right of the = operator is the value stored in the variable.

For example –

```
a= 100  #an integer is  signed
b = 1000.0 # A floating point value
c = "John"      # A string
d = ‘’’ Hello   world
          good morning’’’    # multi line string
```

Multiple Assignments:

Python allows you to assign a single value to several variables simultaneously.

For example: a = b = c = 1

Here, an integer object is created with the value 1, and all three variables are assigned to the same memory location. You can also assign multiple objects to multiple variables.

For example −

```
a,b,c = 1,2,"kvsch“
```

Here, two integer objects with values 1 and 2 are assigned to variables a and b respectively, and one string object with the value "john" is assigned to the variable c.

Also, it is possible to define and initialize multiple variables with different values as given below

```
a, b, c = 5,10,20
```

The above instruction will create three variables named a,b,c and initialize them with values 5, 10 20 respectively.

Output Variables:

The Python print statement is often used to output variables.Variables do not need to be declared with any particular type. Different type of data can be assigned even after initializing them with some type of data.

```
x = 5   # x is of type int
x = "kve "      # x is now of type str print(x)
```

Output: kve

To combine both text and a variable, Python uses the “+” character:

Example

```
  x = "awesome"
 print("Python is " + x)
```

Output of the above code is:

```
Python is awesome
```

You can also use the + character to add a variable to another variable:

Example

```
x = "Python is "
y = "awesome"
z = x + y print(z)
```

Output:

```
Python is awesome
```

Expressions:

An expression is a combination of values, variables, and operators.

An expression is evaluated using assignment operator.

Examples: Y=x + 17

```
>>> x=10
>>> z=x+20
>>> z
30
```

```
>>> x=10
>>> y=20
>>> c=x+y
>>> c
30
```

A value all by itself is a simple expression, and so is a variable.

```
>>> y=20
>>> y
20
```

Python also defines expressions with identifiers, literals, and operators. So,

Operators:

In Python you can implement the following operations using the corresponding operators. Operators trigger an operation. Some operators need one operand, some need more than one operand to perform the operation.

Python supports different types of operators including Arithmetic Operators, Relational Operators, Logical Operators etc.

Operator	**Token**
addition	+
subtraction	-
multiplication	*
Integer Division	/
remainder	%
Binary left shift	<<
Binary right shift	>>
and	&
or	\
Less than	<
Greater than	>
Less than or equal to	<=
Greater than or equal to	>=

Check equality	==
Check not equal	!=

Statements:

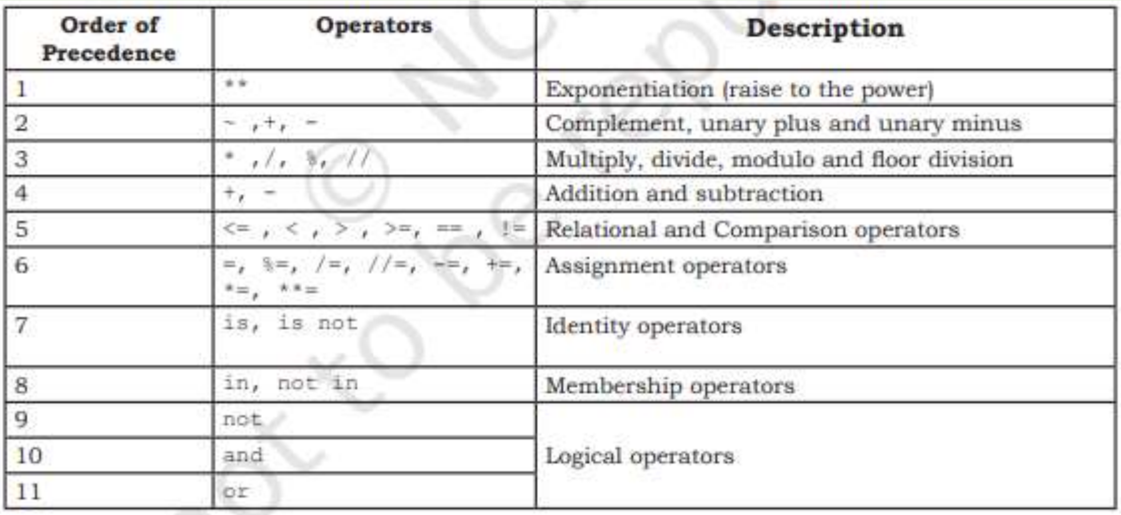
A statement is an instruction that the Python interpreter can execute. We have normally two basic statements, the assignment statement and the print statement. Some other kinds of statements that are if statements, while statements, and for statements generally called as control flows.

Examples:
An assignment statement creates new variables and gives them values:

```
>>> x=10
>>> school="kve"
```

A ‘print’ statement is to display something on the screen/monitor. We use the print() function for this purpose. And input() function is used to receive the input from the user through the keyboard.

Precedence of Operators:



Order of Precedence	Operators	Description
1	**	Exponentiation (raise to the power)
2	~ ,+, -	Complement, unary plus and unary minus
3	* ,/, %, //	Multiply, divide, modulo and floor division
4	+, -	Addition and subtraction
5	<= , < , > , >=, == , !=	Relational and Comparison operators
6	=, %=, /=, //=, -=, +=, *=, **=	Assignment operators
7	is, is not	Identity operators
8	in, not in	Membership operators
9	not	Logical operators
10	and	
11	or	

Operator precedence affects how an expression is evaluated. Each operator is having a priority when used in an expression in combination with other operators.
For example, x = 7 + 3 * 2;
here, x is assigned 13, not 20 because operator * has higher precedence than +, so it first multiplies 3*2 and then adds into 7.
Example 1:

```
>>> 3+4*2 11
```

Multiplication gets evaluated before the addition operation

```
>>> (10+10)*2 40
```

Parentheses () overriding the precedence of the arithmetic operators

Example 2:

```
a = 20
b = 10
c = 15
d = 5
e = 0

e = (a + b) * c / d       #( 30 * 15 ) / 5
print("Value of (a + b) * c / d is ", e)
e = ((a + b) * c) / d    # (30 * 15 ) / 5
print("Value of ((a + b) * c) / d is ",  e)
e = (a + b) * (c / d)    # (30) * (15/5)
print("Value of (a + b) * (c / d) is ",  e)

e = a + (b * c) / d; # 20 +(150/5)
print("Value of a + (b * c) /d is", e)
```

Output:

```
Value of (a + b) * c / d is  90.0
Value of ((a + b) * c) / d is  90.0
Value of (a + b) * (c / d) is  90.0
Value of a + (b * c) /d is 50.0
```

Comments:

Comments are discarded by the Python interpreter. Comments acts as some message to the Programmer. It can be used as documents.

Single-line comments begins with a hash(#) symbol and is useful in mentioning that the whole line should be considered as a comment until the end of line.
A Multi line comment is useful when we need to comment on many lines. In python, triple double quote(“ “ “) and single quote(‘ ‘ ‘)are used for multi-line commenting.
Example:

```
     # this is a single line comment
    ‘’’ I am a multi line comment’’’
```

Errors and Exceptions:

Python Errors and Built-in Exceptions: Python (interpreter) raises exceptions when it encounters errors. When writing a program, we, more often than not, will encounter errors. Error caused by not following the proper structure (syntax) of the language is called syntax error or parsing error.

ZeroDivisionError: ZeroDivisionError in Python indicates that the second argument used in a division (or modulo) operation was zero.

Overflow Error: OverflowError in Python indicates that an arithmetic operation has exceeded the limits of the current Python runtime. This is typically due to excessively large float values, as integer values that are too big will opt to raise memory errors instead.

Import Error: It is raised when you try to import a module which does not exist. This may happen if you made a typing mistake in the module name or the module doesn't exist in its standard path. In the example below, a module named "non_existing_module" is being imported but it doesn't exist, hence an import error exception is raised.

Index Error: An IndexError exception is raised when you refer a sequence which is out of range. In the example below, the list abc contains only 3 entries, but the 4th index is being accessed, which will result an IndexError exception.

Type Error: When two unrelated type of objects are combined, TypeErrorexception is raised.In example below, an int , and a string is added, which will result in TypeError exception.

Indentation Error: Unexpected indent - As mentioned in the "expected an indented block" section, Python not only insists on indentation, it insists on consistent indentation. You are free to choose the number of spaces of indentation to use, but you then need to stick with it.

Syntax errors: These are the most basic type of error. They arise when the Python parser is unable to understand a line of code. Syntax errors are almost always fatal, i.e. there is almost never a way to successfully execute a piece of code containing syntax errors

Run-time error: A run-time error happens when Python understands what you are saying, but runs into trouble when following your instructions.

Key Error : Python raises a KeyError whenever a dict() object is requested (using the format a= adict[key]) and the key is not in the dictionary.
Value Error: In Python, a value is the information that is stored within a certain object. To encounter a ValueError in Python means that is a problem with the content of the object you tried to assign the value to.

Python has many built-in exceptions which forces your program to output an error when something in it goes wrong. In Python, users can define such exceptions by creating a new class. This exception class has to be derived, either directly or indirectly, from Exception class.

Different types of exceptions:

- ArrayIndexOutOfBoundException.
- ClassNotFoundException.
- FileNotFoundException.
- IOException.
- InterruptedException.
- NoSuchFieldException.
- NoSuchMethodException

Modules:

Python module can be defined as a python program file which contains a python code including python functions, class, or variables. In other words, we can say that our python code file saved with the extension (.py) is treated as the module. We may have a runnable code inside the python module. A module in Python provides us the flexibility to organize the code in a logical way. To use the functionality of one module into another, we must have to import the specific module.

Syntax:

import <module-name>

Every module has its own functions, those can be accessed with . (dot)

Note: In python we have help ()

Enter the name of any module, keyword, or topic to get help on writing Python programs and using Python modules. To quit this help utility and return to the interpreter, just type "quit".

Some of the modules like os, date, and calendar so on……

```
>>> import sys
>>> print (sys.version)
3.8.0 (tags/v3.8.0:fa919fd, Oct 14 2019, 19:21:23) [MSC v.1916 32 bit (Intel)]
```

```
>>> print(sys.version_info)
sys.version_info(major=3, minor=8, micro=0, release level='final', serial=0)
```

```
>>> print(calendar.isleap(2020))
True
>>> print(calendar.isleap(2017))
False
```

Control Structures

Flow of execution means the way in which the instructions are executed. It can be

1. Sequential execution
2. Selection/ Conditional statements
3. Iterations/loop

Sequential execution means executing the statements one by one starting from the first instruction onwards.

Selection:

Selection or Conditional statements helps us to execute some statements based on whether the condition is evaluated to True or False.

Use of if statement:

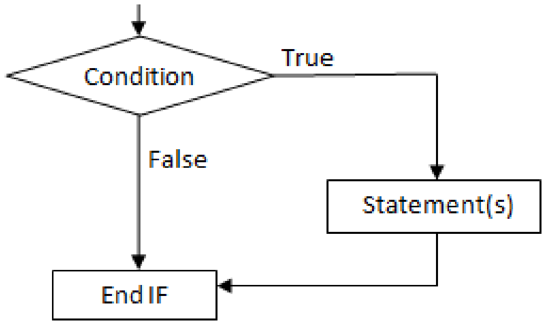
The if statement contains a logical expression using which data is compared and a decision is made based on the result of the comparison.

Syntax:

```
if <expression>:
   statement(s)
```

If the boolean expression evaluates to TRUE, then the block of statement(s) inside the if statement is executed. If boolean expression evaluates to FALSE, then the first set of code after the end of the if statement(s) is executed.

Flow chart representing the execution of 'if' statement:



Example: Python if Statement

```
a = 3
if a > 2:
   print(a, "is greater")
    print("done")
a = -1
if a < 0:
   print(a, "a is smaller")
```

```
    print(“Finish”)
```

Output:

```
3 is greater
done
-1 a is smaller
Finish
```

```
a=10
if a>9:
     print("A is Greater than 9")
```

Output:

```
A is Greater than 9
```

Alternative if (If-else):

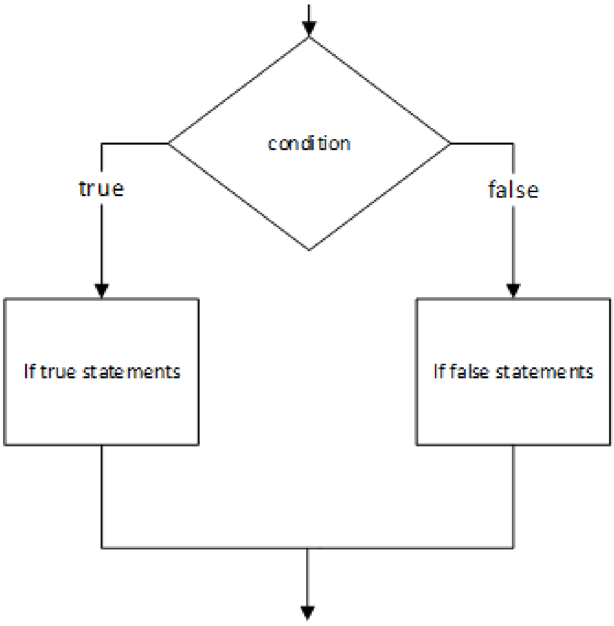
An else statement can be combined with an if statement. An else statement contains the block of code (false block) that executes if the conditional expression in the if statement resolves to 0 or a FALSE value.

The else statement is an optional statement and there could be at most only one else Statement following if.

Syntax of if - else:

```
if test expression:
        Body of if stmts
else:
        Body of else
```

Flow Chart:



Example of if - else:

```
a=int(input('enter the number'))
if a>5:
        print("a is greater")
else:
        print("a is smaller than the input given")
```

Output:

```
enter the number 2
a is smaller than the input given
```

```
a=10 b=20
if a>b:
   print("A is Greater than B")
else:
    print("B is Greater than A")
```

Output:

```
B is Greater than A
```

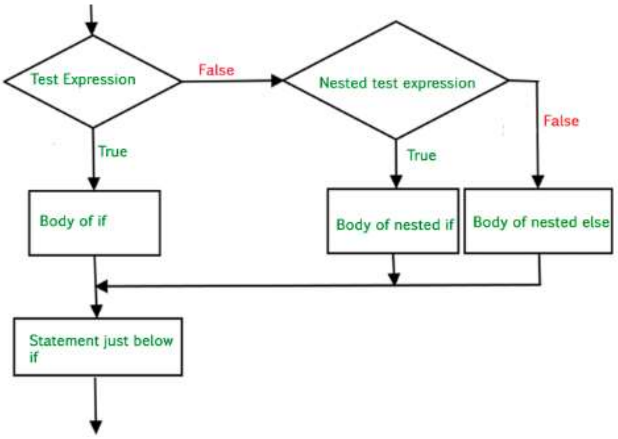
Chained Conditional: (If-elif-else):

The elif statement allows us to check multiple expressions for TRUE and execute a block of code as soon as one of the conditions evaluates to TRUE. Similar to the else, the elif statement is optional. However, unlike else, for which there can be at most one statement, there can be an arbitrary number of elif statements following an if.

Syntax of if – elif - else :

```
If  <test expression>:
        Body of if stmts
elif  < test expression>:
       Body of elif stmts
else:
        Body of else stmts
```

Flow Chart:



Example of if - elif – else:

```
a=int(input('enter the number'))
b=int(input('enter the number'))
c=int(input('enter the number'))
if a>b:
    print("a is greater")
elif b>c:
    print("b is greater")
else:
    print("c is greater")
```

Output:

```
enter the number 5
enter the number  2
enter the number 9
 a is greater
>>>
enter the number 2
enter the number 5
enter the number 9
 c is greater
```

```
var = 100
if var == 200:
    print("1 - Got a true expression value")
    print(var)
elif var == 150:
    print("2 - Got a true expression value")
    print(var)
elif var == 100:
    print("3 - Got a true expression value")
    print(var)
else:
    print("4 - Got a false expression value")
    print(var)
```

Output:

```
3 - Got a true expression value
100
```

Iteration/Repetition:

A loop statement allows us to execute a statement or group of statements multiple times as long as the condition is true. Repeated execution of a set of statements with the help of loops is called iteration.

Loops statements are used when we need to run same code again and again, each time with a different value.

In Python Iteration (Loops) statements are of three types:

- While Loop
- For Loop
- Nested For Loops

While loop:

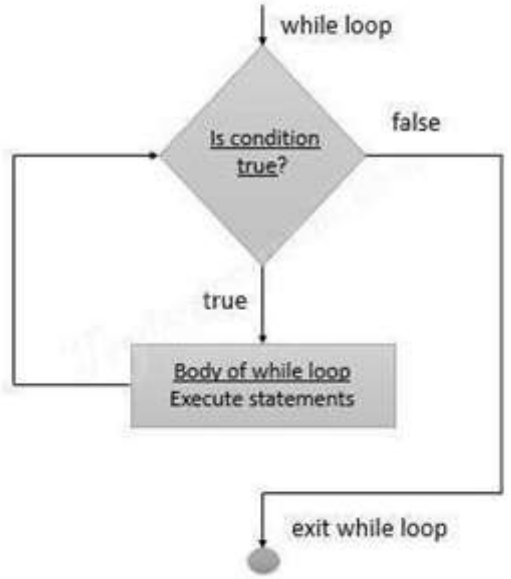
Loops are either infinite or conditional. Python while loop keeps iterating a block of code defined inside it until the desired condition is met.

- The while loop contains a Boolean expression and the code inside the loop is repeatedly executed as long as the Boolean expression is true.
- The statements that are executed inside while can be a single line of code or a block of multiple statements

Syntax:

```
while(expression):
      Statement(s)
```

Flow Chart:



Example Programs:

```
i=1
while i<=6:
        print("KV School")
        i=i+1
```

Output:

```
KV School
KV School
KV School
KV School
KV School
KV  School
```

```
i=1
while i<=3:
    print("KV School",end=" ")
    j=1
    while j<=1:
        print("CS DEPT",end="")
        j=j+1
    i=i+1
    print()
```

Output:

```
KV School CS  DEPT
KV School CS DEPT
KV School CS DEPT
```

For loop:

Python for loop is used for repeated execution of a group of statements for the desired number of times. It iterates over the items of lists, tuples, strings, the dictionaries and other iterable objects.

Syntax:

```
for <loopvariable>  in <sequence>:
        Statement(s)
```

A sequence or iterable object is used to execute the loop. The for loop executes for a definite number of times. During each of the iteration, the loop variable holds a value from the sequence.

Example 1:

```
L = [ 1, 2, 3, 4, 5]
for var  in L:
    print( var, end=’  ‘)
```

Output of the above code is:

```
 1 2 3 4 5
```

Example 2:

```
 for k in range(10):
        print(k, end =’ , ‘)

  output:    0,1,2,3,4,5,6,7,8,9
```

Example 3:

```
 for   k in range(1, 10):
        if  k% 2 ==0:
                 print(k, end = ‘ , ‘)
 Output:    2, 4, 6, 8
```

Example 4:

```
 list = ['K','V','S','C','H']
 i = 1
for item in list:
      print ('School ',i,' is ',item)
       i = i+1
Output:
School 1 is K
School 2 is V
School 3 is S
School 4 is C
School 5 is H
```

Example 5:

```
#Iterating over a Tuple:
tuple = (2,3,5,7)
print ('These are the first four prime numbers ')
#Iterating over the tuple
for a in tuple:
     print (a)

Output:
These are the first four prime numbers 2
3
5
7
```

Example 6:

```
# Iterating over a dictionary:
#creating a dictionary
college = {"ces":"block1","it":"block2","ece":"block3"}
#Iterating over the dictionary to print keys
print ('Keys are:')
 for keys in college:
       print (keys)
#Iterating over the dictionary to print values
print ('Values are:')
for blocks in college.values():
     print(blocks)
```

```
Output:
Keys are:

ces
it
ece
Values are:
block1
block2
block3
```

Example 7:

```
#Iterating over a String:
#declare a string to iterate over
college = 'KVSCH'
#Iterating over the string
for name in college:
    print (name)
```

```
Output:
K
V
 S
C
H
```

Nested For loop:

When one Loop is defined within another, it is called Nested Loops.

Syntax:

for val in sequence1:

```
        for val in sequence2:
                statements
```

```
# Example 1 of Nested For Loops (Pattern Programs)
for i in range(1,6):
     for j in range(0,i):
         print(i, end=" ")
     print('')
Output:
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
```

```
# Example 2 of Nested For Loops (Pattern Programs)
for i in range(1,6):
   for j in range(5,i-1,-1):
      print(i, end=" ")
   print('')

Output:
1 1 1 1 1 1
2 2 2 2
3 3 3
4 4
5
```

break and continue:

In Python, break and continue statements can alter the flow execution of a normal loop. Sometimes we wish to terminate the current iteration or even the whole loop without checking test expression/completing it. The break and continue statements are used in these cases.

'break:' statement

The break statement terminates the loop containing it and control of the program flows to the statement specified immediately after the body of the loop. If 'break' statement is written within a nested loop (loop inside another loop), break will terminate the innermost loop.

'continue' Statement

The 'continue' statement is used to skip the current iteration / rest of the code inside a loop for the current iteration only. Loop does not terminate but continues on with the next iteration.

'pass' statement:

In Python programming, pass is a null statement. The difference between a comment and pass statement in Python is that, while the interpreter ignores a comment entirely, pass is not ignored. 'pass' is just a place holder for functionality to be added later.

Strings:

A string is a group/ a sequence of characters. Since Python has no provision for arrays, we simply use strings. A sequence of any type of characters enclosed within quotes, is a string. This is how we declare a string. We can use a pair of single or double quotes. Every string object is of the type ‘str’.

```
>>> type("name")
<class 'str'>
>>> name=str()
>>>name
''
>>> a=str('kvsch')
>>> a
' kvsch'
>>> a=str(kvsch)
>>> a[2]
's'
>>> fruit = 'banana'
>>> letter = fruit[1]
```

The second statement selects character number 1 from fruit and assigns it to letter. The expression in brackets is called an index. The index indicates which character in the sequence we want.

String slices:

A segment or a part of a string is called a slice. Selecting a slice is similar to selecting a character:

Subsets of strings can be taken using the slice operator ([ ] and [:]) with indexes starting at 0 in the beginning of the string and working their way from -1 at the end.

Slice out substrings, sub lists, sub Tuples using index.

Syntax:

[Start: stop: steps]

- Slicing will start from index and will go up to stop in step of steps.
- Default value of start is 0,
- Stop is last index of list
- And for step default is 1

For example, 1:

```
str = 'Hello World!'
print (str) # Prints complete string
print (str[0] )# Prints first character of the string
print (str[2:5] ) # Prints characters starting from 3rd to 5th
print (str[2:]) # Prints string starting from 3rd character
print ( str * 2 )# Prints string two times
print (str + "TEST") # Prints concatenated string
```

Output:

```
Hello World!
H
llo
llo World!
Hello World!Hello World!
Hello World!TEST
```

Example 2:

```
>>> x='computer'
>>> x[1:4]
'omp'
>>> x[1:6:2]
'opt'
>>> x[3:]
puter
>>> x[:5]
'compu'
>>> x[-1]
'r'
>>> x[-3:]
'ter'
>>> x[:-2]
'comput'
>>> x[::-2]
'rtpo'
>>> x[::-1]
'retupmoc'
```

Note: strings are immutable, which means we can’t change an existing string. The best we can do is create a new string that is a variation from the original:

String concatenation and repetition operators:

‘+’ is called concatenation operator. When both the operands are strings, it will combine them.

‘*’ is called repetition operator and it causes repetition of the characters the specified number of times.

String functions and methods:

There are many methods to operate on String.

S.no	Method name	Description
1.	isalnum()	Returns true if string has at least 1 character and all characters are alphanumeric and false otherwise.
2.	isalpha()	Returns true if string has at least 1 character and all characters are alphabetic and false otherwise.
3.	isdigit()	Returns true if string contains only digits and false otherwise.
4.	islower()	Returns true if string has at least 1 cased character and all cased characters are in lowercase and false otherwise.
5.	isnumeric()	Returns true if a string contains only numeric characters and false otherwise.
6.	isspace()	Returns true if string contains only whitespace characters and false otherwise.
7.	istitle()	Returns true if string is properly "titlecased" and false otherwise.
8.	isupper()	Returns true if string has at least one cased character and all cased characters are in uppercase and false otherwise.
9.	replace(old,new [, max])	Replaces all occurrences of old in string with new or at most max occurrences if max given.
10.	split()	Splits string according to delimiter str (space if not provided) and returns list of substrings;
11.	count()	Occurrence of a string in another string
12.	find()	Finding the index of the first occurrence of a string in another string
13.	swapcase()	Converts lowercase letters in a string to uppercase and viceversa
14.	startswith(str,beg=0, end=len(string))	Determines if string or a substring of string (if starting index beg and ending index end are given) starts with substring str; returns true if so and false otherwise.

List:

With the help of lists, we can store different types of items in a variable. It helps us to store collection of data. List items are ordered, changeable and can be repeated. Indexes are used to access the items.

Lists are mutable.

Python has a set of built-in methods that you can use on lists

Method	**Descriptio n**
append()	Adds an element at the end of the list
clear()	Removes all the elements from the list
copy()	Returns a copy of the list
count()	Returns the number of elements with the specified value
extend()	Add the elements of a list (or any iterable), to the end of the current list
index()	Returns the index of the first element with the specified value
insert()	Adds an element at the specified position
pop()	Removes the element at the specified position
remove()	Removes the first item with the specified value
reverse()	Reverses the order of the list
sort()	Sorts the list

Whenever we need an array, we can use lists. We can construct / create list in many ways.

Example 1:

```
list1=[1,2,3,'A','B',7,8,[10,11]]
print(list1)
```

output:

```
[1, 2, 3, 'A', 'B', 7, 8, [10, 11]]
```

Example 2:

```
x=list()
print(x)
```

output:
x []

Example 3:
Name = " abcdefg"
L = list(Name)
print(L)
Output:
['a', 'b', 'c', 'd', 'e', 'f', 'g']

Basic List Operations:

Lists respond to the + and * operators much like strings; they mean concatenation and repetition here too, except that the result is a new list, not a string.

Python Expression	**Results**	**Description**
len([1, 2, 3])	3	Length
[1, 2, 3] + [4, 5, 6]	[1, 2, 3, 4, 5, 6]	Concatenation
['Hi!'] * 4	['Hi!', 'Hi!', 'Hi!', 'Hi!']	Repetition
3 in [1, 2, 3]	True	Membership
for x in [1, 2, 3]: print x,	1 2 3	Iteration

Note: indexing and slicing are similar to that of strings.
For example:
L= ['kvsch', 'school', 'KVSCH!']

Python Expression	**Results**	**Description**
L[2]	KVSCH	Offsets start at zero
L[-2]	school	Negative: count from the right
L[1:]	['school', 'KVSCH!']	Slicing fetches sections

List slices: - Examples

```
>>> list1= [1,2,3,4,5,6,7,8,9,10]
>>> list1[1:]
Output: [2, 3, 4, 5, 6, 7, 8, 9, 10]
  >>>list1[:1]
 Output:[1]
>>> list1[2:5]
 Output: [3, 4, 5]
>>> list1[:6]
 Output: [1, 2, 3, 4, 5, 6]
 >>>list1 [1:2:4]
 Output:[2]
 >>> ist1[1:8:2]
 output: [2, 4, 6, 8]
```

List Comprehension:

This is a way to create a new list using an existing list.

Syntax:

NewList = [expression   for variable in iterableobject   if condition == True]

Example1:

```
list1=[x**2 for x in range(10)]
print(list1)
output:
[ 0,1,4,9,16,25,36,49,64,81]
```

Note: in the above given code, a new list named list1 is created using the expression x**2, where the for loop is used to assign different values to ‘x’ using the range( ) function.

Example 2:

```
x=[z**2  for z in range(10) if z>4]
print(x)
Output: [25, 36, 49, 64, 81]
```

Example 3:

```
x=[x ** 2 for x in range (1, 11)   if x % 2 == 1]
print(x)
Output: [1, 9, 25, 49, 81]
```

Tuples:

A tuple is a collection which is ordered and unchangeable. In Python tuples are created by enclosing items within round brackets.

- Supports all operations for sequences.
- Immutable, but member objects may be mutable.
- If the contents of a list shouldn’t change, use a tuple to prevent items from accidently being added, changed, or deleted.
- Tuples are more efficient than list due to python’s implementation.

We can create tuple in different ways.

```
X=( )  # an empty tuple
X=(1,2,3) # a tuple with three elements
X=tuple(list1)
X=1,2,3,4
```

Accessing elements of a tuple:

We can use index to access the elements of a tuple. From left to right, index varies from 0 to n-1, where n is total number of elements in the tuple. From right to left, the index starts with -1 and the index of the leftmost element will be –n, where n is the number of elements.

Syntax:

<Name of the tuple>[index]

Example:

```
T = ( 1, 2,4,6)
print(T[0])
Output: 1
print(T[--1])
Output: 6
```

Modifying the elements of a tuple:

Once created, we cannot modify the elements of a tuple. Tuple is immutable. If we try to modify the Element, it will generate ‘TypeError’

Some important functions which can be used with tuple:

count (): Returns the number of times a specified value occurs in a tuple

Example:

```
>>> x=(1,2,3,4,5,6,2,10,2,11,12,2)
>>> x.count(2)
output: 4
```

index (): Searches the tuple for a specified value and returns its position in the tuple.

```
>>> x=(1,2,3,4,5,6,2,10,2,11,12,2)
>>> x.index(2)
Output: 1
```

(Or)

```
>>> x=(1,2,3,4,5,6,2,10,2,11,12,2)
>>> y=x.index(2)
>>> print(y)
Output: 1
```

len(): To know the number of items or values present in a tuple, we use len().

```
>>> x=(1,2,3,4,5,6,2,10,2,11,12,2)
>>> y=len(x)
>>> print(y)
output: 12
```

Tuple comprehension:

Tuple Comprehensions are special: The result of a tuple comprehension is special. You might expect it to produce a tuple, but what it does is produce a special "generator" object that we can iterate over.

For example:

```
>>> x = (i for i in 'abc') #tuple comprehension
>>> x
<generator object <genexpr> at 0x033EEC30>

>>> print(x)
<generator object <genexpr> at 0x033EEC30>
```

You might expect this to print as ('a', 'b', 'c') but it prints as <generator object <genexpr> at 0x02AAD710>. The result of a tuple comprehension is not a tuple: it is actually a generator. The only thing that you need to know now about a generator is that you can iterate over it, but ONLY ONCE.

Example:

```
>>> x = (i for i in 'abc')
>>> for i in x:
print(i)

Output:
a
b
c
```

Example:

Create a list of 2-tuples like (number, square):

```
>>> z=[(x, x**2) for x in range(6)]
>>> z
[(0, 0), (1, 1), (2, 4), (3, 9), (4, 16), (5, 25)]
```

Set comprehension:

Similarly, to list comprehensions, set comprehensions are also supported by Python:

Example:

```
>>> a = {x for x in 'abracadabra' if x not in 'abc'}
>>> a
```

Output:

```
{'r', 'd'}
```

```
>>> x={3*x for x in range(10) if x>5}
>>> x
```

Output:

```
{24, 18, 27, 21}
```

Note: In case of set, elements are enclosed within { }.

Dictionaries:

A dictionary is a collection which is unordered, changeable and indexed. In Python dictionaries are created with curly brackets, and they have keys and values. It is a mapping. Values can be of any type. Only immutable types are allowed as keys. It is not possible to repeat keys. They are to be unique.

- Key-value pairs
- Unordered

We can construct or create dictionary like:
X={1:’A’,2:’B’,3:’c’} X=dict([(‘a’,3) (‘b’,4)]
X=dict(‘A’=1,’B’ =2)

Example:

```
>>> dict1 = {"brand":"mrcet","model":"college","year":2004}
>>> dict1
{'brand': 'mrcet', 'model': 'college', 'year': 2004}
```

Functions available for handling dictionary:

Method	Description
clear()	Remove all items form the dictionary.
copy()	Return a shallow copy of the dictionary.
fromkeys(seq[, v])	Return a new dictionary with keys from seq and value equal to v (defaults to None).
get(key[,d])	Return the value of key. If key does not exit, return d (defaults to None).
items()	Return a new view of the dictionary's items (key, value).
keys()	Return a new view of the dictionary's keys.
pop(key[,d])	Remove the item with key and return its value or d if key is not found. If d is not provided and key is not found, raises KeyError.
popitem()	Remove and return an arbitary item (key, value). Raises KeyError if the dictionary is empty.
setdefault(key[,d])	If key is in the dictionary, return its value. If not, insert key with a value of d and return d (defaults to None).
update([other])	Update the dictionary with the key/value pairs from other, overwriting existing keys.
values()	Return a new view of the dictionary's values
remove():	It removes or pop the specific item of dictionary
del( )	Deletes a particular item
len( )	we use len() method to get the length of dictionary

Note: keys are used to access or modify the items of a dictionary.

Comprehension:

Dictionary comprehensions can be used to create dictionaries from arbitrary key and value expressions

Syntax:

```
        <dictionary> = { expression   for <variable> in sequence }
```

Example:

```
>>> z={x: x**2 for x in (2,4,6)}
>>> z
{2: 4, 4: 16, 6: 36}
>>> dict11 = {x: x*x for x in range(6)}
>>> dict11
{0: 0, 1: 1, 2: 4, 3: 9, 4: 16, 5: 25}
```

Important Points to remember:

- Data types: int, float, complex, Boolean, string, list, tuple, set, dictionary
- Variables are used to hold data. They are having memory address and values of particular types are stored.
- It is not possible to modify immutable types/objects.
- Mutable objects/types can be modified.
- Dictionary uses keys to identify values. Key can be of any immutable type and is not possible to change or duplicate.
- Values of a dictionary can be of any type.
- Three different programming constructs supported by Python are Sequence, Selection, and Iteration.
- For selection, Python uses if, if..else,   and if,,,elif…elif,,,else constructs.
- Different types of loops are while loop and for loop.
- While loop will continue till the specified number of times or till the given test expression is True.
- ‘for’ loop is used when we have to iterate over a sequence. How many times it is to be repeated is Known.
- Strings are sequence of characters. If enclosed in single or double quotes, it must be confined to a line.
- String enclosed within triple quotes can be of multiple lines and are also called as docstrings.
- List comprehension is a way to create a new list using an existing sequence.
- Examples for immutable types: int, float, string, tuple
- Examples for mutable types: list

Multiple Choice Questions

1. Which of the following is not a keyword?

   a) Eval b) assert c) nonlocal d) pass

2. What is the order of precedence in python?

   (i) Parentheses ii) Exponential iii) Multiplication iv) Division v) Addition vi) Subtraction

   a) i,ii,iii,iv,v,vi
   b) ii,i,iii,iv,v,vi
   c) ii,i,iv,iii,v,vi
   d) i,ii,iii,iv,vi,v

3. What error occurs when you execute the following Python code snippet?

```
apple = mango
```

a) SyntaxError
b) NameError
c) ValueError
d) TypeError

4. Find the output.

```
def example(a):
        a = a+2
            a=a*2
 return a
>>>example("hello")
```

a. indentation Error
b. cannot perform mathematical operation on strings
c. hello2
d. hello2hello2

5. What will be the value of X in the following Python expression?

   X = 2+9*((3*12)-8)/10

   a) 30.0
   b) 30.8
   c) 28.4
   d) 27.2

6. Select all options that print. hello-how-are-you

   a. print(‘hello’, ‘how’, ‘are’, ‘you’)
   b. print(‘hello’, ‘how’, ‘are’, ‘you’ + ‘-‘ * 4)

c. print(‘hello-‘ + ‘how-are-you’)
d. print(‘hello’ + ‘-‘ + ‘how’ + ‘-‘ + ‘are’ + ‘you’)

7. Which of the following can be used as valid variable identifier(s) in Python?

i. total
ii. 7Salute
iii. Que$tion
iv. global

8. Which of the following statement is correct for an AND operator?
a) Python only evaluates the second argument if the first one is False
b) Python only evaluates the second argument if the first one is True
c) Python only evaluates True if any one argument is True
d) Python only evaluates False if any one argument is False

9. Which of the following forces an expression to be converted into specific type?
a) Implicit type casting b) Mutable type casting
c) Immutable type casting d) Explicit type casting

10. Which point can be considered as difference between string and list?

a. Length c. Indexing and Slicing
b. Mutability d. Accessing individual elements

11.Which of the following statement is true for extend () list method?
a) adds element at last c) adds multiple elements at last
b) adds element at specified index d) adds elements at random index

12.The statement del l[1:3] do which of the following task?
a) delete elements 2 to 4 elements from the list
b) delete 2nd and 3rd element from the list
c)deletes 1st and 3rd element from the list
d)deletes 1st, 2nd and 3rd element from the list

13.If l=[11,22,33,44], then output of print(len(l)) will be
a)4 b)3 c) 8 d) 6

14.Which of the following method is used to delete element from the list?
a)del()
b)delete()
c)pop()
d)All of these

15.The step argument in range() function_____________.
a. indicates the beginning of the sequence
b. indicates the end of the sequence

c. indicates the difference between every two consecutive numbers in the sequence

d. generates numbers up to a specified value

16. If D=1,2,3

What will be the data type of D?

a)List

b)tuple

c)set

d)invalid type

17.Consider the following code

```
L=[‘a’,’b’,’c’,’d’]
L.pop(-1)
print(L)
```

What would be the output?

a)’d’

b)[‘a’,’b’,’c’,’d’]

c)[‘a’,’b’,’c’]

d)error

18 What is the output when the following code is executed?

```
>>>name=[‘Aadi’,’Beena’,’Charlie’,’David’]
>>>print(name[-1][-1])
```

a)’d’

b)’D’

c)’i’

d)Charlie

19.T=(“See You”)

What is the data type of T?

a)tuple

b)string

c)List

d)set

20.Assertion(A):After adding element in a list, its memory location remains same

Reason(R):List is a mutable data type

a)Both A and R are True and R is the correct explanation for A

b) Both A and R are True and R is not the correct explanation for A

c)A is True but R is False

d)A is False but R is True

21. Assertion(A):An identifier may be combination of letters and numbers.

Reason(R):No special symbols are permitted in an identifier name

a)Both A and R are True and R is the correct explanation for A

b) Both A and R are True and R is not the correct explanation for A

c)A is True but R is False

d)A is False but R is True

Answers of MCQ:

1)a 2)a 3)b 4)a 5)d 6)c 7)a 8)b 9) d 10)b 11)b 12)b 13)a 14)c
15)c 16)b 17)c 18)a 19)b 20)a 21)c

Very Short Answer Type Questions

1. Give the output of the following

```
Sum = 0
for k in range(5):
      Sum = Sum+k
print(Sum)
```

2. Give the output of the following

```
Sum = 0
for k in range(10 , 1, -2):
    Sum = Sum+k
print(Sum)
```

3. Give the output of the following

```
for k in range(4):
    for j in range(k):
          print(‘*’, end = ‘  ‘)
    print()
```

4. Give the output of the following

```
for k in range(5,0, -1):
     for j in range(k):
            print(‘*’, end=’ ‘)
     print()
```

5. How many times the following loop will execute? Justify your answer

```
A = 0
while True:
    print(A)
    A =A+1
```

6. Give the output of the following. Also find how many times the loop will execute.

```
A = 0
while A<10:
            print(A, ‘  , ‘)
   A =A+1
```

7. Give the output of the following. Also find how many times the loop will execute.

```
A = 0
while A<10:
   print(A, ‘  , ‘)
   A =A+1
print(‘\n’, A)
```

8. Give the output of the following

```
T = (5)
T = T*2
print(T)
```

9. Give the output of the following

```
T = (5, )
T = T*2
print(T)
```

10. What kind of error message will be generated if the following code is executed

```
A = 5
B = ‘hi’
d = A+B
print(D)
```

11. Give the output of the following

```
L = [1,2,3,4,5,6,7,8,9]
print(L[:])
```

12. Give the output of the following

```
L = [1,2,3,4,5,6,7,8,9]
print(L[: -1])
```

13. Find the output of the following

```
S = ‘abcdefgh’
L = list(S)
print(L[1:4])
```

14. Give the output of the following

```
L = [1,2,3,4,5,6,7,8,9]
print(L.count(2))
print(L.index(2)
```

15. Write python code to sort the list, L, in descending order.

16. Give the output of the following

```
x=[4,5,66,9]
y=tuple(x)
print( y)
```

17.Guess the output?

```
if 8:
    print(“Hello”)
if(-5):
     print(“Hai”)
```

18.Find output.

```
vowels=['a','e','i']
vowels.append('o')
vowels.append(['u'])
vowels.extend(['A','E'])
print("New List:",vowels)
```

19.Given the following declaration, what will be the output of the following:

```
Lst=(1,2,3,4,5)
del Lst[2]
print(Lst)
```

20.Identify the statement(s) from the following options which will raise TypeError exception.

a)print(‘5’*’3’)

b)print(5*3)

c)print(‘5’+3)

d)print(‘5’+’3’)

e)print(‘5’*3)

Answers for VSA Questions

1)10
2)30
3)

```
*
*  *
*   *   *
```

4)

```
 * * * * *
  *  * * *
  *  *  *
  *  *
  *
```

5)infinite loop. Condition / test expression is always Ture.

6) 0,1,2,3,4,5,6,7,8,9

7) 0,1,2,3,4,5,6,7,8,9
10

8) 10 Note: here T is an integer

9) (5, 5), Note: here T is tuple

10)TypeError

11) [1,2,3,4,5,6,7,8,9]

12) [1,2,3,4,5,6,7,8]

13) ['b', 'c', 'd']

14) 1
1

15) L.sort(reverse= True)

16) (4, 5, 66, 9)

17)Hello
Hai

18) New List: ['a', 'e', 'i', 'o', ['u'], 'A', 'E']

19)Error

20) a)print('5'*'3')

c)print('5'+3)

Short Answer Type Questions

1.Consider the following dictionary ‘D’. Display all the items of the dictionary as individual tuple

D = {‘A’: 20, ‘B’: 30, ‘C’:40. ‘D’: 50}

2.Write Python code to remove an element as entered by the user form the list, L

3.Create a list k, by selecting all the odd numbers within the given range, m and n. User will input the values of m , n at run time

4.Write Python code to create and add items to a dictionary. Ask the user to input key value pairs. Continue as long as the user wishes.

5.Write Python code to find whether the given item is present in the given list using for loop.

6.Create a list, L, with the squares of numbers from 0 to 10

7.Write a program to accept a list of numbers and to create a dictionary with two keys ODD and EVEN whose values are all the odd numbers from the list and all the even numbers from the list respectively.

8.Mr. Rahul wants created a dictionary to store the details of his students and to manipulate the data.

He wrote a code in Python, help him to complete the code:

```
 studentDict = ______       # stmt 1
 n = int(input("How Many Students you Want To Input?"))
for i in range(___):         # stmt 2 - to enter n number of students data
        rollno = input("Enter Roll No:")
        name = input("Enter Name:")
        physicsMarks = int(input("Enter Physics Marks:"))
         chemistryMarks = int(input("Enter Chemistry Marks:"))
        mathMarks = int(input("Enter Maths Marks:"))
        studentDict[rollno]=__________ # stmt 3
```

Answers /Hints: Short answer Type Questions.

```
1) for k in D.items():
          print(k)
```

```
2) a =int('input the item to be deleted')
   l.remove(a)
```

```
3) m = int(input('lower limit'))
   n = int(input('upper limit'))
   n = n+1
   L = [x for x in range(m, n) if x%2!=0]
```

```
4)
  D = { }
  while Ture:
        K = input('type a key')
        V = int(input('type the value')
        D[K] = V
        C = input('type 'y' to add more')
        if C!='y':
                break
```

```
5)      flag = 0

        L = eval(input('input a list on numbers'))

        E = int(input('item to be searched')

        K  =len(L)

        for p in range(K):

                if E ==L(p):

                        flag = 1

                        print('found and index is  ',p)

        if flag==0:

                print('not found')
```

```
 6)
    list1=[]
   for x in range(10):
      list1.append(x**2)

    list1
   Output:
   [0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
  OR
   list1=list(map(lambda x:x**2, range(10)))
```

7)

```
L=eval(input("Enter a list"))
D={"ODD":[],"EVEN":[]}
for i in L:
    if i%2!=0:
        D["ODD"].append(i)
    else:
        D["EVEN"].append(i)
print(D)
```

8)

Statement 1 : StudentDict = dict( )

Statement 2 = for i in range(n):

Statement 3: studentDict[rollno]=[name, physicsMarks, chemistryMarks, mathMarks]

FUNCTIONS

- A function is a block of code that performs a specific task.
- A function usually performs a single task of a large program.
- Functions help in the division of our program into smaller, modular portions known as divide and conquer approach.

FUNCTION DEFINITION

A function is a named sequence of statement(s) that performs a computation. It contains line of code(s) that are executed sequentially from top to bottom by Python interpreter. They are the most important building blocks for any software in Python.

Functions can be categorized as –

i. Built-in functions
ii. Functions defined in module
iii. User defined functions

a) Built-in functions

Built in functions are the function(s) that are built into Python and can be accessed by a programmer.

These are always available and for using them, we don’t have to import any module (file).

Name of the function	Description	Example

abs (x)	It returns distance between x and zero, where x is a numeric expression.	>>>abs(-45)
		45
		>>>abs(119)
		119

max(x, y, z,)	It returns the largest of its arguments: where x, y and z are numeric variable/expression.	>>>max(80, 100, 1000)
		1000
		>>>max(-80, -20, -10)
		-10

min(x, y, z,)	It returns the smallest of its arguments; where x, y, and z are numeric Variable/expression.	>>> min(80, 100, 1000) 80 >>>min(-80, -20, -10) -80
Name of the function	**Description**	**Example**

abs (x)	It returns distance between x and zero, where x is a numeric expression.	>>>abs(-45) 45 >>>abs(119) 119

max(x, y, z,)	It returns the largest of its arguments: where x, y and z are numeric variable/expression.	>>>max(80, 100, 1000)
		1000
		>>>max(-80, -20, -10)
		-10

min(x, y, z,)	It returns the smallest of its arguments; where x, y, and z are numeric Variable/expression.	>>> min(80, 100, 1000) 80 >>>min(-80, -20, -10) -80

cmp(x, y)	It returns the sign of the difference of two numbers: -1 if x < y, 0 if x == y, or 1 if x > y, where x and y are numeric variable/expression.	>>>cmp(80,100)
		-1
		>>>cmp(180, 100)
		1

divmod (x,y)	Returns both quotient and remainder by division through a tuple, when x is divided by y; where x & y are variable/expression.	>>> divmod (14,5) (2,4) >>>divmod (2.7, 1.5) (1.0, 1.20000)

range(start, stop[, step])	This is a versatile function to create lists containing arithmetic progressions. It is most often used in for loops. The arguments must be plain integers. If the step argument is omitted, it defaults to 1. If the start argument is omitted, it defaults to 0. The full form returns a list of plain integers [start,start + step, start + 2 * step, ...]. Step must not be zero (or else Value Error is raised).	>>> range(10)
		[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
		>>> range(1, 11)
		[1,2, 3, 4, 5,6, 7, 8, 9, 10]
		>>>range(0, 30, 5)
		[0, 5, 10,15,20, 25]
		>>> range(0, 10, 3)
		[0,3, 6, 9]
		>>> range(0, -10, -1)
		[0,-1,-2, -3, -4, -5, -6, -7, -8, -9]
		>>> range(0) - []
		>>> range(1, 0) - []

round(x [, n])	It returns float x rounded to n digits from the decimal point, where x and n are numeric expressions. If n is not provided then x is rounded to 0 decimal digits.	>>>round(80.23456, 2) 80.23 >>>round(-100.000056, 3) -100.0 >>> round (80.23456) 80

b) Functions defined in module

A module is a file containing Python definitions (i.e., functions) and statements.

Standard library of Python is extended as module(s) to a programmer. Definitions from the module can be used within the code of a program. To use these modules in the program, a programmer needs to import the module.

How to import a module

There are many ways to import a module in your program, they are:

- ◉ Import
- ◉ From

Import

It is simplest and most common way to use modules in our code. Its syntax is:

import modulename1 [, modulename2,   ]

Example

>>> import math

To use/ access/invoke a function, you will specify the module name and name of the function- separated by dot (.). This format is also known as dot notation.

Example

>>> value= math. sqrt (25) # dot notation

From Statement

It is used to get a specific function in the code instead of the complete module file. If we know beforehand which function(s), we will be needing, then we may use from. For modules having large no. of functions, it is recommended to use from instead of import.

Its syntax is:

>>> from modulename import functionname [, functionname…..]

>>>from modulename import * (Import everything from the file)

Example

>>> from math import sqrt

value = sqrt (25)

Some functions from math module are:

Name of function	Description	Example
ceil(x)	It returns the smallest integer not less than x, where x is a numeric expression.	math.ceil(-45.17) -45.0 math.ceil(100.12) 101.0 math.ceil(100.72) 101.0
floor(x)	It returns the largest integer not greater than x, where x is a numeric expression.	math.floor(-45.17) -46.0 math.floor(100.12) 100.0 math.floor(100.72) 100.0
fabs(x)	It returns the absolute value of x, where x is a numeric value.	math.fabs(-45.17) 45.17 math.fabs(100.12) 100.12 math.fabs(100.72) 100.72

exp(x)	It returns exponential of x: e x, where x is a numeric expression.	math.exp(-45.17)
		2.41500621326e-20
		math.exp(100.12)
		3.03084361407e+43
		math.exp(100.72)
		5.52255713025e+43

pow(x, y)	It returns the value of xy, where x and y are numeric expressions.	math.pow(100, 2) 10000.0 math.pow(100, -2) 0.0001 math.pow(2, 4) 16.0 math.pow(3, 0) 1.0
sqrt (x)	It returns the square root of x for x > 0, where x is a numeric expression.	math.sqrt(100) 10.0 math.sqrt(7)- 2.6457513110645907

Some functions from random module are:

Name of the function	Description	Example

random ( )	It returns a random float x, such that 0 ≤ x	>>random.random( )
		0.281954791393
		>>>random.random( )
		0.309090465205

randint (a, b)	It returns a int x between a & b such that a ≤ x ≤ b	>>> random.randint (1,10)
		5
		>>> random.randint (2,20)
		-1

uniform (a,b)	It returns a floating point number x, such that a <= x < b	>>>random.uniform (5, 10) 5.52615217015
randrange([start,] stop[,step])	It returns a random item from the given range	>>>random.randrange(100,1000, 3) 150

Some functions from statistics module are:

Name of the function	Description	Example
mean(x)	It returns arithmetic mean	>>>statistics.mean([11,24,32,45,51]) 32.6
median(x)	It returns median (middle value) of x	>>>statistics.median([11,24,32,45,51]) 32

mode(x)	It returns mode (the most repeated value)	>>>statistics.mode([11,24,11,45,11])
		11
		>>>statistics.mode(("red","blue","red"))
		red

Some of the other modules, which you can explore, are: string, time, date.

c) User defined functions

a. The keyword def is used to define a function.

b. After the keyword comes an identifier i.e. name of the function, followed by parenthesized list of parameters and the colon which ends up the line.

c. Next follows the block of statement(s) that are the part of function.

Creating User Defined Functions

The def (short for define) keyword is used for creating user defined functions.

Syntax:

```
def function_name([parameter1, parameter2,....]) :
                statement1
                statement2
                ...
                ...
                [return <value1, value2, ...> ]
```

Points about User Defined Functions:

- Every User Defined Function start with the Keyword def.
- The first line of the function is called Function Definition.
- Function name must follow norms of valid identifier.
- Every function header end with parenthesis and colon sign followed by function name
- We can provide parameters inside parenthesis, which is optional.
- The return statement is used to return a value from function. The return statement is also optional. The function should return a value only at the place from where it gets called.
- To use a function, it must be called or invoked in main program.

Type of user defined functions:

The user defined functions can be categorized in following 4 categories:

- Function which accepts arguments and return some value
- Function which accepts arguments but return no value
- Functions which do not accept any argument but return some value
- Functions which do not accept any argument and even not return any value

Parameters/Arguments passed to functions:

These are the values passed to user defined functions from the calling function. If you are passing values of immutable types (i.e., numbers, strings, etc) to the Called function, then the called function cannot alter their values.

- If you are passing values of mutable types (i.e., List, Dictionary, etc.) to the Called function, then the called function can make changes to them.

They are categorized in following two types:

Formal Parameters or Formal Arguments: These are the values provided in Parenthesis when we write a function Header.Actual Parameters or Actual Arguments: These are the values passed to the function when it is called/invoked.

Example:

```
def area_of_circle( radius ): # here radius is the Formal Parameter or Formal Argument
    print(“Area of circle = “,3.1416*radius*radius)
    r=int(input(“Enter radius of circle”))
ar=area_of_circle(r) # here r is the Actual Parameter or Actual Argument output:
```

```
OUTPUT
Enter radius of circle 10
Area of circle = 314.16
```

Types of Arguments:

In Python following four types of arguments allowed:

- Default Arguments
- Positional Arguments
  (As per Syllabus)

(i) Default Parameters

The parameters which are assigned with a value in function header while defining the function are known as default parameters. This value is optional for the parameter. If a user explicitly passes the value in function call, then the value which is passed by the user will be taken by the default parameter. If no value is provided, then the default value will be taken by the parameter. **Default parameters will be written in the end of the function header, means positional parameter cannot appear to the right side of default parameter.**

Example:

```
def SICal(amount, rate, time=10):   #Function definition header
```

(ii) Positional Parameters/ Required Arguments

When the function call statement must match the number and positional order of arguments as defined in the function definition, this is called the positional parameters.

Example:

```
def Test(x,y,z): #function definition header
```

Then we can call function using following statements p,q,r=3,4,6

```
        Test(p,q,r)      #3 variables are passed
        Test(4,q,r)      # 1 literal and 2 variables are passed
        Test(5,6,7)      # 3 literals are passed.
```

So x,y,z are positional parameters and the value must be provided to these parameters

What is the difference between parameter and argument?

Ans. Parameters are temporary variable names within functions. The argument can be thought of as the value that is assigned to that temporary variable. For instance, let's consider the following simple function to calculate sum of two numbers.

```
def sum(a,b):
    return a+b
sum(10,20)
```

Here a, b are the parameters for the function 'sum'.

Arguments are used in procedure calls, i.e., the values passed to the function at run- time.10,20 are the arguments for the function sum

Function returning value(s)

A function can return multiple values. The return values should be a comma-separated list of values. The multiple return values are returned as a tuple. We can unpack the received value by specifying the same number of variables on the left side of the function call.

Example 1:

```
def add10(x,y,z):
   return x+10,y+10,z+10
x=10
y=20
z=30
result=add10(x,y,z)
print(result)
```

Output

```
(20, 30, 40)
```

Example 2:

```
def add10(x,y,z):
   return x+10,y+10,z+10
x=10
y=20
z=30
a,b,c=add10(x,y,z) #unpack
```

Output:

```
20 30 40
```

SCOPE OF VARIABLES:

Scope of a variable is the portion of a program where the variable is recognized and can be accessed therein.

There are two types of scope for variables:

- Local Scope
- Global Scope

1.Local Scope:

A variable declared in a function-body is said to have local scope. It cannot be accessed outside the function. In this scope, the lifetime of variables inside a function is as long as the function executes.

2. Global Scope:

A variable declared in top level segment (main) of a program is said to have a global scope. In this scope, lifetime of a variable inside a program is as long as the program executes.

Example 1:

```
def Sum(x , y) :
    z = x + y
    return z
a = 5
b = 7
s = Sum(a , b)
print(s)
```

In the above program x, y and z are local variables and are destroyed once we return from the function. a, b and s are global variables and remains in the memory until the program executes.

Example 2:

```
def my_func( ):
  x=10      # local variable print("Value inside function:",x)
x=20        # global variable my_func( )
print("Value outside function:",x)
```

Output:

```
Value inside function:10
Value outside function:20
```

Here, we can see that the value of x is 20 initially. Even though the function my_func() changed the value of x to 10, it did not affect the value outside the function. This is because the variable x inside the function is different (local to the function) from the one outside. Although they have same names, they are two different variables with different scope. On the other hand, variables outside of the function are visible from inside. They have a global scope.

Practice Questions (1 MARK)

1	Which of the following is a valid function name? a) Start_game() b) start game() c) start-game() d) All of the above
2	If the return statement is not used in the function then which type of value will be returned by the function? a)int b) str c) float d) None
3	Richa is working with a program where she gave some values to the function. She doesn't know the term to relate these values. Help her by selecting the correct option. a) function value b) arguments or parameters c) return values d) function call

4	What is the minimum and maximum value of c in the following code snippet?
	import random
	a=random.randint(3,5)
	b = random.randint(2,3)
	c = a + b
	print(c)
	a) 3 , 5 b) 5, 8 c) 2, 3 d) 3, 3

5	In python function, the function calling another function is known as__________and the function being called is known __________________________________ a) main, keyword b) caller, called c) called, caller d) executer, execute
6	What will be the output of the following code? print("100+200") a) 300 b) 100200 c) 100+200 d) 200
7	pow( ) function belongs to which library ? a) math b) string c) random d) maths
8	What data type is the object below? L = (1, 23, 'hello',1) a) list b) dictionary c) array d) tuple
9	What is returned by int(math.pow(3, 2))? a) 6 b) 9 c) error, third argument required d) error, too many arguments
10	Which of the following is not a type conversion functions? a) int() b) str() c) input() d) float()

11 Write the output of the following:

print(float())

12 Identify the module to which the following function load () belong to?

a) math b) random c) pickle d) sys

13 How many argument(s) a function can receive

a) Only one b) 0 or many c) Only more than one d) At least one

14 Give the output

```
def fun():
        global a
        a=10
        print(a)
a=5
fun()
print(a)
```

```
a)  10          b) 5       c) 5            d) 10
    10             10         5                5
```

15 Value returning functions should be generally called from inside of an expression

a) True b) False

16 The variable declared inside the function is called a________________variable

a) global b) local c) external d) none of the above

17 These are predefined functions that are always available for use. For using them we don’t need to import any module

a) built in function b) pre-defined function
c) user defined function d) none of the above

18 The____________of a variable is the area of the program where it may be referenced

a) external b) global c) scope d) local

19 If you want to communicate between functions i.e. calling and called statement, then you should use _______

a) values b) return c) arguments d) none of the above

20 Which of the following function header is correct?

a) def mul(a=2, b=5,c) b) def mul(a=2, b, c=5)
c) def mul(a, b=2, c=5) d) def mul(a=2, b, c=5)

21 Find the flow of execution of the following code:

1. def calculate (a, b):

	2. res=a**b 3. return res 4. 5. def study(a): 6. ans=calculate(a,b) 7. return ans 8. 9. n=2 10. a=study(n) 11. print(a) a) 1 > 5 > 9 > 10 >6 > 2 > 3 > 7 > 11 b) 5 > 9 > 10 > 6 > 2 > 3 > 7 > 11 c) 9 > 10 > 5 > 1 > 6 > 2 > 3 > 7 > 11 d) None of the above
22.	Python resolves the scope of a name using the LEGB rule a) True b) False
23	A void function internally returns legal empty value ______________ a) None b) Close() c) Return d) all
24	When you use multiple type argument in function, then default argument take place ___________ a) at beginning b) at end c) anywhere d) none of the above
25	A___________can be skipped in the function call statements a) named parameter b) default parameter c) keyword parameters d) all of the above

	ANSWERS
1	a)Start_game()
2	d) None
3	b) arguments or parameters
4	b) 5, 8
5	b) caller, called
6	c) 100+200
7	a) math
8	d) tuple
9	b) 9
10	c) input()
11	b) 0.0
12	c)pickle
13	b) 0 or many
14	a) 10 10
15	a) True
16	b) local
17	a) built in function
18	c) scope
19	c) arguments
20	c) def mul(a, b=2, c=5)
21	a) 1 > 5 > 9 > 10 >6 > 2 > 3 > 7 > 11
22	a) True
23	a) None
24	b) at end
25	b) default parameter

Practice Questions (2 MARKS)

1. What is the output of the following code?

```
def cube(x):
    return x * x * x
x = cube(3)
print( x)
```

a) 9 b)3 c)27 d) 30

2 Which of the following items are present in the function header?

a) function name
b) parameter list
c) return value
d) Both A and B

3 Choose correct answer

```
def fun1(num):
    return num+5
print(fun1(5))
print(num)
```

a) Print value 10 b) Print value 5 c) Name Error d) 25

4. Predict the output of the following code

```
def func1(list1):
   for x in list1:
       print(x.lower(),end="#")
func1(["New","Dehli"])
```

A. [New,Dehli]
B. new#dehli#
C. newdehli#
D. New#Dehli#

5. What will be the output of the following python code?

```
def mul (num1, num2):
     x = num1 * num2
x = mul(20,30)
```

A. 600 B. None C. No Output D. 0

6 Which of the following function header is Correct:

A. def fun(x=1,y)
B. def fun(x=1,y,z=2)
C. def fun(x=1,y=1,z=2)
D. def fun(x=1,y=1,z=2,w)

7 What is the output of the program given below?

```
x = 50
def func (x) :
        x = 2
func (x)
print ('x is now', x)
```

A) x is now 50   B) x is now 2 C) x is now 100   D) Error

8 Choose the correct output from the options given below.

```
print('Welcome!')
print('Iam',__name__) #    is double underscore
```

```
a) Welcome!                          b) Error
   Iam __main__
c) Welcome!                          d)None of these
   Iam __name__
```

9 Predict the output of the following code fragment

```
def update(x=10):
    x+=15
    print("x=",x)
x=20
update()
print("x=",x)
```

```
a)  x=20                      b) x=25
    x=25                         x=25
c) x=20                       d) x=25
  x=25                           x=20
```

10 Predict the output of the following code fragment

```
def display(x=2,y=3):
      x=x+y
      y+=2
      print(x,y)
display( )
display(5,1)
display(9)
```

```
a)  5 5                  b)12 5
    6 3                     6 3
    12 5                    5 5

c) 5 6                   d) 5    5
   12 5                     7    7
```

6 3   6   6

11 Find the output print(pow(5,4,9))

a)   7   b)0   c)4   d) error

12 Give the output of the following program

```
def check(a):
        for i in range(len(a)):
                a[i]=a[i]+5
                return a
b=[1,2,3,4]
c=check(b)
print(c)
```

a) [6, 8, 8, 9]   b) [6,7,8,9]

c) [7, 7, 8, 9]   d) [6,7,9,9]

13 Give the output

```
def abc(x,y=60):
    return x+y
a=20
b=30
a=abc(a,b)
print(a,b)
b=abc(a)
print(a,b)
a=abc(b)
print(a,b)
```

a) 50   30 50 110   170   b) 3 5   110 170 110
                                     0 0

c) 50 30   d) 5 3   50 110 170 110
   50 110   0 0

   170 110

14

Predict the output of the following code snippet:

```
def Execute(M):
    if M%3==0:
        return M*3
    else:
        return M+10;
def Output(B=2):
    for T in range (0,B):
        print(Execute(T),"*",end="")
    print()
Output(4)
Output()
Output(3)
```

15.

Find the output of the following program:

```
def ChangeIt(Text,C):
    T=""
    for K in range(len(Text)):
        if Text[K]>='F' and Text[K]<='L':
            T=T+Text[K].lower();
        elif Text[K]=='E' or Text[K]=='e':
            T=T+C;
        elif K%2==0:
            T=T+Text[K].upper()
        else:
            T=T+T[K-1]
    print(T)
OldText="pOwERALone"
ChangeIt(OldText,"%")
```

(a) PPW%RRLLN%   (b) PPWCRRllNC

(c) PPW%RRllN%   (d) PpW%RrllN%

16

What possible outputs are expected to be displayed on screen at the time of execution of the program from the following code? Also specify the maximum value that can be assigned to each of the variables L and U.

```
import random

Arr=[10,30,40,50,70,90,100]

L=random.randrange(1,3)

U=random.randrange(3,6)

for i in range(L,U+1):

    print(Arr[i],"@",end="")
```

i) 40 @50 @ ii) 10 @50 @70 @90 @

iii) 40 @50 @70 @90 @ iv) 40 @100 @

17

Find the output of the following code

```
def disp(str):
   m=' '
   for i in range(0,len(str)):
      if(str[i].isupper()):
         m=m+str[i].lower()
      elif str[i].islower():
         m=m+str[i].upper()
      else:
         if i%2==0:
            m=m+str[i-1]
         else:
            m=m+"@"
   print(m.swapcase())
disp('StudyBag$2021')
```

a) StudyBagG@2@2
b) sTUDYbAGg@2@2
c) StudyBagG$2$2
d) None

18

What will be the output of the following code

```
 total=0
def add(a,b):
        global total
        total=a+b
        print(total)
add(6,6)
print(total)
```

a) 12 b) 12c) 0 d) None of these

19 Find and write the output of the following Python code:

```
def makenew(mystr):
    newstr = " "
    count = 0
    for i in mystr:
        if count%2 ==0:
            newstr = newstr+i.lower()
        else:
            if i.islower():
                newstr = newstr+i.upper()
            else:
                newstr = newstr+i
        count +=1
    newstr = newstr+mystr[:3]
    print ("The new string is :", newstr)
makenew("cbseEXAMs@2022")
```

20 What possible output(s) are expected to be displayed on screen at the time of execution of the following code? Also specify the maximum and minimum value that can be assigned to variable X.

```
import random
L=[10,7,21]
X=random.randint(1,2)
for i in range(X):
    Y=random.randint(1,X)
print(L[Y],"$",end=" ")
```

(i)10 $ 7 $ (ii) 21 $ 7 $ (iii) 21 $ 10 $ (iv) 7 $

21 Choose the correct option:

Statement1: Local Variables are accessible only within a function or block in which it is declared.

Statement2: Global variables are accessible in the whole program.

a. Statement1 is correct but Statement2 is incorrect
b. Statement2 is correct but Statement1 is incorrect
c. Both Statements are Correct
d. Both Statements are incorrect

22 Consider the following code and choose correct answer:

```
def nameage(name, age):
    return [age,name]
t=nameage('kishan',20)

  print(type(t))
```

a) tuple   b) list   c) (kishan,20)   d) None of all

23 Write the output of the following:

```
a=(10, 12, 13, 12, 13, 14, 15)

print(max(a) + min(a) + a.count(2))
```

a) 13 b) 25   c) 26   d) Error

24 Consider the code given below and Identify how many times the message “**Hello All**” will be printed.

```
def prog(name):
    for x in name:
        if x.isalpha():
            print('Alphabet')
        elif x.isdigit():
            print('Digit')
        elif x.isupper():
            print('Capital Letter')
        else:
            print('Hello All')
prog('vishal123@gmail.com')
```

a)   0   b) 2   c) 1   d) 3

25 Find and write the output of the following Python code:

```
def changer(p,q=10):
    p=p/q
    q=p%q
    print(p,"#",q)
    return p
a=200
b=20
a=changer(a,b)
print(a,"$",b)
a=changer(a)
print(a,"$",b)
```

26 What will be the output for the below code snippet?

```
def div(lst,n):
    for i in range(0,n):
        if lst[i]%5==0:
            lst[i]+=5
        else:
            lst[i]=lst[i]//2
lt=[45,20,23,54,5]
div(lt, len(lt))
for i in lt:
    print(i,end='#')
```

a) 50#25#11.5#27.0#10#   b) 50#25#11#27#10#

c) 50#25#1#0#10#   d) 225#100#1#0#25#

ANSWERS	
1.	Ans. C) 27
2	Ans.d Both A and B
3	Ans c)Name Error
4.	Ans. B)new#dehli#
5.	Ans. C)No Output
6	Ans.c) def fun(x=1,y=1,z=2)
7	Ans. A) A) x is now 50
8	Ans. a)Welcome! Iam __main__

9	Ans. d)
	x=25
	x=20

10	Ans. a) 5 5 6 3 12 5
11	Ans. C) 4 Hint: pow(x, y, z) is equal to x^y % z
12	Ans. b) [6, 7, 8, 9]
13	Ans.C) c) 50 30 50 110 170 110
14	Ans. 0 *11 *12 *9 * 0 *11 * 0 *11 *12 *

15.	Ans: c) PPW%RRllN%
16	Ans. Options i and iii i)40 @50 @ iii) 40 @50 @70 @90 @ Maximum value of L and U L=2 ,U=5
17	Ans : a) StudyBagG@2@2 Note: The swapcase() method returns a string where all the upper case letters are lower case and vice versa. Syntax. string.swapcase().
18	Ans : a) 12 12
19	Ans. The new string is : cBsEeXaMs@2022cbs
20	Ans. iv) 7 $ Maximum value of x is 2 Minimum value of x is 1
21	Ans.c) Both Statements are correct
22	Ans : b)
23	Ans : b) 25
24	Ans: b) 2
25	Ans. 10.0 # 10.0 10.0 $ 20 1.0 # 1.0 1.0 $ 20
26	Ans: b) 50#25#11#27#10#

ASSERTION AND REASONING QUESTIONS(1 mark) (a) Both A and R are true and R is the correct explanation for A (b)Both A and R are true and R is not the correct explanation for A (c) A is True but R is False (d)A is false but R is True	
1	Assertion(A): Python Standard Library consists of various modules. Reasoning(R): A function in a module is used to simplify the code and avoids repetition.
2	Assertion(A): If the arguments in a function call match the number and order of argumets as defined in the function definition,such arguments are called the positional arguments. Reasoning(R): During a function call,the argument list first contains default arguments followed by positional arguments.
3	Assertion(A): The random module is a built-in module to generate the pseudo-random variables. Reason(R): The randrange() function is used to generate a random number between the specified range in its parameter.
4	Assertion (A): Global variable is declared outside the all the functions. Reasoning (R): It is accessible through out all the functions.
5	Assertion (A): The math.pow(2,4)gives the output: 16.0 Reason (R): The math.pow() method receives two float arguments, raise the first to the second and return the result.
6	Assertion (A): A variable declared as global inside a function is visible with changes made to it outside the function. Reasoning (A): All variables declared outside are not visible inside a function till they are redeclared with global keyword.
7	Assertion (A): Built-in function are predefined in the language that are used directly. Reason (R): print() and input() are built-in functions
8	Assertion (A):- In Python, statement return [expression] exits a function. Reasoning (R):- Return statement passes back an expression to the caller. A return statement with no arguments is the same as return None.
9	Assertion (A): When passing a mutable sequence as an argument, function modifies the original copy of the sequence. Reasoning (R): Function can alter mutable sequences passes to it.
10	Assertion (A) : A function is a block of organized and reusable code that is used to perform a single related action. Reason (R) : Function provides better modularity for your application and a high degree of code reusability.

ANSWERS	
1	(b)Both A and R are true and R is not the correct explanation for A
2	(c) A is True but R is False
3	(b)Both A and R are true and R is not the correct explanation for A
4	(a) Both A and R are true and R is the correct explanation for A
5	(b)Both A and R are true and R is not the correct explanation for A
6	(a) Both A and R are true and R is the correct explanation for A
7	(a) Both A and R are true and R is the correct explanation for A
8	(a) Both A and R are true and R is the correct explanation for A
9	(b)Both A and R are true and R is not the correct explanation for A
10	(a) Both A and R are true and R is the correct explanation for A

Additional Practice Questions (2/3 MARKS)	
1	Python function oddeven(L) to print positive numbers in a list L. Example: Input: [4, -1, 5, 9, -6, 2, -9, 8] Output: [4, 5, 9, 2, 8]
2	Write a function listchange(Arr,n)in Python, which accepts a list Arr of numbers and n is an numeric value depicting length of the list. Modify the list so that all even numbers doubled and odd number multiply by 3 Sample Input Data of the list: Arr= [10,20,30,40,12,11], n=6 Output: Arr = [20,40,60,80,24,33]
3	Write definition of a method/function AddOddEven(VALUES) to display sum of odd and even values separately from the list of VALUES. For example : If the VALUES contain [15, 26, 37, 10, 22, 13] The function should display Even Sum: 58 Odd Sum: 65
4	Write a function in Shift(Lst), Which accept a List 'Lst' as argument and swaps the elements of every even location with its odd location and store in different list eg. if the array initially contains 2, 4, 1, 6, 5, 7, 9, 2, 3, 10 then it should contain 4, 2, 6, 1, 7, 5, 2, 9, 10, 3
5	Define a function ZeroEnding(SCORES) to add all those values in the list of SCORES, which are ending with zero (0) and display the sum. For example : If the SCORES contain [200, 456, 300, 100, 234, 678] The sum should be displayed as 600

6	Write a function countNow(PLACES) in Python, that takes the dictionary, PLACES as an argument and displays the names (in uppercase)of the places whose names are longer than 4 characters. For example, Consider the following dictionary PLACES={1:"Delhi",2:"London",3:"Paris",4:"New York",5:"Doha"} The output should be: LONDON NEW YORK
7	Write a function, lenWords(STRING), that takes a string as an argument and returns a tuple containing length of each word of a string. For example, if the string is "Come let us have some fun", the tuple will have (4, 3, 2, 4, 4, 3)
8	Write the Python statement for each of the following tasks using BUILT-IN functions/methods only: (i) To insert an element 200 at the third position, in the list L1. (ii) To check whether a string named, message ends with a full stop / period or not
9	A list named studentAge stores age of students of a class. Write the Python command to import the required module and (using built-in function) to display the most common age value from the given list.
10	Write a function LISTELE(L), where L is the list of elements passed as argument to the function. The function returns another list named 'indexList' that stores the indices of all Non-Zero Elements of L. For example: If L contains [12,4,0,11,0,56] The indexList will have - [0,1,3,5]

EXCEPTION HANDLING :

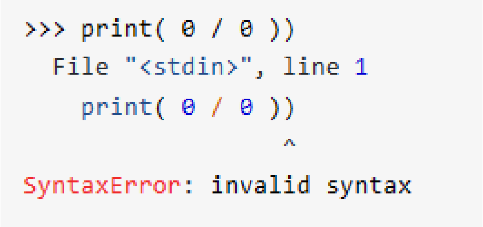
Introduction, Handling exceptions using try – except-finally blocks

Introduction:

A Python program terminates as soon as it encounters an error. In Python, an error can be a syntax error or an exception. However exception differs from a syntax error. Here we will see how to raising exceptions and making assertions and discuss how try and except block work along with finally block.

Exceptions versus Syntax Errors

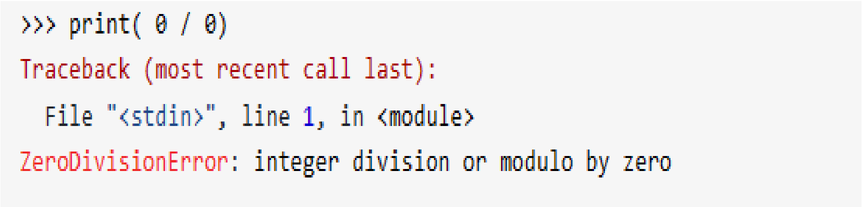
Syntax errors or parsing errors are detected when we have not followed the rules of the particular programming language while writing program. When syntax error encountered, python displays the name of the error and a small description about the error. The execution of the program will start only after the syntax error is rectified.
Eg:



```
>>> print( 0 / 0 ))
  File "<stdin>", line 1
    print( 0 / 0 ))
                  ^
SyntaxError: invalid syntax
```

An exception is a python object that represents an error. Syntax errors are also handled as exceptions. The exception needs to be handled by the programmer so that the program does not terminate abruptly. When an exception occurs during execution of a program and there is a built -in exception defined for that, the error message written in that exception is displayed. The programmer then has to take appropriate action and handle it.

Eg:

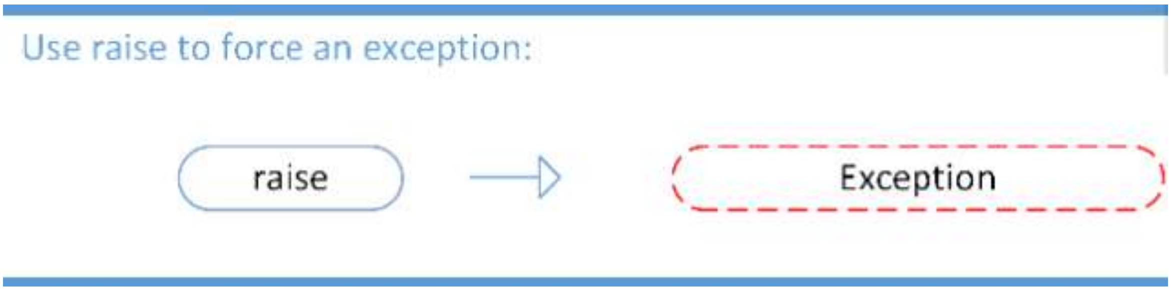


```
>>> print( 0 / 0)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ZeroDivisionError: integer division or modulo by zero
```

This type of error occurs whenever syntactically correct Python code results in an error. The last line of the message indicated what type of exception error you ran into. Python details what type of exception error was encountered. In this case, it was a ZeroDivisionError.

Raising Exception:

The raise statement is used to throw an exception.



Use raise to force an exception:

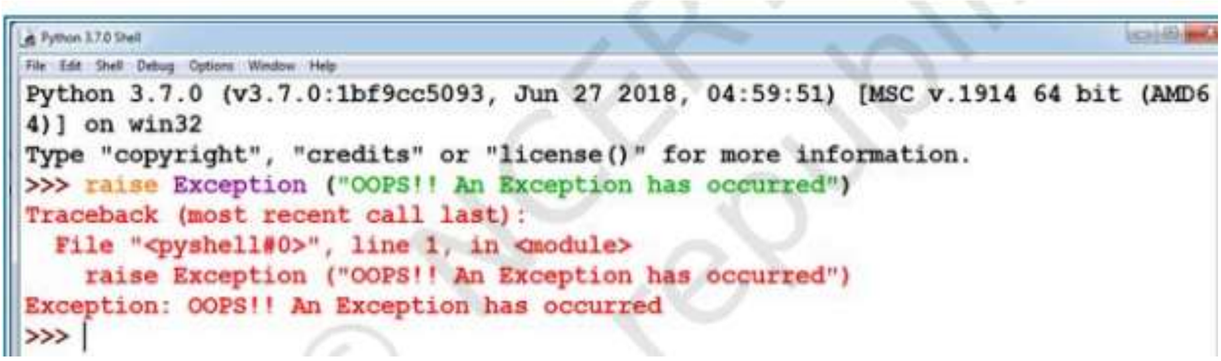
The Syntax of raise statement is :

Raise exception-name [(optional argument)]

The argument is generally a string that is displayed when the exception is raised.

Eg1:

when an exception is raised as shown below, the message “OOPS : An Exception has occurred” is displayed along with a brief description of the error.



```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD6
4)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> raise Exception ("OOPS!! An Exception has occurred")
Traceback (most recent call last):
  File "<pyshell#0>", line 1, in <module>
    raise Exception ("OOPS!! An Exception has occurred")
Exception: OOPS!! An Exception has occurred
>>> |
```

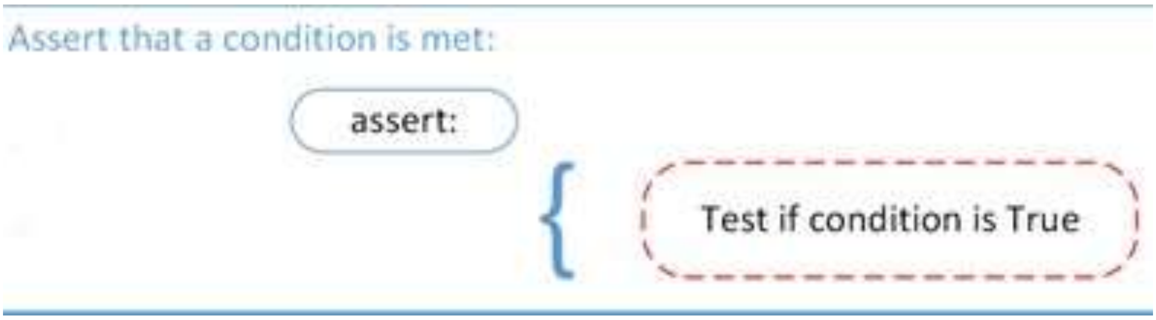
Eg2:

Code to accept two numbers and print quotient :

```
n = int(input("Enter Number 1 :"))
m = int(input("Enter Number 2 : "))
if m == 0:
    raise ZeroDivisionError
else:
    print("Quotient : ", n / m)
```

The assert statement:

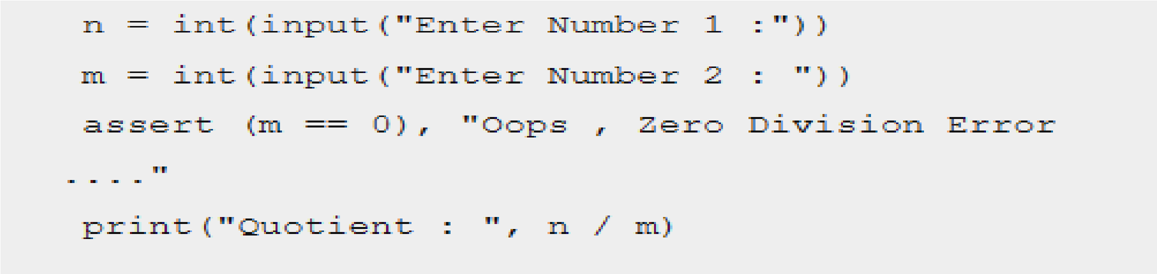
An assert statement in python is used to test an expression in the program code. If result after testing comes false, then the exception is raise. This statement is generally a function call to check for valid input.



The syntax for assert statement is :

assert Expression[,arguments]

On encountering an assert statement, python evaluates the expression given immediately after the assert keyword. If this expression is false. An AssertionError exception is raised which can be handled like any other exception.



```
    n = int(input("Enter Number 1 :"))
    m = int(input("Enter Number 2 : "))
    assert (m == 0), "Oops , Zero Division Error
   ...."
    print("Quotient : ", n / m)
```

Output-1:

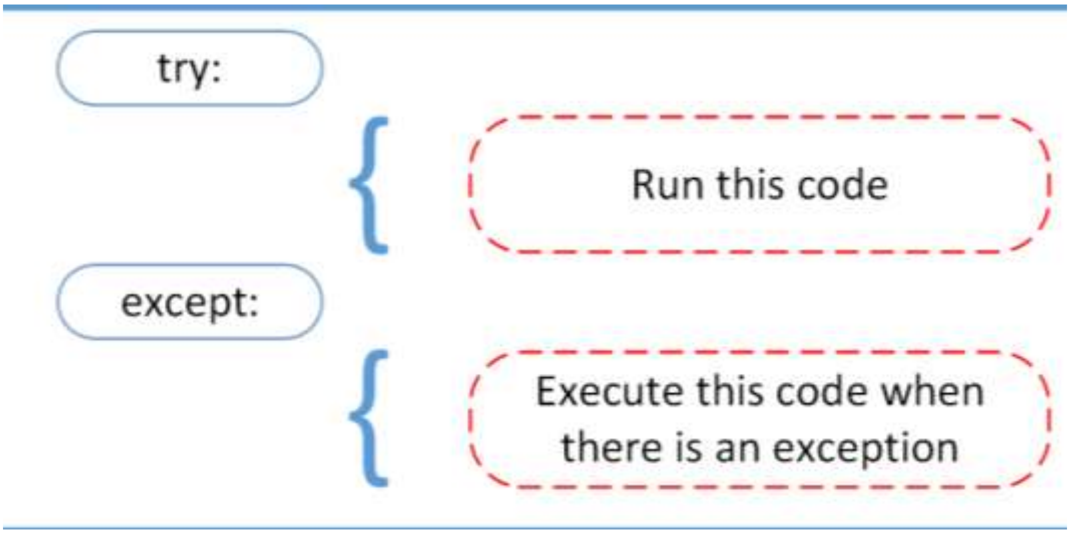
```
Enter Number 1:10

Enter Number 2 : 2

Traceback(most recent call last):

File"D://PythonProg/ExceptionHandling/DivideByZero.py",line 4, in assert
(m==0),"Opps, ZeroDivisionError"

AssertionError:Opps, ZeroDivisionError
```

Output-2:

```
Enter Number 1:10
Enter Number 2 : 0
Traceback(most recent call last):
File"D://PythonProg/ExceptionHandling/DivideByZero.py",line 5, in
print("Quotient:",n/m)
ZeroDivisionError : division by zero
```

Try and Except Block:

An exception is caught in the try block and handles in except block. Exception handling is being used not only in Python programming but in most programming languages like C++, Java etc. It is a useful technique that helps in capturing runtime errors and handling them so as to avoid the program getting crashed. While writing or debugging a program a user might doubt an exception to occur in a particular part of the code. Such suspicious lines of code are put inside a try block . Every try block is followed by a exception block, the appropriate code to handle each of the possible exceptions (in the code inside try block) are written the except clause.

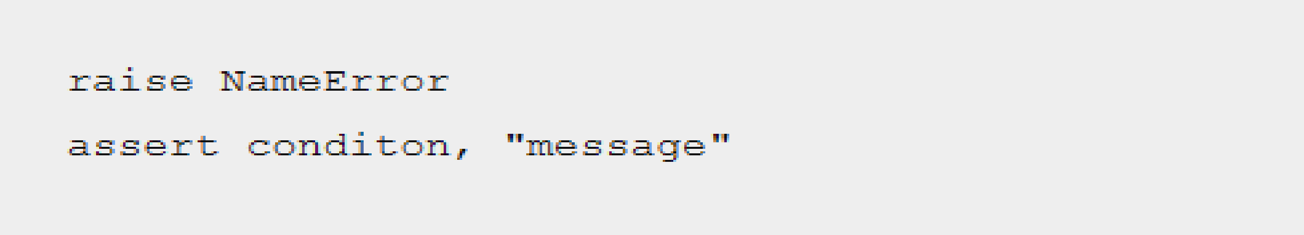


- **Exception Handling** : Writing additional code in a program to give proper messages or instructions to the user encountering an exception called exception handling.

```
try:
    statements
except Exception_Name:
    statements for handling exception
```

- **Throwing an exception** :

Throwing an exception means raising an exception. Each time an error detected in a program, the python interpreter raises (throws) an exception . The exception handlers are designed to execute when a specific exception is raised. Programmers can also forcefully raise exception in program, using the raise and assert statement. Once an exception is raised no further statements in the current block of code is executed

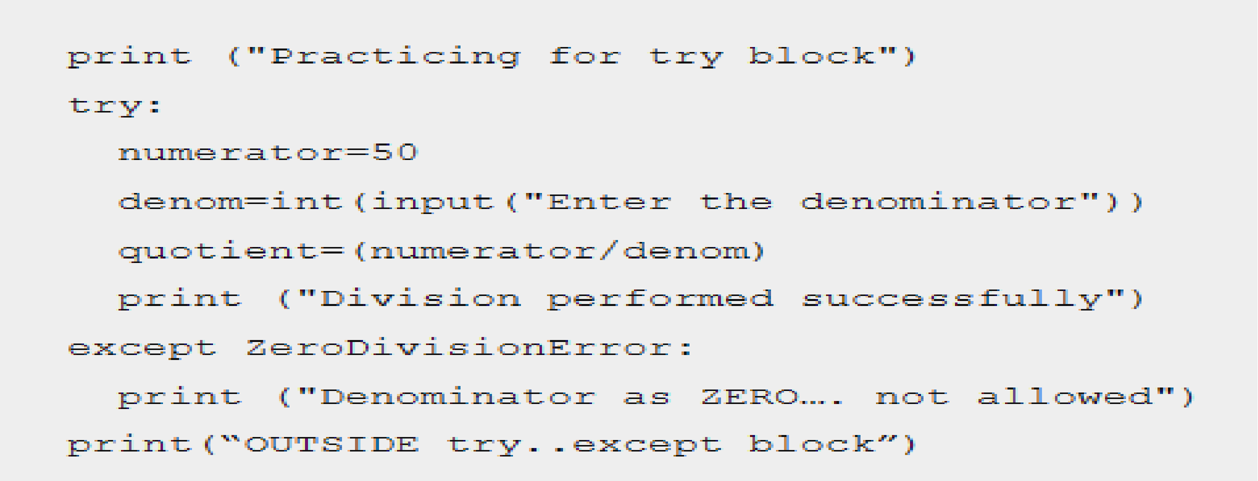


```
raise NameError

assert conditon, "message"
```

- **Catching an exception**: catching an exception means handling an exception by an exception handlers. An exception is said to be caught when a code designed to handle an particular exception is executed . Exception if any are caught in try block are handled in exception block.

.

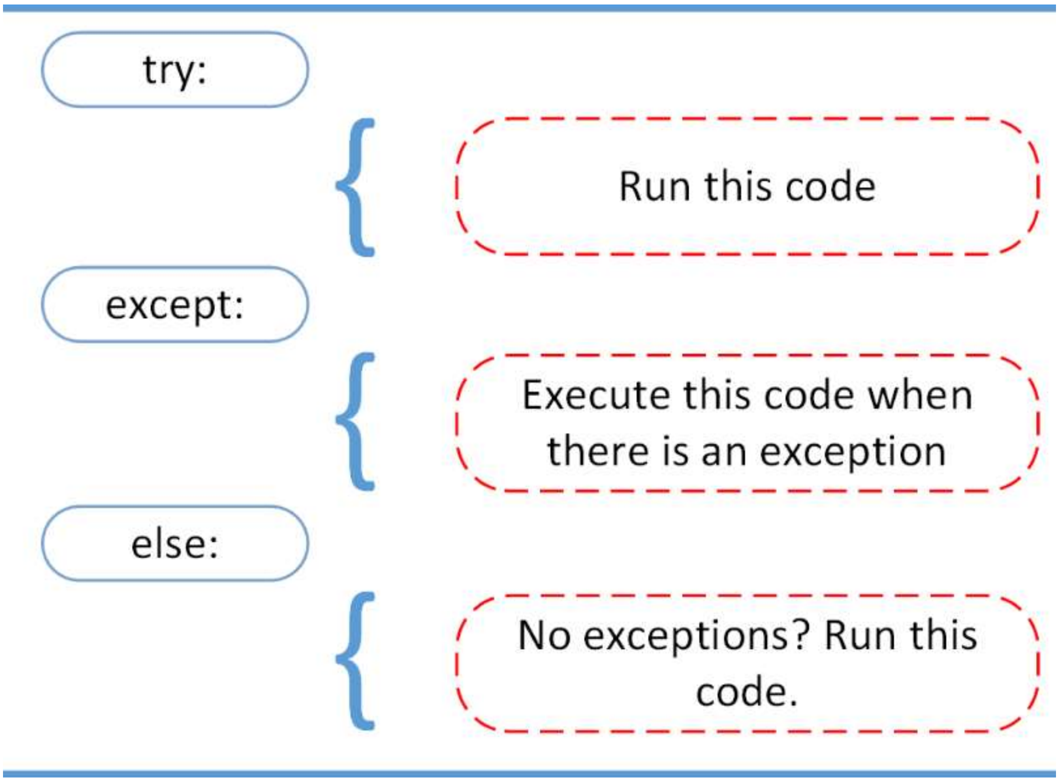
While executing the program if an exception is encountered further execution of the code inside the try block is stopped and the control is transferred to the except block.



```
print ("Practicing for try block")
try:
   numerator=50
   denom=int(input("Enter the denominator"))
   quotient=(numerator/denom)
   print ("Division performed successfully")
except ZeroDivisionError:
   print ("Denominator as ZERO…. not allowed")
print(“OUTSIDE try..except block”)
```

- **try...except…else clause:**

We can put an optional else clause along with the try...except clause. An except block will be executed only if some exception is raised in the try block. But if there is no error then none of the except blocks will be executed. In this case, the statements inside the else clause will be executed.

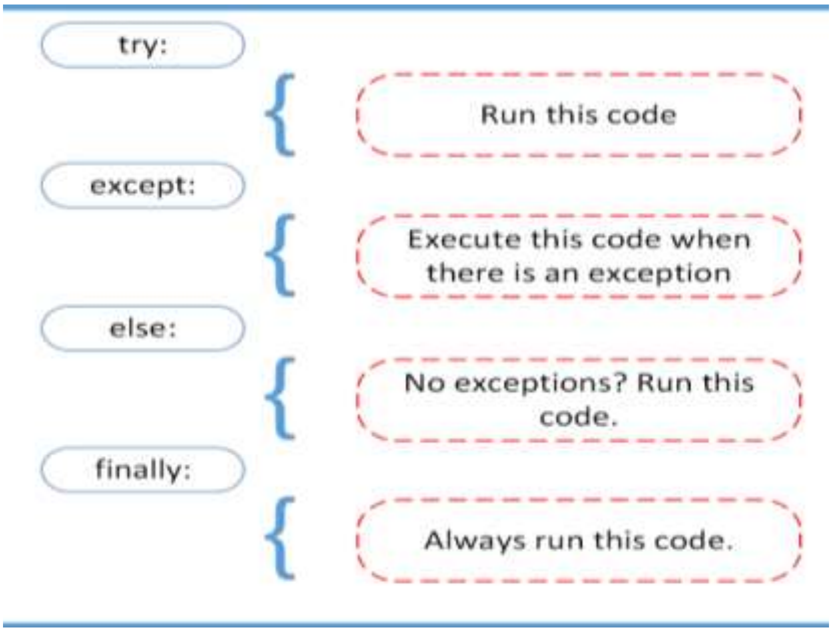

try:
Run this code
except:
Execute this code when there is an exception
else:
No exceptions? Run this code.

<u>Eg:</u>

```
try:
        numerator=50
        denom=int(input("Enter the denominator: "))
        quotient=(numerator/denom)
         print ("Division performed successfully")
except ZeroDivisionError:
        print ("Denominator as ZERO is not allowed")
except ValueError:
        print ("Only INTEGERS should be entered")
else:
        print ("The result of division operation is ", quotient)
```

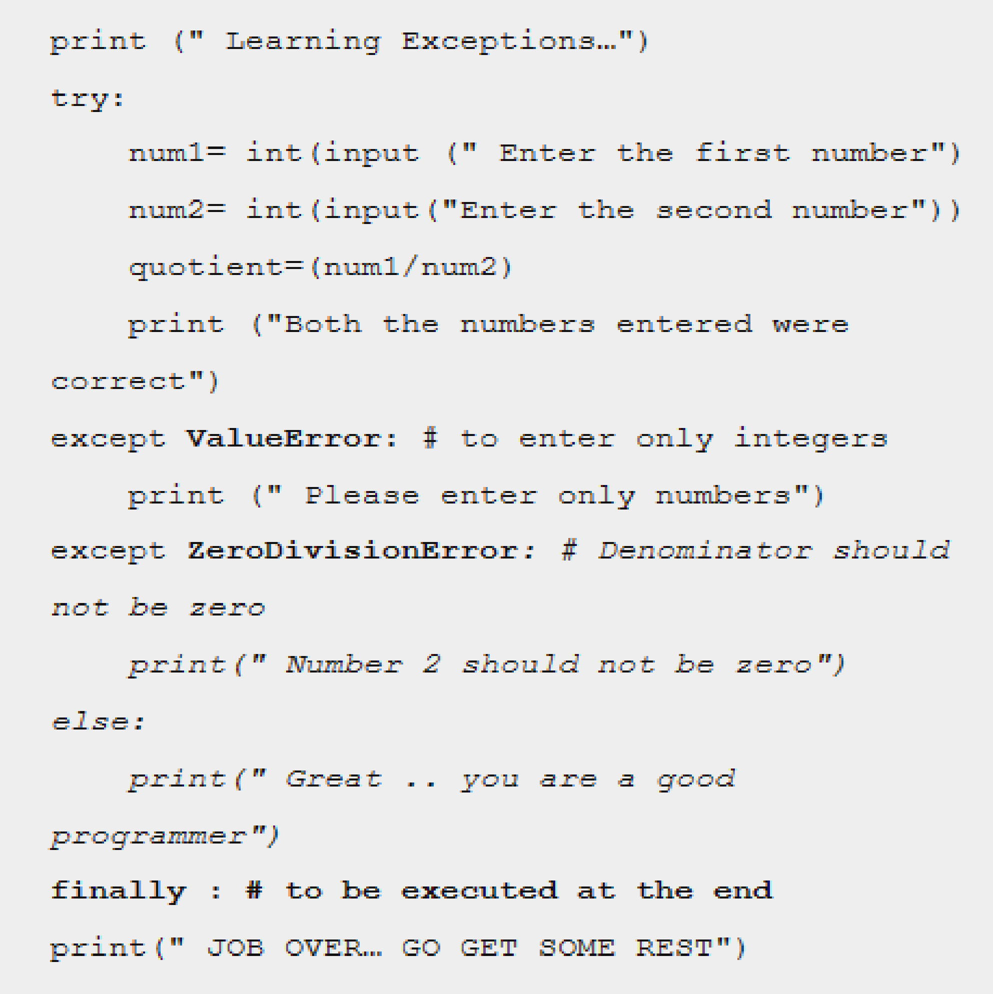
<u>Finally clause:</u>

The try statement in Python can also have an optional finally clause. The statements inside the finally block are always executed regardless of whether an exception has occurred in the try block or not.


try:
Run this code
except:
Execute this code when there is an exception
else:
No exceptions? Run this code.
finally:
Always run this code.

It is a common practice to use finally clause while working with files to ensure that the file object is closed. If used, finally should always be placed at the end of try clause, after all except blocks and the else block.

Eg:



```
print (" Learning Exceptions…")
try:
    num1= int(input (" Enter the first number")
    num2= int(input("Enter the second number"))
    quotient=(num1/num2)
    print ("Both the numbers entered were
correct")
except ValueError: # to enter only integers
    print (" Please enter only numbers")
except ZeroDivisionError: # Denominator should
not be zero
    print(" Number 2 should not be zero")
else:
    print(" Great .. you are a good
programmer")
finally : # to be executed at the end
print(" JOB OVER… GO GET SOME REST")
```

Some of the commonly occurring built-in exceptions are :

S. No	Name of the Built-in Exception	Explanation
1	SyntaxError	It is raised when there is an error in the syntax of the Python code.
2	ValueError	It is raised when a built-in method or operation receives an argument that has the right data type but mismatched or inappropriate values.
3	IOError	It is raised when the file specified in a program statement cannot be opened.
4	KeyboardInterrupt	It is raised when the user accidentally hits the Delete or Esc key while executing a program due to which the normal flow of the program is interrupted.
5	ImportError	It is raised when the requested module definition is not found.
6	EOFError	It is raised when the end of file condition is reached without reading any data by input().
7	ZeroDivisionError	It is raised when the denominator in a division operation is zero.
8	IndexError	It is raised when the index or subscript in a sequence is out of range.
9	NameError	It is raised when a local or global variable name is not defined.
10	IndentationError	It is raised due to incorrect indentation in the program code.
11	TypeError	It is raised when an operator is supplied with a value of incorrect data type.
12	OverFlowError	It is raised when the result of a calculation exceeds the maximum limit for numeric data type.
13	Key-Error	Python raises a KeyError whenever a dict() object is requested (using the format a= adict[key]) and the key is not in the dictionary.

MCQ Question based on Exception Handlihng

1. Which block lets you test a block of code for errors?

A. try
B. except
C. finally
D. None of the above

Ans : A
Explanation: The try block lets you test a block of code for errors.

2. What will be output for the folllowing code?

```
try:
  print(x)
except:
  print(""An exception occurred"")
```

A. x

B. An exception occurred

C. Error

D. None of the above

Ans : B

Explanation: An exception occurred be the output for the followinng code because the try block will generate an error, because x is not defined.

3. What will be output for the folllowing code?

```
x = ""hello""
if not type(x) is int:
  raise TypeError(""Only integers are allowed"")
```

A. hello

B. garbage value

C. Only integers are allowed

D. Error

Ans : C

Explanation: TypeError: Only integers are allowed

4. How many except statements can a try-except block have?

A. 0

B. 1

C. more than one

D. more than zero

Ans : D

Explanation: There has to be at least one except statement.

5. Can one block of except statements handle multiple exception?

A. yes, like except TypeError, SyntaxError [,…]

B. yes, like except [TypeError, SyntaxError]

C. No

D. None of the above

Ans : A

Explanation: Each type of exception can be specified directly. There is no need to put it in a list.

6. What will be output for the folllowing code?

```
x=10
y=8
assert x>y, 'X too small’
```

A. Assertion Error
B. 10 8
C. No output
D. 108

Ans : C

Explanation: The code shown above results in an error if and only if xy, there is no error. Since there is no print statement, hence there is no output.

7. What will be the output of the following Python code?

```
lst = [1, 2, 3]
lst[3]
```

a) NameError
b) ValueError
c) IndexError
d) TypeError

Answer: c

Explanation: The snippet of code shown above throws an index error. This is because the index of the list given in the code, that is, 3 is out of range. The maximum index of this list is 2.

8. What will be the output of the following Python code?

```
t[5]
```

a) IndexError
b) NameError
c) TypeError
d) ValeError

Answer: b

Explanation: The expression shown above results in a name error. This is because the name ‘t’ is not defined.

9. What will be the output of the following Python code, if the time module has already been imported?

```
4 + '3'
```

a) NameError
b) IndexError
c) ValueError
d) TypeError

Answer: d

Explanation: The line of code shown above will result in a type error. This is because the operand '+' is not supported when we combine the data types 'int' and 'str'. Sine this is exactly what we have done in the code shown above, a type error is thrown.

10. What will be the output of the following Python code?

```
int('65.43')
```

a) ImportError
b) ValueError
c) TypeError
d) NameError

Answer: b

Explanation: The snippet of code shown above results in a value error. This is because there is an invalid literal for int() with base 10: '65.43'.

DATA FILE HANDLING

File- A file is a sequence of bytes on the disk/permanent storage where a group of related data is stored. File handling in Python enables us to create, update, read, and delete the files stored on the file system through our python program.

Data File handling takes place in the following order.

1- Opening a file.
2- Performing operations (read, write) or processing data.
3- Closing the file.

Types of File in Python:
Python allows us to create and manage three types of data files.

1- **Text file**
2- **Binary file**
3- **CSV file**

Text file: A text file is simply a sequence of ASCII or Unicode characters. A line is a sequence of characters, stored on permanent storage. In a text file, each line is terminated by a special character, known as End of Line (EOL). Text file can be created using any text editor. Ex. Myfile.txt.

Binary file: A binary file stores the data in the same way as stored in the memory. The .exe files, mp3 file, image files, word documents are some of the examples of binary files. We can't read a binary file using a text editor.

CSV file: CSV (Comma Separated Values) is a simple text file format for storing data in tabular form such as spread sheet or database. The data is organized with one record on each line and each field is separated by comma.

Compare text files, binary files and csv files

Sl.no	Text file	Binary file	CSV file
1	It is capable to handle textual data.	It is capable to handle large file.	It is very common format and platform independent.
2	It consists of series of lines of a set of letters, numbers or symbols (String)	It consists of data with a specific pattern without any delimiter.	It consists of plain text with a list of data with a delimiter.

3	Any text editors like notepad can be used to read them.	No specific programs can be used to read them, python provides functions to read data.	It can be read using text editors like notepads and spreadsheet software.
4	Every line ends with EOL.	There is no specific EOL character.	It terminates a line automatically when the delimiter is not used after data.

Absolute Path – It is a full path of the file from the root directory. Absolute paths ensure that Python can find the exact file on your computer.
Ex : - C:\\Users\\Tanmay\\Desktop\\Delete\\file_handling.txt
f=open(“C:\\Users\\Tanmay\\Desktop\\Delete\\file_handling.txt”,r)

Relative Path – It is the path of the file from the current working directory.

Relative Path is the hierarchical path that locates a file or folder on a file system starting from the current directory. The relative path is different from the absolute path, which locates the file or folder starting from the root of the file system.
f=open(“file_handling.txt”,r)

Operation On Files In python

1. Opening a file -open() function
2. Read/Write file -Read Functions – read() , readline(), readlines()
   Write Functions – write() , writelines()
3. Close the File – close() Function
4. Remove the file – remove() #Function of OS Module
5. Rename the file–rename(“oldname”,”newname”) #Function of OS Module

Read the Contents of An Existing File

1. Open the file–

file_object = open(“read_sample.txt”, ’r’) # Relative Path
file_object = open(“C:\\Desktop\\Notes\\sample.txt”) #Absolute Path

2. Reading the file–

rf = file_object.read()
print(rf)

3. Closing the file–

file_object.close()

Working of open() function

We use open () function in Python to open a file in read or write mode. open ( ) will return a file object. To return a file object we use open() function along with two arguments, that accepts file name and the mode, whether to read or write. So, the syntax being: open(filename, mode). There are three kinds of mode, that Python provides and how files can be opened:
" r ", for reading.
" w ", for writing.
" a ", for appending.

Modes Description

1. **r** - Opens a file for reading only. The file pointer is placed at the beginning of the file. This is the default mode. Gives error if file does not exist.
2. **r+** - Opens a file for both reading and writing. The file pointer placed at the beginning of the file.
3. **w** - Opens a file for writing only. Overwrites the file if the file exists. If the file does not exist, creates a new file for writing.
4. **w+** - Opens a file for both writing and reading. Overwrites the existing file if the file exists. If the file does not exist, creates a new file for reading and writing.
5. **a** - Opens a file for appending. The file pointer is at the end of the file if the file exists. That is, the file is in the append mode. If the file does not exist, it creates a new file for writing.
6. **a+** - Opens a file for both appending and reading. The file pointer is at the end of the file if the file exists. The file opens in the append mode. If the file does not exist, it creates a new file for reading and writing.

Reading from a file

There are three ways to read data from a text file.

read() : Returns the read bytes in form of a string. Reads n bytes, if no n specified, reads the entire file.

File_object.read([n])

readline() : Reads a line of the file and returns in form of a string. For specified n, reads at most n bytes. However, does not reads more than one line, even if n exceeds the length of the line.

File_object.readline([n])

readlines() : Reads all the lines and return them list in which each line as a string element.

File_object.readlines()

Note: '\n' is treated as a special character of two bytes.

Writing onto text files:

write() function

The write() function will write the content in the file without adding any extra characters.

```
file_name.write(content)
```

writelines() function

This function writes the content of a list to a file. file_name.

```
writelines(list_of_lines)
```

# Program to show various ways to read data from a file.

# Creating a file

```
    file1 = open("myfile.txt", "w")
    L = ["This is Delhi \n", "This is Paris \n", "This is London \n"]
```

# Writing data to a file

```
    file1.write("Hello \n")
    file1.writelines(L)
    file1.close()
```

# To show difference between read and readline

```
    print("Output of Read(9) function is ")
    print(file1.read(9))
    print()

    file1.seek(0)

    print("Output of Readline(9) function is ")
    print(file1.readline(9))
    print()

    file1.seek(0)

    print("Output of Readlines function is ")
    print(file1.readlines())
```

Output:

```
    Output of Read() function is
     Hello
    This is Delhi
    This is Paris
    This is London

    Output of Readline function is
```

```
 Hello

Output of Read(9) function is Hello
Th

Output of Readline(9) function is
 Hello


Output of Readlines function is

['Hello \n', 'This is Delhi \n', 'This is Paris \n', 'This is London \n']
```

Setting Offsets in a File

If we want to access data in a random fashion, then Python gives us seek() and tell() functions to do so

The tell() method

This function returns an integer that specifies the current position of the file object in the file. The position so specified is the byte position from the beginning of the file till the current position of the file object. The syntax of using tell() is:

A=file_object.tell()

The seek() method

This method is used to position the file object at a particular position in a file. The syntax of seek() is:

file_object.seek(offset ,[reference_point])

In the above syntax, offset is the number of bytes by which the file object is to be moved. reference_point indicates the starting position of the file object. That is, with reference to which position, the offset has to be counted. It can have any of the following values:

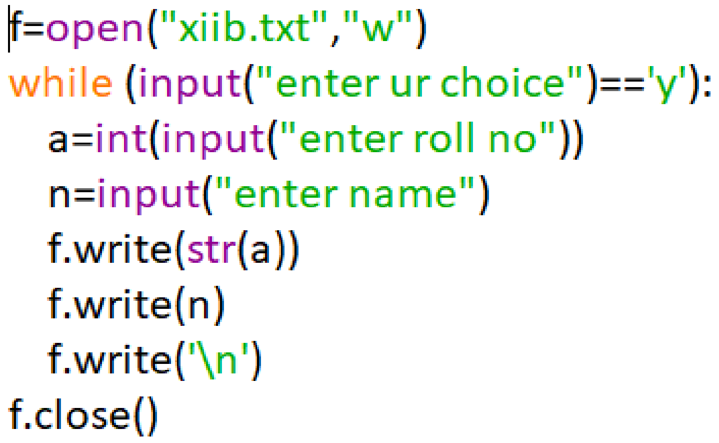
0	- beginning of the file
1	- current position of the file
2	- end of file

By default, the value of reference_point is 0, i.e. the offset is counted from the beginning of the file.

For example, the statement fileObject.seek(5,0) will position the file object at 5th byte position from the beginning of the file.

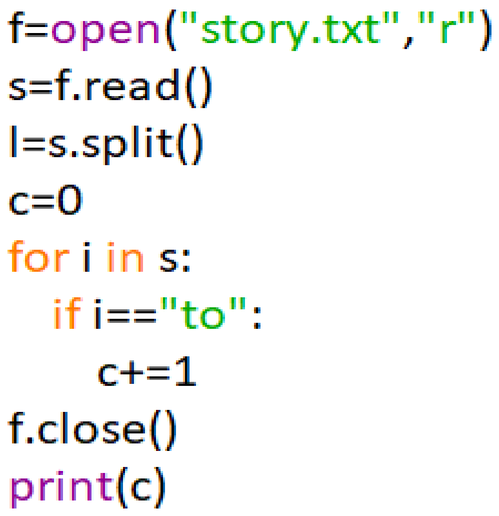
SAMPLE PROGRAMS

1. Write a program to write roll no and name to xiib.txt



```
f=open("xiib.txt","w")
while (input("enter ur choice")=='y'):
   a=int(input("enter roll no"))
   n=input("enter name")
   f.write(str(a))
   f.write(n)
   f.write('\n')
f.close()
```

2.Write read contents from story.txt and count no: of independent words “to” in the file



```
f=open("story.txt","r")
s=f.read()
l=s.split()
c=0
for i in s:
   if i=="to":
      c+=1
f.close()
print(c)
```

3.______ is the return type of readline()

Ans: string

PRACTICE QUESTIONS

1. Write a program to read contents from the text file story.txt and count no:of vowels in it
2. Write a program to read contents from the text file myfile.txt and find average word count
3. Write a program to read contents from the text file library.txt and count “is” as independent word
4. Write a program to read contents from the text file diary.txt and count number of lines with Starting letter “T” or “M”
5. Write a program to read contents from the text file mydiary.txt and count number of lines with ending letter “r”

MULTIPLE CHOICE QUESTIONS

1	If a text file is opened in **w+** mode, then what is the initial position of file pointer/cursor? a. Beginning of file b. End of the file c. Beginning of the last line of text file Undetermined
2	Which of the following statements are true? a. When you open a file for reading, if the file does not exist, an error occurs b. When you open a file for writing, if the file does not exist, a new file is created c. When you open a file for writing, if the file exists, the existing file is overwritten with the new file d. All of the mentioned
3	To read the entire remaining contents of the file as a string from a file object myfile, we use a. myfile.read(2) b. myfile.read() c. myfile.readline() d. myfile.readlines()
4	A text file is opened using the statement **f = open('story.txt').** The file has a total of 10 lines. Which of the following options will be true if statement 1 and statement 2 are executed in order. Statement 1: L1 = f.readline( ) Statement 2: L2 = f.readlines( ) a. L1 will be a list with one element and L2 will be list with 9 elements. b. L1 will be a string and L2 will be a list with 10 elements. c. L1 will be a string and L2 will be a list with 9 elements. d. L1 will be a list with 10 elements and L2 will be an empty list.
5	Which function of a file object can be used to fetch the current cursor position in terms of number of bytes from beginning of file? a. seek( ) b. bytes( ) c. tell( ) d. fetch( )

6. What will be the output of the following code?

```
f = open(‘test.txt’, ‘w+’)
L = [‘First Line\n’, ‘Second Line\n’, ‘Third Line’]
f.writelines(L)
f.flush()
f.seek(0)
O = f.readlines()
print(len(O))
```

a. 33
b. 31
c. 28
d. 3

7. The contents of a text file named ‘quote.txt’ is as shown below

All the kings horses and all the kings men
cannot fix what isn’t broken.

What will be the output of the following code?

```
fin = open('fracture.txt’)
data = fin.read(10)
print(data[0:3], end= ‘’)
data = fin.readline(5)
print(data[0:3] , end= ‘’)
fin.seek(0)
data = fin.read(4)
print(data[0:3] , end= ‘’)
```

a. AllngsAll
b. AllcanAll
c. Allcancan
d. Allngscan

8 What will be the most correct option for possible output of the following code, given that the code executes without any error.

```
f = open(‘cricket.txt’)
data = f.read(150)
print(len(data))
```

a. It will always be 150
b. 151
c. More than or equal to 150
d. Less than or equal to 150

9. For the following python code, what will be the datatype of variables x, y, z given that the code runs without any error?

```
f = open(‘story.txt’)
x = f.read(1)
y = f.readline()
z = f.readlines()
```

a. string, list, list

b. None, list, list

c. string, string, list

d. string, string, string

10 The contents of a text file named ‘fracture.txt’ is as shown below:

Do you dare stay out, Do you dare go in
How much can you lose, How much can you win
And if you go in, should you turn left or right
You will get so confused that you will start in to race
Down long wiggled roads at a breaknecking pace

What will be the output of the following code?

```
fin = open('fracture.txt’)
x = 0
for line in fin:
        words = line.split( )
        for w in words:
              if len(w)>x:
                    x = len(w)
print(x)
```

a.	12
b.	11
c.	13
d.	10

ANSWERS:		
	1.	a. Beginning of file
	2.	d. All of the mentioned
	3.	b. myfile.read()
	4.	c. L1 will be a string and L2 will be a list with 9 elements.
	5.	c. tell( )
	6.	d. 3
	7.	a. AllngsAll
	8.	d. Less than or equal to 150
	9.	c. string, string, list
	10.	a. 12

VERY SHORT ANSWER TYPE QUESTIONS

1. **Differentiate between file modes r+ and w+ with respect to python?**

Ans: r+ opens a text file for reading and writing.

w+ opens a text file for reading and writing. It overwrites the file if it exists, create a file if it doesn’t.

2. **Write a statement in Python to open a text file “ABC.TXT” in reading mode.**

Ans: F=open("ABC.TXT","r")

3. **In ____________ files each line terminates with EOL or ‘\n’ or carriage return, or ‘\r\n’.**

Ans: Text File

4. **Observe the following code and answer the questions that follow.**

```
File=open(“MyData”,”a”)
______________ #Blank1
File.close()
```

a) What type (text/binary) of file is MyData ?

b) Fill the Blank1 with statement to write “ABC” in the file “Mydata”

Ans: a) Text File

b) File.write(“ABC”)

5. **What are files?**

Ans: A named entity, usually stored on a hard drive that contains a stream of characters are called files.

2 SHORT ANSWER TYPE QUESTIONS

1. **Explain seek() method in python.**

In Python, seek() function is used to change the position of the File Handle to a given specific position.

Syntax: fi.seek(offset, from_where), where fi is the file pointer

Offset: This is used for defining the number of positions to move forward.

from_where: This is used for defining the point of reference. It can take **0:** beginning of the file. **1:** current position of the file. **2:** end of the file.

2. **Write a function in Python that counts the number of "the" or "this" words present in a text file "myfile.txt".**

Example: If the "myfile.txt" contents are as follows:

This is my first class on Computer Science. File handling is the easiest topic for me and Computer Networking is the most interesting one.

The output of the function should be: Count of the/this in file: 3

Answer:

```
def displayTheThis():
  num=0
  f=open("myfile.txt","r")
  N=f.read()
  M=N.split()
  for x in M:
    if x=="the" or x== "this":
      print(x)
      num=num+1
  f.close()
  print("Count of the/this in file:",num)
```

3. **Write a function countVowels() in Python, which should read each character of a text file “myfile.txt”, count the number of vowels and display the count.**

Example: If the “myfile.txt” contents are as follows:

This is my first class on Computer Science.

The output of the function should be: Count of vowels in file: 10

Answer:

```
def countVowels():
  fobj = open(“myfile.txt”)
  data = fobj.read()
  count = 0
  vowels=[‘a’,’e’,’I’,’o’,’u’]
  for ch in data:
    if ch in vowels:
       count +=1
  print(“Count of vowels in file:”, count)
```

4. **Write a Python program to count all the line having 'a' as last character**.

Answer:

```
count =0
f=open('fracture.txt',"r")
data=f.readlines()
for line in data:
   if line[-2] == 'a':
        count=count+1
print("Number of lines having 'a' as last character is/are : " ,count)
      f.close()
```

5. **Assume that a text file named TEXT1.TXT already contains some text written into it, write a program with a function named vowelwords(),that reads the file TEXT1.TXT and create a new file named TEXT2.TXT ,which shall contain only those words from the file TEXT1.TXT which don't start with an uppercase vowel(i.e. with 'A','E','I','O','U')** ,

for example if the file TEXT1.TXT contains

Carry Umbrella and Overcoat When it Rains

then the file TEXT2.TXT shall contain

Carry and when it Rains.

Answer:

```
def vowelwords ():
        file1=open('TEXT1.txt','r')
        file2=open('TEXT2.txt','w')
        text = file1.read()
        text=text.split()
        vowels=['A','E','I','O','U']
        for i in text:
              if i[0] not in vowels:
              file2.write(i)
              file2.write(" ")
        file1.close()
        file2.close()
vowelwords()
```

Binary file

A binary file is a file whose content is in a binary format consisting of a series of sequential bytes, each of which is eight bits in length. The content must be interpreted by a program or a hardware processor that understands in advance exactly how that content is formatted and how to read the data. Binary files include a wide range of file types, including executables, libraries, graphics, databases, archives and many others.

There are mainly two types of data files — text file and binary file.

A text file consists of human readable characters, which can be opened by any text editor.

On the other hand, binary files are made up of non-human readable characters and symbols, which require specific programs to access its contents.

Binary files are also stored in terms of bytes (0s and 1s), but unlike text files, these bytes do not represent the ASCII values of characters.

Binary files are stored in a computer in a sequence of bytes.

Differences between text files and binary files.

S. No.	Text file	Binary File
1.	The text files can easily be transferred from one computer system to another.	Binary files cannot easily be transferred from one computer system to another due to variations.
2.	It stores data using ASCII format i.e. human-readable graphic characters.	It stores data in binary format i.e. with the help of 0 and 1.
3.	These files are easily readable and modifiable because the content written in text files is human readable.	These files are not easily readable and modifiable because the content written in binary files is not human-readable and it is encrypted content.
4.	These files create portability problems.	These files are easily portable.
5.	Error in a textual file can be easily recognized and eliminated.	Error in a binary file corrupts the file and is not easily detected.

S. No.	Text file	Binary File
6.	Text files are used to store data more user friendly.	Binary files are used to store data more compactly.
7.	Mostly .txt and .rtf are used as extensions to text files.	Can have any application defined extension.

Basic operations on a binary file

There are 4 types of operations that can be handled by Python on files:

- Opening a file
- Reading data from a file
- Writing data into a file
- Closing a file

Opening a Binary file in Python

Opening a file refers to getting the file ready either for reading or for writing.
To open a file in Python, we use the open () function.

The syntax of open () is as follows:

file_object= open (file_name, access_mode)

Mode	Description
rb	Open file in binary mode for reading only. The file pointer stands at the beginning of the file.
rb+	Open file in binary mode for both reading and writing. The file pointer stands at the beginning of the file.
wb	Open file in binary mode for writing only. It creates the file if it does not exist. If the file exists, then it erases all the contents of the file. The file pointer stands at the beginning of the file.
wb+	Open file in binary mode for both reading and writing. It creates the file if it does not exist. If the file exists, then it erases all the contents of the file. The file pointer stands at the beginning of the file.
ab	Open file in binary mode for appending data. Data is added to the end of the file. It creates the file if it does not exist. The file pointer stands at the end of the file.
ab+	Open a file in binary mode for reading and appending data. Data is added to the end of the file. It creates the file if it does not exist. The file pointer stands at the end of the file.

Example1:

myObject=open(“myfile.txt”, “a+”)

The file myfile.txt is opened in append and read modes. The file object will be at the end of the file. That means we can write data at the end of the file and at the same time we can also read data from the file using the file object named myObject.

Example 2:

```
# open the file using open() function
file = open("sample.txt")
# Reading from file
print(file.read())
```

Closing a Binary file

Python has a **close()** method to close a file.

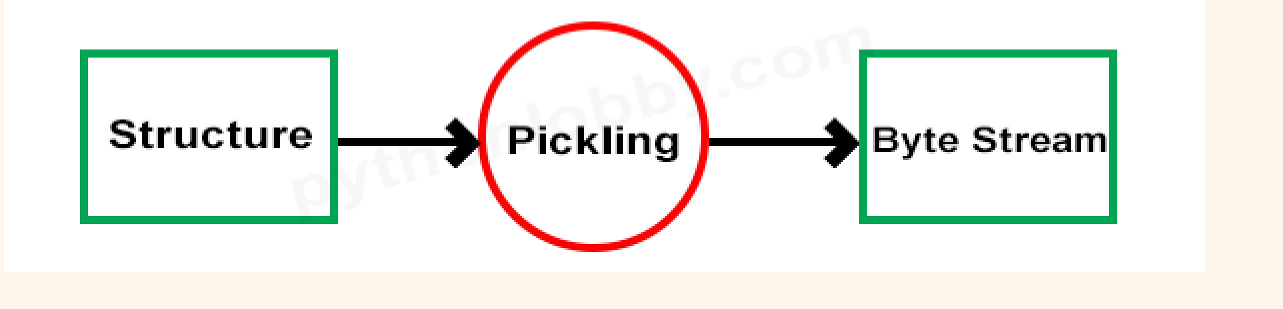
Example 1:

```
file = open("sample.txt")
print(file.read())        # Reading from file
file.close()     # closing the file
```

Import Pickle Module in Python

Pickling: The process of converting the structure (lists and dictionary etc.) into a byte stream just before writing to the file. This is also called as object serialization. It convert a Python object (list, dictionary, etc.) into a character stream.

Unpickling: A reverse of pickling process where information from byte stream gets converted into object structure.



Dump() function: We use dump() method to perform pickling operation on our **Binary Files.** It returns the object representation in byte mode. The dump() method belongs to pickle module.

Python pickle module is used for serializing and de-serializing a Python object structure. Any object in Python can be pickled so that it can be saved on disk.

“Pickle” method can be used to store any kind of object in a file as it allows us to store Python objects with their structures. So for storing data in binary format we will use “pickle” method.
First, we need to import the module.

import pickle

It provides two main methods for the purpose- dump and load.

1. **pickle.dump():** This function is used to pickle the data object into byte stream and stores it in the binary file.
2. Pickle.load() : This function reads pickled objects from a file, whereas loads() deserializes them from a bytes-like object.

Example:

```
import pickle

file = open('important', 'rb')    # open a file, where you stored the pickled data

data = pickle.load(file) # dump information to that file

file.close()  # close the file

print('Showing the pickled data:')

cnt = 0

for item in data:

    print('The data ', cnt, ' is : ', item)

    cnt += 1
```

Reading data from a Binary file

- This module can be used to store any kind of object in file.
- This module provides the dump( ) and load( ) methods, to write and read data from a binary file.
- Python provided read ( ) and write ( ) methods works with string parameters and will not directly work with binary files. Conversion of data at the time of reading and writing is required.

Follow these steps to read data:

1. Open the file in read mode using “rb” Ex.: f = open(“File.dat”, “rb”)
2. Use while loop with True statement to read the entire contents of the file individually.
3. Use try – except for Exception handling to avoid runtime EOFError
4. Now load data into an object through the load function
5. Print data as per need
6. Close the file

Syntax

•dump(object , file_handler) - used to write any object to the binary file.

• Object=load(file_handler) - used to read object from the binary file.

Example

```
 import pickle
# This will write the object List into the file .
x=[1,2,3,4,5]
pickle.dump(x,f)
# This will read the object from the file and store in variable.
x=pickle.load(f)
print(x)
```

Writing the Binary File

```
Example
import pickle
f=open("abc.bin","wb")
x="Hello 1"
pickle.dump(x, f)      # writing a string to file
x=[1,2,3,4,5]
pickle.dump(x, f)     # writing a list to file
x={"Name":"Ajay", "Age":15,"Class":9}
pickle.dump(x, f)    # writing a Dictionary to file
f.close()
```

Search Records from binary file

Follow these steps to search the record in the binary file:

1. Open the file in reading mode using “rb”
2. Prompt a message to ask unique field from data to search
3. Declare a boolean variable flag to store **False** for the record not found and **True** when the record found
4. Use a while loop to access records individually
5. Now load the data into the dictionary object using load() function
6. Use if condition to compare the data with the variable taken in step 2
7. Print the record found
8. Assign True to the flag variable declared in step 3
9. Use the except block to handle EOFError and terminate the loop using the break
10. Print record not found message when Flag is False
11. Finally, close the file using f.close()

Example:

```
def bf_search():
        f = open("Sports.dat","rb")
        pc = int(input("Player to code to search:"))
        flag=False
        while True:
           try:
              rec= pickle.load(f)
              if rec['Pcode']==pc:
                   print("Player Name:",rec['Pname'])
                   print("Individual Score:",rec['Score'])
                   print("Rank:",rec['Rank'])
                   flag = True
            except Exception:
               f.close()
        if flag==False:
           print("Record not found...")
        f.close()
bf_search()
```

Append data in Binary File

To append data in binary follow these steps:

1. Open the file in append mode using “ab” Ex.: f = open (“file.dat”,”ab”)
2. Enter data to append
3. Append entered data into the dictionary/list object
4. Use pickle.dump() method to write the dictionary/list data
5. Close the file

Example:

```
def   bf_append():
        f = open("sports.dat","ab")
        print("Append Data")
        pcode = int(input("Enter the Player code:"))
        pname = input("Enter Player Name:")
        score = int(input("Enter individual score:"))
        rank = int(input("Enter Player Rank:"))
        rec={'Pcode':pcode,'Pname':pname,'Score':score,'Rank':rank}
        pickle.dump(rec,f)
        f.close()
bf_append()
```

Update record in Binary file

To update record you can use the search record code if you wish. To update the record follow these steps:

1. Open the file using read mode
2. Declare a variable for unique value to be updated
3. Use try-except and while loop as explained above
4. Add record fetched from binary file into a list
5. Enter the new record information to update
6. Compare the fetched records with entered record and assign the new values to update
7. Write the data using dump() function
8. Close the file

Example:

```
def bf_update():
    f = open('student.dat','rb')
    reclst = []
    while True:
        try:
            rec = pickle.load(f)
            reclst.append(rec)
        except EOFError:
            break
    f.close()
pc=int(input("Enter player code to update:"))
pn=input("Enter new name:")
ps=int(input("Enter Player Score:"))
pr=int(input("Enter Player Rank:"))
for i in range (len(reclst)):
  if reclst[i]['Pcode']==pc:
    reclst[i]['Pname'] = pn
    reclst[i]['Score'] = ps
    reclst[i]['Rank'] = pr
f = open('sports.dat','wb')
for i in reclst:
  pickle.dump(i,f)
f.close()
bf_update()
```

Question and Answers

1. Which functions is used to close a file in python?

(A) close (B) cloose() (C) Close() (D) close()

Ans: (D) close()

2. Vishnu is a Python programmer. He has written a code and created a binary file record.dat with studentid, subjectcode and marks. The file contains 10 records.He now has to update a record based on the studentid id entered by the user and update the marks. The updated record is then to be written in the file temp.dat. The records which are not to be updated also have to be written to the file temp.dat. If the student id is not found, an appropriate message should to be displayed. As a Python expert, help him to complete the following code based on the requirement given above:

```
import _______ #Statement 1
def update_data():
    rec={}
    fin=open("record.dat","rb")
    fout=open("_____________") #Statement 2
    found=False
    sid=int(input("Enter student id to update his marks :: "))
    while True:
        try:
            rec = ______________ #Statement 3
            if rec["studentid"]==sid:
                found=True
                rec["marks"]=int(input("Enter new marks :: "))
                pickle.____________ #Statement 4
            else:
                pickle.dump(rec,fout)
        except:
            break
if found==True:
    print ("The marks of studentid ", sid ," has been updated.")
else:
    print("No student with such id is not found")
```

```
                fin.close()
                fout.close()
```

(i) Which module should be imported in the program? (Statement 1)

(ii) Write the correct statement required to open a temporary file named temp.dat. (Statement 2)

(iii) Which statement should Aryan fill in Statement 3 to read the data from the binary file, record.dat and in Statement 4 to write the updated data in the file, temp.dat?

Answer:

(i) pickle (1 mark for correct module)
(ii) fout=open('temp.dat', 'wb') (1 mark for correct statement)
(iii) pickle.load(fin)
(iv) pickle.dump(rec,fout)

3. Which of the following python statement will bring the read pointer to 10th character from the end of a file containing 100 characters, opened for reading in binary mode.

a) File.seek(10,0) b) File.seek(-10,2) c) File.seek(-10,1) d) File.seek(10,2)

Ans: File.seek(-10,2)

4. Which of the file opening mode will open the file for reading and writing in binary mode.

a) rb b) rb+ c) wb d) a+

Ans: b) rb+

5. ASSERTION AND REASONING based questions.

Mark the correct choice as

(a) Both A and R are true and R is the correct explanation for A

(b) Both A and R are true and R is not the correct explanation for A

(c) A is True but R is False

(d) A is false but R is True

Assertion (A): A binary file in python is used to store collection objects like lists and dictionaries that can be later retrieved in their original form using pickle module.

Reasoning (A): A binary files are just like normal text files and can be read using a text editor like notepad.

Ans: A is True and R is False

6.A binary file data.dat needs to be created with following data written it in the form of Dictionaries.

Rollno	Name	Age
1001	TOM	17
1002	BOB	16
1003	KAY	16

Write the following functions in python accommodate the data and manipulate it.

a) A function insert() that creates the data.dat file in your system and writes the three dictionaries.

b) b) A function read() that reads the data from the binary file and displays the dictionaries whose age is 16.

Ans:

```
a)import pickle
def insert():
    d1 = {'Rollno':1001, 'Name':'TOM', 'Age':17}
    d2 = {'Rollno':1002, 'Name':'BOB', 'Age':16}
    d3 = {'Rollno':1003, 'Name':'KAY', 'Age':16}
    f = open("data.dat","wb")
    pickle.dump(d1,f)
    pickle.dump(d2,f)
    pickle.dump(d3,f)
    f.close()
b)
def read():
    f = open("data.dat", "rb")
    while True:
        try:
            d = pickle.load(f)
            if d['Age']==16:
                print(d)
        except EOFError:
    break
 f.close()
```

7. Which of the following mode will refer to binary data?

(a)r (b) w (c) + (d) b

Ans: (d) b

8. Which file mode can be used to open a binary file in both append and read mode?

a) w+ b) wb+ c) ab+ d) a+

Ans: (c) ab+

9. Sheela is a Python programmer. She has written a code and created a binary file “book.dat” that has structure [BookNo, Book_Name,Author, Price]. The following user defined function CreateFile() is created to take input data for a record and add to book.dat and another user defined function CountRec(Author) which accepts the Author name as parameter and count and return number of books by the given Author.As a Python expert, help her to complete the following code based on

the requirement given above:

```
import ___________ #statement1
def createFile():
        fobj = open("book.dat","________") #statement2
        BookNo = int(input("Book number:"))
        Book_Name = input("Book Name:")
        Author = input("Author:")
        Price = int(input("Price:"))
        rec = [BookNo, Book_Name, Author, Price]
        pickle.__________________ #statement3
        fobj.close()
       def Count_Rec(Author):
             fobj = open("book.dat","rb")
             num = 0
             try:
             while True:
                rec = pickle.________________ #statement4
                if Author == rec[2]:
                       num=num+1
             except:
 fobj.close()
```

```
        return num
```

(i) Which module should be imported in the program? (Statement 1)

(ii) Write the correct statement required to open a file named book.dat.(Statement 2)

(iii) Which statement should be filled in Statement 3 to write the data into the binary file, book.dat?

(iv) Statement 4 to read the data from the file, book.dat?

Ans:

(i) pickle

(ii) ab

(iii) dump(rec, fobj)

(iv)load(fobj)

10. Mr. Deepak is a Python programmer. He has written a code and created a binary file “MyFile.dat” with empid, ename and salary. The file contains 15 records. He now has to update a record based on the employee id entered by the user and update the salary. The updated record is then to be written in the file “temp.dat”. The records which are not to be updated also have to be written to the file “temp.dat”. If the employee id is not found, an appropriate message should to be displayed. As a Python expert, help him to complete the following code based on the requirement given above:

```
import ________ #Statement 1
def update_rec():
    rec={}
    fin=open("MyFile.dat","rb")
    fout=open("_____________") #Statement 2
    found=False
    eid=int(input("Enter employee id to update salary : "))
    while True:
        try:
            rec=______________ #Statement 3
            if rec["empid"]==eid:
                found=True
                rec["salary"]=int(input("Enter new salary : "))
                pickle.____________ #Statement 4
            else:
                pickle.dump(rec,fout)
        except:
            break
if found==True:
        print("The salary of employee id ",eid," has been updated.")
else:
        print("No employee with such id is not found")
fin.close()
fout.close()
```

(i) Which module should be imported in the program? (Statement 1)

(ii) Write the correct statement required to open a temporary file named temp.dat.(Statement 2)

(iii) Which statement should Deepak fill in Statement 3 to read the data from the binary file, record.dat and in Statement 4 to write the updated data in the file, temp.dat?

Ans: (i) Ans: pickle

(ii) fout=open(‘temp.dat’, ‘wb’)

(iii) Statement 3: pickle.load(fin)

iv)Statement 4: pickle.dump(rec,fout)

CSV FILE

- A Comma Separated Values (CSV) file is a plain text file that contains the comma-separated data.
- These files are often used for exchanging data between different applications.
- CSV files are usually created by programs that handle huge amounts of data.
- They are used to export data from spreadsheets (ex:- excel file) and databases (Ex:- Oracle, MySQL). It can be used to import data into a spreadsheet or a database.

CSV File Structure

## sample.csv file structure

```
Name, DOB, City
Ram, 12-Jul-2001, Delhi
Mohan, 23-Jan-2005, Delhi
```

Python CSV Module

- CSV Module is available in Python Standard Library.
- The CSV module contains classes that are used to read and write tabular form of data into CSV format.
- To work with CSV Files, programmer have to import CSV Module.

import csv

Methods of CSV Module :

- writer( )
- reader( )

Both the methods return an Object of writer or reader class. Writer Object again have two methods – writerow( ) , writerows( ).

writer( ) Methods

This function returns a writer object which is used for converting the data given by the user into delimited strings on the file object.

writer( ) Object Methods –

- w_obj . writerow(<Sequence>) : Write a Single Line
- w_obj . writerows (<Nested Sequence>) : Write Multiple Lines

Example:-

## writerow()

```
import csv
row=['Nikhil', 'CEO', '2', '9.0']
f=open("myfile.csv", 'w')
w_obj = csv.writer(f)
w_obj.writerow(row)
f.close()
```

## writerows()

```
import csv

rows = [['Nikhil','CEO','2','9.0'],

        ['Sanchit','CEO','2','9.1']]

f=open("myfile.csv",'w')

w_obj = csv.writer(f)

w_obj.writerows(rows)

f.close()
```

reader( ) Methods

This function returns a reader object which will be used to iterate over lines of a given CSV file.

```
r_obj = csv.reader(csvfile_obj)
```

To access each row, we have to iterate over this Object.

```
for i in r_obj:
    print(i)

import csv

f=open("myfile.csv",'r')

r_obj = csv.reader(f)

for data in r_obj:

    print(data)

f.close()
```

SAMPLE QUESTIONS:

1. Write a program to read entire data from file data.csv

```
import csv f=open("data.csv", 'r')
d=csv.reader(f)
for row in d:
        print(row)
```

OUTPUT:

```
 ['Admno', 'Name', 'Class', 'Sec', 'Marks']
 ['1231', 'Amit', 'XII', 'A', '45']
['1224', 'Anil', 'XII', 'B', '49']
 ['1765', 'Suman', 'XI', 'A', '42']
 ['2132', 'Naman', 'XII', 'C', '38']
```

2.Write a program to add/insert records in file “data.csv”. Structure of a record is roll number, name and class.

```
import csv
field = ["Roll no" , "Name" , "Class"]
f = open("data.csv" , 'w')
d=csv.writer(f)
d.writerow(field)
rn=int(input("Enter Roll number: "))
nm = input("Enter name: ")
cls = input("Enter Class: ")
 rec=[rn,nm,cls]
 d.writerow(rec)
 f.close()
```

Multiple Choice Questions:

1. CSV stands for :
   a. Comma Separated Values
   b. Comma Separated Variables
   c. Comma Stored Values
   d. Comma Stored Variables

2. The separator character of CSV Files is called a
   a. EOL
   b. Delimiter
   c. EOF
   d. Default

3. The default delimiter for a CSV file is :

a. Semi colon
b. Colon
c. Comma
d. Hyphen

4. _____ module of Python provides the functionality to read and write tabular data in CSV file.

a. pickle
b. csv
c. file
d. ccv

5. Which function is used to open a csv file ?

a. Open()
b. csv.open()
c. writer()
d. csv.writer()

6. Name the function to read from CSV file.

a. read()
b. csv.reader()
c. csv.read()
d. readline()

7. Which among the following is not a function of csv module?

a. reader()
b. read()
c. writer()
d. writerows()

8. In Python, default newline character is :

a. \f
b. \t
c. \n
d. \v

9. CSV module allows to write multiple rows using function.

a. writerows( )
b. writerow( )
c. writer( )
d. None of the above

10. The opening function of a csv file is similar to the opening of:

a. Binary file
b. Text File
c. Both of them
d. None of them

11. The ________argument of open function is to specify how Python handle the newline characters in csv file

a. newline
b. line
c. mode
d. char

12. Which among the following is an iterable object?

a. writer
b. file
c. reader
d. All of the above

13. To specify a different delimiter while writing into a csv file,argument is used with writer object:

a. newline
b. separator
c. character
d. delimiter

14. Which mode opens the file for exclusive creation, which fails in the case where file already exists

a. a
b. w
c. x
d. r

15. The file mode to open a CSV file for reading as well as writing is .

a. a+
b. w+
c. r+
d. All the above

16. Identify the line number which may cause an error:

```
import csv      #Line1
line=[[1,2,3],[4,5,6]]#Line 2
with open("sample.csv","w",newline="") as csvfile: #Line3
        writer=csv.writer(csvfile,delimiter="|") #Line4
        for line in writer: #Line5
                writer.writerow(line)
```

a. Line1
b. Line2
c. Line 4
d. Line

17. The CSV files are files.

a. Plain text file
b. Binary
c. Data
d. Python

18. The writer () function has how many mandatory parameters?

a. 1
b. 2
c. 3
d. 4

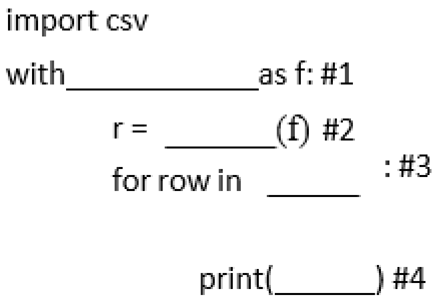
19. Which of the following parameter needs to be added with open function to avoid blank row followed by each record in the CSV file?

a. quotechar
b. quoting
c. newline
d. skiprow

20. Ajaikrishna wants to separate the values by a $ sign. Suggests him a pair of function and parameter to use it.

a. open, quotechar
b. writer, quotechar
c. open, delimiter
d. writer, delimiter

21. Tejalakshmi of Class 12 have written the below code. Observe and fill in the given blanks so that it opens the file “data.csv” and read and print all the records.



```
import csv
with____________as f: #1
        r = _______(f) #2
        for row in  ______  : #3

                print(______) #4
```

I. What must be filled in line 1?

a. Open(“data.csv”,”r”)

b. open(“data.csv”)

c. “data.csv”

d. File

II. What must be filled in Line 2?

a. csv.reader()

b. csv.read()

c. csv.write()

d. csv.writer()

III. What must be filled in line 3?

a. data.csv

b. f

c. r

d. None

IV. What must be filled in line 4?

a. data

b. f

c. “File”

d. row

V. What is the default data type of data read from this file?

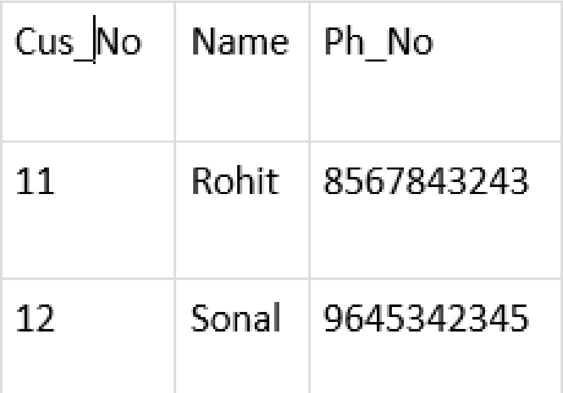
a. List

b. String

c. Tuple

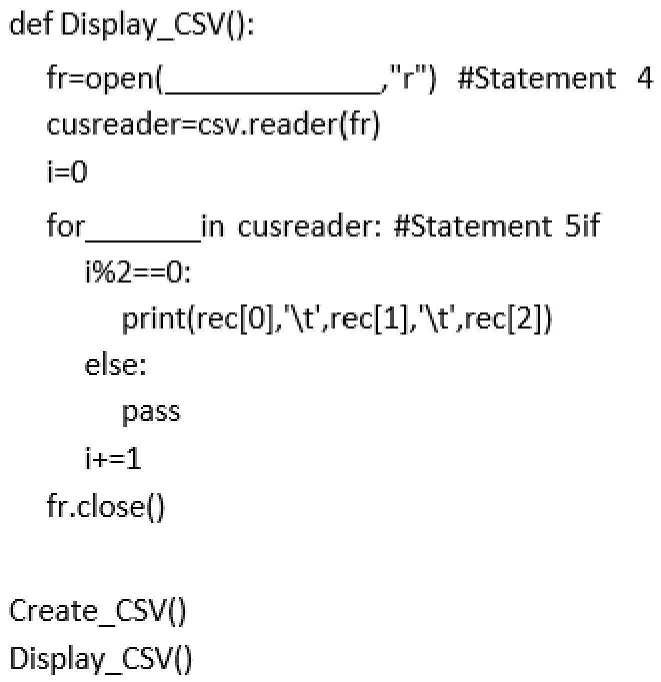
d. Integer

22. Sudev, a student of class 12th, is learning CSV File Module in Python. During examination, he has been assigned an incomplete python code to create a CSV file ‘customer.csv’. Help him in completing the code which creates the desired CSV file.



Cus_No	Name	Ph_No
11	Rohit	8567843243
12	Sonal	9645342345

```
_________csv  #Statement  1
def Create_CSV():
    fw=open("customer.csv","w")
     __________=csv.writer(fw)                    #Statement2
    csvwriter.writerow(["Cus_No","Name","Ph_No"])
    n=int(input("Enter total number of Customer"))
   for i in range(n):
         cusno=int(input("Enter     Customer     no."))
         Name=input("EnterName")
         Ph_No=int(input("EnterPhone           No."))
         Rec=[cusno,Name,Ph_No]
         csvwriter.writerow(_______) #Statement 3
   fw.close()
```



```
def Display_CSV():
   fr=open(_____________,"r")  #Statement  4
   cusreader=csv.reader(fr)
   i=0
   for_______in cusreader: #Statement 5if
      i%2==0:
         print(rec[0],'\t',rec[1],'\t',rec[2])
      else:
         pass
      i+=1
   fr.close()


Create_CSV()
Display_CSV()
```

I. Identify suitable code for the blank space in line marked as Statement-1.

a) include
b) add
c) Import
d) import

II. Identify the missing code for the blank space in line marked as Statement-2.

a) Customer
b) reader
c) csvwriter
d) writer

III. Identify the argument name for the blank space in line marked as Statement-3?

a) Row
b) Rec
c) row
d) rec

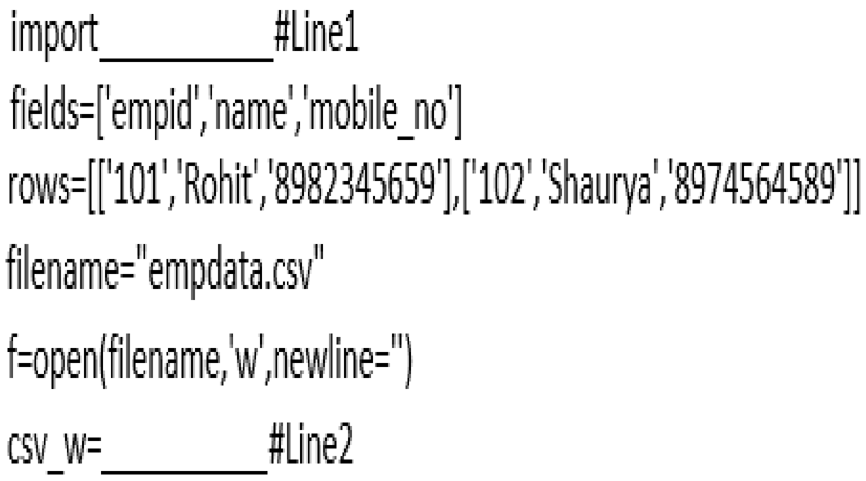
IV. Identify the missing file name for the blank space in line marked as Statement-4?

a) customer
b) customer.csv
c) customer.txt
d) customer.dat

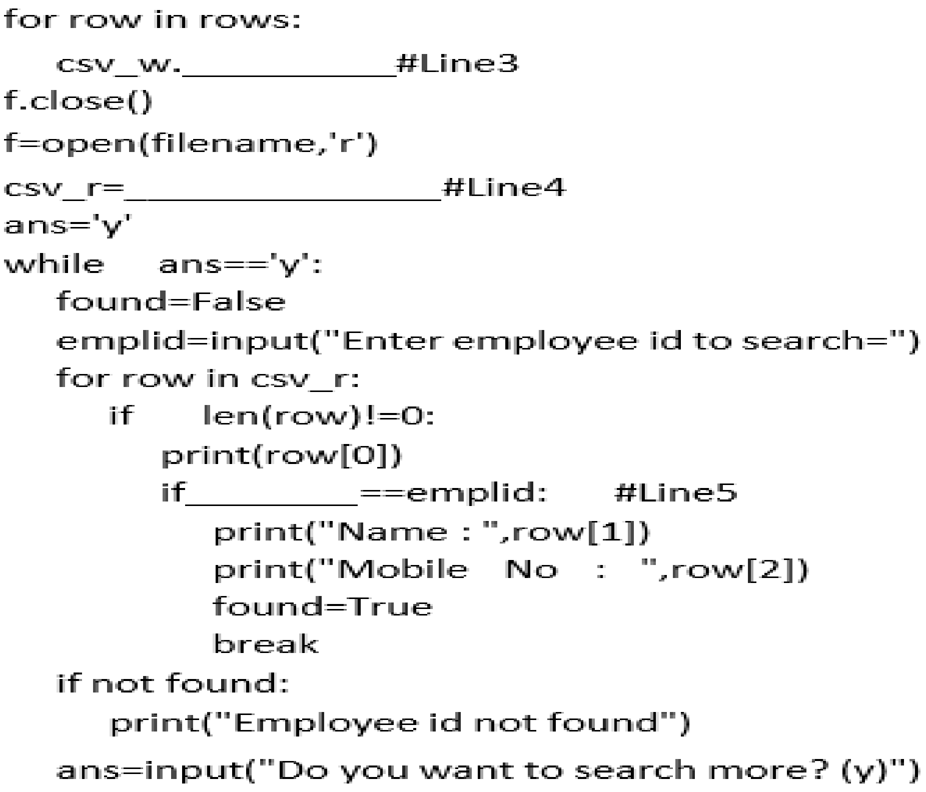
V .Identify the object name for the blank space in line marked as Statement-5?

a) i
b) Rec
c) row
d) rec

23. Daya of class 12 is writing a program to create a CSV file “empdata.csv” with empid, name & mobile number. Also, to search a particular empid and display its record details. He has written the following code. As a programmer, help him to successfully execute the given task.



```
import_________ #Line1
fields=['empid','name','mobile_no']
rows=[['101','Rohit','8982345659'],['102','Shaurya','8974564589']]
filename="empdata.csv"
f=open(filename,'w',newline='')
csv_w=_________ #Line2
```



```
for row in rows:
    csv_w.____________#Line3
f.close()
f=open(filename,'r')
csv_r=__________________#Line4
ans='y'
while    ans=='y':
    found=False
    emplid=input("Enter employee id to search=")
    for row in csv_r:
        if      len(row)!=0:
            print(row[0])
            if__________==emplid:      #Line5
                print("Name : ",row[1])
                print("Mobile   No   :   ",row[2])
                found=True
                break
    if not found:
        print("Employee id not found")
    ans=input("Do you want to search more? (y)")
```

Choose the module he should import in Line1.

a) math
b) pickle
c) csv
d) random

II. Choose a code to write the column heading from fields list in Line2.

a) writerows(fields)
b) writerow(field)
c) writerow(fields)
d) writerows(fields)

III. Choose a code to write the row from rows list in Line3.

a) writerows(row)
b) writerow(row)
c) writerow(rows)
d) write_row(row)

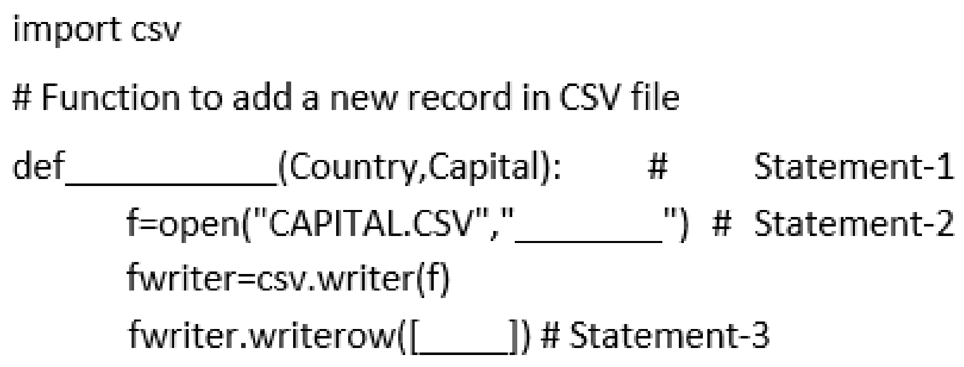
IV. Choose a code for line 4 to read the data from a csv file.

a) csv.reader(f) b) csv.read(f) d) pickle.load(f) e)f.read()

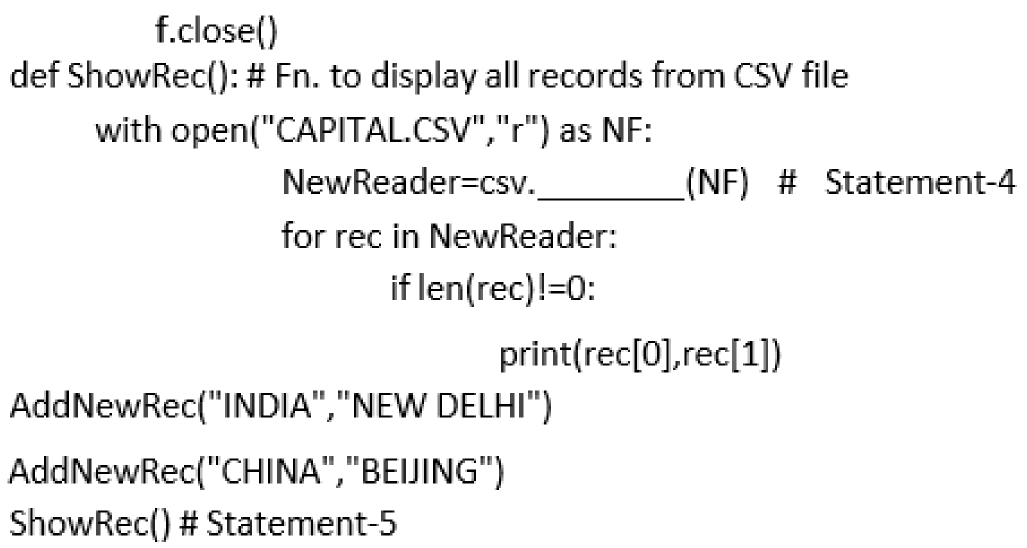
V. Choose the correct variable (list value) to check "emplid" in Line5

a) Row[0]
b) Rec[0]
c) row[0]
d) rec[0]

24.Viraj is making a software on "Countries and their Capitals" in which various records are to be stored/retrieved in "CAPITAL.CSV" data file. It consists of few records of Countries and their Capitals. He has written the following code in python. As a programmer, you have to help him to successfully execute the program.



```
import csv
# Function to add a new record in CSV file
def___________(Country,Capital):      #     Statement-1
        f=open("CAPITAL.CSV","________")  #  Statement-2
        fwriter=csv.writer(f)
        fwriter.writerow([_____]) # Statement-3
```



```
            f.close()
def ShowRec(): # Fn. to display all records from CSV file
        with open("CAPITAL.CSV","r") as NF:
                NewReader=csv.________(NF)   #   Statement-4
                for rec in NewReader:
                        if len(rec)!=0:

                                print(rec[0],rec[1])
AddNewRec("INDIA","NEW DELHI")

AddNewRec("CHINA","BEIJING")
ShowRec() # Statement-5
```

I. Choose the Name of the function in Statement-1.
   a) AddNewRec
   b) Addnew
   c) Addrec
   d) AddNewRec()

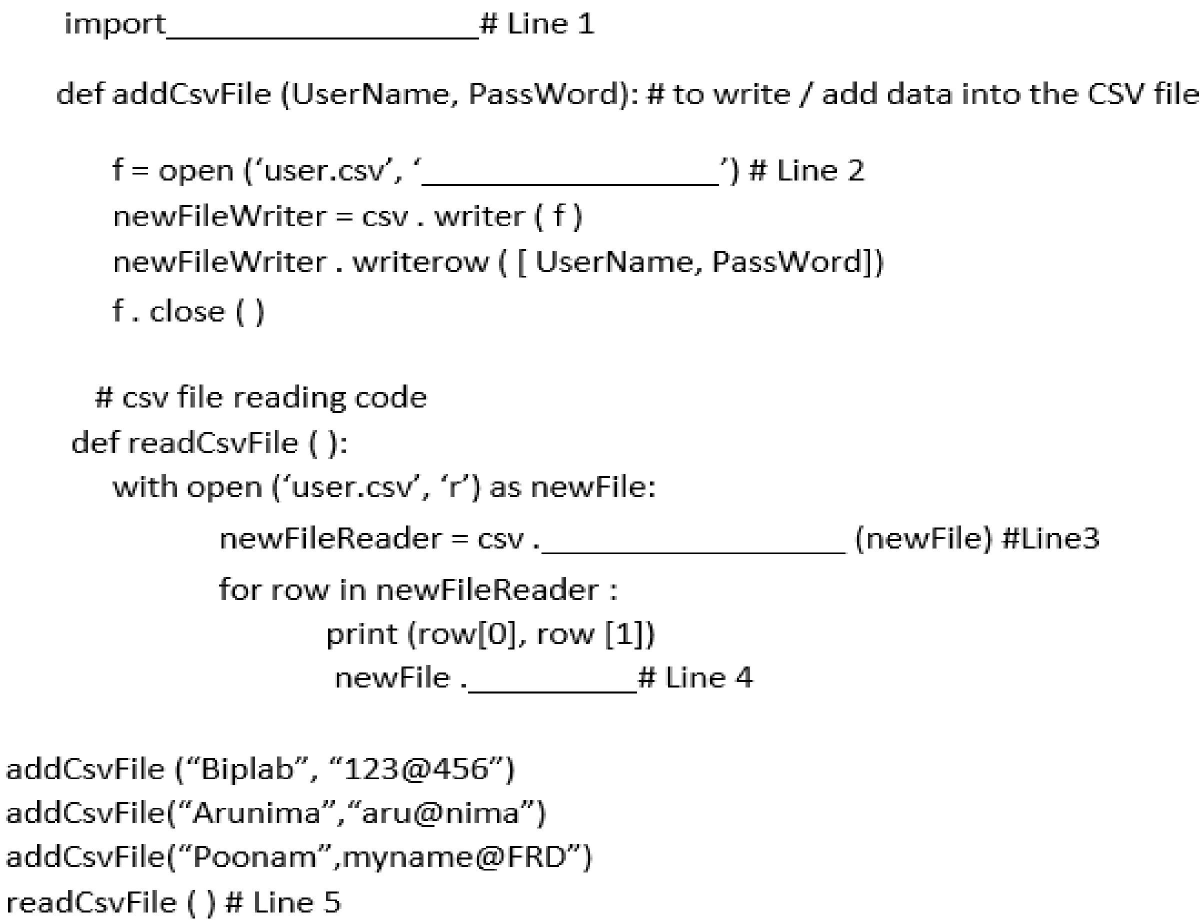
II. Choose the file mode to be passed to add new records in Statement-2.
   a) w
   b) r
   c) w+
   d) a

III. Identify the correct variables in Statement-3 to store data to the file.
   a) country,capital
   b) Country,Capital
   c) Coun,Cap
   d) [Country,Capital]

IV. Choose the correct option for Statement-4 to read the data from a csv file.
   a) Reader()
   b) reader()
   c) read
   d) reader

V. Choose the output which will come after executing Statement-5.
   a) ‘INDIA NEW DELHI’ ‘CHINA BEIJING’
   b) ‘CHINA’ ‘BEIJING’
   c) INDIA NEW DELHI
   d)None of the above

25. Rinsha of class 12 is writing a program to create a CSV file “user.csv” which will contain user name and password for some entries. She has written the following code. As a programmer, help her to successfully execute the given task.



```
  import__________________# Line 1

 def addCsvFile (UserName, PassWord): # to write / add data into the CSV file

     f = open (‘user.csv’, ‘_________________’) # Line 2
     newFileWriter = csv . writer ( f )
     newFileWriter . writerow ( [ UserName, PassWord])
     f . close ( )

   # csv file reading code
  def readCsvFile ( ):
     with open (‘user.csv’, ‘r’) as newFile:
            newFileReader = csv ._________________ (newFile) #Line3
            for row in newFileReader :
                   print (row[0], row [1])
                    newFile .__________# Line 4

addCsvFile (“Biplab”, “123@456”)
addCsvFile(“Arunima”,“aru@nima”)
addCsvFile(“Poonam”,myname@FRD”)
readCsvFile ( ) # Line 5
```

I. Name the module she should import in Line 1.

a. CSV
b. csv
c. math
d. File

II. In which mode, Rinsha should open the file to add data into the file.

a. r
b. a
c. w
d. w+

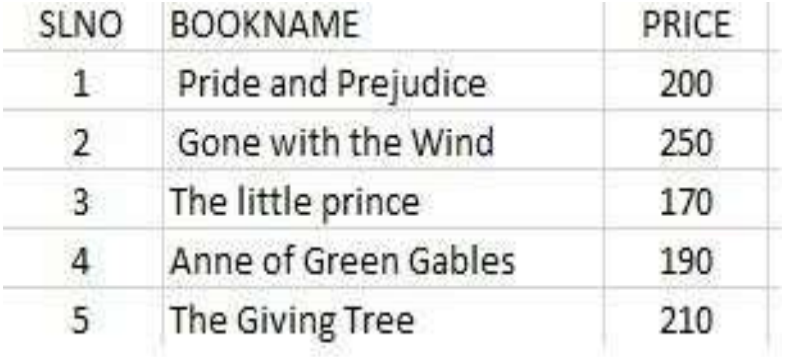
III. Fill in the blank in Line 3 to read the data from a csv file.

a. writer()
b. reader()
c. write()
d. read()

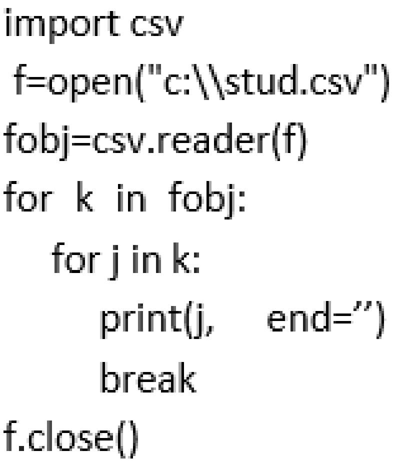
IV. Fill in the blank in Line 4 to close the file.

a. End()
b. Close()
c. close()
d. end()

26. Consider the following csv file and the code fragment associated with the following csv file :



SLNO	BOOKNAME	PRICE
1	Pride and Prejudice	200
2	Gone with the Wind	250
3	The little prince	170
4	Anne of Green Gables	190
5	The Giving Tree	210



```
import csv
 f=open("c:\\stud.csv")
fobj=csv.reader(f)
for  k  in  fobj:
    for j in k:
        print(j,    end=’’)
        break
f.close()
```

I.What will be the output printed by the above code?

a. SLNO12345
b. SLNO
c. The entire content
d. Error

II. What will be the output printed by the above code if the break is replaced with continue?

a. SLNO12345
b. SLNO
c. The entire content
d. Error

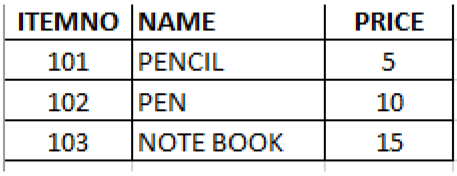
III. What will occur if the file stud.csv is not existing in the mentioned path?

a. It will create a new one
b. It will create an error
c. None of the above
d. It will cause a system reboot

IV. Which statement in the above code will help to move to the next record?

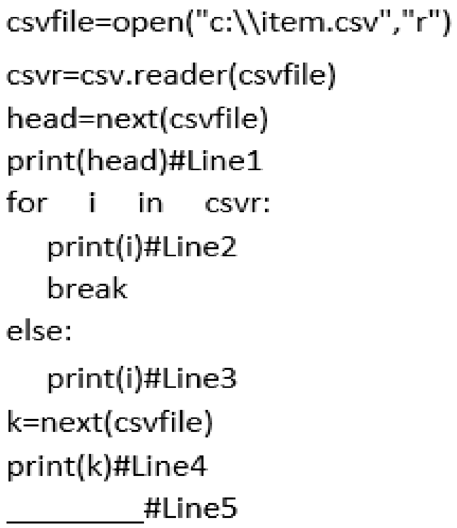
a. fobj.next()
b. next(fobj)
c. fobj.move()
d. fobj.forward()

27. Sai Krishna has created the following csv file named item.csv:



ITEMNO	NAME	PRICE
101	PENCIL	5
102	PEN	10
103	NOTE BOOK	15

He has written the following program for managing the file. Help him to find the answers for the following questions.



```
csvfile=open("c:\\item.csv","r")
csvr=csv.reader(csvfile)
head=next(csvfile)
print(head)#Line1
for   i   in   csvr:
   print(i)#Line2
   break
else:
   print(i)#Line3
k=next(csvfile)
print(k)#Line4
_________#Line5
```

I. What will be printed by Line1?

a. All the records
b. ITEMNO, NAME, PRICE
c. item.csv
d. None of the above

II. What will be printed by Line2?

a. 101,PENCIL,5
b. ITEMNO, NAME, PRICE
c. 102,PEN,10
d. 103,NOTEBOOK,156

III. What will be printed by Line3?

a. 103,NOTEBOOK,15
b. Line 3 will not be executed
c. Line 3 will be executed , but nothing will be printed
d. 102,PEN,10

IV. What will be printed by Line4?

a. 101,PENCIL,5
b. ITEMNO, NAME, PRICE
c. 102,PEN,10
d. 103,NOTEBOOK,15

V. What must be written in Line 5?

a. F.close()
b. f.close()
c. fobj.close()
d. csvfile.close()

ANSWERS

QNO	OPT	QNO	OPT	QNO	OPT	QNO	OPT
1	A	6	b	11	a	16	d
2	B	7	b	12	c	17	a
3	C	8	c	13	d	18	a
4	B	9	a	14	c	19	c
5	B	10	b	15	d	20	d

QNO		OPT	QNO		OPT	QNO		OPT
21	I	b	22	I	d	23	I	c
	II	a		II	c		II	c
	III	c		III	b		III	b
	IV	d		IV	b		IV	a
	V	b		V	d		V	c

QNO		OPT	QNO		OPT	QNO		OPT
24	I	d	25	I	b	26	I	b
	II	d		II	b		II	c
	III	b		III	b		III	b
	IV	D		IV	c		IV	b
	V	b						

QNO		OPT
27	I	b
	II	a
	III	b
	IV	c
	V	b

28. Vedansh is a Python programmer working in a school. For the Annual Sports Event, he has created a csv file named Result.csv, to store the results of students in different sports events. The structure of Result.csv is : [St_Id, St_Name, Game_Name, Result]

Where

St_Id is Student ID (integer)

ST_name is Student Name (string)

Game_Name is name of game in which student is participating(string)

Result is result of the game whose value can be either 'Won', 'Lost' or 'Tie'

For efficiently maintaining data of the event, Vedansh wants to write the following user defined functions:

Accept() – to accept a record from the user and add it to the file Result.csv. The column headings should also be added on top of the csv file.

wonCount() – to count the number of students who have won any event.

As a Python expert, help him complete the task.

ANSWER:

```
def Accept():
    sid=int(input("Enter Student ID "))
    sname=input("Enter Student Name")
    game=input("Enter name of game")
    res=input("Enter Result")
    headings=["Student ID","Student Name","Game Name","Result"]
    data=[sid, sname, game, res]
    f=open("Result.csv", "a", newline="")
    csvwriter=csv.writer(f)
    csvwriter.writerow(headings)
    csvwriter.writerow(data)
    f.close()


def wonCount():
    f=open("Result.csv","r")
    csvreader=csv.reader(f,delimiter=",")
    head=list(csvreader)
```

```
print(head[0])
for x in head:
    if x[3]=="WON":
        print (x)
f.close()
```

29. Kiran is a Python programmer working in a shop. For the employee's tax collection, he has created a csv file named protax.csv, to store the details of professional tax paid by the employees. The structure of protax.csv is : [empid, ename, designation, taxpaid]

Where,

empid is employee id (integer)

ename is employee name (string)

designation is employee designation(string)

taxpaid is the amount of professional tax paid by the employee(string)

For efficiently maintaining data of the tax, Kiran wants to write the following user defined functions:

Collection() – to accept a record from the employee and add it to the file protax.csv.

Totaltax() – to calculate and return the total tax collected from all the employees.

As a Python expert, help him complete the task.

ANSWER:

```
def Accept():
    empid=int(input("Enter Employee ID "))
    ename=input("Enter Employee Name")
    designation=input("Enter the Designation")
    taxpaid=input("Enter the tax amount")
    data=[empid, ename, designation, taxpaid]
    f=open("protax.csv", "a", newline="")
    csvwriter=csv.writer(f)
    csvwriter.writerow(data)
    f.close()
```

```
def Totaltax():
    f=open("protax.csv","r")
    csvreader=csv.reader(f,delimiter=",")
    content=list(csvreader)
    total=0
    for row in content:
        total+=int(row[3])
    f.close()
    return total
```

30. Arun of class 12 was writing a program to read the CSV file “mydata.csv” which will contain user name and password for some entries. As a programmer, help him to write the following user defined functions:

adddata() – to add a new entry to the file “mydata.csv.

display() – to display the content of the file “mydata.csv”.

ANSWER:

```
def adddata():
    userid=input("Enter Username:")
    passwd=input("Enter th password: ")
    entry=[userid,passwd]
    f=open("mydata.csv", "a", newline="")
    csvwriter=csv.writer(f)
    csvwriter.writerow(entry)
    f.close()


def display():
    f=open("mydata.csv","r")
    csvreader=csv.reader(f,delimiter=",")
    content=list(csvreader)
    for rec in content:
        print(rec)
    f.close()
```

DATA STRUCTURE

INTRODUCTION

Computers store and process data with an extra ordinary speed and accuracy. So, it is highly essential to store data efficiently, for accessing it faster. Processing of data should happen in the least possible time, without losing the accuracy.

Data structures deal with how the data is organized and held in the memory when a program processes it.

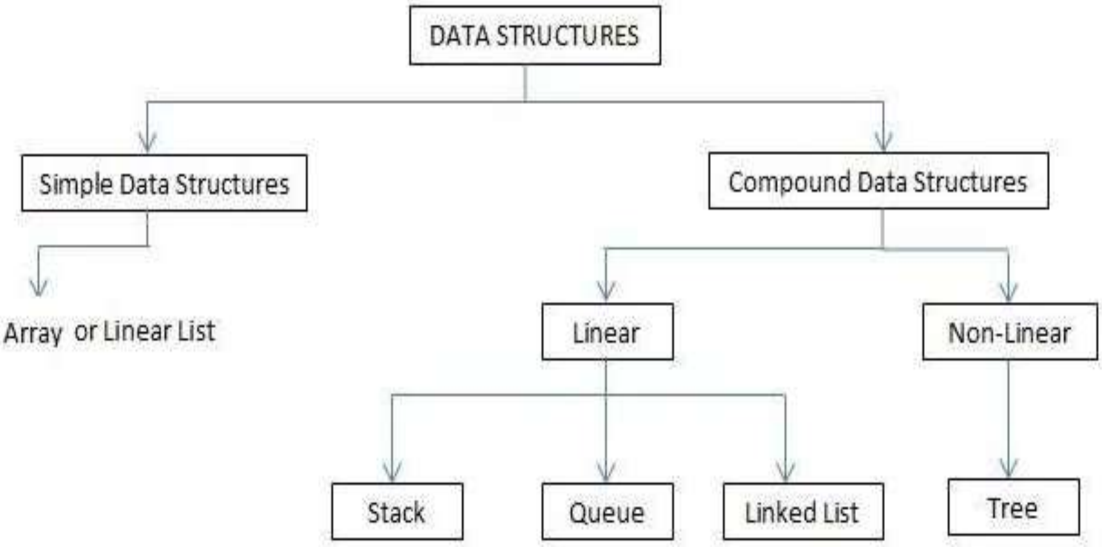
- Data structures, helps in writing efficient programs in any language.
- A data structure is a named group of data of different data types which is stored in a specific way and can be processed as a single unit.
- A data structure has well-defined operations, behaviour and properties.

Data type vs Data structure

- A data type defines a set of values along with well-defined operations stating its input-output behaviour.
- A data structure is a physical implementation that clearly defines a way of storing, accessing, manipulating data stored in a data structure.

For example, List as a data type which can store heterogeneous data type of elements, but when implemented as data structure its behaviour is clearly prescribed (searching, sorting etc.)

List



- Lists in Python are used to store collection of heterogeneous items.
- These are mutable, which means that one can change their content without changing their identity.
- one can recognize lists by their square brackets [and] that hold elements separated by a comma.

• Lists are built into Python: you do not need to invoke them separately.

Stack

A stack is a container of objects that are inserted and removed according to the Last-In-First-Out (LIFO) concept.

For Example: At a dinner party where there is a stack of plates, plates are always added or removed from the top of the pile.
A pile of books, a stack of coins where we can remove only the top book or the coin placed at the top.
This concept is used for evaluating expressions and syntax parsing, scheduling algorithms/routines, etc.

Applications of Stack

1. Reversing a word/line: This can be accomplished by pushing each character on to a stack as it is read. When the line is finished, characters are popped off the stack and they will come off in the reverse order.

2.The compilers use stacks to store the previous state of a program when a function is called during recursion.

2. Backtracking is a form of recursion. But it involves choosing only one option out of possibilities. Used in solving Puzzle Sudoku.

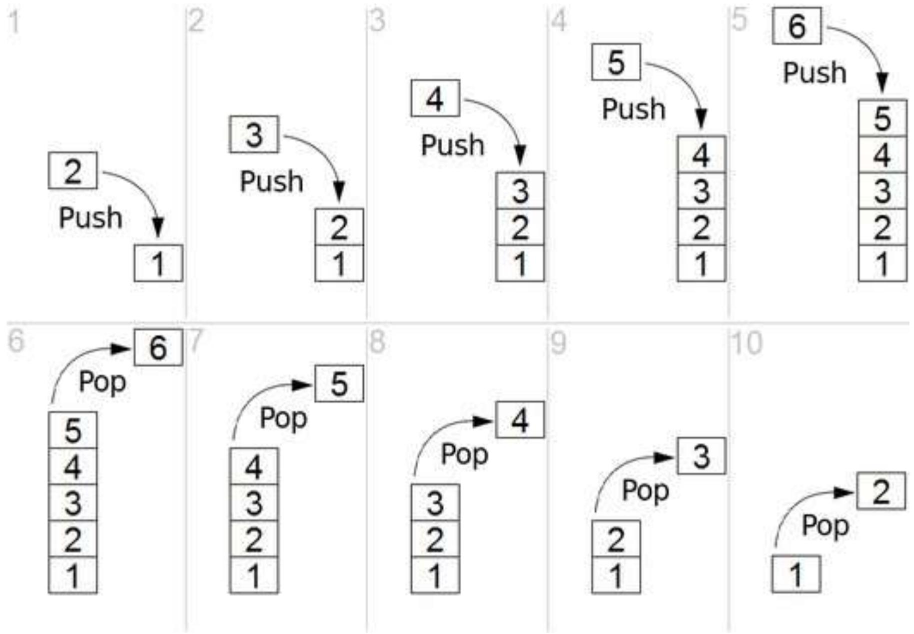
3. Undo mechanism in Text editors by keeping all the text changes in a stack.

Linear List

- A linear data structure is that, whose elements form a sequence.
- When elements of linear structure are homogeneous and are represented in memory by means of sequential memory locations, these linear structures are called arrays.
- Linear List size: Length=UB-LB+1

Stack

Stack is a linear/sequence structure in which insertion and deletion can take place only at one end, i.e., stack’s top. Because of this, stack is called LIFO (Last in First out) data structure.



Implementation of stack using list

- The implementation of stack using list in Python is the easiest of all programming language.
- Basic operations performed on stack are:

1. Creating a stack
2. Push/Adding elements to the stack
3. Checking for empty stack
4. Pop/Deleting elements from a stack
5. Traversal/Displaying a stack

List methods used and Important things to remember

1) list.append(element) – It is used to implement push operations(used to append or add elements at the end of the list)

2) list.pop() –It is used to implement pop operations(removing elements at the end)

3) list[::-1]-List slicing is used to print the elements in the reverse order from top position to list[0]

4) top=len(list)-1 (length of list -1)

5) stack - LIFO(Last In First Out)

6) Push and Pop through one end(Top)

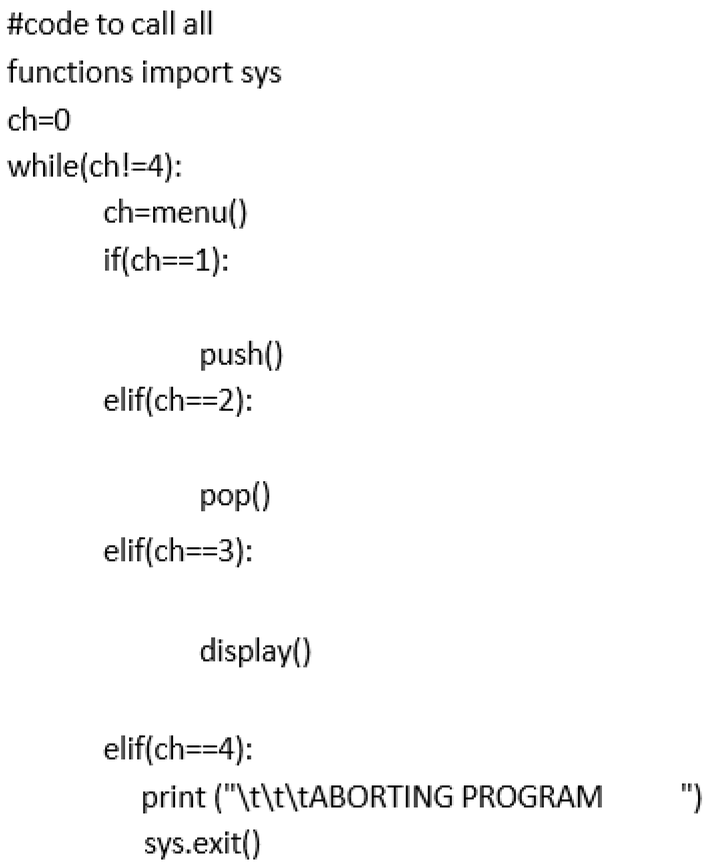
Implementation of stack – menu oriented program

1. Write the Python Code to implement stack.

```
L = []
def PUSH():
        N = int(input("Enter the Number:"))
        L.append(N)
def POP():
        if L == []:
            print("List is Empty - UNDERFLOW")
        else:
            N = L.pop()
            print("The Element Deleted in Stack is :", N)

def DISPALL():
        if L == []:
            print("List is Empty")
        else:
            print("List in LIFO ORDER:")
            for i in range(len(L)-1, -1, -1):
                print(L[i], end='<--')
while True:
        print("\n\n\tMENU:")
        print("\t----")
        print("1. PUSH")
        print("2. POP")
        print("3. DISPLAY ALL")
        print("4. EXIT")
        ch= int(input("Enter the Choice:"))
        if ch == 1:
            PUSH()
        elif ch == 2:
            POP()
        elif ch == 3:
            DISPALL()
        elif ch == 4:
            break
         else:
              print("Give correct choice")
```

2.

```
S=[]
from os import system
def menu():#menu display
            ch=0
            while(ch<1 or ch>4):
            anyvar=system('cls')
            #print ("\n"*100)
            print ("\n\n\n\n\n")
            print ("\t\t\t1: PUSH")
            print ("\t\t\t2: POP")
            print ("\t\t\t3: DISPLAY")
             print ("\t\t\t4: EXIT")
            ch=int(input("\n\t\t\tEnter a choice (1-4):"))
            return ch
def push():#code to push an item
            item=int(input("\t\t\tEnter an item to push: "))
            S.append(item)
            print ("\t\t\tITEM " , item ," PUSHESD IN THE STACK")
 def pop():#code to pop from stack
 if (S==[]):
             print ("\t\t\tNO ITEM TO POP")
  else:
             item=S.pop()
      print ("\t\t\tITEM " , item ," POPPED FROM THE STACK")
 def display():#code to display stack
             if (S==[]):
                        print ("\t\t\tEMPTY STACK")
             else:

                        print ("\t\t\t",)
             for i in S:
                        print( i ,' ',end="")

 #code to call all
```



```
#code to call all
functions import sys
ch=0
while(ch!=4):
        ch=menu()
        if(ch==1):

                push()
        elif(ch==2):

                pop()
        elif(ch==3):

                display()

        elif(ch==4):
           print ("\t\t\tABORTING PROGRAM        ")
           sys.exit()
```

QUESTIONS – BASED ON CBSE PATTERN

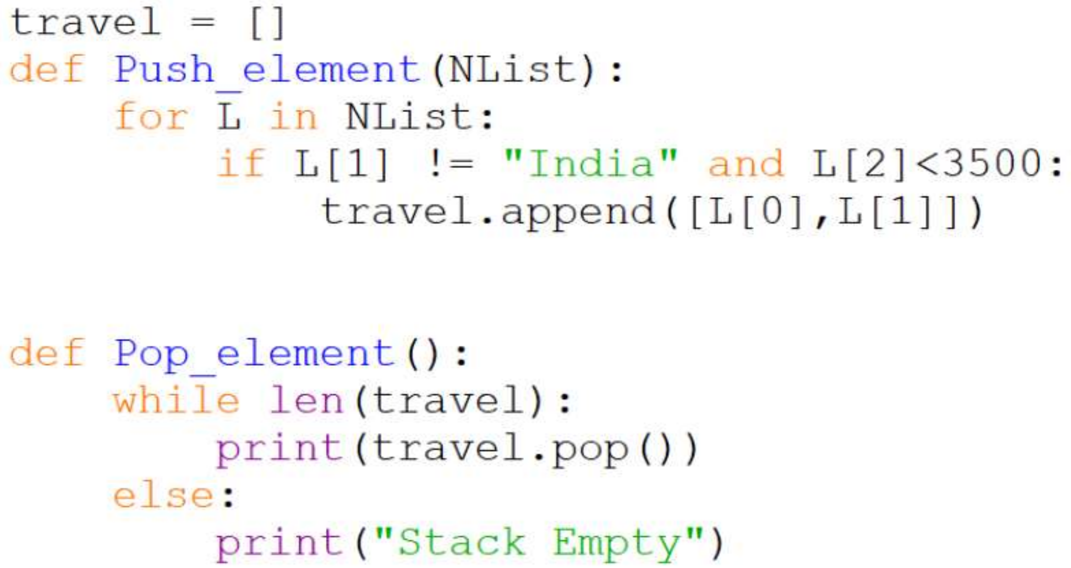
1	A list, NList contains following record as list elements: [City, Country, distance from Delhi] Each of these records are nested together to form a nested list. Write the following user defined functions in Python to perform the specified operations on the stack named travel**.** **(i) Push_element(NList):** It takes the nested list as an argument and pushes a list object containing name of the city and country, which are not in India and distance is less than 3500 km from Delhi. **(ii) Pop_element():** It pops the objects from the stack and displays them. Also, the function should display “Stack Empty” when there are no elements in the stack. For example: If the nested list contains the following data: NList=[["New York", "U.S.A.", 11734], ["Naypyidaw", "Myanmar", 3219], ["Dubai", "UAE", 2194], ["London", "England", 6693], ["Gangtok", "India", 1580], ["Columbo", "Sri Lanka", 3405]] The stack should contain: ['Naypyidaw', 'Myanmar'], ['Dubai', 'UAE'],	3

['Columbo', 'Sri Lanka']

The output should be:

```
['Columbo', 'Sri Lanka']
['Dubai', 'UAE']
['Naypyidaw', 'Myanmar']
Stack Empty
```

Ans



```
travel = []
def Push_element(NList):
    for L in NList:
        if L[1] != "India" and L[2]<3500:
            travel.append([L[0],L[1]])


def Pop_element():
    while len(travel):
        print(travel.pop())
    else:
        print("Stack Empty")
```

2 Write a function in Python Popstack (names), where **names** is a stack implemented by a list of names. The function returns the name deleted from the stack. 3

Ans

```
def popStack(names) :
    L = len(names)
    if L==0:
        print("Stack Underflow")
    else:
        val = names[L-1]
        names.pop()
    return val
res=popStack(['Harikaran','Balaji','Nikhil','Cathrine'])
print('Value returned:',res)
```

Note : Any correct code giving the same result is also accepted

3.	A stack S contains Name of countries.Write a function POP(A), to remove Name from stack. Function should handle underflow condition and should return removed element.	3

Ans	def POP(A):
	    if len(A) = = 0:
	        print(“UNDERFLOW”)
	    else:
	        val=A.pop( )
	    return val	

4.	Write a function PUSH(STACK,Name) to add new name to the STACK data structure . Also display all the names having at least 4 characters from the stack.	3

Ans	def PUSH(STACK,Name):
	        STACK.append(Name)
	        #code for displaying
	        if len(STACK)==0:
	                print(“Empty stack”)
	        else:
	                top=len(STACK)-1
	                for i in range(top,-1,-1):
	                        if len(STACK[i])>=4:
	                                print(STACK[i])	

5.	**UNSOLVED EXERCISE:** Coach Abhishek stores the races and participants in a dictionary. Write a program, with separate user defined functions to perform the following operations: a. Push() Push the names of the participants of the dictionary onto a stack, where the distance is more than 100. b. PoP() Pop and display the content of the stack. For example: If the sample content of the dictionary is as follows: Races ={100:'Varnika', 200 :'Jugal', 400:'Kushal', 800:'Minisha'}} The output from the program should be: Minisha Kushal Jugal	3

SAMPLE QUESTIONS:

MCQ/Very Short Answer Type Questions(1-mark)

1. Choose the correct output for the following stack operation(* top position)

Push
(5)
Push
(8)
Pop()
Push
(2)
Push
(5)
Pop()
Push
(1)

(a) 8 5 2 5 1*
(b) 8 5 5 2 1*
(c) 2 5 5 1*
(d) 5 2 1*

2. Which list method can be used to perform Push operation in a stack implemented by list?

a) append()
b) extend()
c) push()
d) insert()

3. Which list method can be used to perform Pop operation in a stack implemented by list?

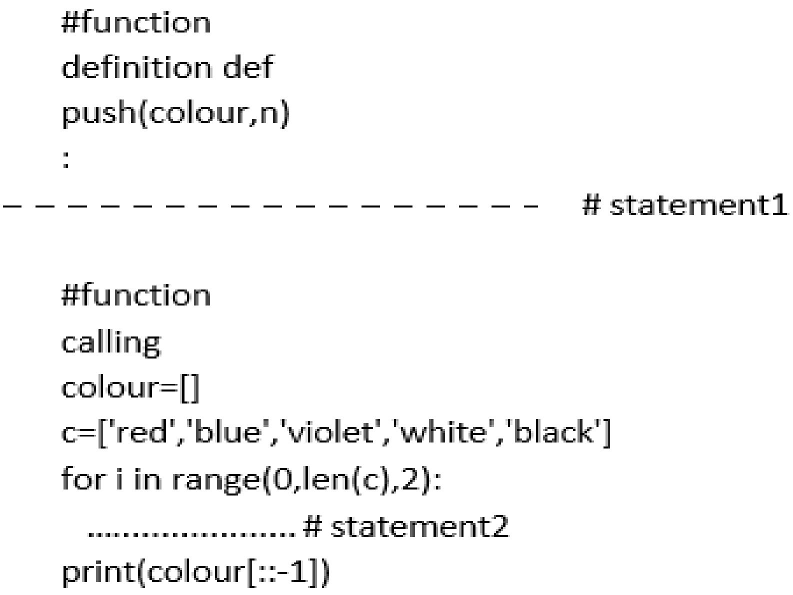
a) pop()
b) pop(1)
c) remove()
d) pop(0)

4. Consider the following operation performed on a stack of size 3, What will be the output? (* top position)

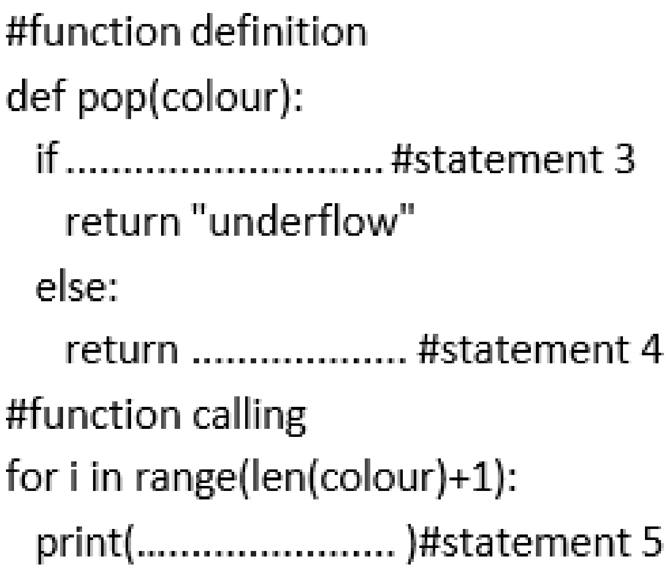
Push(10)
Push(20)
Push(30)
Pop()
Push(40)
Push(50)

(a) overflow
(b) underflow
(c) 10 20 30 40 50*
(d) 10 20 40 50*

5. Based on the below given code, Write answer to the following questions i to v



```
    #function
    definition def
    push(colour,n)
    :
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _    # statement1

    #function
    calling
    colour=[]
    c=['red','blue','violet','white','black']
    for i in range(0,len(c),2):
      ..................... # statement2
    print(colour[::-1])
```



```
#function definition
def pop(colour):
  if ............................ #statement 3
    return "underflow"
  else:
    return ................... #statement 4
#function calling
for i in range(len(colour)+1):
  print(...................... )#statement 5
```

(i) Identify the suitable code for statement 1?

a) colour.insert(len(colour),n)
b) colour.append(len(colour),n)
c) colour.append()
d) colour.extend()

(ii) Identify the suitable code for statement 2?

a) push(colour,c[i])
b) push(colour)
c) push(c[i])

d)push(colour,i)

(iii) Identify the suitable code for statement 3?

a)colour==[]:
b)colour.isEmpty():
c) len(colour)=0:
d)None of the above

(iv) Fill in the statement to delete an element from the stack?

a) colour.pop(1)
b) colour.pop()
c) del colour[1]
d) colour.delete(1)

(v) Fill the statement 5,to call pop function
a) pop(c)
b)pop(colour)
c) call pop(colour)
d)def pop(colour)

6. What do you mean by Data Structure?
7. LIFO data structure is?
8. Can we have nested list?
9. Name one linear data structure.
10. Name one non-linear data structure.
11. Name the operation for insertion in a stack.
12. Name the operation for deletion from a stack.
13. Name the function to find length of a list.
14. Indexing in list starts from?

Short Answer Type Questions(2-marks)

1. How is Data Structure different from Data Type?
2. Define Stack and Queue
3. Name some operations commonly performed on data structures?
4. What is a list?
5. What is traversing? Write python code to traverse a list.
6. Name the methods used for inserting and deleting elements from a list.
7. Write some applications of stack.

Application based Short Answer Type Questions(2-marks)

8. Predict the output with respect to the list L=[40,20,30,10,50]

(a) print(L)
(b) print(len(L))
(c) L.pop() ; print(L)
(d) L.append(70); print(L)
(e) L.sort(); print(L)

9. Find the output:

(a) secondlist=[1,2,3,[4,5,[6,7,8],9],10,11]
print(len(secondlist))

L=[1, 2, 3, [4, 5, [6, 7, 8], 9] ,10, 11]

(b) L[1]
(c) L[3]
(d) L[3][1]
(e) L[3][2][0]
(f) L[3][2]
(g) L[3][2][1]
(h) L[3][3]

10. Predict the output:

(a) b=[[9,6],[4,5],[7,7]] X=b[:2]
X.append(10)
print(X)

(b) b=[[9,6],[4,5],[7,7]] X=b[:2]
X[1].append(10)
print(X)

11. Consider STACK=['a','b','c','d']. Write the STACK content after each operations:

a) STACK.pop( )
b) STACK.append('e')
c) STACK.append('f')
d) STACK.pop( )

12. Write a program to implement a stack for the students(studentno, name). Just implement Push.

13. Write a program to implement a stack for the students(studentno, name). Just implement Pop and display.

14. If L=["Python", "is", "a", ["modern", "programming"], "language", "that", "we", "use"] , then find the output:

a) L[0][0]

b) L[3][0][2]

c) L[3:4][0]

d) L[3:4][0][1]

e) L[3:4][0][1][3]

f) L[0:9][0]

g) L[0:9][0][3]

h) L[3:4][1]

15. What is the difference between pop() and pop(0)?

Long Answer Type Questions (3-marks)

1. Vedika has created a dictionary containing names and marks as key-value pairs of 5 students. Write a program, with separate user-defined functions to perform the following operations:

Push the keys (name of the student) of the dictionary into a stack, where the corresponding value (marks) is greater than 70.

Pop and display the content of the stack.

The dictionary should be as follows:

d={“Ramesh”:58, “Umesh”:78, “Vishal”:90, “Khushi”:60, “Ishika”:95}

Then the output will be: Umesh Vishal Ishika

2. Write a function in Python Push (nums), where **nums** is a list of numbers. From this list push all odd numbers into a stack implemented by using a list. Display the stack if it has at least one element, otherwise display appropriate error message.

3. Write PushOn(Book) and Pop(Book) methods/functions in Python to add a new Book and delete a Book from a list of Book titles, considering them to act as push and
pop operations of the Stack data structure.

4. Write functions in python for performing stack operations implemented by using list and also write a function for the menu driven operations to implement all the stack operations? (Hint: Use global variables)

5. Write a function in python to perform Push Operation in a stack implemented by using list. Maximum size of the stack can be input by the user along with original stack and also display the top position and stack elements after push operation.

```
                    >>> li=[3,4,5]
                    >>> push(li,5)
                    enter the element:20
Top position after Push operation: 3 stack after Push operation: [20, 5, 4, 3]
```

6. Write a function in python to perform Pop Operation in a stack implemented by using list. Print the element deleted, top position and also display the stack elements before and after the pop operation?

```
>>> li=[20,5,4,3]
>>> pop(li)
Original stack [3, 4, 5, 20]
Top position: 3
Deleted element 3
stack after Pop Operation: [4, 5, 20]
```

7. Write a function in python to display elements of a stack implemented by using list. Use both traditional and python methods. Print stack empty message, if elements are not there.

```
>>> li=[20,5,4,3]
>>> display(li)
Stack elements: [3, 4, 5, 20]
>>> displaytraditional(li) Stack elements:
3 4 5 20
>>> l2=[]
>>> display(l2) Stack empty
```

8. Write a function in Python PUSH(Arr),where Arr is a list of numbers, from this list push all even numbers into a stack implemented by using a list. Display the stack if it has at least one element, otherwise display “stack empty” message.

```
                    >>> li=[1,6,89,100,25,29]
                    >>> push(li)
```

The item 1 can't be inserted because it is not an even number The item 89 can't be inserted because it is not an even number The item 25 can't be inserted because it is not an even number The item 29 can't be inserted because it is not an even number

Stack elements after push operation : [100, 6]

9. Write a function in Python PUSH(mydict,maxsize=5),where mydict is a dictionary of phone book(name and mobile numbers), from this dictionary push all phone numbers to a stack implemented by using list and also the display the numbers that Display the stack if it has at least one element, otherwise display “stack empty” message

```
          >>>mydict={1234567890:"Shyam",94567892:"Ram",8657456789012:
          "Karun",9674123789:"Tharun"}
          >>> push(mydict)
```

Stack elements after push operation :
[[1234567890, 94567892, 8657456789012, 9674123789]]

10. Write a function in Python PUSH(mydict),where mydict is a dictionary of phone book(name and mobile numbers), from this dictionary push only phone numbers having last digit is greater than 5 to a stack implemented by using list and also display the stack if it has at least one element, otherwise display “stack empty” message.

```
>>> mydict={9446789123:"Ram",8889912345:"Sam",7789012367:"Sree"}
>>> push(mydict)
```

Phone number: 9446789123 last digit is less than five which can't be pushed Stack elements after push operation : [7789012367, 8889912345]

11. Write a function in Python PUSH(mydict),where mydict is a dictionary of phone book(name and mobile numbers), from this dictionary push only phone numbers having 10 digits into a stack implemented by using list . Display the stack if it has at least one element, otherwise display “stack empty” message.

```
>>>mydict={1234567890:"Shyam",94567892:"Ram",8657456789012:"Karun",
9674123789:"Tharun"}
>>> push(mydict)
```

Digit of phone number 8
Phone number: 94567892 doesn't have 10 digits,which can't be pushed Digit of phone number 13
Phone number: 8657456789012 doesn't have 10 digits,which can't be pushed Stack elements after push operation : [9674123789, 1234567890]

12. Write a function in Python PUSH(Arr),where Arr is a list of numbers, From this list push all numbers divisible by 5 in to a stack implemented by using a list. Display the stack if it has at least one element, otherwise display appropriate error message.

```
>>> li=[10,2,5,6,15,30]
>>> push(li)
```

The item 2 can't be inserted because it is not divisible by 5 The item 6 can't be inserted because it is not divisible by 5 Stack elements: [30, 15, 5, 10]

13. Write a function in Python POP(Arr),where Arr is a stack implemented by a list of numbers. The function returns the value deleted from the stack and stack status after pop operation.

```
>>> li=[10,20,30,40,50]
>>> dele,st=pop(li)
Original stack [50, 40, 30, 20, 10]
>>> print("Element deleted=",dele,"Stack after deletion",st[::-1])
Element deleted= 50
Stack after deletion [40, 30, 20, 10]
```

Or

```
>>> pop(li)
Original stack [40, 30, 20, 10]
    (40, [10, 20, 30])
```

ANSWER KEY

I. MCQ/ Very Short Answer Type Questions(1-mark)

1. (d) 5 2 1*
2. (a) append()
3. (a) pop()
4. (a) overflow
5.
   (i). a colour.insert(len(colour),n)
   (ii). a push(colour,c[i]) (iii).a colour==[]
   (iv). b colour.pop()
   (v). b pop(colour)
6. Data Structure means organization of data. A data structure has well defined operations or behaviour.
7. STACK
8. Yes
9. Lists
10. Graphs

11 PUSH

12 pop

13. len()
14. 0

Short Answer Type Questions(2-marks)

1. Data Structure provides information regarding organization of data whereas Data Type provides information regarding the domain of values and operations that can be performed on data.
2. Stack – A stack is a linear list also known as LIFO list with the special property that items can be added or removed from only one end called the top.

Queue – A queue is a linear list also known as FIFO list with the special property that items can be added at one end and removed from the other.

3. Traversal, Insertion, Deletion, Searching, Sorting, Merging etc.
4. A list is a mutable sequence of data elements indexed by their position. A list is represented using [ ] . e.g L=[10,20,30]
5. Traversing means accessing or visiting or processing each element of any data structure.

```
 L=[10,20,30,40,50]
 for x in L:
        print(x)
```

6. Various methods for inserting elements in a list are - insert(), append(), extend() and methods used for deleting items from a list are – pop() , remove(), clear()

7. Reversing a string, compilers use stack to store previous state of program, undo mechanism in text editors and backtracking.

Application based Short Answer Type Questions(2-marks)

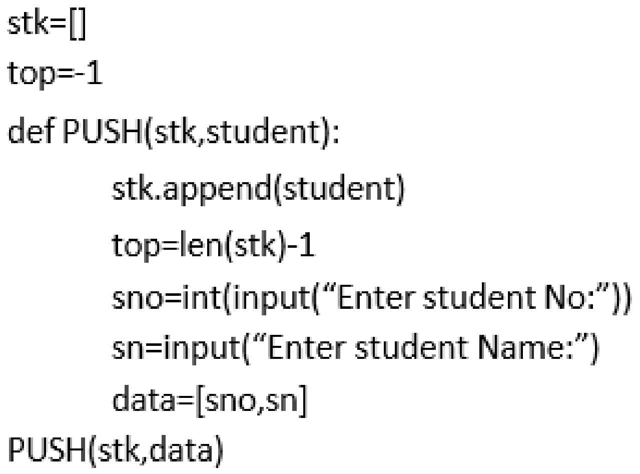
8. (a) Ans: [40, 20, 30, 10, 50]
   (b) Ans: 5
   (c) Ans:50
   [40, 20, 30, 10]
   (d) Ans: [40, 20, 30, 10, 70]
   (e) Ans: [10, 20, 30, 40, 70]

9. (a) Ans:6
   (b) Ans: 2
   (c) Ans:[4, 5, [6, 7, 8], 9]
   (d) Ans:5
   (e) Ans:6
   (f) Ans: [6, 7, 8]
   (g) Ans:7
   (h) Ans:9

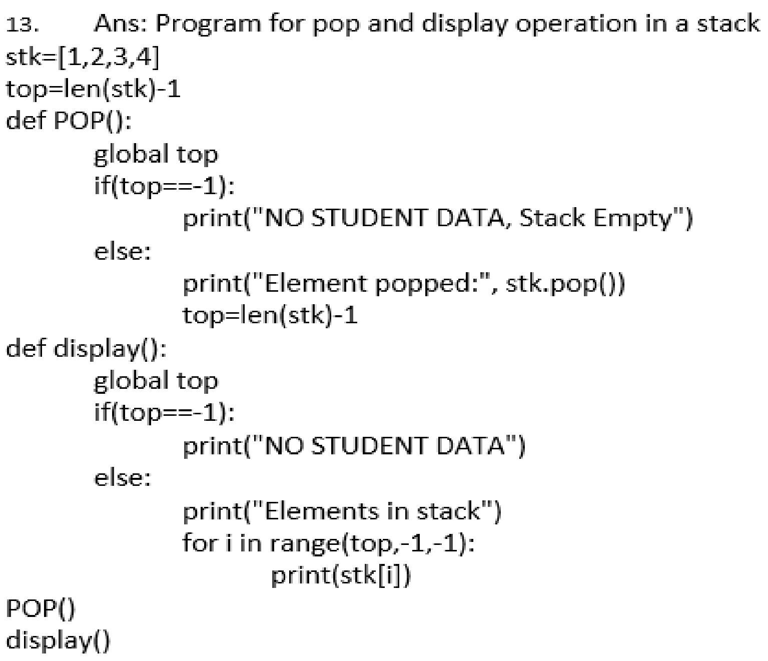
10. (a) Ans: [[9, 6], [4, 5], 10]
    (b) Ans: [[9, 6], [4, 5, 10]]

11. Ans: ['a', 'b', 'c']
    Ans: ['a', 'b', 'c',’e’]
    Ans: ['a', 'b', 'c',’e’,’f’]
    Ans: ['a', 'b', 'c',’e’]

12.



```
stk=[]
top=-1
def PUSH(stk,student):
        stk.append(student)
        top=len(stk)-1
        sno=int(input(“Enter student No:”))
        sn=input(“Enter student Name:”)
        data=[sno,sn]
PUSH(stk,data)
```



```
13.     Ans: Program for pop and display operation in a stack
stk=[1,2,3,4]
top=len(stk)-1
def POP():
        global top
        if(top==-1):
                print("NO STUDENT DATA, Stack Empty")
        else:
                print("Element popped:", stk.pop())
                top=len(stk)-1
def display():
        global top
        if(top==-1):
                print("NO STUDENT DATA")
        else:
                print("Elements in stack")
                for i in range(top,-1,-1):
                        print(stk[i])
POP()
display()
```

14. (a) Ans: ‘P’

(b) Ans: ‘d’

(c) Ans: ['modern', 'programming']

(d) Ans: 'programming'

(e) Ans: 'g'

(f) Ans: 'Python'

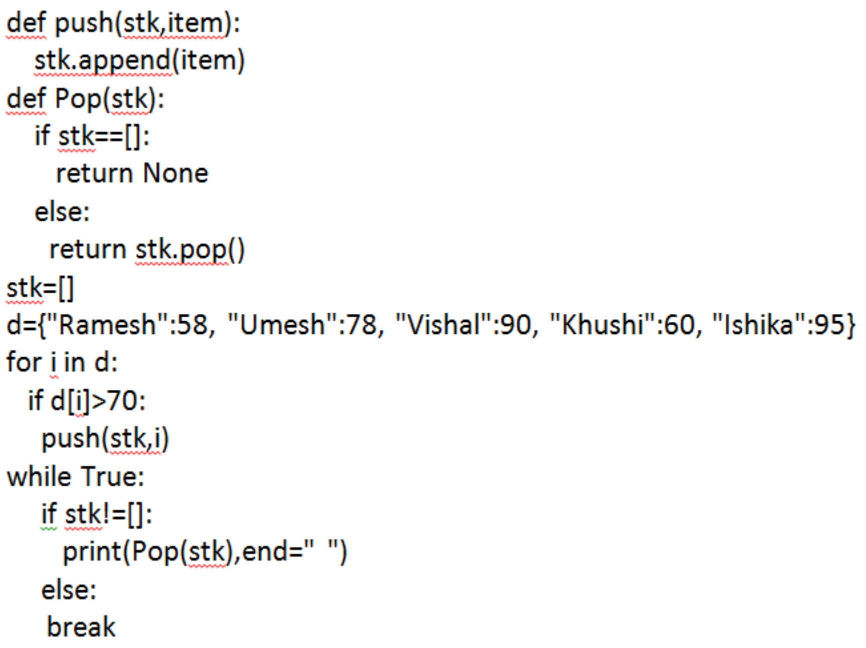
(g) Ans: 'h'

(h) Ans: IndexError: list index out of range

15. Ans: pop() will delete the last element of a list whereas pop(0) will delete element at index zero of a list

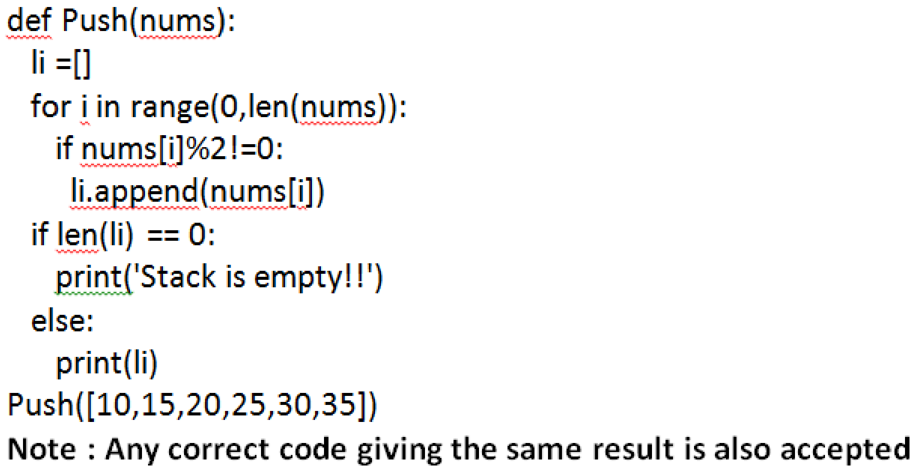
Long Answer Type Questions (3-marks)

1.



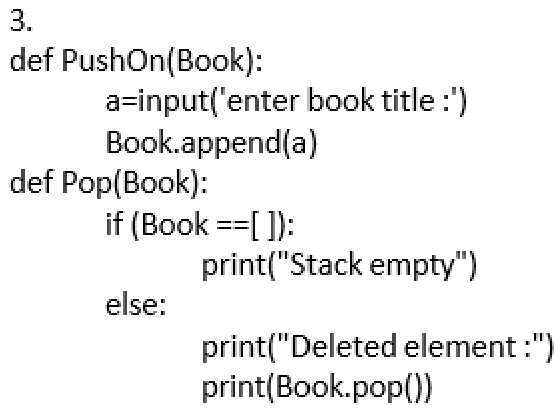
```
def push(stk,item):
   stk.append(item)
def Pop(stk):
   if stk==[]:
     return None
   else:
    return stk.pop()
stk=[]
d={"Ramesh":58, "Umesh":78, "Vishal":90, "Khushi":60, "Ishika":95}
for i in d:
  if d[i]>70:
    push(stk,i)
while True:
   if stk!=[]:
     print(Pop(stk),end=" ")
   else:
    break
```

2.



```
def Push(nums):
  li =[]
  for i in range(0,len(nums)):
    if nums[i]%2!=0:
      li.append(nums[i])
  if len(li) == 0:
    print('Stack is empty!!')
  else:
    print(li)
Push([10,15,20,25,30,35])
```

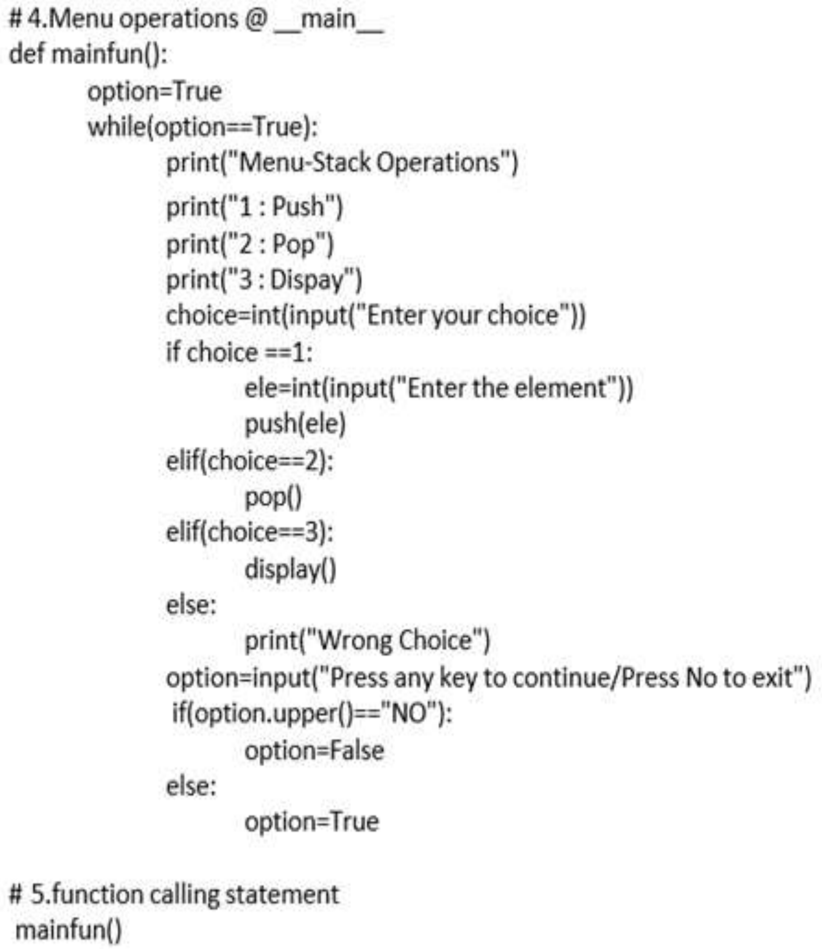
Note : Any correct code giving the same result is also accepted



3.

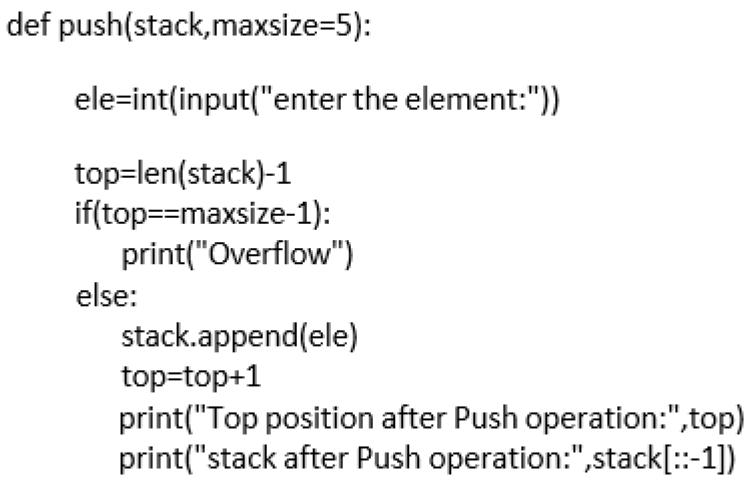
```
def PushOn(Book):
        a=input('enter book title :')
        Book.append(a)
def Pop(Book):
        if (Book ==[ ]):
                print("Stack empty")
        else:
                print("Deleted element :")
                print(Book.pop())
```

```
4
# declaration of global variables
stack=list()
maxsize=5
# 1.Push operation
def push(ele):
        top=len(stack)-1
        if(top==maxsize-1):
                print("Overflow")
        else:
                stack.append(ele)
# 2.Pop operation
def pop():
        top=len(stack)-1
        if(top== -1):
                print("Underflow")
        else:
                print("Deleted element",stack.pop())
# 3.Display operation
def display():
        if(stack==[]):
                print("Stack empty")
        else:
                print("Stack  elements",stack[::-1])
```



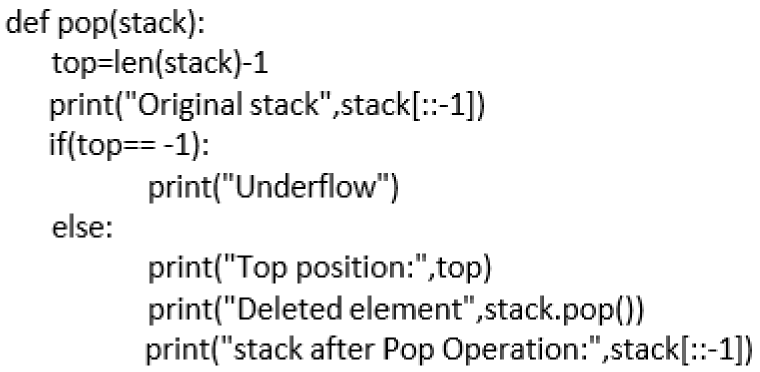
```
# 4.Menu operations @ __main__
def mainfun():
        option=True
        while(option==True):
                print("Menu-Stack Operations")
                print("1 : Push")
                print("2 : Pop")
                print("3 : Dispay")
                choice=int(input("Enter your choice"))
                if choice ==1:
                        ele=int(input("Enter the element"))
                        push(ele)
                elif(choice==2):
                        pop()
                elif(choice==3):
                        display()
                else:
                        print("Wrong Choice")
                option=input("Press any key to continue/Press No to exit")
                 if(option.upper()=="NO"):
                        option=False
                else:
                        option=True

# 5.function calling statement
 mainfun()
```

5. #Stack operation-Push



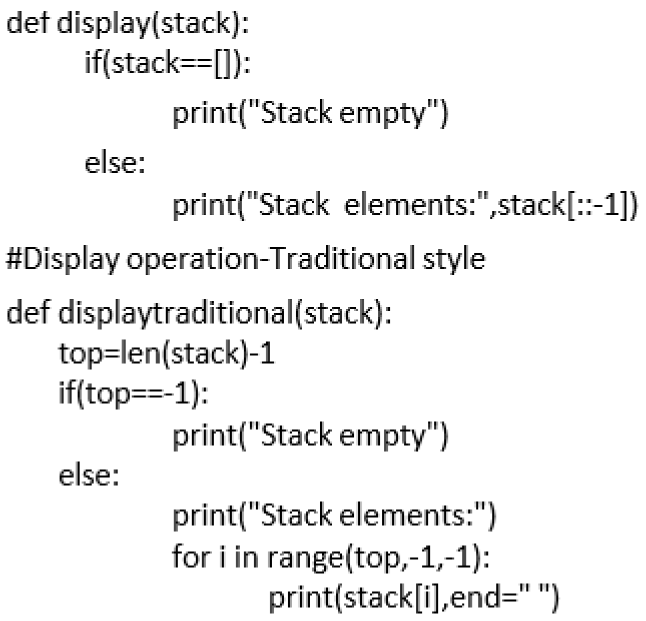
```
def push(stack,maxsize=5):

        ele=int(input("enter the element:"))

        top=len(stack)-1
        if(top==maxsize-1):
              print("Overflow")
        else:
              stack.append(ele)
              top=top+1
              print("Top position after Push operation:",top)
              print("stack after Push operation:",stack[::-1])
```

6. #stack operation-Pop



```
def pop(stack):
      top=len(stack)-1
      print("Original stack",stack[::-1])
      if(top== -1):
                  print("Underflow")
      else:
                  print("Top position:",top)
                  print("Deleted element",stack.pop())
                  print("stack after Pop Operation:",stack[::-1])
```

7. Ans: Program for pop and display operation in a stack



```
def display(stack):
          if(stack==[]):
                      print("Stack empty")
          else:
                      print("Stack  elements:",stack[::-1])
#Display operation-Traditional style
def displaytraditional(stack):
       top=len(stack)-1
       if(top==-1):
                      print("Stack empty")
       else:
                      print("Stack elements:")
                      for i in range(top,-1,-1):
                                  print(stack[i],end=" ")
```

8.

```
def push(Arr):
    stack=list()
    maxsize=5
    for ele in Arr:
        top=len(stack)-1
        if(top==maxsize-1):
            print("Overflow")
        elif(ele%2==0):
            stack.append(ele)
        else:
            print("The item",ele," can't be inserted because it is not an even number")
    if(stack==[]):
        print("Stack empty")
    else:
    print("Stack elements after push operation :",stack[::-1])
```

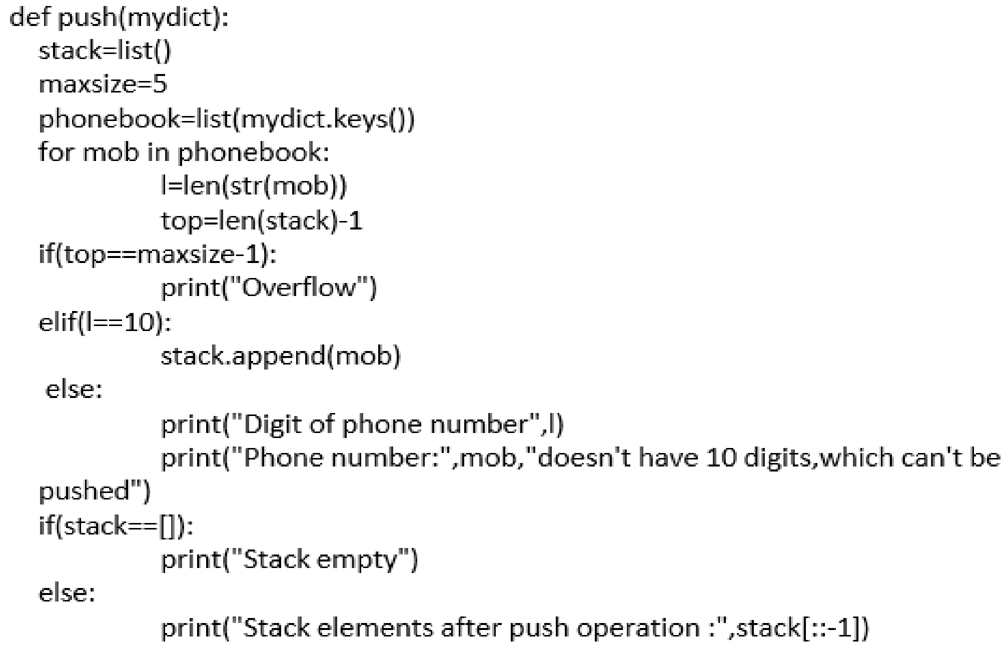
9.

```
def push(mydict,maxsize=5):
    stack=list()
    phonebook=list(mydict)
    top=len(stack)-1
    if(top==maxsize-1):
        print("Overflow")
    else:
        stack.append(phonebook)
    if(stack==[]):
        print("Stack empty")
    else:
        print("Stack elements after push operation :",stack[::-1])
```

10.

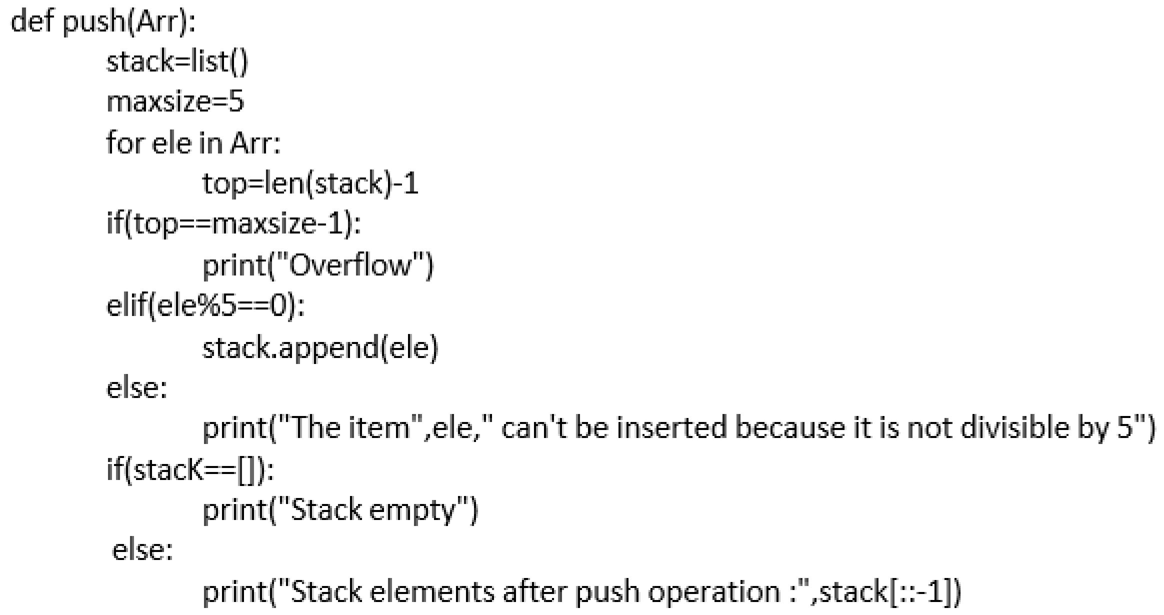
```
def push(mydict):
    stack=list()
    maxsize=5
    phonebook=list(mydict.keys())
    for mob in phonebook:
        lastdig=mob%10
        top=len(stack)-1
        if(top==maxsize-1):
            print("Overflow")
        elif(lastdig>=5):
            stack.append(mob)
        else:
            print("Phone number:",mob,"last digit is less than five which can't be pushed")
    if(stack==[]):
        print("Stack empty")
    else:
    print("Stack elements after push operation :",stack[::-1])
```

11.



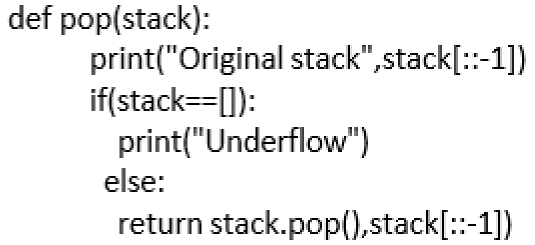
```
def push(mydict):
    stack=list()
    maxsize=5
    phonebook=list(mydict.keys())
    for mob in phonebook:
                l=len(str(mob))
                top=len(stack)-1
    if(top==maxsize-1):
                print("Overflow")
    elif(l==10):
                stack.append(mob)
     else:
                print("Digit of phone number",l)
                print("Phone number:",mob,"doesn't have 10 digits,which can't be
    pushed")
    if(stack==[]):
                print("Stack empty")
    else:
                print("Stack elements after push operation :",stack[::-1])
```

12



```
def push(Arr):
            stack=list()
            maxsize=5
            for ele in Arr:
                        top=len(stack)-1
            if(top==maxsize-1):
                        print("Overflow")
            elif(ele%5==0):
                        stack.append(ele)
            else:
                        print("The item",ele," can't be inserted because it is not divisible by 5")
            if(stacK==[]):
                        print("Stack empty")
             else:
                        print("Stack elements after push operation :",stack[::-1])
```

13



```
def pop(stack):
          print("Original stack",stack[::-1])
          if(stack==[]):
            print("Underflow")
           else:
            return stack.pop(),stack[::-1])
```

COMPUTER NETWORKS

INTRODUCTION

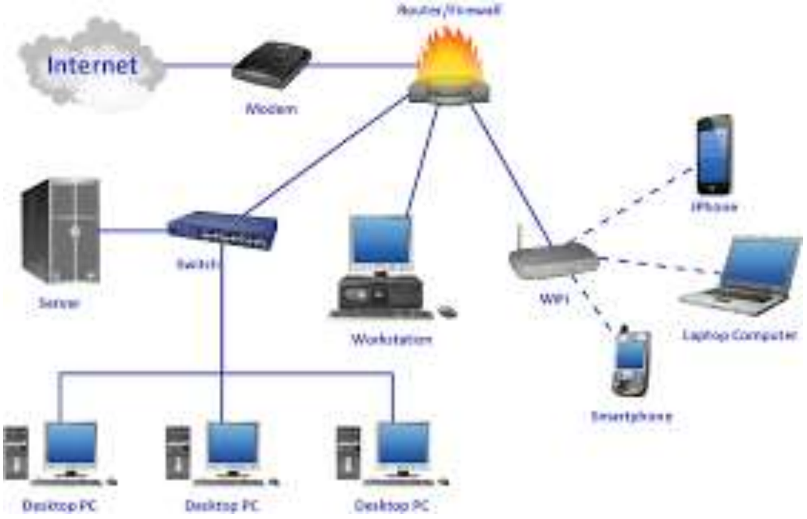
We are living in a connected world. Information is being produced, exchanged, and traced across the globe in real time. It's possible as almost everyone and everything in the digital world is interconnected through one way or the other.

A group of two or more similar things or people interconnected with each other is called network. Some of the examples of network in our everyday life includes:

- Social network
- Mobile network
- Network of computers
- Airlines, railway, banks, hospitals networks

What is Computer Network?

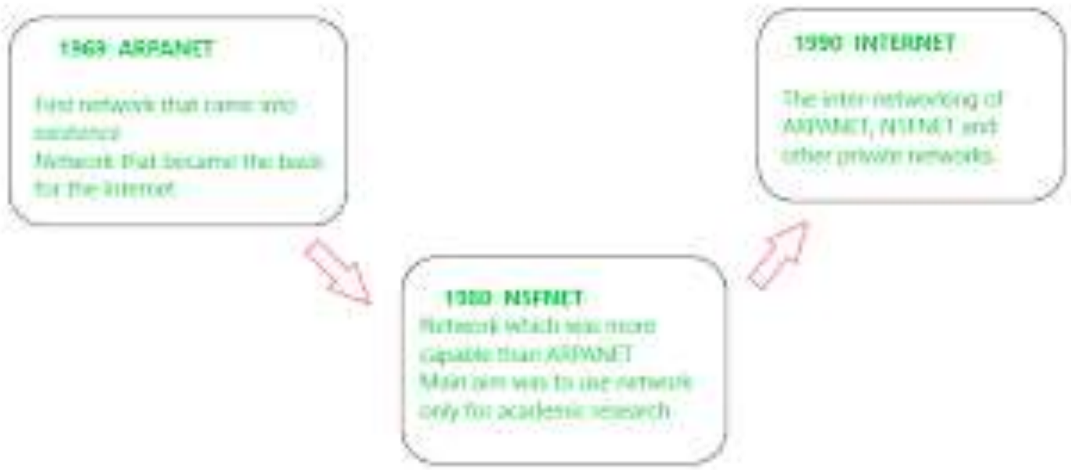
A Computer network is an interconnection among two or more computers or computing devices. Such interconnection allows computers to share data and resources among each other Resources such as files, applications, printers and software are common information shared in a networking.



The advantage of networking is

- Resource Sharing
- Collaborative Interaction
- Cost Saving
- Increased storage
- Time Saving

Evolution of Network:


1969 ARPANET
First network that came into existence
Network that became the basis for the internet
1980 NSFNET
Network which was more capable than ARPANET
Main aim was to use network only for academic research
1990 INTERNET
The inter-networking of ARPANET, NSFNET and other private networks.

(I)ARPANET (Advanced Research Project Agency Network)

- It came into existence in 1960s
- A project for interconnecting, US department of defense with academic and research organization across different places for scientific collaboration.

(II)NSFNET (National Science Foundation Networks)

- It came into existence in 1986
- It was the first large-scale implementation of Internet technologies in a complex environment of many independently operated networks

(III) INTRANET

- It is a local or restricted communication system
- It is managed by a person or organization.
- Intranet users can avail services from internet but Internet user cannot access intranet directly

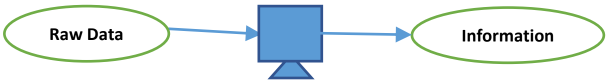
(III) INTERNET

- It came into existence in 1960s
- It is known as Network of Networks
- A global computer network providing variety of information and communication facilities consisting of interconnected networks using standardized communication protocols.

DATA COMMUNICATION TERMINOLOGIES

DATA

Data means information in digital form which is stored processed and exchanged between digital devices like computer, mobile phones or laptop.
Data can be text, image, audio, video or multimedia files.
Computers stores raw data and process these data into meaningful information. Hence, we can define Information as processed data.



COMMUNICATION

The exchange of information between two or more networked or interconnected devices is called communication. These devices must be capable of sending /receiving data over a communication medium.

COMPONENTS OF DATA COMMUNICATION

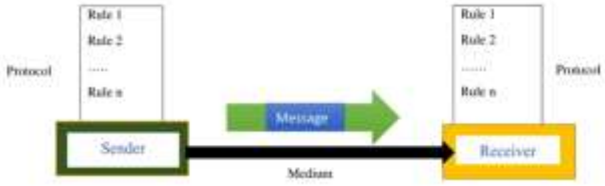
The five main components of data communication are as follows:
SENDER: Sender is a device which is capable of sending data over a communication network. In data communication Sender is also called Source.

RECEIVER: Receiver is a device which is capable of receiving data over a communication network. In data communication Receiver is also called Destination.

MESSAGE: message is the information being exchanged between a sender and a receiver over a communication network.

COMMUNICATION MEDIUM: Communication medium is the path or channel through which the information is moved from the sender to the receiver. A communication medium can be either wired/guided or wireless/unguided.

PROTOCOLS: The set of standard rules which are followed in data communication are known as Data Communication Protocols. All the communicating devices like sender receiver and other connected devices in the network should follow these protocols.



Why Protocols are needed?

The communicating devices may be in different geographical areas. The speed of these devices may be different. Also, the data transfer rates of different networks may be different. These complexities make it necessary to have a common set of rules to ensure the secure communication of data.

Examples of some commonly used Protocols in data communication are given below:

- Transmission Control Protocol (TCP)
- Internet Protocol (IP)
- File Transfer Protocol (FTP)
- Simple Mail Transport Protocol (SMTP)
- Hyper Text Transfer Protocol (HTTP)

MEASURING CAPACITY OF COMMUNICATION MEDIA

Capacity of a communication channel means the maximum quantity of signals that a communication channel can carry. The capacity of a communication medium is measured by its bandwidth and data transfer rate.

BANDWIDTH

Bandwidth is the difference between the highest and lowest frequencies a transmission media can carry.

The unit of bandwidth is Hertz.

DATA TRANSFER RATES

Data transfer rate is the number of bits transmitted through a channel per unit of time. Data transfer rate is measured in bits per second (bps). It is also measured in Kilobits per second (Kbps), Megabits per second (Mbps) or Gigabits per second (Gbps).

IP ADDRESS

IP address or Internet Protocol address is a unique numeric address assigned to every device connected to a network. It uniquely identifies every node connected to a local network or internet.

An IP address allows computers to send and receive data over the internet. They can also be used to track down a user's physical location.There are two versions for IP address IPV4 and IPV6. IP addresses are binary numbers but are typically expressed in decimal form (IPv4) or hexadecimal form (IPv6) to make reading and using them easily.

The commonly used IP address is IPV4. An IPv4 address consists of four numbers, each of which contains one to three digits, with a single dot (.) separating each set of digits. Each of the four numbers can range from 0 to 255.

Example IP address: 24.171.248.170

SWITCHING TECHNIQUES

In large networks, there may be more than one paths for transmitting data from sender to receiver. The process of selecting a path of data out of the available paths is called switching. There are two popular switching techniques – circuit switching and packet switching.

1. Circuit Switching

In circuit switching, whenever a source end node wants to send a message to the destination end node a physical link is first established between the source and the destination. Then only the data transmission takes place. After the complete transmission of data this physical link is terminated.

Simple example of a circuit switching is telephone network in which a person calls another person. When the call receiving person receives the call, then only the connection is established. Then the message is conveyed and finally the connection is terminated.

Advantages:

1. Since a dedicated communication channel is set up before communicating the message, the data transmission is reliable and is suitable for long and continuous communication.

2 Circuit switching uses fixed bandwidth as well as data rates.

3. As the data is communicated continuously, no need of sequencing or re ordering it at the receiving end.

Disadvantages:

1. Time required to setup a physical connection between the sender and the receiver makes delay in communication

2. Since a communication channel is dedicated for a particular transmission, it cannot be utilized for other communication, even if the channel is free.

3. More expensive since connection has to be established every time before communication.

2. Packet Switching

In the packet switching technique, the whole message is split into small packets. Now, these packets are transmitted one by one from sender to the receiver through the intermediary switches in the network. The packets will take shortest path as possible.

Every packet will have a sequence number in order to identify their order at the receiving end. The packets will also contain information like source address, intermediate node address, destination address etc.

Advantages:

1. Packet switching is effective type of data transmission technique as it effectively utilizes the communication channel. Multiple users can share the channel simultaneously utilizing the bandwidth effectively.

2. It is cost effective and easy to implement compared to circuit switching.

3. As the messages are sent as small sized packets, the data transmission is quick and easy.

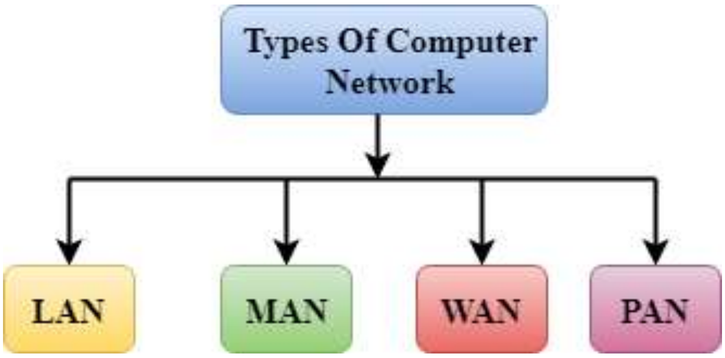
Disadvantages:

1. In packet switching the movement of packets may not be in correct order. Hence it is not suitable for voice communication.
2. Unorganized movement of packets makes it necessary to implement proper sequencing and reordering techniques.
3. As the packets flow through multiple paths from the source to the destination, complex security protocols are required to ensure reliable communication.

Computer Network Types

A computer network is a group of computers linked to each other that enables the computer to communicate with another computer and share their resources, data, and applications.

A computer network can be categorized by their size, complexity and geographical spread. A computer network is mainly of four types:



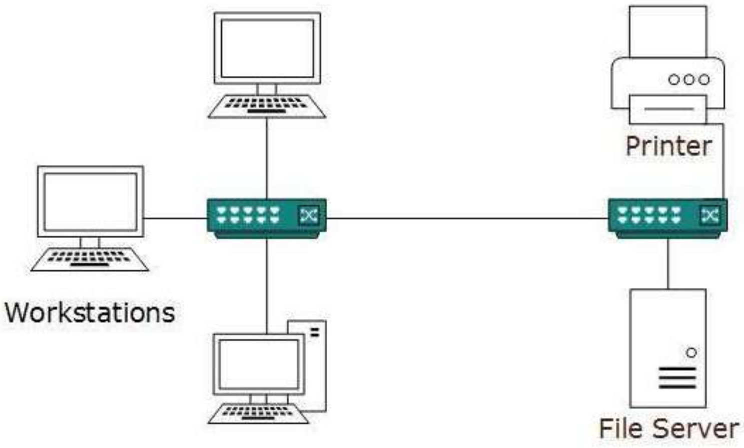
- LAN (Local Area Network)
- PAN (Personal Area Network)
- MAN (Metropolitan Area Network)
- WAN (Wide Area Network)

PAN (Personal Area Network)

- Personal Area Network is a network of information technology devices (laptop, mobile phones, media player and play stations) arranged within the range of an individual person, typically within a range of 10 meters / covers an area of 30 feet.
- Thomas Zimmerman was the first research scientist to bring the idea of the Personal Area Network.

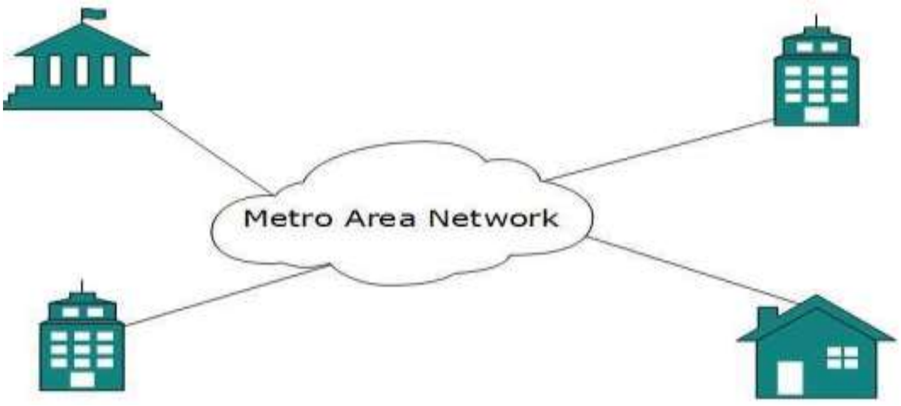
LAN (Local Area Network)

- Local Area Network is a group of computers connected to each other in a small area such as a building, office through a communication medium such as twisted pair, coaxial cable, etc to share resources. Shared resources can be data, information, programs, printers, hard-disks, modem etc. The data is transferred at an extremely faster rate, and provides higher security
- It is less costly as it is built with inexpensive hardware such as hubs, network adapters, and Ethernet cables.



MAN (Metropolitan Area Network)

- A metropolitan area network is a network that covers a larger geographic area that is spread over an area as big as a city by interconnecting different LAN to form a larger network through a telephone exchange line.
- Government agencies use MAN to connect to the citizens and private industries.
- The most widely used protocols in MAN are RS-232, Frame Relay, ATM, ISDN, OC-3, ADSL, etc.

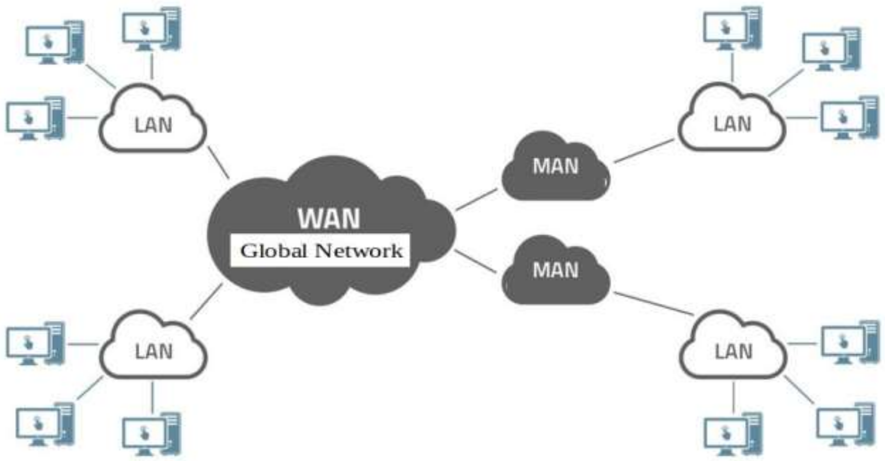


Uses of Metropolitan Area Network:

- MAN is used in communication between the banks in a city.
- It can be used in an Airline Reservation.
- It can be used in a college within a city.
- It can also be used for communication in the military.

WAN (Wide Area Network)

- A Wide Area Network is a network that extends over a large geographical area such as states or countries through a telephone line, fiber optic cable or satellite links.
- The internet is one of the biggest WAN in the world.
- A Wide Area Network is widely used in the field of Business, government, and education.



Advantages of Wide Area Network:

Following are the advantages of the Wide Area Network:

- Geographical area: A Wide Area Network provides a large geographical area. Suppose if the branch of our office is in a different city then we can connect with them through WAN. The internet provides a leased line through which we can connect with another branch.
- Centralized data: In case of WAN network, data is centralized. Therefore, we do not need to buy the emails, files or back up servers.
- Get updated files: Software companies work on the live server. Therefore, the programmers get the updated files within seconds.
- Exchange messages: In a WAN network, messages are transmitted fast. The web application like Facebook, WhatsApp, Skype allows you to communicate with friends.
- Sharing of software and resources: In WAN network, we can share the software and other resources like a hard drive, RAM.
- Global business: We can do the business over the internet globally.
- High bandwidth: If we use the leased lines for our company then this gives the high bandwidth. The high bandwidth increases the data transfer rate which in turn increases the productivity of our company.

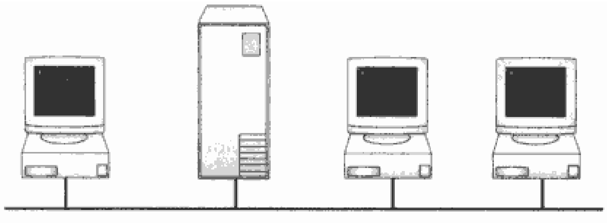
Disadvantages of Wide Area Network:

- Security issue: A WAN network has many security issues as compared to LAN and MAN network as all the technologies are combined together that creates the security problem.
- Needs Firewall & antivirus software: The data is transferred on the internet which can be changed or hacked by the hackers, so the firewall needs to be used. Some people can inject the virus in our system so antivirus is needed to protect from such a virus.
- High Setup cost: An installation cost of the WAN network is high as it involves the purchasing of routers, switches.
- Troubleshooting problems: It covers a large area so fixing the problem is difficult.

Networking Topologies

Bus

The "common bus" topology is often referred to as a "linear bus". This topology belongs to the simplest and most widespread topologies. It uses a single cable, called a trunk or segment, along which all the computers of the network are connected

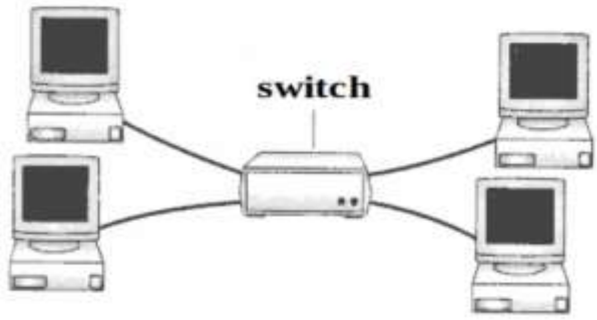


Disadvantage:

1. If the segment is broken the network comes to a standstill and devices cannot interact with each other.

Star

All computers are connected using cable segments to a central component called a switch. The signals from the transmitting computer go through the switch to all the others.



able connection and network configuration management are centralized.

Disadvantages:

1. Since all computers are connected to a central point, cable consumption increases significantly for large networks.

2. If the central component or switch fails, the entire network will be disrupted.

3. If only one computer fails (or the cable connecting it to the switch), then only this computer will not be able to transmit or receive data over the network. The rest of the computers on the network will not be affected.

Advantages:

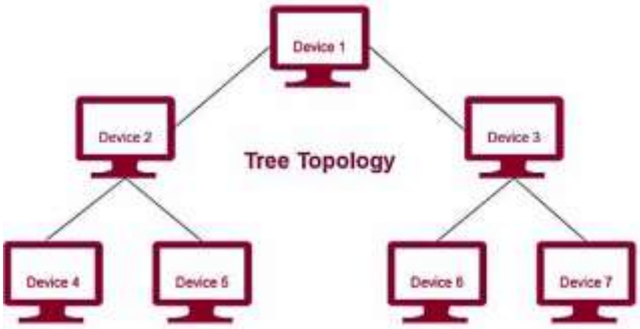
1. Each device requires just one port i.e. to attach to the switch.

2. No disruptions to the network when connecting or removing devices.

3. Robust in nature

Tree

Tree Topology is a topology which is having a tree structure in which all the computers are connected like the branches which are connected with the tree.



Disadvantages:

1. Due to the presence of a large number of nodes, the network performance of tree topology becomes a bit slow.

2. Requires a large number of cables compared to star and bus topology.

3. The establishment cost is very high.

Advantages:

1. Tree topology provides easy maintenance and easy fault identification can be done.

2. Point-to-point wiring for individual segments.

3. Tree Topology is highly secure.

NETWORK PROTOCOL:

A protocol means the rules that are applicable for a network. Protocols defines standardized formats for data packets, techniques for detecting and correcting errors etc.

TCP/IP (Transmission Control Protocol/ Internet Protocol

- It is a set of standardized rules that uses a client-server model of communication in which a user or machine (a client) requests a service by a server in the network.
- The IP protocol ensures that each computer or node connected to the Internet is assigned an IP address, which is used to identify each node independently.
- TCP ensures that the message or data is broken into smaller chunks, called IP packets. Each of these packets are routed (transmitted) through the Internet, along a path from one router to the next, until it reaches the specified destination. TCP guarantees the delivery of packets on the designated IP address. It is also responsible for ordering the packets so that they are delivered in sequence.

FTP (File Transfer Protocol)

- It is a standard internet protocol provided by TCP/IP used for transmitting the files from one host to another. It is mainly used for transferring the web page files from their creator to the computer that acts as a server for other computers on the internet. It is also used for downloading the files to computer from other servers.

Objectives of FTP

- It provides the sharing of files.
- It is used to encourage the use of remote computers.
- It transfers the data more reliably and efficiently.

SMTP (Simple Mail Transfer Protocol.)

- SMTP is a set of communication guidelines that allow software to transmit an electronic mail over the internet is called Simple Mail Transfer Protocol.
- It is a program used for sending messages to other computer users based on e-mail addresses.It provides a mail exchange between users on the same or different computers, and it also supports:

I. It can send a single message to one or more recipients.
II. Sending message can include text, voice, video or graphics.
III. It can also send the messages on networks outside the internet.

- The main purpose of SMTP is used to set up communication rules between servers. They also have a way of handling the errors such as incorrect email address. For example, if the recipient address is wrong, then receiving server reply with an error message of some kind.

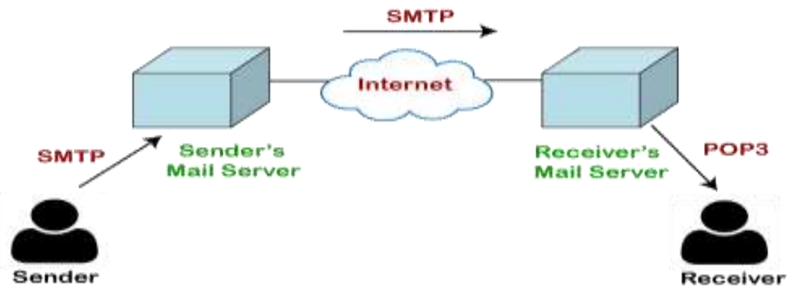
Point-to-Point Protocol (PPP) is protocol that is used to connect one computer system to another. Computers use PPP to communicate over the telephone network or the Internet.

A PPP connection exists when two systems physically connect through a telephone line. You can use PPP to connect one system to another. For example, an established PPP connection between a branch office and a central office allows either office to transfer data to the other through the network.

POP3 Protocol

Post Office Protocol version 3 (POP3) is a standard mail protocol used to receive emails from a remote server to a local email client. POP3 allows you to download email messages on your local computer and read them even when you are offline.

How is mail transmitted?



Telnet

- The main task of the internet is to provide services to users. For example, users want to run different application programs at the remote site and transfers a result to the local site. This requires a client-server program such as FTP, SMTP. But this would not allow us to create a specific program for each demand.
- The better solution is to provide a general client-server program that lets the user access any application program on a remote computer. Therefore, a program that allows a user to log on to a remote computer. A popular client-server program Telnet is used to meet such demands. Telnet is an abbreviation for Terminal Network.
- Telnet provides a connection to the remote computer in such a way that a local terminal appears to be at the remote side.

There are two types of login:

Local Login- When a user logs into a local computer, then it is known as local login.

Remote login- When the user wants to access an application program on a remote computer, then the user must perform remote login.

VoIP :

VoIP stands for **Voice over Internet Protocol**. It is also referred to as IP telephony, internet telephony, or internet calling. It's an alternative to a public switched telephone network (PSTN).

Introduction to Web Services

Everyone is using internet irrespective of their background & interest. The types of services available on internet are as diverse as the interest of the people. Web Services means the services provided by World Wide Web. The World Wide Web provides services like chatting, emailing, video conferencing, e-learning, e-shopping, e-reservation, e-groups and social networking. These services may mean different things to different people but in simple words, they are the ways to communicate and here the communication takes place between two electronic devices.

World Wide Web (WWW)

The World Wide Web commonly referred to as WWW, W3, or the Web is an interconnected system of public webpages accessible through the Internet.

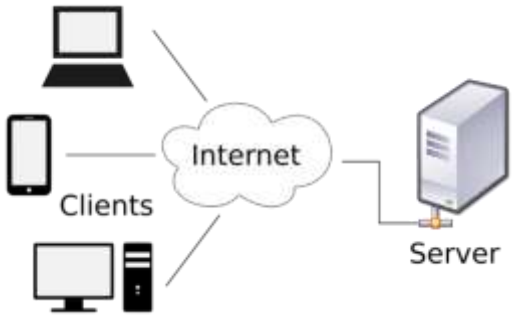
The Web is not the same as the Internet: the Web is one of many applications built on top of the Internet.It's invented by Tim Berners-Lee in 1989. Since that time, the Web has changed the world. It has perhaps become the most powerful communication medium the world has ever known.

Major components of WWW are,

1. Web Server – It is a computer that stores website resources (web pages, images, videos, etc.).
2. Web Browser (Client) - A software application used to access the web resources.
3. Webpage - Hypertext documents formatted in Hypertext Mark-up Language (HTML) and displayed in a web browser.
4. Website - A website is a collection of inter-linked web pages that is identified by a common domain name (website name) and stored on a web server.
5. HTTP Protocol - It governs data (web page) transfer between a server and a client.
6. HTML- A mark-up language used to specify the structure of a webpage.
7. URL- Address used to identify documents and other web resources on the internet.

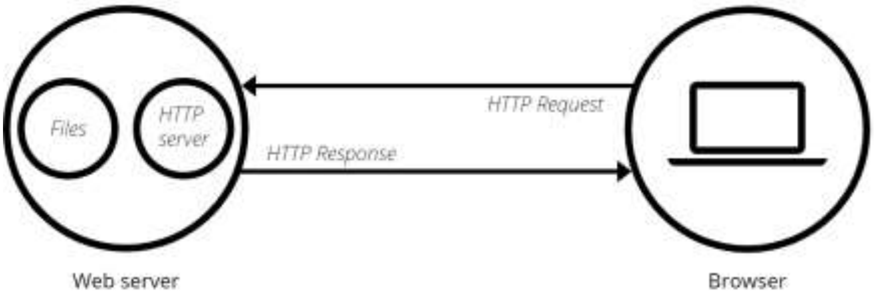
Web Architecture

Web is working based on a client-server architecture.



Client: It is a computer capable of requesting, receiving & displaying information in the form of web pages or using a particular service from the service providers (Servers). It's also called as web browsers.

Servers: It is a remote computer which provides/transfers information to the client (in the form of web pages) or access to particular services.



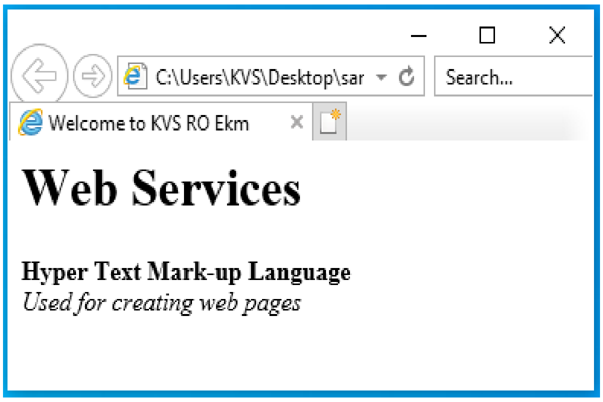
So, it's basically the Client requesting web pages and the Server serving it as long as the requested content is available in the server. i.e. a request-response model.

Difference between Internet and WWW

Internet	World Wide Web(WWW)
Internet stands for Interconnected Networks	WWW stands for World wide Web
Internet is a means of connecting a computer to any other computer anywhere in the world.	World Wide Web which is a collection of information which is accessed via the Internet.
Internet is infrastructure.	WWW is service on top of that infrastructure.
Internet is primarily hardware-based.	WWW is more software-oriented as compared to the Internet.
Internet uses TCP/IP protocol.	WWW uses HTTP Protocol.

HTML (Hypertext Mark-up Language)

- It is not a programming language. It is a mark-up language that tells web browsers how to structure the web pages you visit.
- HTML is language the helps in creating and designing web content.
- It has a variety of tags and attributes for defining the layout and structure of the web document.
- It is designed to display the data in formatted manner.
- A HTML document has the extension .htm or .html.
- Hypertext is a text which is linked to another document.

HTML Code	**As Rendered in the Browser**
<html> <head> <title> Welcome to KVS RO Ekm </title> </head> <body> <h1>Web Services</h1> <b>Hyper Text Mark-up Language</b><br> <br><i>Used for creating web pages</i> </body> </html>	 C:\Users\KVS\Desktop\sar Search... Welcome to KVS RO Ekm # Web Services **Hyper Text Mark-up Language** *Used for creating web pages*

XML (eXtensible Mark-up Language)

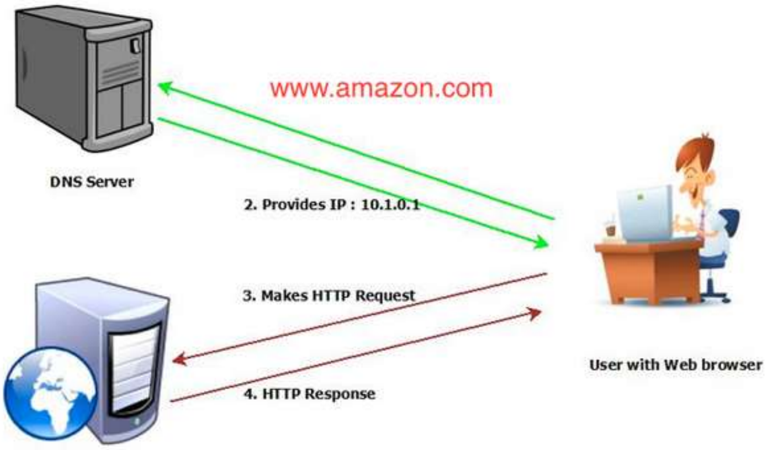
- XML is a mark-up language like HTML.
- It is designed to transport or store data.
- In contrast to HTML, it's used to transport the data and not for displaying the data.
- Unlike HTML, it does not have predefined tags.
- It is possible to define new tags in XML.
- It allows the programmer to use customized tags.
- XML is case sensitive.
- A XML document has the extension .xml.

HTML v/s XML

HTML	XML
HTML stands for Hyper Text Mark-up Language	XML stands for eXtensible Mark-up Language
HTML is a case insensitive.	XML is case sensitive.
Predefined tags (commands).	User defined tags (commands).
It is used for presentation of the Data.	It is used for transportation of the Data.
Small errors can be ignored.	Errors not allowed.
Closing tags are optional.	Compulsory to use closing tags.

HTTP - Hyper Text Transfer Protocol

- HTTP is used to transfer data across the web. HTTP specifies how to transfer hypertext (linked web documents) between two computers.
- It allow users of the World Wide Web to exchange information found on web pages.
- When accessing any web page entering http:// in front of the address tells the browser to communicate over HTTP.
- A Protocol is a set of rules for communication between two computers. HTTP is a textual, stateless protocol.



- Clients (web browsers) send requests through request object of http to web servers for web pages, resources etc.
- Web server respond accordingly through response object of http.
- After this cycle (request – response), the connection between client and server across the Internet is disconnected.
- A new connection must be made for each request (means for each web page).
- Hypertext Transfer Protocol Secure (HTTPS) is an extension of the HTTP.

- It is used for secure communication over a computer network.
- In HTTPS, the communication protocol is encrypted using Transport Layer Security (TLS).
- HTTPS provides,
  1) Encryption: Encrypting the exchanged data to keep it secure from eavesdroppers.
  2) Data integrity: Data cannot be modified or corrupted during transfer.
  3) Authentication: Proves that your users communicate with the intended website.

Domain Names

- The Internet is a vast ocean where information is available in the form of millions of websites.
- Each website is stored on a server which is connected to the Internet, which means each server has an IP address. Every device connected to the Internet has an IP address.
- To access a website, we need to enter its IP address on our web browser.
- But it is very difficult to remember the IP addresses of different websites as they are in terms of numbers or strings.

- However, it is easier to remember names, and therefore, each computer server hosting a website or web resource is given a name against its IP address. These names are called the Domain names or hostnames corresponding to unique IP addresses assigned to each server.

- For easy understanding, it can be considered as the phonebook where instead of remembering each person's phone number, we assign names to their numbers.

For example, IP addresses and domain names of some websites are as follows:

Domain Name	IP Address
ncert.nic.in	164.100.60.233
cbse.nic.in	164.100.107.32
kvsangathan.nic.in	164.100.166.91
wikipedia.org	198.35.26.96

- The process of converting a hostname (such as www.google.com) into the corresponding IP address (such as 172.217.14.196) is called domain name resolution.

- Specialized DNS servers are used for domain name resolution (DNS resolution), so whenever you type some web address (domain name) on the address bar of a web browser, it will be translated into an equivalent IP address with the help of one or more DNS servers including local DNS cache.

URL-Uniform Resource Locator

- Every web page that is displayed on the Internet has a specific address associated with it, this address is known as the URL.
- Like postal address is required to reach a letter to a person staying anywhere on earth.

The URL consists of four basic parts, namely, server type, hostname, folder name and the filename. Each one of these has a specific function.

1) The "server type" indicates the type of Internet server (Protocol) being accessed. The server type is always followed by "://" and the host name.
2) The host name/domain name is the Internet address of a remote computer on which the files reside.
3) The folder name indicates the name of the directory in which the files are located.
4) The filename specifies the name of the specific document to be displayed in the browser. The filename itself consists of two pieces of information, the name of the file to be displayed and the file extension, which specifies the file type (.htm for HTML file, .txt for a text file, .bmp for a bitmap image, etc.)

The structure of a URL can be represented as follows:

Websites

- A website is a collection of linked web pages (plus their associated resources) that share a unique domain name.
- Thus, a Web site is a collection of related Web pages.
- Each web page of a given website provides explicit links, most of the time in the form of clickable portion of text, that allow the user to move from one page of the website to another.
- Each Web site is owned and updated by an individual, company, or an organization.
- Web is a dynamically moving and changing entity, today web sites generally change on a daily or even hourly basis.
- To access a website, type its domain name in your browser address bar, and the browser will display the website's main web page, or homepage (casually referred as "the home")

Web page

- Web page is an electronic document designed using HTML.
- It displays information in textual or graphical form.
- It may also contain downloadable data files, audio files or video files.
- Traversal from one webpage to another web page is possible through hyperlinks.

Difference between Webpage and Website

A web page is one single page of information, while a website is made up of a number of related web pages inter-connected by links known as hyperlinks.

Web Browser

- Web browser is software program to navigate the web pages on the internet.
- A bowser interprets the coding language of the web page and displays it in graphic form.
- A web browser allows anyone to access the web without even knowing commands used in software languages to design a web page.
- Internet works on client -server model.
- A web browser is a client which requests the information from the web server.
- The web server sends the information back to the client.
- The web address of the webpage written on the address bar tells the web browser which page to access.
- The most popular web browsers are: Google Chrome, Mozilla Firefox, Internet Explorer, Safari, Opera, Netscape and UC Browser.

What is a cookie?

- A cookie is information stored on your computer by a website you visit.
- Cookies often store your settings for a website, such as your preferred language or location.
- When you return to the site, browser sends back the cookies that belong to the site.
- This allows the site to present you with information customized to fit your needs.
- Cookies can store a wide range of information, including personally identifiable information (such as your name, home address, email address, or telephone number).
- However, this information can only be stored if you provide it - websites cannot gain access to information you didn't provide to them, and they can't access other files on your computer.

Web Server

- A *web server* is a computer or a group of computers hosting one or more *websites.*
- "Hosting" means that all the *web pages* and their supporting files are available on that computer.
- The *web server* will send any *web page* from the *website* it is hosting to any user's browser, per user request.
- More importantly, since a web server can host multiple websites, the term *web server* is never used to designate a website, as it could cause great confusion.
- It works on client/server model.
- It delivers the requested web page to web browser.
- Web servers use special programs such as Apache or IIS to deliver web pages over the http Protocol

Web Hosting

- Web hosting is the process of uploading/saving the web content on a web server to make it available on WWW.
- In case an individual or a company wants to make its website available on the internet, it should be hosted on a web server.
- A good way to think about this is if the domain name is the address of our house, then web hosting is the actual house that address points to. All websites on the internet, need web hosting.
- Domain names and web hosting are two different services. However, they work together to make websites possible. It is possible with the system known as DNS.

Model Questions

I. Multiple Choice Questions

1. What are the three common types of computer networks?

a. ROM, MAN, LAN b. RAM, WAN, LAN

c. MAN, LAN, WAN d. None of the above

2. What is the acronym for LAN?

a. Local Area Network b. Local Access Network

c. Line And Networking d. Line-less Networking

3. Define what a LAN is?

a. Connected devices share the resources of a single processor or server within a small geographic area

b. Normally find within a business and school

c. These are computers that share resources over a large area

d. None of the above

4. Mr. John is a small businessman who runs Hardware. He has been experiencing problems with his small accounting department, which he depends on to provide sales reports. Mr. John wants to share information between his 7 computer stations and have one central printing area. What type of network would you recommend to Mr. John?

a. MAN b. LAN

c. WAN d. SAN

5. WAN covers a larger geographical area than MAN?

a. True b. False

6. A network that consists of both LANs and MANs is called a Wide area network?

a. True b. False

7. Arrange the Following Types of Networks according to their size, from largest to smallest?

a. LAN, WAN, MAN b. WAN, LAN, MAN

c. MAN, LAN, WAN d. WAN, MAN, LAN

8. You are a member of a club that deals with computer networks. The club has to take a project to build a MAN. Where would this project likely take place?

a. A small building/organization b. University or college

c. Home d. None of the above

9. What is the acronym MAN stand for?

a. Magnetic Access Network b. Metropolitan Area Network

c. Multi-Area Network d.. Multi-Access net

10. In your school there is a library, and you can use the internet to do research, this library will most likely be a WAN network?

a. True b. False

11. Types of Networks are Categories by their Geographical Area cover?

a. True b. False

12. Metropolitan Area Network Normally Require an Internet connection?

a. True b. False

13. What’s a web browser?

a) A kind of spider

b) A computer that store www files

c) A person who likes to look at websites

d) A software program that allows you to access sites on the World Wide Web

14. A _____ is a document commonly written and is accessible through the internet or other network using a browser?

a) Accounts b) Data

c) Web page d) Search engine

15 Which of the following is used to read HTML code and to render Webpage?

a) Web Server b) Web Browser

c) Web Matrix d) Weboni

16. Which of these tech company owns Firefox web browser?

a) Lenovo b) IBM

c) Apple d) Mozilla

17. Which of the following browsers were/are available for the Macintosh?

a) Opera b) Safari

c) Netscape d) All of these

18. What is the name of the browser developed and released by Google?

a) Chrome b) Googly

c) Heetson d) Titanium

19. Which of the following is a Web Browser?

a) MS-office b) Notepad c) Firefox d) Word 2007

20. Which of the following is not a web browser?

a) Chrome b) Microsoft Edge c) Safari d) Ubuntu

21.URL stands for

(a) Uniform Research Limited (b) Uniform Resource Locator

(c) Uniform Resource Labs (d) Uniform Research Locator

22. LinkedIn is an example for ______________________ website.

(a) E-learning (b) E-commerce

(c) Video conferencing (d) Social networking

23. Which of the following is not a web service?

(a) Distance Learning (b) E-mailing

(c) Video conferencing (d) Social networking

24. Web browsers are also called as ________________

(a) Web Servers (b) Web Clients (c) Web Hosting (d) Web Designing

25. Working of WWW based on ____________________ architecture.

(a) Peer-To-Peer architecture (b) Client-Client architecture

(c) Client-Server architecture (d) Server-Server architecture

26. _________________ is a computer software capable of requesting, receiving & displaying information in the form of webpages.

(a) Web Servers (b) Web Browsers

(c) Web Designers (d) Web Camera

27. A ____________ is a program or automated script which browses the World Wide Web in a methodical, automated manner.

(a) Web Servers (b) Web Browsers

(c) Web Designers (d) Web Crawler

28.________________ is a mark-up language that helps in developing web pages.

(a) HTTP (b) HTML (c) XML (d) C++

29. ________________ is a language used to transport data over internet.

(a) HTTP (b) HTML (c) XML (d) C++

30.___________ is a set of rules for communication between two computers over a network.

(a) Modem (b) Protocol

(c) Switch (d) IP address

31.In web services, the communication takes place between

(a) Two electronic devices (b) Two human beings

(c) Two spiders (d) None of the above

32.Web services means services provided by ________________________

(a) Microsoft (b) Google

(c) BSNL (d) World Wide Web

II. Very Short Answer Type Questions(1-mark)

1. Name The transmission media best suitable for connecting to hilly areas.

2. How many pair of wires are there in twisted pair cable (Ethernet)?

3. Name a device that forwards data packets along networks.

4. What is the full form of WWW?

5. What is the full form of Internet?

6. Who invented the WWW in 1989?

7. Special software's that is used to view webpages are ____________

8. ________ are used to store webpages, so whenever a request, it will serve the request.

9.____________ are programs /computers used to store information's in the form of webpages.

10. Web pages that are linked to each other via __________

11. ____________ protocol is used to transfer web pages over internet.

12. Full form of HTTP?

13._____________ is a massive collection of digital pages to access information over the Internet

14. Write any 2 differences between HTML & XML?

15.___________ is a real-time communication between two or more users via computer.

16.___________ helps us to learn anywhere using Internet.

17.___________ allows customers to conduct financial transactions on a secure Website.

18. Internet can be used to get reservation of trains and air planes through _______ service.

19. ___________ helps to create and maintain social relationship over web.

20. Expand the following abbreviations:

a. HTTP
b. XML
c. HTTPS
d. HTML
e. VoIP

21. Name any two common web browsers.

22. Full form of Email is ______________________

23.What out of the following, you will use to have an audio visual chat with an expert sitting in a faraway place to fix-up technical issues?

(i) E-mail (ii) VoIP (iii) FTP

24. Match the following

Web Services		Description	
A	Video conferencing	P	Without ever having to go booking office
B	E-Shopping	Q	Self-paced learning modules allow students to work at their own speed
C	E-mail	R	Each of the end user has a camera as well as microphone to capture video and audio in real time and it will be transmitted over internet
D	E-reservation	S	Purchasing products through computers/mobile devices
E	E-learning	T	Messages normally reaches a recipients account within seconds

25. Match the following

Web Services		**Applications**	
A	Video conferencing	P	IRCTC
B	E-Shopping	Q	Diksha App
C	E-mail	R	GEM
D	E-reservation	S	Gmail
E	E-learning	T	Instagram
F	Social Networking	U	VConSol

III.Short Answer Type (Theory with Answers)

1. What are Protocols? Name the protocol used to transfer a file from one device to the other.

Protocols are set of rules that are followed while transmitting data through a computer network. Protocols determines how to data can be moved securely from a source device to a destination device. The protocol used for transferring a file from one device to another is the File Transfer Protocol (FTP)

2. What is meant by an IP Address? Give an example for IP Address.

An IP Address is a numerical address that uniquely identifies every device connected to a network or internet. The user's physical location can be tracked by using an IP Address. IP V4 (IP Version 4) is a popular version of IP Address. IP Address (in IP V4) consists of four set of numbers separated by a dot. These numbers can range from 0 to 255.

An example IP Address format is given below:

192.158.12.38

3. Explain how an IP Address become helpful in investigating cyber-crimes.

IP address can be used to trace the physical location of a user connected to a network. By this many cyber crime can be investigated and traced out efficiently tracking the exact location from where the cybercrime is carried out.

4. Why Protocols are needed in the case of Data Communication?

The communicating devices may be in different geographical areas. The speed of these devices may be different. Also, the data transfer rates of different networks may be different. These complexities make it necessary to have a common set of rules i.e., Protocols to ensure the secure communication of data

5. What is the difference between World Wide Web & Internet?

Internet means interconnected networks that spread all over the world (i.e. the physical infrastructure), while WWW means the information's (available in the form of webpages) that can be accessed through internet.

6. What is a protocol, give some examples?

Protocols are set of rules that are followed while transmitting data through a computer network. Protocols determines how to data can be moved securely from a source device to a destination device. The protocol used for transferring a file from one device to another is the File Transfer Protocol (FTP)

7. What is the difference between E-mail and chat?

In order to chat, you need to have an account on the same service as the person you are chatting with. e.g. on the other hand, in case of E-mail, it is not necessary, i.e. you can have an account from any provider and you can establish your own.

8. What are cookies?

Cookies are files stored temporarily on www browser's computer, which allow the www server to store persistent information associated with browsing user on user's system.

9. What is the difference between domain name and IP address?

IP addresses look like this: 192.168.12.134.

Domain names look like this: "www.google.com"

Domain names are easier for us to remember and use, while computers are quite handy with numbers. Thus, we use DNS (Domain Naming System) to translate domain names into the IP addresses.

IP address is a unique identifier for a computer or device on internet. A domain name (website name) is a name that identifies one or more IP addresses (when hosted at different servers for load balancing).

10. Give one suitable example of each URL and domain name?

URL: https://kvsangathan.nic.in/hq-gyan-kosh

Domain name: kvsangathan.nic.in

11. Differentiate between XML and HTML.

XML was designed to describe data and to focus on what data is.

HTML was designed to display data and to focus on how data looks.

HTML is about displaying information while XML is about describing information.

12. Distinguish between website and web browser.

Website:- It is a collection of inter-linked web pages stored in a server.

Web Browser:- It is a software application for retrieving, presenting and traversing through information resources in the form of web pages available on the World Wide Web.

13. Differentiate between the terms Domain Name and URL in context of web services. Also write one example of each to illustrate the difference.

Domain Name	URL
A domain name or website name is a human-friendly text form of the IP address.	URL is a string that represents the complete web address of any web page. It's used to locate a webpage.
It is the part of the URL that is more human friendly.	It is the string that represents a complete web address that contains the domain name.
Example: kvsangathan.nic.in	Example: https://kvsangathan.nic.in/contact-us

22.Differentiate between communication using Optical Fiber and Ethernet Cable in context of wired medium of communication technologies.

Optical Fibre - Very Fast - Expensive - Immune to electromagnetic interference

Ethernet Cable - - Slower as compared to Optical Fiber - Less Expensive as compared to Optical Fiber - prone to electromagnetic interference

IV. Previous CBSE Board Questions

1) Ravi was trying to log-in to his internet-banking account. He noticed that the URL of the net banking starts with 'https'. The 's' in 'https' stands for _______
   (i) Simple 0(ii) Smart (iii) Secure (iv) Strength

2) What is mean by the homepage of a website?

3) What is the significance of the URL?

4) Which of the following is not a network protocol?
   (i) HTML (ii) HTTP (iii) SMTP (iv) FTP

5) Which of the following internet protocols provides secure data transmission between server and browser with the help of encryption.
   a) HTTP b) HTTPS c) TELNET d) ARPANET

6) Devanand, a student of Class XII, is not able to understand the difference between web client and web-server. Help him in understanding the same by explaining their role and giving suitable example of each.

7) Write the full form of Cc and Bcc (used in email communication). Explain the difference between them.

8) Define Internet and write its two uses in our daily life. How is it different from the World Wide Web (WWW).

9) Web _________ is a software used to view web pages.

10) In a network, _____________ is a computer that provides data and resources to other computers.

11) How is a website and a webpage related?

12) Microsoft Edge, Apple Safari are examples for __________________

13) What is the use of a server in a computer network?

14) Among the following service available on the World Wide Web are?

i) Email ii) HTML iii) XML iv) Video conferencing

(a) (i) and (ii) (b) (i) and (iv)

(c) (ii) and (iii) (d) None of the above

15) HTML and XML are _______________

(a) Programming Languages (b) Scripting Languages

(c) Mark-up Languages (d) None of the above

16) XML uses _____________

(a) User defined tags (b) Predefined Tags

(c) Both user defined and predefined tags (d) None of the above

17) XML was not designed to ____________________

(a) store the data (b) present the data

(c) carry the data (d) both a & c

18) Which of the following will you suggest to establish the online face to face communication between the people in the Regional Office Ernakulum and Delhi Headquarter?

(a) Cable TV (b) Email

(c) Text chat (d) Video Conferencing

19) What kind of data gets stored in cookies and how is it useful?

20) What do the following top level domains signify?

(a) .com

(b) .org

21) "With XML you invent your own tags", Explain this statement with the help of example.

22) Define Domain Name Resolution?

23) ____________ tags are case sensitive and _____________ tags are not case sensitive.

(a) HTML, XML (b) HTTP, XML

(c) XML, HTTP (d) XML,HTML

24) Which of the following is not a web browser ?

(a) Google Chrome (b) Internet Explorer (c) Mozilla Firefox (d) Photoshop

25) Which protocol helps us to browse through web pages using internet browsers?

26) Name any one internet browser.

27) XML stands for

(A) Xtra Markup Language

(B) Extensible Markup Language

(C) Extensible Marking Language

(D) Extensive Marked Language

28) We can upload a file to a web server using a protocol called __________.

(A) FPT (B) IP (C) TCP (D) FTP

29) ______________ delivers the requested web page to web browser.

30) MyPace University is setting up its academic blocks at Naya Raipur and is planning to set up a network. The University has 3 academic blocks and one Human Resource Center as shown in the diagram below: Study the following structure and answer questions (a) to (e)

Center to Center distances between various blocks/center is as follows:

Law Block to business Block	40m
Law block to Technology Block	80m
Law Block to HR center	105m
Business Block to technology Block	30m
Business Block to HR Center	35m
Technology block to HR center	15m

Number of computers in each of the blocks/Center is as follows:

Law Block	15
Technology Block	40
HR center	115
Business Block	25

a) Suggest the most suitable place (i.e., Block/Center) to install the server of this University with a suitable reason.

b) Suggest an ideal layout for connecting these blocks/centers for a wired connectivity.

c) Which device will you suggest to be placed/installed in each of these blocks/centers to efficiently connect all the computers within these blocks/centers?

d) Suggest the placement of a Repeater in the network with justification.

e) The university is planning to connect its admission office in Delhi, which is more than 1250km from university. Which type of network out of LAN, MAN, or WAN will be formed? Justify your answer.

Test Your Self : PART I

1 Write any 1 advantage and 1 disadvantage of Bus topology.

2. What happens to the Network with Star topology if the following happens:

(i) One of the computers on the network fails?

(ii) The central hub or switch to which all computers are connected, fails?

3. Two doctors have connected their mobile phones to transfer a picture file of a person suffering from a skin disease. What type of network is formed?

4. SunRise Pvt. Ltd. is setting up the network in the Ahmadabad. There are four departments named as MrktDept, FunDept, LegalDept, SalesDept.

Distance between various buildings is given as follows:

MrktDept to FunDept	80 m
MrktDept to LegalDept	180 m
MrktDept to SalesDept	100 m
LegalDept to SalesDept	150 m
LegalDept to FunDept	100 m
FunDept to SalesDept	50 m

Number of Computers in the buildings

MrktDept	20
LegalDept	10
FunDept	8
SalesDept	42

i) Suggest a cable layout of connections between the Departments and specify topology.

ii) Suggest the most suitable building to place the server with a suitable reason.

iii) Suggest the placement of i) modem ii) Hub /Switch in the network.

iv) The organization is planning to link its sale counter situated in various part of the same city/ which type of network out of LAN, WAN, MAN will be formed? Justify.

5. Name the protocol Used to transfer voice using packet switched network.

6. What is HTTP?

7. Write the purpose of the following devices:

(i). Network Interface Card

(ii). Repeater

Test Your Self : PART II

1. The 'Grand Plaza ' Mall has a computer network. The network is in one building. Name this type of network(out of LAN/MAN/WAN).

2. Identify the type of topology on the basis of the following:

a. Since every node is directly connected to the server, a large amount of cable is needed which increases the installation cost of the network.

b. It has a single common data path connecting all the nodes

3. The following is a 32 bit binary number usually represented as 4 decimal values, each representing 8 bits, in the range 0 to 255 (known as octets) separated by decimal points.

192.158.1.38

What is it? What is its importance?

4. Dinsey has to share the data among various computers of his two offices branches situated in the same city. Name the network (out of LAN, WAN, PAN and MAN) which is being formed in this process.

5. . Global Pvt. Ltd. is setting up the network in the Bangalore . There are four departments

Distances between various buildings are as follows:

Accounts to Research Lab	55 m
Accounts to Store	150 m
Store to Packaging Unit	160 m
Packaging Unit to Research Lab	60 m
Accounts to Packaging Unit	125 m
Store to Research Lab	180 m

Number of Computers

Accounts	25
Research Lab	100
Store	15
Packaging Unit	60

i) Suggest a cable layout of connections between the buildings.

ii) Suggest the most suitable place (i.e. buildings) to house the server of this organization.

iii) Suggest the placement of the following device with justification:

a) Repeater b) Hub/Switch

6. Write one example each of URL and IP address.

ANSWER KEY

I. **Multiple Choice Questions(MCQ)**

1. **c**	2. **a**	3. **a**	4. **b**	5. **a**
6. **a**	7. **d**	8. **d**	9. **b**	10. **a**
11. **a**	12. **a**	13. **d**	14. **c**	15. **b**
16. **d**	17. **b**	18. **a**	19. **c**	20. **d**
21. **b**	22. **d**	23. **a**	24. **b**	25. **c**
26. **b**	27. **d**	28. **b**	29. **c**	30. **b**
31. **a**	32. **d**			

II. **Very Short Answer Type Questions(1-mark)**

1. Microwave / Radio wave
2. Two insulated copper wires
3. Router
4. World Wide Web
5. Interconnected Networks
6. Tim Berners-Lee
7. Web browsers
8. Web servers
9. Web servers
10. hyperlinks
11. HTTP- HyperText Transfer Protocol
12. HyperText Transfer Protocol
13. World Wide Web(WWW) or Web
14. Refer comparison table
15. Chat
16. E-learning

17. Internet banking

18. E-reservation

19. Social networking websites

20.

   a. HTTP- HyperText Transfer Protocol
   b. XML – eXtensible Mark-up Language
   c. HTTPS - HyperText Transfer Protocol Secure
   d. HTML - HyperText Mark-up Language
   e. VoIP-Voice over Internet Protocol

21. Google Chrome, Mozilla Firefox

22. Electronic mail

23. Ii.VoIP

24.

A	Video conferencing	P	Each of the end user has a camera as well as microphone to capture video and audio in real time and it will be transmitted over internet
B	E-Shopping	Q	Purchasing products through computers/mobile devices
C	E-mail	R	Messages normally reaches a recipients account within seconds
D	E-reservation	S	Without ever having to go booking office
E	E-learning	T	Self-paced learning modules allow students to work at their own speed

25.

Web Services		Applications	
A	Video conferencing	P	VConSol
B	E-Shopping	Q	GEM
C	E-mail	R	Gmail
D	E-reservation	S	IRCTC
E	E-learning	T	Diksha App
F	Social Networking	U	Instagram

IV.Previous CBSE Board Questions - Answers

1. (iii) Secure
2. The default (first) page of a website is called a Homepage.
3. URL specifies unique address of each document on the internet.
4. (i) HTML

Internet Protocols for communication over the Internet, the communicating devices must follow certain rules. These rules are called Internet protocols.

For email communication, we use SMTP and POP.

For communication between browser and server HTTP and HTTPS protocols are used.

We can use TELNET to access services available on a remote computer.

5. b) HTTPS
6. Web-Client: An application that requests for services from a webserver. Example: Web Browsers, Chatting Applications

Web-Server: Web-server is a software (or any dedicated computer running this software) that serves the request made by web-clients. Example: Apache Server.

7. Cc : Carbon Copy: every recipient can check who else has received the mail.

Bcc : Blind Carbon Copy: no recipient can check who else has received the mail.

8. The Internet is a worldwide network that links many smaller computer-networks. Uses of the Internet 1. E-learning 2. E-commerce

The difference between the internet and www:

Internet means interconnected networks that spread all over the world (i.e. the physical infrastructure), while WWW means the information's (available in the form of webpages) that can be accessed through internet.

9. Web Browser
10. Server
11. A website is a collection of interlinked webpages.
12. Web Browsers
13. Role of server is to serve various clients (sharing data or resources among multiple clients)
14. (b) (i) and (iv)

15. (c) Mark-up Languages

16. (a) User defined tags

17. (b) present the data

18. (d) Video Conferencing

19. Cookies can store a wide range of information, including personally identifiable information (such as your name, home address, email address, or telephone number).

Cookies often store your settings for a website, such as your preferred language or location. When you return to the site, browser sends back the cookies that belong to the site. This allows the site to present you with information customized to fit your needs.

20. (a) .com - commercial

(b) .org - organization

21. XML tags are created by the user as there are no standard tags.

Ex : <name>Nayana<name>

22. The process of converting domain names (website names) into corresponding IP address with the help of DNS servers is called domain name resolution.

23. (d) XML,HTML

24. (d) Photoshop

25. HTTP - Hyper Text Transfer Protocol

26. Google Chrome or any other valid browser name

27. (B) Extensible Markup Language

28. (D) FTP

29. Web Server

30 a. Most suitable place to install the server is HR center, as this center has maximum number of computers.

c. Switch

d. Repeater may be placed when the distance between 2 buildings is more than 70 meter.

e. WAN, as the given distance is more than the range of LAN and MAN.

Test Yourself: PART I

1. **Advantage**: Since there is a single common data path connecting all the nodes, the bus topology uses a very short cable length which considerably reduces the installation cost. **Disadvantage**: Fault detection and isolation is difficult. This is because control of the network is not centralized in any particular node. If a node is faulty on the bus, detection of fault may have to be performed at many points on the network. The faulty node has then to be rectified at that connection point.

2. (i). failure in one cable will not affect the entire network

(ii). If the central hub or switch goes down, then all the connected nodes will not be able to communicate with each other.

3. PAN

4. i. Star Topology should be used.

ii) As per 80 – 20 rule, SalesDept because it has maximum no. of computers.

iii)Each building should have hub/switch and Modem in case Internet connection is required.

iv)MAN (Metropolitan Area Network)

5. VoIP

6. HTTP is a protocol that is used for transferring hypertext(i.e. text, graphic, image, sound, video, etc,) between 2 computers and is particularly used on the World Wide Web (WWW)

7.

(i) Network Interface Card (NIC) is a network adapter used to set up a wired network. It acts as an interface between computer and the network.

(ii) A repeater is a device that amplifies a signal being transmitted on the network.

Test Yourself: PART II

1. LAN

2.

a. Star Topology

b. Bus Topology

3. It is an IP Address. It is used to identify the computers on a network

4. MAN

5. (i)

(ii)The most suitable place/ building to house the server of this organization would be building Research Lab, as this building contains the maximum number of computers

(iii). a)Repeater : distance between Store to Research Lab is quite large, so a repeater would ideally be placed.

b)Hub/Switch : Each would be needed in all the buildings to interconnect the group of cables from the different computers in each building.

6.IP address 192.168.1.1

URL : https://www.apple.com/

DBMS-DATABASE MANAGEMENT SYSTEM

DBMS:

A database management system (DBMS) is a software that is responsible for storing, maintaining and utilizing databases. The collection of data, usually referred to as the database, contains information relevant to an enterprise. A database along with a DBMS is referred to as a database system.

Keeping organizational information in a file- processing system has a number of major disadvantages like:

a) Data redundancy and inconsistency
b) Difficulty in accessing data
c) Data isolation
d) Integrity problems
e) Concurrent access anomalies
f) Security problems

These difficulties among others, prompted the development of database systems.

- Database systems are designed to manage large bodies of information.
- Management of data involves both defining structures for storage of information and providing mechanisms for the manipulation of information.
- The database system also must ensure the safety of the information stored, despite system crashes or attempts at unauthorized access. If data are to be shared among several users, the system must avoid possible anomalous results.
- DBMS lets users to create a database, store, manage, update/modify and retrieve data from that database by users or application programs.

Some examples of open source and commercial DBMS include MySQL, Oracle, PostgreSQL, SQL Server, Microsoft Access, MongoDB.

Database systems are widely used. Some representative applications are :

a) Banking
b) Airlines
c) Universities
d) Telecommunications
e) Sales
f) Finance, etc.

A data model describes the structure of the database, including how data are defined and represented, relationships among data, and the constraints.

Database schema: Database Schema is the design of a database. It is the skeleton of the database that represents the structure (table names and their fields/columns), the type of data each column can hold, constraints on the data to be stored (if any), and the relationships among the tables.

Database schema is also called the visual or logical architecture as it tells us how the data are organized in a database.

Data constraint: Restrictions or limitations are put on the type of data that can be inserted in one or more columns of a table to ensure accuracy and reliability of data in the database.

Meta – Data or Dictionary: A database catalogue that has data about data.

Database instance: The state of database at any time, after loading of data. A database schema can have different instances at different times.

Query: A query is a request to a database for obtaining information in a desired way.

Data Manipulation: Modification of database consists of three operations viz. Insertion, Deletion or Updation.

Relational model:

The relational model uses a collection of tables to represent both data and the relationships among those data.

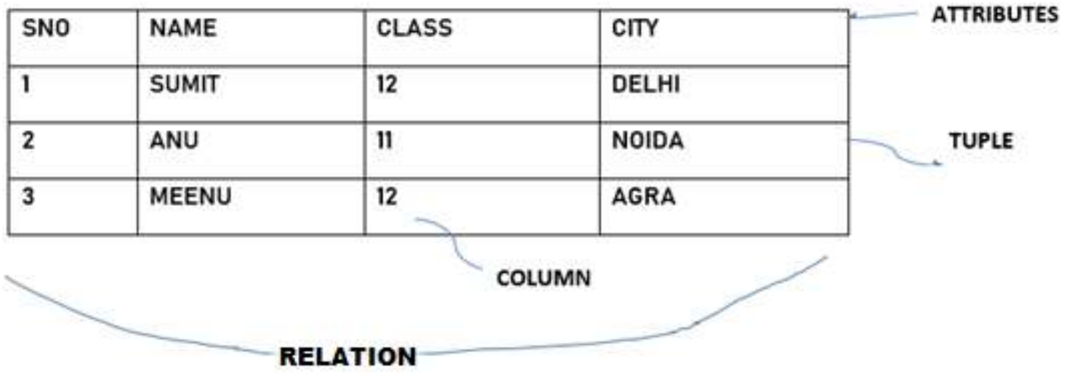
Each table has multiple columns, and each column has a unique name. The relational model is an example of a record – based model.

Record- based models are so named because the database is structured in fixed- format records of several types. Each table contains records of particular type. Each record type defines a fixed number of fields/attributes.

The columns of the table correspond to the attributes of the record type.

A row in a table represents a relationship among a set of values. The relational data model is the *most widely used data model*.

Other types of data models include object-oriented data model, entity-relationship data model, document model and hierarchical data model.



SNO	NAME	CLASS	CITY
1	SUMIT	12	DELHI
2	ANU	11	NOIDA
3	MEENU	12	AGRA

Relation: A Relation is logically related data organized in the form of tables.

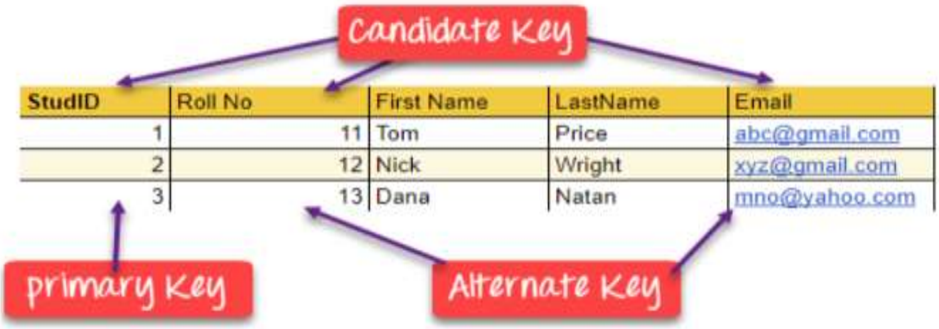
Attribute/ Field: Column of a table is called Attribute or Field.

Tuple/ Entity/ Record: Rows of a table is called Tuple or Record.

Domain: It is collection of values from which the value is derived for a column.

Degree - Number of columns (attributes) in a table.

Cardinality - Number of rows (Records) in a table.



StudID	Roll No	First Name	LastName	Email
1	11	Tom	Price	abc@gmail.com
2	12	Nick	Wright	xyz@gmail.com
3	13	Dana	Natan	mno@yahoo.com

Primary Key – A primary is an attribute or set of attributes in a relation that uniquely identifies tuples (rows) in that relation.

Candidate Key –It is an attribute or a set of attributes or keys participating for Primary Key, to uniquely identify each tuples in that relation.

Alternate Key – A candidate key that is not the primary key is called alternate key or secondary key.

Foreign Key – Foreign keys are the attributes of a relation that points to the primary key of another relation

Model Questions

I. Multiple choice Questions(MCQ):

1. DBMS stands for_____________
a) Data Base Management Software
b) Data Base Maintenance System
c) Data Basic Management System
d) Data Base management system

2. In RDBMS, R stands for_________
a) Relational
b) Rotational
c) Rational
d) None of the above

3. A Database contains one or more_______
a) Data
b) Tables
c) Files
d) Links

4. What is not true in respect of DBMS?
a) Database enforces standards
b) Database increases redundancy
c) Database facilitates sharing of data
d) Database helps to maintain integrity

5. Cardinality is total ___________
a) number of rows in a table
b) number of columns in a table
c) number of data items in a table
d) none of the above

6. Degree refers to total_________
a) number of rows in a table
b) number of columns in a table
c) number of data items in a table
d) none of the above

7. Data about data is __________
a) Data redundancy
b) Meta Data
c) Database schema
d) None of the above

8. Repetition of data is called ....
a) Data redundancy
b) Data Description
c) Data inconsistency
d) None of the above

9. Mismatched redundant copies of data is known as data ....
a) Dependence
b) Inconsistency
c) Isolation
d) Redundancy

10. A .... is an organized collection of structured data.
a) Database
b) File
c) DBMS
d) Information

11. A data .... is a set of rules that define valid data.
a) Query
b) Constraint
c) Dictionary
d) All of the above

12. A relational database consists of a collection of ....
a) Fields
b) Records
c) Keys
d) Tables

13. A row in a database.
a) Field
b) Record
c) Key

d) Table

14. The term _____________ is used to refer to a field in a table.

a) Attribute

b) Row

c) Tuple

d) Instance

15. Which of the following statements is not true about relational database?

a) Relational data model is the most widely used data model.

b) The data is arranged as a collection of tables in relational database.

c) Relational database increases data redundancy and inconsistency.

d) None of the above.

16. Which of the following is a disadvantage of file processing system?

a) Data redundancy

b) Data isolation

c) Data inconsistency

d) All of the above

17. Table: Employee

EMPID	ENAME	SALARY	DEPTNO	Email
1001	George	30000	10	George23@yahoo.co.in
1002	Mary	32000	12	Mary_123@yahoo.co.in
1003	Alwin	25000	12	Al_win@gmail.com
1004	Sumit	20000	11	Sumit_222@yahoo.co.in

Answer the questions based on the table Employee.

a) The cardinality of the table Employee is ……

i) 5

ii) 4

iii) 3

iv) 6

b) The degree of the table Employee is …….

i) 3

ii) 4

iii) 5

iv) 6

c) Which column can be made as the primary key in the table Employee?

i) EMPID
ii) EMAIL
iii) Both i and ii
iv) None of the above

d) If two columns are added to the table Employee, then the cardinality and degree of the table is …… and …… respectively.
i) 4 , 7
ii) 7, 4
iii) 6,5
iv) 5,6

18. An attribute in a relation is a foreign key if it is the ________ key in any other relation.
a) Candidate
b) Primary
c) Super
d) Sub

19. A(n) ________ in a table represents a logical relationship among a set of values.
a) Column
b) Key
c) Row
d) Attribute

II. Very Short Answer Questins:

1) What is meant by a database?
2) Define primary key?Give an example.
3) What do you mean by candidate key?
4) What is meant by degree and cardinality of a table?
5) What is meant by DBMS?
6) What is meant by database schema?
7) What is meant by data constraint?
8) What is meant by relation?

ANSWERS:

I. Multiple Choice Questions (MCQ)

1	d	2	a	3	b	4	b
5	a	6	b	7	b	8	a
9	b	10	a	11	b	12	d
13	b	14	a	15	c	16	d
17a)	(ii)	17b)	(iii)	17c)	(iii)	17d)	(ii)
18	b	19	c				

II. Very Short Answer Answers:

1	A database is an organized collection of structured information, or inter-related data, typically stored in a computer system.
2	A primary key is a column or set of columns that contain values that uniquely identify each row in a table. For example Rno can be primary key of the table student. Table:Student <table><tr><td>RNO</td><td>NAME</td><td>MARK</td></tr><tr><td>100</td><td>Tanay</td><td>30</td></tr><tr><td>101</td><td>Kiran</td><td>50</td></tr><tr><td>102</td><td>Manu</td><td>30</td></tr></table>
3	It is an attribute or a set of attributes capable of being the Primary Key, to uniquely identify each record in that table.
4	Degree refers to the number of attributes/columns in a relation. Cardinality refers to the number of tuples/rows in a relation.
5	RDBMS (Relational Database Management System) is the software used to store, manage, query, and retrieve data stored in a relational database.
6	Database schema is also called the visual or logical architecture as it tells us how the data are organized in a database.
7	Restrictions or limitations on the type of data that can be inserted in one or more columns of a table to ensure accuracy and reliability of data in the database.
8	A relation is a named, two dimensional table storing logically related data.

STRUCTURED QUERY LANGUAGE(SQL)

Structured Query Language: introduction, Data Definition Language and Data Manipulation Language, data type (char(n), varchar(n), int, float, date), constraints (not null, unique, primary key), create database, use database, show databases, drop database, show tables, create table, describe table.

1.1 What is SQL?

SQL(Structured Query Language) is a language that is used to manage data that is held in a relational database management system. It uses tables to manipulate and retrieve information from databases for analysis.

By using SQL commands, one can search for any data in the database and perform other functions like creating tables, adding records, modifying data, removing rows, dropping tables etc.

1.2 What Can SQL do?

SQL can execute queries against a database

SQL can retrieve data from a database

SQL can insert records in a database

SQL can update records in a database

SQL can delete records from a database

SQL can create new databases

SQL can create new tables in a database

SQL can create stored procedures in a database

SQL can create views in a database

SQL can set permissions on tables, procedures, and views

1.3 ADVANTAGES OF SQL

There are the following advantages of SQL:

- **High speed**

Using the SQL queries, the user can quickly and efficiently retrieve a large amount of records from a database.

- **No coding needed**

In the standard SQL, it is very easy to manage the database system. It doesn't require a substantial amount of code to manage the database system.

- **Well defined standards**

Long established are used by the SQL databases that are being used by ISO and ANSI.

- **Portability**

SQL can be used in laptop, PCs, server and even some mobile phones.

- **Interactive language**

SQL is a domain language used to communicate with the database. It is also used to receive answers to the complex questions in seconds.

- **Multiple data view**

Using the SQL language, the users can make different views of the database structure

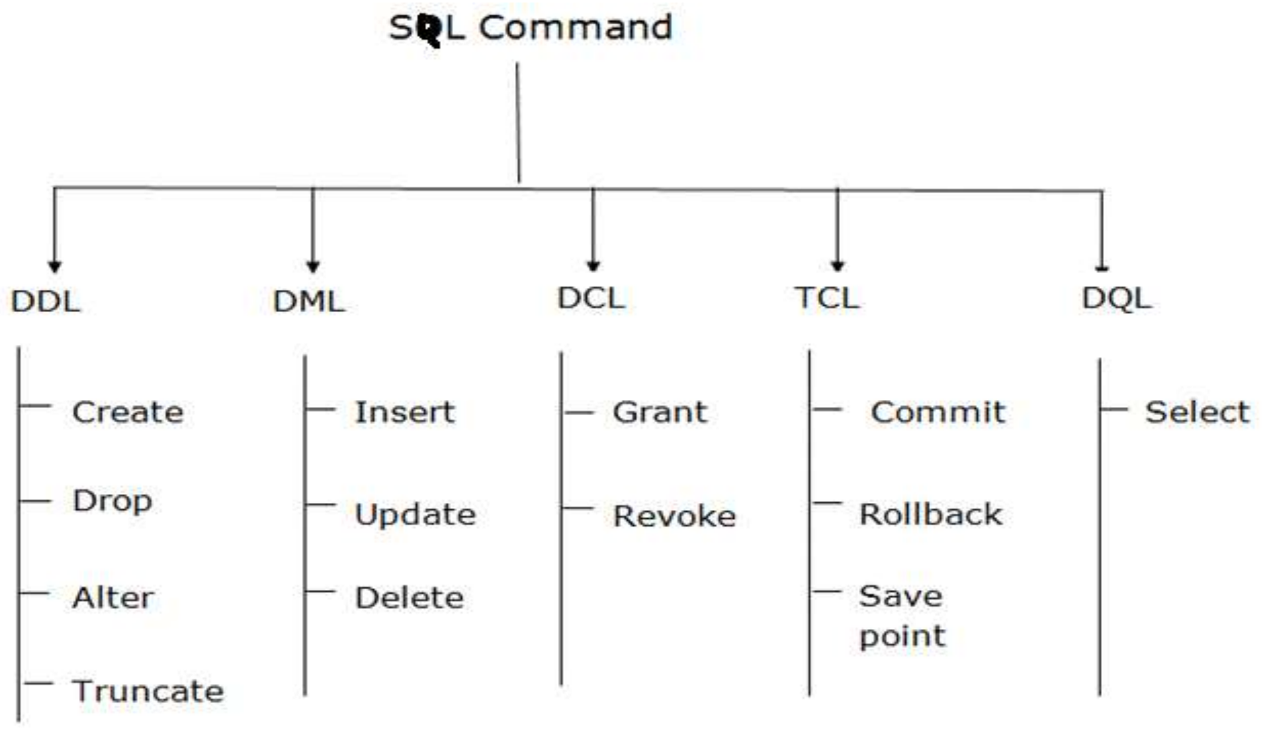
1.4 SQL Commands

SQL commands are instructions. It is used to communicate with the database. It is also used to perform specific tasks, functions, and queries of data.

SQL can perform various tasks like create a table, add data to tables, drop the table, modify the table, set permission for users.

1.5 Types of SQL Commands

There are five types of SQL commands: DDL, DML, DCL, TCL, and DQL.



1.5.1 DDL or Data Definition Language

DDL or Data Definition Language actually consists of the SQL commands that can be used to define the database schema. It simply deals with descriptions of the database schema and is used to create and modify the structure of database objects in the database. DDL is a set of SQL commands used to create, modify, and delete database structures but not data.

List of DDL commands:

CREATE: This command is used to create the database or its objects (like table, index, function, views, store procedure, and triggers).

DROP: This command is used to delete objects from the database.

ALTER: This is used to alter the structure of the database.

TRUNCATE: This is used to remove all records from a table, including all spaces allocated for the records are removed.

COMMENT: This is used to add comments to the data dictionary.

RENAME: This is used to rename an object existing in the database.

1.5.2 DML (Data Manipulation Language):

The SQL commands that deals with the manipulation of data present in the database belong to DML or Data Manipulation Language and this includes most of the SQL statements. It is the component of the SQL statement that controls access to data and to the database. Basically, DCL statements are grouped with DML statements.

List of DML commands:

INSERT : It is used to insert data into a table.

UPDATE: It is used to update existing data within a table.

DELETE : It is used to delete records from a database table.

Difference between DDL and DML:

DDL	DML
It stands for Data Definition Language.	It stands for Data Manipulation Language.
It is used to create database schema and can be used to define some constraints as well.	It is used to add, retrieve or update the data.
It basically defines the column (Attributes) of the table.	It add or update the row of the table. These rows are called as tuple.

1.6 DATATYPES

- **Text Data types**

  Char(size) – fixed length of *size* bytes

  Varchar(size)-variable length of *size* bytes

  Varchar2(size)-variable length of *size* bytes

- **Number Data types**

  Integer(size)or Int- It represents a number without decimal point

  Float(Size)-It represents a floating point number

  Real-Same as float but no size argument is used

- **Date data type**

  Date , Time

1.7 CONSTRAINT

A Constraint is a condition or check applicable on a field or set of fields.

Types of Constraints:

- **Unique Constraint** :-This ensures that no rows have the same value in the specified column(s)

Syntax:

Create table EMP (ecode integer unique, ename char(20),sex char(1),grade char(2));

Unique constraint applied on ecode of EMP table ensures that no rows have the same ecode value

- **Primary key Constraint**:- This declares a column as the primary **key** of the table This is similar to **unique** constraint except that one column (or one group of columns)can be applied in this constraint .

The primary key cannot allow NULL values but Unique key allows NULL values.

The following SQL creates a PRIMARY KEY on the "ID" column when the "Persons" table is created:

```
CREATE TABLE Persons (   ID int NOT NULL,   LastName
varchar(255) NOT NULL,
   FirstName varchar(255),   Age int,   PRIMARY KEY (ID));
```

- **Not null**:-This constraint ensures column should not be **NULL**

Syntax: Create table EMP(ecode integer Not null unique, ename char(20),sex char(1),grade char(2));

1.8 DATABASE COMMANDS IN MYSQL

- **CREATE DATABASE**

CREATE DATABASE is the SQL command used for creating a database in MySQL.

Imagine you need to create a database with name “movies”. You can create a database in MySQL by executing following SQL command

Syntax: mysql>CREATE DATABASE movies;

- **SHOW DATABASES**

You can see list of existing databases by running following SQL command.

Syntax: mysql>SHOW DATABASES;

- **USE**

You can use SQL command **USE** to select a particular database.

Syntax: mysql>USE database_name;

- **DROP DATABASE**

The DROP DATABASE statement is used to drop an existing SQL database.

Syntax :mysql>DROP DATABASE database_name*;*

CREATE TABLE

The CREATE TABLE statement is used to create a new table in a database.

Syntax: CREATE TABLE *table_name* (*column1 datatype*, *column2 datatype*, *column3 datatype*, ...);

Example: The following example creates a table called "Persons" that contains five columns:

PersonID, LastName, FirstName, Address, and City:

```
CREATE TABLE Persons (
    PersonID int,
    LastName varchar(255),
    FirstName varchar(255),
    Address varchar(255),
    City varchar(255)
);
```

- **SHOW TABLES;**

We can get the number of table information of a database using the following statement:

```
mysql> SHOW TABLES;
```

- **DESCRIBE TABLE**

Use the **DESCRIBE** command to show the structure of the table, such as column names, constraints on column names, etc. The **DESC** command is a short form of the DESCRIBE command. Both DESCRIBE and DESC commands are equivalent.

Syntax The following are the syntax to display the table structure:

```
mysql>  DESCRIBE | DESC    table_name;
```

ALTER TABLE Statement

The ALTER TABLE statement is used to add, delete, or modify columns in an existing table.The ALTER TABLE statement is also used to add and drop various constraints on an existing table.

ALTER TABLE - ADD Column/Attribute **To add a column in a table, use the following syntax:**

ALTER TABLE *table_name* ADD *column_name datatype*;

Example **ALTER TABLE Customers ADD Email varchar(255);**

ALTER TABLE - DROP COLUMN

To delete a column in a table, use the following syntax

ALTER TABLE *table_name* DROP COLUMN *column_name*;

Example ALTER TABLE Customers DROP COLUMN Email;

SQL PRIMARY KEY on ALTER TABLE

To create a PRIMARY KEY constraint on the "ID" column when the table is already created, use the following SQL:

```
ALTER TABLE table_name    ADD PRIMARY KEY (Column_name);
```

Example **ALTER TABLE Persons ADD PRIMARY KEY (ID);**

DROP a PRIMARY KEY Constraint

To drop a PRIMARY KEY constraint, use the following SQL:

```
ALTER TABLE table_name     DROP PRIMARY KEY;
```

Example : **ALTER TABLE Persons DROP PRIMARY KEY;**

The SQL DROP TABLE Statement

The DROP TABLE statement is used to drop an existing table in a database.

Syntax **DROP TABLE *table_name*;**

DROP TABLE Shippers;

INSERT INTO Statement:

The INSERT INTO statement is used to insert new records in a table.

INSERT INTO Syntax:

It is possible to write the INSERT INTO statement in two ways:

1. Specify both the column names and the values to be inserted:

INSERT INTO *table_name* (*column1*, *column2*, *column3*, ...) VALUES (*value1*, *value2*, *value3*, ...);

2. If you are adding values for all the columns of the table, you do not need to specify the column names in the SQL query. However, make sure the order of the values is in the same order as the columns in the table. Here, the INSERT INTO syntax would be as follows:

INSERT INTO *table_name* VALUES (*value1*, *value2*, *value3*, ...);

The SQL DELETE Statement:

The DELETE statement is used to delete existing records in a table.

DELETE Syntax:

DELETE FROM *table_name* WHERE *condition*;

Note: Be careful when deleting records in a table! Notice the WHERE clause in the DELETE statement. The WHERE clause specifies which record(s) should be deleted. If you omit the WHERE clause, all records in the table will be deleted!

The following SQL statement deletes all rows in the "Customers" table, without deleting the table:

DELETE FROM Customers;

The SQL SELECT Statement

The SELECT statement is used to select data from a database.

The data returned is stored in a result table, called the result-set.

SELECT Syntax: SELECT *column1*, *column2, ...* FROM *table_name*;

Here, column1, column2, ... are the field names of the table you want to select data from. If you want to select all the fields available in the table, use the following syntax:

SELECT * FROM *table_name*;

The SQL WHERE Clause:

The WHERE clause is used to filter records.

It is used to extract only those records that fulfill a specified condition.

WHERE Syntax: SELECT *column1*, *column2, ...*FROM *table_name* WHERE *condition*;

Operators in The WHERE Clause

The following operators can be used in the WHERE clause:

Operator	**Description**
=	Equal
>	Greater than
<	Less than
>=	Greater than or equal
<=	Less than or equal
<>	Not equal. Note: In some versions of SQL this operator may be written as !=
BETWEEN	Between a certain range
LIKE	Search for a pattern
IN	To specify multiple possible values for a column

The SQL AND, OR and NOT Operators

The WHERE clause can be combined with AND, OR, and NOT operators.

The AND and OR operators are used to filter records based on more than one condition:

- The AND operator displays a record if all the conditions separated by AND are TRUE.
- The OR operator displays a record if any of the conditions separated by OR is TRUE.

The NOT operator displays a record if the condition(s) is NOT TRUE.

AND Syntax

```
SELECT column1, column2, ...

FROM table_name

WHERE condition1 AND condition2 AND condition3 ...;
```

OR Syntax

```
SELECT column1, column2, ...

FROM table_name

WHERE condition1 OR condition2 OR condition3 ...;
```

NOT Syntax

```
SELECT column1, column2, ...

FROM table_name

WHERE NOT condition;
```

The SQL IN Operator

The IN operator allows you to specify multiple values in a WHERE clause.

The IN operator is a shorthand for multiple OR conditions.

IN Syntax

```
SELECT column_name(s)

FROM table_name

WHERE column_name IN (value1, value2, ...);
```

or:

SELECT *column_name(s)*

FROM *table_name*

WHERE *column_name* IN (*SELECT STATEMENT*);

The SQL BETWEEN Operator

The BETWEEN operator selects values within a given range. The values can be numbers, text, or dates.

The BETWEEN operator is inclusive: begin and end values are included.

BETWEEN Syntax

SELECT *column_name(s)*

FROM *table_name*

WHERE *column_name* BETWEEN *value1* AND *value2;*

The SQL ORDER BY Keyword

The ORDER BY keyword is used to sort the result-set in ascending or descending order.

The ORDER BY keyword sorts the records in ascending order by default. To sort the records in descending order, use the DESC keyword.

ORDER BY Syntax

SELECT *column1*, *column2, ...*

FROM *table_name*

ORDER BY *column1*, *column2, ...* ASC|DESC;

Example

SELECT * FROM Customers

ORDER BY Country;

ORDER BY DESC Example

The following SQL statement selects all customers from the "Customers" table, sorted DESCENDING by the "Country" column:

```
SELECT * FROM Customers

ORDER BY Country DESC;
```

ORDER BY Several Columns Example

The following SQL statement selects all customers from the "Customers" table, sorted by the "Country" and the "CustomerName" column. This means that it orders by Country, but if some rows have the same Country, it orders them by CustomerName:

Eg: SELECT * FROM Customers ORDER BY Country, CustomerName;

SQL Aliases

SQL aliases are used to give a table, or a column in a table, a temporary name.

Aliases are often used to make column names more readable.

An alias only exists for the duration of that query.

An alias is created with the AS keyword.

Alias Column Syntax: SELECT *column_name* AS *alias_name* FROM *table_name;*

Alias Table

Syntax: SELECT *column_name(s)* FROM *table_name* AS *alias_name;*

The SQL UPDATE Statement

The UPDATE statement is used to modify the existing records in a table.

UPDATE Syntax: UPDATE *table_name* SET *column1* = *value1*, *column2* = *value2*, ...

WHERE *condition*;

UPDATE Table

The following SQL statement updates the first customer (CustomerID = 1) with a new contact person *and* a new city.

UPDATE Customers

SET ContactName = 'Alfred Schmidt', City= 'Frankfurt'

WHERE CustomerID = 1;

<u>The SQL SELECT DISTINCT Statement</u>

The SELECT DISTINCT statement is used to return only distinct (different) values.

Inside a table, a column often contains many duplicate values; and sometimes you only want to list the different (distinct) values.

SELECT DISTINCT Syntax:

SELECT DISTINCT *column1*, *column2, ...*FROM *table_name*;

SELECT Example Without DISTINCT

The following SQL statement selects all (including the duplicates) values from the "Country" column in the "Customers" table:

Eg: **SELECT Country FROM Customers;**

Now, let us use the **SELECT DISTINCT statement**

and see the result.

SELECT DISTINCT Examples

The following SQL statement selects only the DISTINCT values from the "Country" column in the "Customers" table:

SELECT DISTINCT Country FROM Customers;

The following SQL statement lists the number of different (distinct) customer countries:

SELECT COUNT(DISTINCT Country) FROM Customers;

<u>The SQL LIKE Operator</u>

The LIKE operator is used in a WHERE clause to search for a specified pattern in a column.

There are two wildcards often used in conjunction with the LIKE operator:

- The percent sign (%) represents zero, one, or multiple characters
- The underscore sign (_) represents one, single character

The percent sign and the underscore can also be used in combinations!

LIKE Syntax

SELECT *column1*, *column2, ...*

FROM *table_name*

WHERE *columnN* LIKE *pattern*;

LIKE Operator	**Description**
WHERE CustomerName LIKE 'a%'	Finds any values that start with "a"
WHERE CustomerName LIKE '%a'	Finds any values that end with "a"
WHERE CustomerName LIKE '%or%'	Finds any values that have "or" in any position
WHERE CustomerName LIKE '_r%'	Finds any values that have "r" in the second position
WHERE CustomerName LIKE 'a_%'	Finds any values that start with "a" and are at least 2 characters in length
WHERE CustomerName LIKE 'a__%'	Finds any values that start with "a" and are at least 3 characters in length
WHERE ContactName LIKE 'a%o'	Finds any values that start with "a" and ends with "o"

What is a NULL Value?

A field with a NULL value is a field with no value.

If a field in a table is optional, it is possible to insert a new record or update a record without adding a value to this field. Then, the field will be saved with a NULL value.

How to Test for NULL Values?

It is not possible to test for NULL values with comparison operators, such as =, <, or <>.

We will have to use the IS NULL and IS NOT NULL operators instead.

IS NULL Syntax

SELECT *column_names*

FROM *table_name*

WHERE *column_name* IS NULL;

IS NOT NULL Syntax

SELECT *column_names*

FROM *table_name*

WHERE *column_name* IS NOT NULL;

The IS NULL Operator

The IS NULL operator is used to test for empty values (NULL values).

The following SQL lists all customers with a NULL value in the "Address" field:

SELECT CustomerName, ContactName, Address

FROM Customers

WHERE Address IS NULL;

The IS NOT NULL Operator

The IS NOT NULL operator is used to test for non-empty values (NOT NULL values).

The following SQL lists all customers with a value in the "Address" field:

SELECT CustomerName, ContactName, Address

FROM Customers

WHERE Address IS NOT NULL;

Aggregate (Group) function

- Aggregate functions are the functions that operate on a set of rows to give one result per group.
- These sets of rows on which group function is applied may be the whole table or the table split into groups.

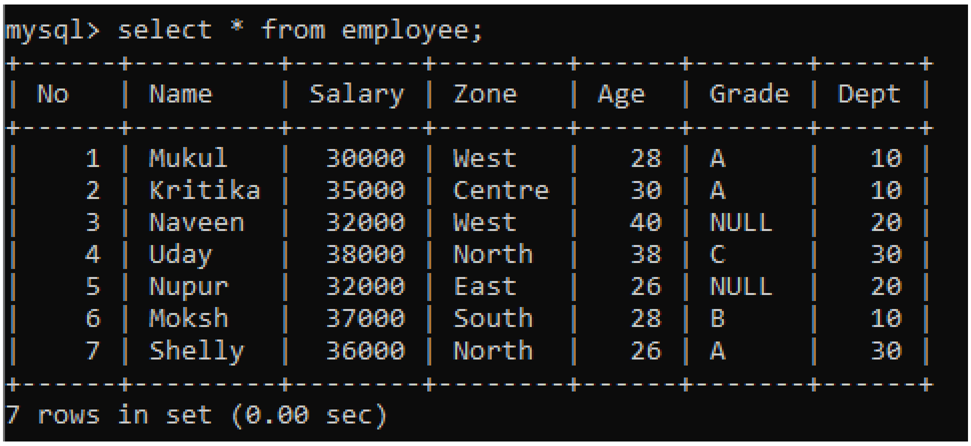
Types of Group Functions

Function	Description
sum()	Find the sum of numeric values
avg()	Find the average of numeric values
count()	Counts the number of rows in a table
max()	Find the maximum among all the values in a column
min()	Find the minimum among all the values in a column

Remember the following points about group functions:

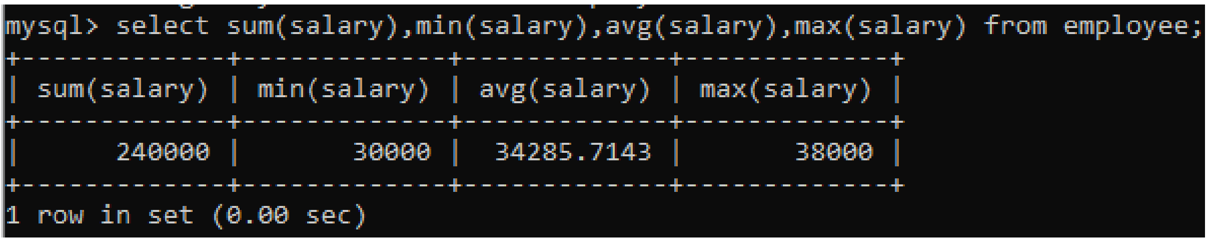
- All group functions,except count(*) ignore NULL values
- Functions -sum(),avg() are used with NUMERIC data.
- Functions -min() and max() can be used with any data type

Example:Consider the following table employee



```
mysql> select * from employee;
+------+---------+--------+--------+------+-------+------+
| No   | Name    | Salary | Zone   | Age  | Grade | Dept |
+------+---------+--------+--------+------+-------+------+
|    1 | Mukul   |  30000 | West   |   28 | A     |   10 |
|    2 | Kritika |  35000 | Centre |   30 | A     |   10 |
|    3 | Naveen  |  32000 | West   |   40 | NULL  |   20 |
|    4 | Uday    |  38000 | North  |   38 | C     |   30 |
|    5 | Nupur   |  32000 | East   |   26 | NULL  |   20 |
|    6 | Moksh   |  37000 | South  |   28 | B     |   10 |
|    7 | Shelly  |  36000 | North  |   26 | A     |   30 |
+------+---------+--------+--------+------+-------+------+
7 rows in set (0.00 sec)
```

Q: Find the sum, average, minimum, maximum value of salaries of employees in the employee table



```
mysql> select sum(salary),min(salary),avg(salary),max(salary) from employee;
+-------------+-------------+-------------+-------------+
| sum(salary) | min(salary) | avg(salary) | max(salary) |
+-------------+-------------+-------------+-------------+
|      240000 |       30000 |  34285.7143 |       38000 |
+-------------+-------------+-------------+-------------+
1 row in set (0.00 sec)
```

Count() function

Count () has got three formats:

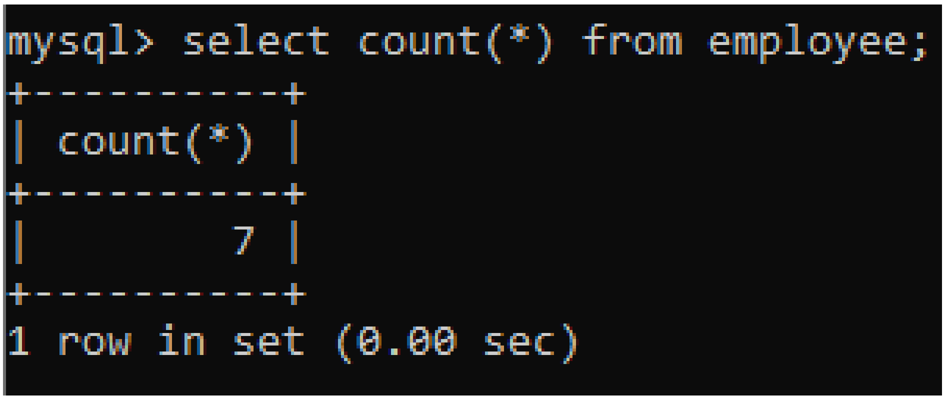
Count(*)

This function returns the number of rows in the table that satisfy the criteria of select statement.

In its counting, it includes duplicate rows and rows with NULL values in any of the column

Example:

Q: Count the number of employees in the employee table.



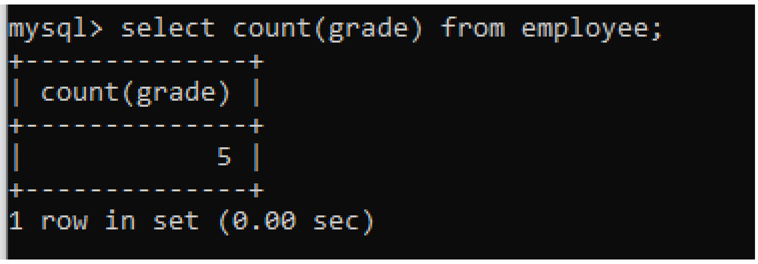
```
mysql> select count(*) from employee;
+----------+
| count(*) |
+----------+
|        7 |
+----------+
1 row in set (0.00 sec)
```

Count(<col name>)

- This function returns the number of not null values in the specified column, but includes duplicate values in counting

Example

Q; count the number of grades of employees in the employee table.



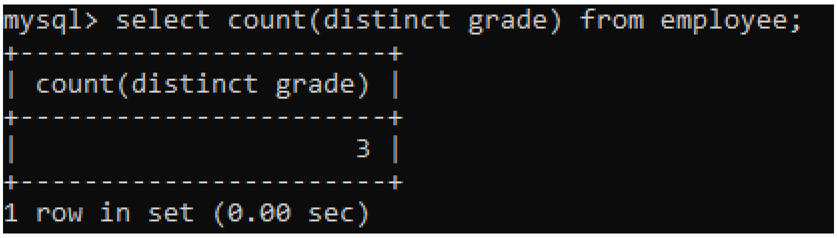
```
mysql> select count(grade) from employee;
+--------------+
| count(grade) |
+--------------+
|            5 |
+--------------+
1 row in set (0.00 sec)
```

Count(DISTINCT <col name>)

This function returns the number of unique, not null values in the specified column.

Example

Q Count the number of different grades of the employee



```
mysql> select count(distinct grade) from employee;
+-----------------------+
| count(distinct grade) |
+-----------------------+
|                     3 |
+-----------------------+
1 row in set (0.00 sec)
```

Grouping Records (Group by clause)

- To divide the rows in a table into smaller groups of information, group by clause is used.
- It combines all identical rows in a group of fields.
- A column name is used for grouping

Syntax;-

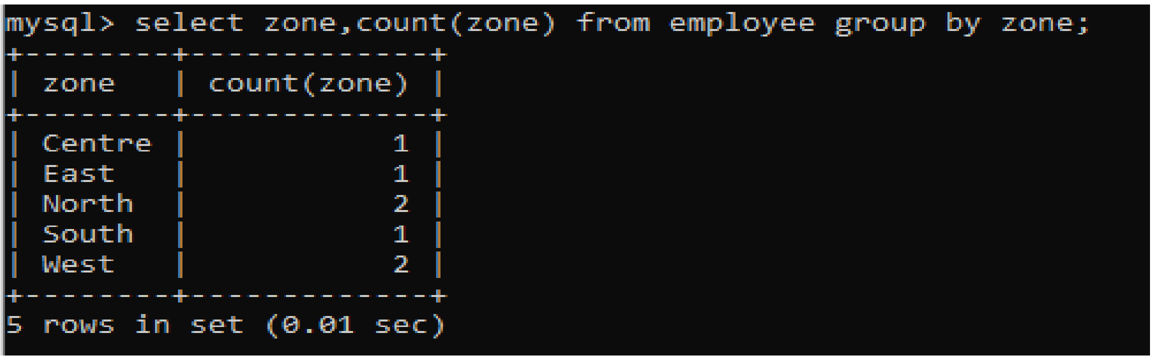
SELECT [DISTINCT] <COL LIST> FROM <TABLE NAME>
[WHERE <CONDITION>]
[GROUP BY < GROUP BY EXPR>]
[HAVING <CONDITION>]
[ORDER BY <COL NAME>/<EXPR> ASC/DESC];

NOTE -

- Group by expression specifies the columns whose values determine the basics for grouping rows
- WHERE clause is always before GROUP BY if required.

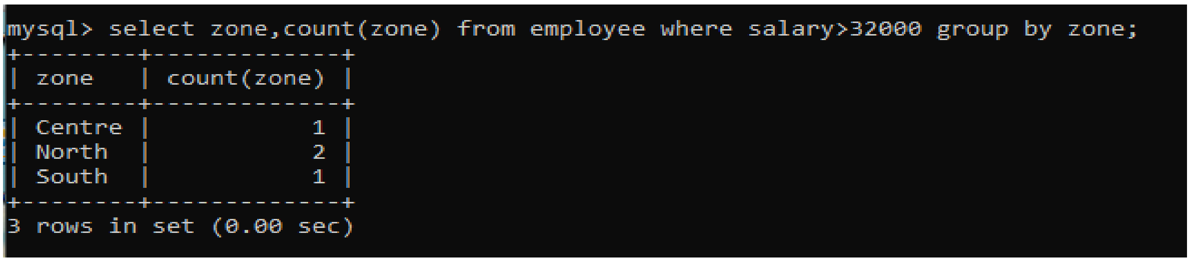
Example

Q. Display the no of employees in each zone.



```
mysql> select zone,count(zone) from employee group by zone;
+--------+-------------+
| zone   | count(zone) |
+--------+-------------+
| Centre |           1 |
| East   |           1 |
| North  |           2 |
| South  |           1 |
| West   |           2 |
+--------+-------------+
5 rows in set (0.01 sec)
```

Q. Display the no of employees in each zone whose salary is greater than 32000



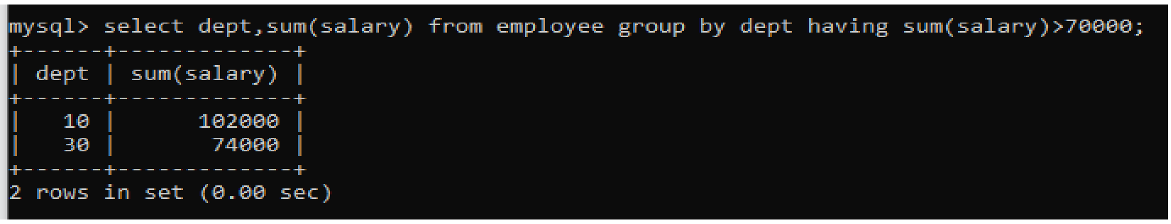
```
mysql> select zone,count(zone) from employee where salary>32000 group by zone;
+--------+-------------+
| zone   | count(zone) |
+--------+-------------+
| Centre |           1 |
| North  |           2 |
| South  |           1 |
+--------+-------------+
3 rows in set (0.00 sec)
```

Having clause

- This clause is used to restrict rows resulting after grouping.
- Steps followed in execution of select with group by and having clause-
  1. Rows are grouped according to the columns in the group by clause.
  2. Then the group function is applied.
  3. Groups matching with Having clauses are displayed.

Example

Q. Display only whose departments with sum of salaries whose total salary is greater than 70000

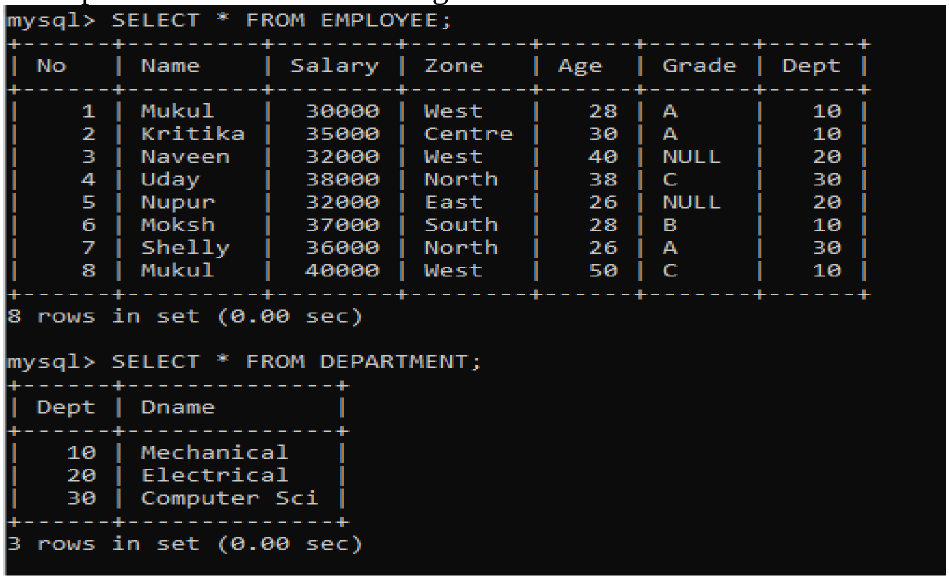


```
mysql> select dept,sum(salary) from employee group by dept having sum(salary)>70000;
+------+-------------+
| dept | sum(salary) |
+------+-------------+
|   10 |      102000 |
|   30 |       74000 |
+------+-------------+
2 rows in set (0.00 sec)
```

Cartesian Product (Cross Join or unrestricted join)

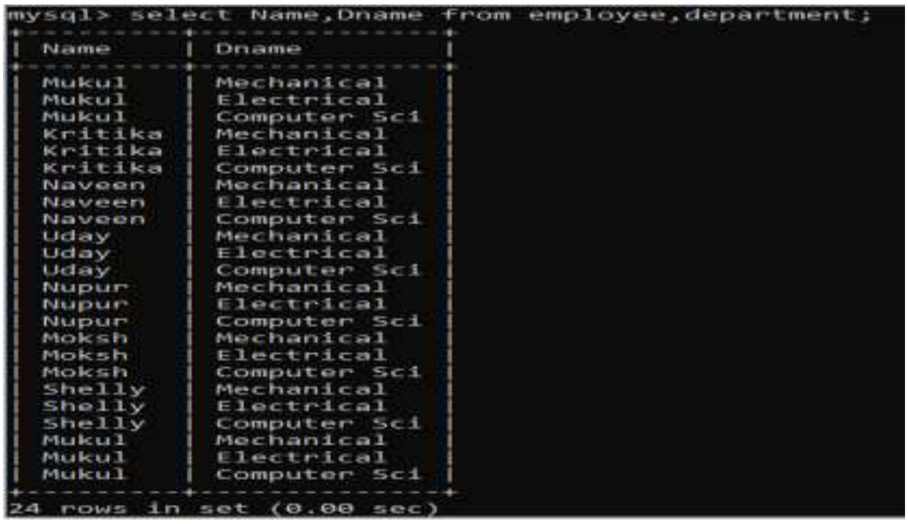
- Returns all the rows in the two tables listed in the query.
- Each row of the first table is paired with all the rows in the second table.
- This happens when there is no relationship between two tables.
- It is rarely useful.

Example- Consider the following tables



```
mysql> SELECT * FROM EMPLOYEE;
+------+---------+--------+--------+------+-------+------+
| No   | Name    | Salary | Zone   | Age  | Grade | Dept |
+------+---------+--------+--------+------+-------+------+
|    1 | Mukul   |  30000 | West   |   28 | A     |   10 |
|    2 | Kritika |  35000 | Centre |   30 | A     |   10 |
|    3 | Naveen  |  32000 | West   |   40 | NULL  |   20 |
|    4 | Uday    |  38000 | North  |   38 | C     |   30 |
|    5 | Nupur   |  32000 | East   |   26 | NULL  |   20 |
|    6 | Moksh   |  37000 | South  |   28 | B     |   10 |
|    7 | Shelly  |  36000 | North  |   26 | A     |   30 |
|    8 | Mukul   |  40000 | West   |   50 | C     |   10 |
+------+---------+--------+--------+------+-------+------+
8 rows in set (0.00 sec)

mysql> SELECT * FROM DEPARTMENT;
+------+--------------+
| Dept | Dname        |
+------+--------------+
|   10 | Mechanical   |
|   20 | Electrical   |
|   30 | Computer Sci |
+------+--------------+
3 rows in set (0.00 sec)
```

Q: To display the name of the employees and their department name



```
mysql> select Name,Dname from employee,department;
+---------+--------------+
| Name    | Dname        |
+---------+--------------+
| Mukul   | Mechanical   |
| Mukul   | Electrical   |
| Mukul   | Computer Sci |
| Kritika | Mechanical   |
| Kritika | Electrical   |
| Kritika | Computer Sci |
| Naveen  | Mechanical   |
| Naveen  | Electrical   |
| Naveen  | Computer Sci |
| Uday    | Mechanical   |
| Uday    | Electrical   |
| Uday    | Computer Sci |
| Nupur   | Mechanical   |
| Nupur   | Electrical   |
| Nupur   | Computer Sci |
| Moksh   | Mechanical   |
| Moksh   | Electrical   |
| Moksh   | Computer Sci |
| Shelly  | Mechanical   |
| Shelly  | Electrical   |
| Shelly  | Computer Sci |
| Mukul   | Mechanical   |
| Mukul   | Electrical   |
| Mukul   | Computer Sci |
+---------+--------------+
24 rows in set (0.00 sec)
```

Joins in MySQL

- A join is used when data from two or more tables is required.
- Rows in one table can be joined to the rows in another table based on the common values existing in corresponding columns of two tables.
- Joins are used to retrieve data from tables related to each other with primary- foreign key relationships.
- There are many types of join

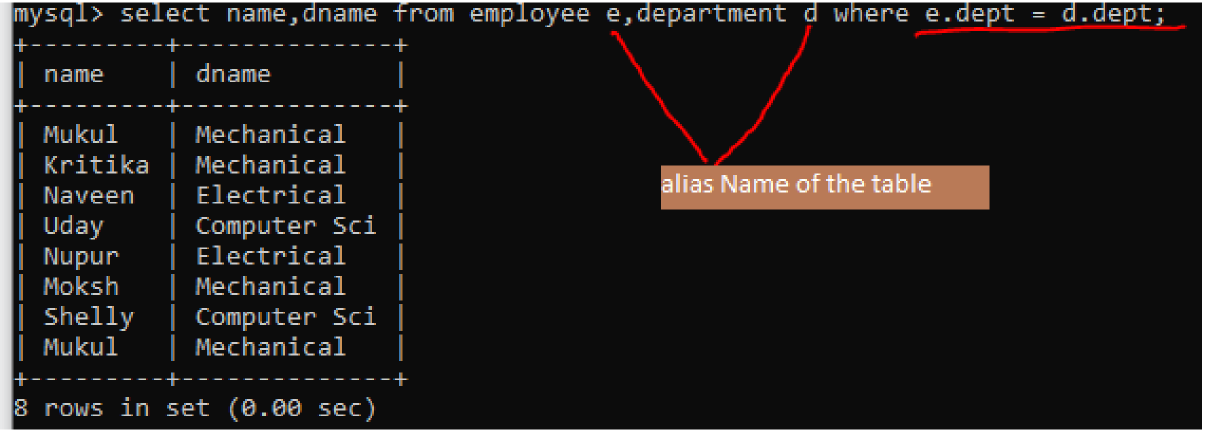
Equi join

- Specified columns from the joining tables are checked for equality.
- Values from joining tables are retrieved only if the condition in where clause is satisfied.

SYNTAX:-

```
SELECT <column_name (s)>
FROM <table_name1>, <table_name2>, ...., <table_nameN>
WHERE <table_name1>.<column_name> = <table_name2>.<column_name>;
```

Q: to display the name of the employee and their department



```
mysql> select name,dname from employee e,department d where e.dept = d.dept;
+---------+--------------+
| name    | dname        |
+---------+--------------+
| Mukul   | Mechanical   |
| Kritika | Mechanical   |
| Naveen  | Electrical   |
| Uday    | Computer Sci |
| Nupur   | Electrical   |
| Moksh   | Mechanical   |
| Shelly  | Computer Sci |
| Mukul   | Mechanical   |
+---------+--------------+
8 rows in set (0.00 sec)
```

Note-

To join n tables together minimum of n-1 join conditions are to be satisfied.

Results of Equi joins are the same as simple joins or inner joins. Inner join (simply join) uses on clause.

You should always qualify the columns when joining tables having the same name as corresponding columns. To qualify the columns we use "." (dot) operator.

Natural Join

This clause is based on all the columns in the two tables that have the same name.

It selects the rows from two tables that have equal values in the matched columns.

SYNTAX:-

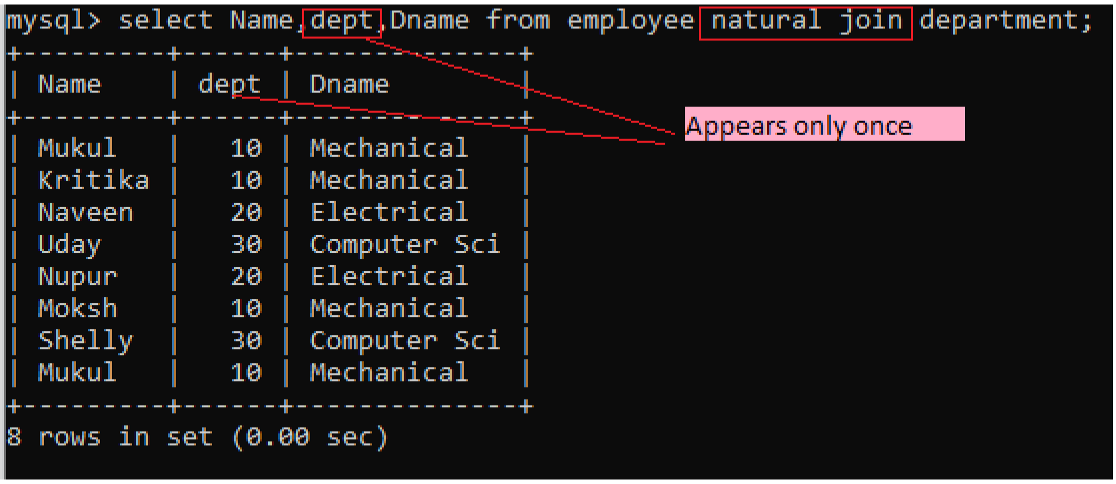
SELECT [column_names | *]

FROM table_name1

NATURAL JOIN table_name2;

Example- consider the same tables employee and department.

Q : To display the name of employee and department of all employee



```
mysql> select Name,dept,Dname from employee natural join department;
+---------+------+--------------+
| Name    | dept | Dname        |
+---------+------+--------------+
| Mukul   |   10 | Mechanical   |
| Kritika |   10 | Mechanical   |
| Naveen  |   20 | Electrical   |
| Uday    |   30 | Computer Sci |
| Nupur   |   20 | Electrical   |
| Moksh   |   10 | Mechanical   |
| Shelly  |   30 | Computer Sci |
| Mukul   |   10 | Mechanical   |
+---------+------+--------------+
8 rows in set (0.00 sec)
```

Appears only once

Note-No need to specify the column names to join. Works with same column name in both the tables. The Resulting table has unique columns.

ASSERTION AND REASONING QUESTIONS

1. Assertion(A): The resultset refers to a logical set of records that are fetched from the database executing an SQL Query.

   Reason (R): Resultset stored in a cursor object can be extracted by using fetch(…) functions.

   (a) Both A and R are true and R is the correct explanation of A
   (b) Both A and R are true and R is not the correct explanation of A
   (c) A is true but R is false
   (d) A is false but R is true

2. Assertion(A): In SQL, aggregate function avg()calculates the average value on a set of values and produce a single result.

   Reason (R): The aggregate functions are used to perform some fundamental arithmetic tasks such as min(), max(), sum() etc.

   (a) Both A and R are true and R is the correct explanation of A
   (b) Both A and R are true and R is not the correct explanation of A
   (c) A is true but R is false
   (d) A is false but R is true

3. Assertion(A): Primary key is a set of one or more attributes that identify tuples in a relation.

   Reason (R): The primary key constraint is a combination of the NOT NULL and UNIQUE constraints.

   (a) Both A and R are true and R is the correct explanation of A
   (b) Both A and R are true and R is not the correct explanation of A
   (c) A is true but R is false
   (d) A is false but R is true

4. Assertion(A): Foreign key is a non-key attribute whose value is derived from primary key of another table.

   Reason (R): Each foreign key refers a candidate key in a relation.

   (a) Both A and R are true and R is the correct explanation of A
   (b) Both A and R are true and R is not the correct explanation of A
   (c) A is true but R is false
   (d) A is false but R is true

5. Assertion(A): The SELECT statement in SQL is used to retrieve data from one or more tables.

   Reason(R): The SELECT statement can be used to retrieve all columns or a subset of columns from a table.

   (a) Both A and R are true and R is the correct explanation of A
   (b) Both A and R are true and R is not the correct explanation of A
   (c) A is true but R is false
   (d) A is false but R is true

6. Assertion(A): The WHERE clause in SQL is used to filter data based on a specified condition.

   Reason(R): The WHERE clause is always required when using the SELECT statement in SQL

   (a) Both A and R are true and R is the correct explanation of A
   (b) Both A and R are true and R is not the correct explanation of A
   (c) A is true but R is false
   (d) A is false but R is true

7. Assertion (A): The ORDER BY clause in SQL is used to sort the result set in ascending or descending order.

   Reason(R): The ORDER BY clause can only be used with the SELECT statement in SQL.

   (a) Both A and R are true and R is the correct explanation of A
   (b) Both A and R are true and R is not the correct explanation of A
   (c) A is true but R is false
   (d) A is false but R is true

8. Assertion (A): The IN operator in SQL is used to match a value with a list of values.

   Reason(R): The IN operator can only be used with the where clause in SQL.

   (a) Both A and R are true and R is the correct explanation of A
   (b) Both A and R are true and R is not the correct explanation of A
   (c) A is true but R is false
   (d) A is false but R is true

9. Assertion (A): The between operator in SQL is used to match a value with in a range of values.

   Reason(R): The between operator can only be used with numerical values in SQL.

   (a) Both A and R are true and R is the correct explanation of A
   (b) Both A and R are true and R is not the correct explanation of A
   (c) A is true but R is false
   (d) A is false but R is true

10. Assertion (A): The LIKE operator in SQL is used to match a value with a pattern.

    Reason(R): The % and _ wildcard characters can be used with the LIKE operator in SQL.

    (a) Both A and R are true and R is the correct explanation of A
    (b) Both A and R are true and R is not the correct explanation of A
    (c) A is true but R is false
    (d) A is false but R is true

ANSWERS

1. (b)
2. (b)
3. (a)
4. (b)
5. (a)
6. (b)
7. (a)
8. (b)
9. (b)
10. (a)

I	**Interface Python with SQL database** **Database connectivity** Database connectivity refers to connection and communication between an application and a database system. The term “front-end” refers to the user interface, while “back-end” means the server, application and database that work behind the scenes to deliver information to the user. ● Mysql.connector Library or package to connect from python to MySQL. Before we connect the program with mysql , we need to install connectivity package named mysql-connector- python ● Command to install connectivity package pip install mysql-connector-python ● Command to import connector import mysql.connector
12	**Steps for creating database connectivity applications** **i) Start Python** start python editor to create our own python script.

ii) Import mysql.connector package

Import statement as shown below:

```
import mysql.connector
                 or
import mysql.connector as con
```

iii) Establishing a connection to MySQL DATABASE.

We need to establish a connection to a mysql database using connect () function of mysql.connector package.

The connect statement creates a connection to the mysql server and returns a

MySQL connection object.

A database connection object controls the connection to the database. It represents a unique session with a database connected from within a program.

Syntax:

```
<Connection object> = mysql.connector.connect
(host=<hostname>,user=<username>, passwd=<password>, database=<database>)
```

Hostname-The hostname of database server or IP address

User -Username on MYSQL

Password-Password of the database user

Database-Name of the database

```
import mysql.connector
con=mysql.connector.connect(host=”localhost”,user=”root”, passwd=”manager “ ,
database=”student12”)
```

iv) Creating a cursor Object

It is a useful control structure of database connectivity. It will let us execute all the queries we need. Cursor stores all the data as a temporary container of returned data and allows traversal so that we can fetch data one row at a time from cursor. Cursors are created by the <connectionobject>.cursor()method.

A Database cursor is a special control structure that facilitates the row by row processing of records in the result set.

Syntax:

<cursor object>=<connectionobject>.cursor()

```
cursor=con.cursor()
```

v) Execute SQL query:

We can execute SQL query using<cursorobject>. execute() function .

Syntax:

<cursor object>.execute(<sql query string>)

```
cursor.execute(“select * from student”)
```

The above code will execute the SQL query and store the retrieved records

(resultset) in the cursor object(cursor).

Result set refers to a logical set of records that are fetched from the database by executing an SQL query and made available in the program.

vi) Extract data from Result set:

The records retrieved from the database using SQL select query has to be extracted as record from the result set. We can extract data from the result set using the following fetch () function.

➢ Multiple ways to retrieve data:

fetchall()

fetchone()

fetchmany()

- ❖ fetchall()

Fetches all the rows from the result set in the form of a tuple containing the records.

- ❖ fetchone()

Fetches only one row from the result set in the form of a tuple containing a record

- ❖ fetchmany (size)

Fetches the number of rows from the result set in the form of a tuple containing the records.

Syntax:

```
<data>=<cursor>.fetchall()

<data>=<cursor>.fetchone()

<data>=<cursor>.fetchmany(size)
```

```
data=cursor.fetchall()
for row in data:
    print(row)
```

vii) Closing cursor and connection

we must close the connection established.

Syntax:

```
  <connectionobject>.close()
    con.close()
```

3 **Python connectivity program to display entire content from a table.**

```
import mysql.connector
mycon=mysql.connector.connect(host="localhost",user="root",passwd="manager",
database="class12")
if mycon.is_connected():
   print("successfully connected")
   cursor=mycon.cursor()
   st="select * from employee"
   cursor.execute(st)
   data=cursor.fetchall()
   count=cursor.rowcount
   print("Number of records in the employee table",count)
   print("Details of the records of the employee")
   for row in data:
      print(row)
   mycon.close()
else:
   print("connection problem")
```

4 **Python connectivity program to display the record from a table by using a parameterised query.**

```
import mysql.connector
mycon=mysql.connector.connect(host="localhost",user="root",passwd="manager",
database="class12")
if mycon.is_connected():
   print("successfully connected")
   cursor=mycon.cursor()
   eid=input("enter the employee id")
   st="select * from employee where empid='{}'".format(eid)
```

```
    cursor.execute(st)
    data=cursor.fetchall()
    print("Details of the records of the employee")
    for row in data:
        print(row)
    mycon.close()
else:
    print("connection problem")
```

5 **Python connectivity program to insert a new record to a table by using a parameterised query.**

```
import mysql.connector
con=mysql.connector.connect(host="localhost",user="root",password="manager",
database="class12")
if con.is_connected():
    print("successfully connected")
    cur=con.cursor()
    empid=int(input("enter empid:"))
    ename=input("enter name:")
    desg=input("enter designation:")
    sex=input("enter sex:")
    salary=float(input("enter salary:"))
    cur=con.cursor()
    str=("insert into
employee(empid,ename,desg,sex,sal)values({},'{}','{}','{}','{}')".format
    (empid,ename, desg,sex,salary))
    cur.execute(str)
    con.commit()
    print("successfully inserted")
    con.close()
```

```
else:
    print("connection problem")
```

6 **Python connectivity program to update a record of a table by using parameterised query.(Update salary of employees using employee id)**

```
import mysql.connector
con=mysql.connector.connect(host="localhost",user="root",password="manager",
database="class12")
if con.is_connected():
    print("successfully connected")
    cur=con.cursor()
    eid=int(input("enter empid:"))
    salary=float(input("enter salary:"))
    cur=con.cursor()
    str=("update employee set sal={}where empid={}").format(salary,eid)
    cur.execute(str)
    con.commit()
    print("successfully updated record")
    con.close()
else:
    print("connection problem")
```

7 **Python connectivity program to delete a record from table by using parameterised query.(Delete employee details by using employee id)**

```
import mysql.connector
con=mysql.connector.connect(host="localhost",user="root",password="manager",database="class12")
if con.is_connected():
    print("successfully connected")
    cur=con.cursor()
    eid=int(input("enter empid:"))
    cur=con.cursor()
    str=("delete from employee where empid={}").format(eid)
    cur.execute(str)
    con.commit()
    print("successfully deleted record:")
    con.close()
else:
    print("connection problem")
```

8 **You can try some thing more……**

i)TO CREATE DATABASE SCHOOL USING PYTHON INTERFACE

```
import mysql.connector
mydb=mysql.connector.connect(host="localhost",user="root",
passwd="system")
 mycursor=mydb.cursor()
mycursor.execute("CREATE DATABASE SCHOOL")
```

ii) SHOW DATABASE

```
import mysql.connector
mydb=mysql.connector.connect(host="localhost",user="root",
passwd="system")
mycursor=mydb.cursor()
mycursor.execute("SHOW DATABASES")
for x in mycursor:
    print (x)
```

ii) TO CREATE A TABLE IN MYSQL USING PYTHON INTERFACE

```
import mysql.connector
mydb=mysql.connector.connect(host="localhost",user="root",

passwd="system",database="student")

mycursor=mydb.cursor()

mycursor.execute("CREATE TABLE FEES (ROLLNO INT,NAME
VARCHAR(20), AMOUNT INT);")
```

iv) TO SHOW THE TABLES IN MYSQL USING PYTHON INTERFACE

```
import mysql.connector
mydb=mysql.connector.connect(host="localhost",user="root",

passwd="system",database="student")

mycursor=mydb.cursor()

mycursor.execute("SHOW TABLES")

for x in mycursor:

        print(x)
```

v) TO DESCRIBE TABLE STRUCTURE USING PYTHON INTERFACE

```
mydb=mysql.connector.connect(host="localhost",user="root",

passwd="system",database="student")

mycursor=mydb.cursor()

mycursor.execute("DESC STUDENT")

for x in mycursor:

  print(x)
```

vi) TO DROP AN ENTIRE TABLE FROM MYSQL DATABASE USING PYTHON INTERFACE

```
import mysql.connector
mydb=mysql.connector.connect(host="localhost",user="root",

passwd="system",database="student")
```

```
mycursor=mydb.cursor()

mycursor.execute("DROP TABLE STUDENT")
```

vii) TO ADD A COLUMN IN THE EXISTING TABLE USING PYTHON INTERFACE

```
import mysql.connector
mydb=mysql.connector.connect(host="localhost",user="root",

passwd="system",database="student")

mycursor=mydb.cursor()

mycursor.execute("ALTER TABLE STUDENT ADD AGE
INT”)

mydb.commit()
```

viii) TO DROP A COLUMN FROM THE TABLE USING PYTHON INTERFACE

```
import mysql.connector
mydb=mysql.connector.connect(host="localhost",user="root",

passwd="system",database="student")

mycursor=mydb.cursor()

mycursor.execute("ALTER TABLE DROP AGE ”)

mydb.commit()
```

ix) TO ALTER THE DATATYPE OF A COLUMN IN A TABLE USING PYTHON INTERFACE

```
import mysql.connector

mydb=mysql.connector.connect(host="localhost",user="root",

passwd="system",database="student")

mycursor=mydb.cursor()

mycursor.execute("ALTER TABLE STUDENT MODIFY GRADE
CHAR(3)")
```

Questions

1 **Write a python connectivity program to display only first three records from a table?**

```
import mysql.connector
mycon=mysql.connector.connect(host="localhost",user="root",passwd="manager",
database="class12")
if mycon.is_connected():
    cursor=mycon.cursor()
    st="select * from employee"
    cursor.execute(st)
    data=cursor.fetchmany(3)
    print("Details of the records of the employee")
    for row in data:
        print(row)
    mycon.close()
else:
    print("connection problem")
```

2 **Write a python connectivity program to display first record from a table?**

```
import mysql.connector
mycon=mysql.connector.connect(host="localhost",user="root",passwd="manager",
database="class12")
if mycon.is_connected():
    cursor=mycon.cursor()
    st="select * from employee"
    cursor.execute(st)
    data=cursor.fetchone()
    print("Details of the records of the employee")
    print(data)
    mycon.close()
```

```
else:
   print("connection problem")
```

3 **Write a python connectivity program to display the records of all female managers from a table named employee and also find out the number of female managers. If the details are not existing, then print “Record Not Found” message?**

```
import mysql.connector
mycon=mysql.connector.connect(host="localhost",user="root",passwd="manager",
database="class12")
if mycon.is_connected():
   cursor=mycon.cursor()
   st="select * from employee where desg='Manager' and sex='F'"
   cursor.execute(st)
   data=cursor.fetchall()
   count=cursor.rowcount
   if count>0:
      print("Number of Female Managers:",count)
      print("Details of the records of the employee")
      for row in data:
         print(row)
      mycon.close()
   else:
      print("Record not found")
else:
   print("connection problem")
```

4 **Write a python connectivity program to display the records of all managers from a table named employee using parameterised query and also findout the number of managers. If the details are not existing, then print “Record Not Found” message?**

```
import mysql.connector
```

```
mycon=mysql.connector.connect(host="localhost",user="root",passwd="manager",
database="class12")
if mycon.is_connected():
    cursor=mycon.cursor()
    designation=input("enter the designation")
    st="select * from employee where desg='{}'".format(designation)
    cursor.execute(st)
    data=cursor.fetchall()
    count=cursor.rowcount
    if count>0:
        print("Number of Managers",count)
        print("Details of managers from employee table")
        for row in data:
            print(row)
        mycon.close()
    else:
        print("Record not found")
else:
    print("connection problem")
```

5 **Write a python connectivity program to update salary from a table named employee .Use parameterised query and count the number of updated record. If the details are not existing,then print “Record Not Found” message?**

```
import mysql.connector
con=mysql.connector.connect(host="localhost",user="root",password="manager",
database="class12")
if con.is_connected():
    print("successfully connected")
    cur=con.cursor()
    eid=int(input("enter empid:"))
    salary=float(input("enter salary:"))
    cur=con.cursor()
    str=("update employee set sal={}where empid={}").format(salary,eid)
    cur.execute(str)
    con.commit()
    count=cur.rowcount
    if count>0:
        print("successfully updated record:",count)
        con.close()
    else:
        print("Record not found")
else:
    print("connection problem")
```

6 **Write a python connectivity program to delete the record from a table named employee using parameterised query and also count the number of deleted record. If the details are not existing,then print "Record Not Found" message?**

```
import mysql.connector
con=mysql.connector.connect(host="localhost",user="root",password="manager",database="class12")
if con.is_connected():
    cur=con.cursor()
    eid=int(input("enter empid:"))
    cur=con.cursor()
    str=("delete from employee where empid={}").format(eid)
    cur.execute(str)
    con.commit()
    count=cur.rowcount
    if count>0:
        print("successfully deleted record:",count)
        con.close()
    else:
        print("Record not found")
else:
    print("connection problem")
```

CBSE TYPE QUESTIONS AND ANSWERS

7 **Shyam has created a table named Student in MYSQL database, SCHOOL:**

- **rno(Roll number)- integer**
- **name(Name) - string**
- **Per(Percentage) – float**

Note the following to establish connectivity between Python and MySQL:

•Username - root • Password - manager • Host – localhost

Shyam, now wants to display the records of students whose percentage is more than 90. Help Shyam to write the program in Python

```
import mysql.connector
mycon=mysql.connector.connect(host="localhost",user="root",passwd="manager",
database="school")
cursor=mycon.cursor()
per=90
st="select * from student where per>{}".format(per)
cursor.execute(st)
data=cursor.fetchall()
print("Details of the records of the employee")
for row in data:
    print(row)
mycon.close()
```

8 **Shyam has created a table named 'Student' in MYSQL database, 'SCHOOL':**

- **rno(Roll number)- integer**
- **name(Name) - string**
- **Per(Percentage) – float**

Note the following to establish connectivity between Python and MySQL:

•Username - root • Password - manager • Host – localhost

Shyam, now wants to insert records of students . Help Shyam to write the program in Python.

```
import mysql.connector
con=mysql.connector.connect(host="localhost",user="root",password="manager",
database="school")
cur=con.cursor()
rno=int(input("Enter the Roll number:"))
name=input("Enter the name")
per=float(input("enter the percentage:"))
cur=con.cursor()
str=("insert into student(rno,name,per)values({},'{}',{})").format(rno,name,per)
cur.execute(str)
con.commit()
print("successfully inserted record:")
con.close()
```

9 **Shyam has created a table named ‘Student’ in MYSQL database, ‘SCHOOL’:**

- **rno(Roll number)- integer**
- **name(Name) - string**
- **Per(Percentage) – float**

Note the following to establish connectivity between Python and MySQL:

•Username - root • Password - manager • Host – localhost

Shyam, now wants to increase the percentage of students by 2% ,who got less than 40 percentage. Help Shyam to write the program in Python.

```
import mysql.connector
con=mysql.connector.connect(host="localhost",user="root",password="manager",
database="school")
cur=con.cursor()
per=40
```

```
cur=con.cursor()

str=("update student set per=per+2 where per<{}").format(per)

cur.execute(str)

con.commit()

print("successfully updated record:")  con.close()
```

10 **Shyam has created a table named ‘Student’ in MYSQL database, ‘SCHOOL’:**

• rno(Roll number)- integer

• name(Name) - string

• Per(Percentage) – float

Note the following to establish connectivity between Python and MySQL:

•Username - root • Password - manager • Host – localhost

Shyam, now wants to delete the records of students who got less than 33 percentage. Help Shyam to write the program in Python.

```
import mysql.connector

con=mysql.connector.connect(host="localhost",user="root",password="manager",

database="school")

cur=con.cursor()

cur=con.cursor()

per=33

str=("delete from student where per<{}").format(per)

cur.execute(str)

con.commit()

print("successfully deleted record:")

con.close()
```

****

Sample Paper - Computer Science (083)

Class : XII

Maximum Marks: 70 **Time Allowed: 3 hours**

General Instructions:

- Please check this question paper contains 35 questions.
- The paper is divided into 4 Sections- A, B, C, D and E.
- Section A, consists of 18 questions (1 to 18). Each question carries 1 Mark.
- Section B, consists of 7 questions (19 to 25). Each question carries 2 Marks.
- Section C, consists of 5 questions (26 to 30). Each question carries 3 Marks.
- Section D, consists of 2 questions (31 to 32). Each question carries 4 Marks.
- Section E, consists of 3 questions (33 to 35). Each question carries 5 Marks.
- All programming questions are to be answered using Python Language only.

SECTION A		
1.	State True or False Python supports dynamic typing.	1
2.	Identify the odd one from this list: a) Tuple b) Dictionary c) List d) String	1

3.	Choose the correct option.
	d = { “Naturals”:100 , “Amul” : 200, “BR”: 300 , “Amul” : 500 }
	print(sorted(d))
	a) { “Naturals”:100 , “Amul” : 200, “BR”: 300 , “Amul” : 500 }
	b) [“Amul”, “Amul”, “BR”, “Naturals”]
	c) [“Amul”, “BR”, “Naturals”]
	d) {“Amul”:500, “BR”:300, “Naturals”:100}
		1

4.	Evaluate the following expression: 20+10* 2**2**3 a) 2580 b) 660 c) 7680 d) 540	1

5.	If S=’computer science model examination ’, what will the followingcommand return? print(S.capitalize()) a) Computer science model examination b) Computer Science Model Examination c) computer science model examination d) COMPUTER SCIENCE MODEL EXAMINATION	
6	Identify the function which is not a MySQL function. a) Sum() b) Count() c) Max() d) Average()	
7.	Choose the correct option from the following: The tell() function a) Places the file handle at the position that we specify b) Sets the file handle at the beginning of the file c) Returns the byte number of the position of the file handle. d) Brings the file handle to the end of the file	1
8.	The command used to insert a new column to an already existing table is: a) Update b) Alter c) Insert d) modify	1
9.	Assuming the table emp to exist, what will the following command do? Delete from emp; a) Will delete the table emp b) Will delete all the attributes in it c) Will delete all the tuples in it. d) Wrong syntax	1

10.	Find the erroneous statement from the following:
	
	Str=’Hard Work Is The Key To Success’
	print(Str) # Statement 1
	for I in len(str): # Statement 2
	  if Str[I]==’ ‘: # Statement 3
	    Str[I]=’ - ’ # Statement 4
	
	Str=’Good Luck’ # Statement 5
	
	(a) Statement 2
	(b) Statement 4
	(c) Statement 5
	(d) Statement 2 and 4
		1

11.	An Alternate key is a -----------------key, but not a Primary Key. a) Candidate b) Unique c) Foreign d) None of these	1
12.	Identify the correct option(s) to open the text file in read mode. i) sf =open('student.txt','w') ii) sf = open('student.txt','r') iii) sf = open('student.txt') iv) sf = open('student.txt','rb') a) Option i b) both iii & iv c) both ii and iii d) both ii and iv	1
13.	The wild card character '_' indicates: a) one character b) Any number of characters c) Specific number of characters mentioned d) One column	1
14.	Fill in the blank: _______Protocol is used to browse the internet. (a) VoIP (b) SMTP (c) PPP (d)HTTP	1
15.	What will be the output of the following: A=”Model Exam” B=A print(id(A)==id(B)) a) True b) False c) 0 d) 1	1
16.	Which of the following is an optional argument of the connect()function in python mysql.connector module? a) Host b) User c) Password d) database	1

Q17 and 18 are ASSERTION AND REASONING based questions. Mark the correct choice as (a) Both A and R are true and R is the correct explanation for A (b) Both A and R are true and R is not the correct explanation for A (c) A is True but R is False (d) A is false but R is True		
17.	Assertion (A):- The default delimiter in CSV files is comma(,). Reasoning (R):- The default delimiter of the CSV file can be overridden with the help of the 'delimiter' attribute.	1
18.	Assertion (A):- If the arguments in a function call statement match the number and order of arguments as defined in the function definition, such arguments are called positional arguments. Reasoning (R):- During a function call, the argument list first contains default argument(s) followed by positional argument(s).	1
	SECTION B	
19.	(i) Expand the following terms: POP3 , URL (ii) Give one difference between Web Server and Web Site. OR (i) Write down any one advantage of computer networks (ii) Define Gateway	1+1

20.	Roshan tried to define a function to input an integer and print its factors. But he encountered certain syntax and logical errors in his code. Rewrite the corrected code and **underline** the corrections made.
	def factors():
	    n=input("Enter the number")
	    for i in range(0,n)
	If n% i = 0 :
	print(i)
		2

21.	Write a function countNow(STUDENT) in Python, that takes the dictionary, STUDENT as an argument and displays the names (in uppercase) having 4 characters. For example, Consider the following dictionary STUDENT={1:"ARUN",2:"AVINASH",3:"PRIYA",4:"NIHA",5:"JEWEL"} The output should be: ARUN NIHA OR Write a function, JUMBLE(STRING), that takes a string as an argument and returns another string where uppercase letter is converted to lowercase and vice	1

	versa. For example, if the string is "HeLlO", then the output will be :hElLo	

22.	Predict the output of the Python code given below:
	
	D = ["P",20,"R",10,"S",30]
	T = 0
	A = ""
	B = 0
	for C in range(1,6,2):
	    T= T + C
	    A= A + D[C-1]+"$"
	    B = B + D[C]
		2

23	Write the Python statement for each of the following tasks using BUILT-IN functions/methods only: (i) To delete last element in the list `L1`. (ii) To find the length of a string named `message` **OR** A list named MARK stores marks obtained by a student in 5 subjects. Write the Python command to import the required module and (using built-in function) to display the average mark of the student.	1+1
24.	Ms. Shalu has just created a table named “Staff” containing Columns Sname, Department and Salary. After creating the table, she realized that she has forgotten to add a primary key in the table. Help her in writing an SQL command to add a primary key - StaffId of integer type to the table Staff. Thereafter, write the command to insert the following record in the table: StaffId - 111 Sname- Shalu Department: Marketing Salary: 45000 **OR** Kiran is working in a database named SCHOOL, in which he has created a table named “STUDENT” containing columns UID, SName, Gender and Category. After creating the table, he realized that the attribute, Category has to be deleted from the table and a new attribute Stream of data type string has to be added. This attribute Stream cannot be left blank. Help him to write the commands to complete both the tasks.	2

25.	Predict the output :
	
	def Find (P, Q = 20) :
	    P = P + Q
	   Q = P - Q
	   print (P,"@",Q)
	return P
	
	R =380
		2

```
S= 120
R=Find(R, S)
print(R,"*",S)
S=Find(S)
print(R,"*",S)
print (T,B,sep='#')
print(A)
```

SECTION C

26. Predict the Output : 3

```
Text1="ChandraYAAN-3"
Text2=""
I=0
while I<len(Text1):
        if Text1[I]>="0" and Text1[I]<="9":
                Val = int(Text1[I])
                Val = Val + 1
                Text2=Text2 + str(Val)
        elif Text1[I]>="A" and Text1[I] <="Z":
                Text2=Text2 + (Text1[I+1])
        elif Text1[I]>="a" and Text1[I] <="z":
            Text2=Text2 + Text1[I].upper()
        else:
                Text2=Text2 + "*"
        I=I+1
print (Text2)
```

27 Consider the given table Employee and answer the following questions

EmpID	EmpName	Designation	Salary	DoJ
E101	Sam	Systems Engineer	30000	2020-09-12
E102	Krish	Technical Lead	35000	2020-03-07
E103	Mia	Lead Consultant	43000	2019-02-16
E104	Isabel	Manager	50000	2018-02-20
E105	Nasrin	Lead Consultant	50000	2018-10-04
E106	Ayan	Systems Engineer	32000	2021-01-15
E107	Sen	Manager	55000	2017-11-25

Write the output of the queries based on Employee table

(i) select EmpName, Designation from Employee where EmpName like '____';

```
(ii)  select distinct Designation from Employee;


 (iii)select EmpID,EmpName, Designation from Employee where Designation in
('Technical Lead', 'Lead Consultant’);
```

28. Write a Python function ReversWords() to read the text file “Python.txt” and display the words starting with ‘P’ or ‘p’ in reverse order and all other words as such. 3

Eg:If the file contains the following text:

Python is a popular programming language.

Then it is to be displayed as->

nohtyP is a ralupop gnimmargorp language.

OR

Write a function AMCount() to display the no.of occurrences of ‘A’ and ‘M’ (‘a’ and ‘m' also)separately of a text file STORY.TXT.

29. **Consider the table EMPLOYEES shown below and write SQL commands for (i) to (iii)** 3

EMP ID	NAME	DESIGNATION	SALARY	BENEFITS	ADDRESS	CITY
200	Akash	Manager	15000	2000	Christ Nagar	Jaipur
201	Ajit	Manager	12000	1500	Delhi Road	Agra
202	Ramesh	Salesman	8000	800	Bandra	Mumbai
203	Deepak	Clerk	5000	500	Holy Palace	Mumbai
204	Aman	Officer	10000	1000	Badarwa	Haryana

(i) To display Name , Designation and Address of all employees whose name starts with character ‘A’

(ii) To display the NAME,DESIGNATION,ADDRESS and TOTAL SALARY of all employees where TOTAL SALARY is calculated as SALARY + BENEFIITS

(iii) To display the number of employees designation wise

30. Write Addnew(Book) and Remove(Book) functions in python to add a new book and remove a book from a List of books, considering them to act as PUSH and POP operations of the data structure Stack. 3

SECTION D

31 Consider the following tables DOCTOR and SALARY. Write SQL commands for the following statements. 4

TABLE : DOCTOR

ID	NAME	DEPT	SEX	EXPERIENCE
101	JOHN	ENT	M	12
104	SMITH	ORTHOPEDIC	M	6
107	F GEORGE	CARDIOLOGY	M	11
114	LARA DUTT	SKIN	F	10
109	K GEORGY	MEDICINE	F	4
105	JOHNSON	ORTHOPEDIC	M	7
117	LUCY	ENT	F	8
111	BILL G	MEDICINE	F	15
130	MORPHY	ORTHOPEDIC	M	16

TABLE : SALARY

ID	BASIC	ALLOWANCE	CONSULTATION
101	12000	1000	300
104	24000	2300	500

107	32000	3200	500
114	12000	1000	600
109	41000	1700	100
105	27100	1100	200
117	11000	1200	300
111	18900	1600	200
130	42000	2600	300

(i) Display NAME of all doctors who are in “ORTHOPEDIC” having more than 10 years’ experience from the table DOCTOR.

(ii) Display the average salary of all doctors working in “ENT” department using the DOCTOR and SALARY. (Salary= Basic + Allowance)

(iii) Display the minimum ALLOWANCE of female doctors.

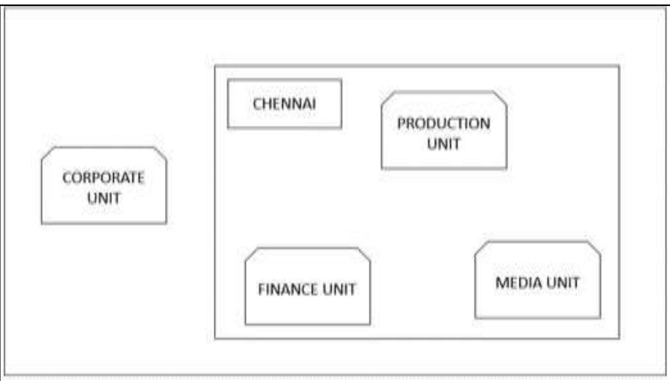
(iv) Display the highest consultation fee amount for all male doctors.

32 Write a Program in Python that defines and calls the following user defined functions: 4

a) **add()** – To accept and add data of an employee to a CSV file ‘furdata.csv’. Each record consists of a list with field elements as fid, fname and fprice to store furniture id, furniture name and furniture price respectively.

b) **search()**- To display the records of the furniture whose price is more than 10000.

33 SAP Methodologies is planning to expand their network in India starting with three cities in India to build infrastructure for research and development of their chemical products. The company has planned to setup their main office in Chennai at three different locations and have named their offices as Production Unit, Finance Unit and Media Unit. The company has one more research office namely Corporate Unit in Mumbai. A rough layout of the same is as follows:



Approximate distance between these units is as follows:

PRODUCTION UNIT	FINANCE UNIT	70M
PRODUCTION UNIT	MEDIA UNIT	15kM
PRODUCTION UNIT	CORPORATE UNIT	2112 Km
FINANCE UNIT	MEDIA UNIT	15Km

WIN G	**NO.COMPUTERS**
PRODUCTION UNIT	150
FINANCE UNIT	35
MEDIA UNIT	10
CORPORATE UNIT	30

Suggest the kind of network required (out of LAN, MAN, WAN) for each of the following units:

(a) Production Unit and Media Unit

(b) Production Unit and Finance Unit

(ii) Name the device you suggest for connecting all computers within each unit?

(iii) Which of the following communication media, will you suggest to be procured by the company for connecting their local office units in Chennai for very effective (high speed) communication?

(a) Telephone cable

(b) Optical fiber

(c) Ethernet cable

(iv) Among the different units in Chennai, which unit can be made the Server. Justify your reason.

(v) Suggest a cable/wiring layout for connecting the company's local office units located in Chennai.

34 i)Differentiate between binary file and csv file: 2+3

ii)Raghav is a Python programmer , created a binary file record.dat with employee id, ename and salary. The file contains 10 records. He now has to update a record based on the employee id entered by the user and update the salary. The updated record is then to be written in the file temp.dat. The records which are not to be updated also have to be written to the file temp.dat. If the employee id is not found, an appropriate message should to be displayed.

OR

i)Differentiate between seek( ) and tell( ) functions

ii)Anushka wants to increase the price by 10% for those items which are less than 250. The details of the items (itemno, iname and price) are stored in a file named 'items.dat'. She wants to store the modified data in another file named items_copy.dat. Anushka has written a function named Price_Change() . Help Anushka to accomplish the task

35 Define the term Domain with respect to RDBMS. Give one example to support your answer.

Sunil wants to write a program in Python to read a record from the table named student and displays only those records who have marks between 80 and 95:

RollNo – integer
Name – string
Clas – integer
Marks – integer
Note the following to establish connectivity between Python and MYSQL:
· Username is root
· Password is tiger
The table exists in a MYSQL database named vidyalaya.

OR

i)Define Candidate Key

ii)Write a program to insert the following record in the table Student:

RollNo – integer

Name – string

Class – integer

Marks – integer

Note the following to establish connectivity between Python and MYSQL:

Username is root

Password is abcd

The table exists in a MYSQL database named XIICS.

The details (RollNo, Name, Class and Marks) are to be accepted from the user.

Sample Paper – Marking Scheme

Computer Science (083)

Class : XII

	SECTION A	
1.	State True or False Python supports dynamic typing **Ans: True**	1
2.	Identify the odd one from this list: a)Tuple b) **Dictionary** c) List d) String	1

3.	Choose the correct option.
	d = { “Naturals”:100 , “Amul” : 200, “BR”: 300 , “Amul” : 500 }
	print(sorted(d))
	a) { “Naturals”:100 , “Amul” : 200, “BR”: 300 , “Amul” : 500 }
	b) [“Amul”, “Amul”, “BR”, “Naturals”]
	c) [“Amul”, “BR”, “Naturals”]
	d) {“Amul”:500, “BR”:300, “Naturals”:100}
		1

4.	Evaluate the following expression: 20+10* 2**2**3 a) **2580** b) 660 c) 7680 d) 540	1
5.	If S=’computer science model examination ’, what will the following command return? print(S.capitalize()) **a) Computer science model examination** b) Computer Science Model Examination c) computer science model examination d) COMPUTER SCIENCE MODEL EXAMINATION	1
6.	Identify the function which is not a MySQL function. a) Sum() b) Count() c) Max() **d) Average()**	1
7.	Choose the correct option from the following: The tell() function ………………………………. a) Places the file handle at the position that we specify. b) Sets the file handle at the beginning of the file. **c) Returns the byte number of the position of the file handle.** d) Brings the file handle to the end of the file.	1

8.	The command used to insert a new column to an already existing table is: a) Update b) Alter c) Insert d) modify	1
9.	Assuming the table ‘emp’ to exist, what will the following command do? Delete from emp; a) Will delete the table emp b) Will delete all the attributes in it c) Will delete all the tuples in it. d) Wrong syntax	1

10.	Find the erroneous statement from the following:
	Str=’Hard Work Is The Key To Success’
	print(Str) # Statement 1
	for I in len(str): # Statement 2 if Str[I]= =’ ‘:
	      # Statement 3
	Str[I]=’ - ’ # Statement 4 Str=’Good
	Luck’ # Statement 5
	(a) Statement 2
	(b) Statement 4
	(c) Statement 3 and 5
	(d) Statement 2 and 4
		1

11.	An Alternate key is a key, but not a Primary Key. **a)** Candidate **b)** Unique **c)** Foreign None of these	1
12.	Identify the correct option(s) to open the text file in read mode. (Assume the file already exists) i) sf =open('student.txt','w') ii) sf = open('student.txt','r') iii) sf = open('student.txt') iv) sf = open('student.txt','rb') a) Option i b) both iii & iv c) both ii and iii d) both ii and iv	1
13.	The wild card character ‘_’ indicates: **a)** one character **b)** Any number of characters **c)** Specific number of characters mentioned **d)** One column	1

14.	Fill in the blank: protocol is used to browse the internet. (a) VoIP (b) SMTP (c) PPP (d)HTTP	1
15.	What will be the output of the following: A=”Model Exam” B=A print(id(A)==id(B)) **a)** True **b)** False **c)** 0 **d)** 1	1
16.	Which of the following is an optional argument of the connect() function in python mysql.connector module? a) Host b) User c) Password d) database	1
Q17 and 18 are **ASSERTION AND REASONING** based questions. Mark the correct choice as (a) Both A and R are true and R is the correct explanation for A		
(b) Both A and R are true and R is not the correct explanation for A (c) A is True but R is False (d) A is false but R is True		
17. **Ans**	Assertion (A):- The default delimiter in CSV files is comma(,). Reasoning (R):- The default delimiter of the CSV file can be overridden with the help of the ‘sep’ attribute. **(a) Both A and R are true and R is the correct explanation for A**	1
18. **Ans**	Assertion (A):- If the arguments in a function call statement match the number and order of arguments as defined in the function definition, such arguments are called positional arguments. Reasoning (R):- During a function call, the argument list first contains default argument(s) followed by positional argument(s). **(c) A is True but R is False**	1
	SECTION B	
19.	(i) Expand the following terms: HTTP , URL HTTP – Hyper Text Transfer Protocol URL – Uniform Resource Locator (ii) A web server is a computer system that stores and delivers web pages to users when they request them. A website is a collection of web pages, images, videos, and other digital assets	1+1

that are hosted on a web server.

Or

i)Write down any one advantage of computer networks

File Sharing / Resource Sharing

ii)Define Gateway

A gateway is a device which connects dissimilar networks

20 Roshan tried to define a function to input an integer and print its factors. But he encountered certain syntax and logical errors in his code. Rewrite the corrected code and **underline** the corrections made. 2

```
def factors():
    n=input("Enter the number")
     for i in range( 0,n)
          If n% i = 0 :
         print(i)
```

Ans:

```
def factors():
    n=int(input("Enter the number"))
     for i in range( 1,n+1):
          if n% i = = 0 :
               print(i)
```

21

```
def countNow(STUDENT) :
    for name in STUDENT.values():
        if len(name)= =4:
            print(name)

STUDENT={1:"ARUN",2:"AVINASH",3:"PRIYA",4:"NIHA",5:"JEWEL
      "}
countNow(STUDENT)
```

or

```
def Jumble(str):
    n=len(str)
    str2=""
    for i in range(0,n):
         if str[i].isupper():
              str2=str2+str[i].lower()
         elif str[i].islower():
              str2=str2+str[i].upper()
          else:
              str2=str2+str[i]
return str2
```

22	9#60 P$R$S$ (1 mark for each correct line with the appropriate symbol)	2

23	(i) To delete last element in the list L1.
	S=L1.pop()
	(ii) To find the length of a string named message
	n=len(message)
	or
	import statistics
	print(statistics.mean(MARK))
		2

24	ALTER TABLE Staff ADD StaffId INT PRIMARY KEY; INSERT INTO Staff VALUES("Shalu","Marketing",45000,111); Or To delete the attribute, category: ALTER TABLE STUDENT DROP category; To add the attribute, Stream ALTER TABLE Sports ADD stream varchar(10) NOT NULL ;	2
25	**500 @ 380** **500 * 120** **140 @ 120** **500 * 140**	2
26	hHANDRAAAN-*4	3
27	(i) select EmpName, Designation from Employee where EmpName like '_ _ _'; **EmpName** **Designation** Sam Systems Engineer Mia Lead Consultant Sen Manager (i) select distinct Designation from Employee; **distinct Designation** Systems Engineer Technical Lead Lead Consultant Manager	3

(iii)select EmpID,EmpName, Designation from Employee where Designation in ('Technical Lead', 'Lead Consultant');

```
  EmpID  EmpName  Designation
   E102    Krish    Technical Lead
   E103    Mia      Lead Consultant
  E105    Nasrin   Lead Consultant
```

28

```
def ReversWords():
    with open("Python.txt") as f:
        ch=f.read()
        l=ch.split()
        for i in l:
            if i[0] in "Pp":
                print(i[::-1],end=' ')
            else:
                print(i,end=' ')
        f.close()
or
def CountAM():
    with open("Story.txt") as f:
        A=M=0
        s=f.read()
        for ch in s:
            if ch in "Aa":
                A+=1
            elif ch in "Mm":
                M+=1
    f.close()
     print("No.of A and M is", A ," & ", M)
```

3

29

(i) To display Name , Designation and Address of all employees whose name starts with character ‘A’

select name,designation,address from employees where name like 'A%';

(ii) To display the NAME,DESIGNATION,ADDRESS and TOTAL SALARY of all employees where TOTAL SALARY

is calculated as SALARY + BENEFIITS

select name,designation,address,salary+benifits as 'total salary' from employees;

(iii) To display the number of employees designation wise

select designation,count(*) from employees group by designation

3

30

Ans. Book=[ ]

```
  def Addnew(s):
```

3

	name=input(“Enter book
	name:”)
	Book.append(name)
	
	def Remove(Book):
	    if(Book==[ ]):
	        print(“Stack Empty.
	    Underflow”) else:
	        print(“Deleted book is :”, Book.pop())	

31	(i) SELECT NAME FROM DOCTOR WHERE DEPT=”ORTHOPEDIC” AND EXPERIENCE>10; (ii) SELECT AVG(BASIC + ALLOWANCE) FROM DOCTOR, SALARY WHERE DOCTOR.ID=SALARY.ID AND DEPT=”ENT”; (iii) SELECT MIN(ALLOWANCE) FROM DOCTOR, SALARY WHERE DOCTOR.ID=SALARY.ID AND SEX=”F”; (iv) SELECT MAX(CONSULTATION) FROM DOCTOR, SALARY WHERE DOCTOR.ID=SALARY.ID AND SEX=”M”;	4

32	import csv
	
	def add():
	    fout=open("furdata.csv","a",newline='\n')
	    wr=csv.writer(fout)
	    fid=int(input("Enter Furniture Id :: "))
	    fname=input("Enter Furniture name :: ")
	    fprice=int(input("Enter price :: "))
	    FD=[fid,fname,fprice]
	    wr.writerow(FD)
	    fout.close()
	
	def search():
	    fin=open("furdata.csv","r",newline='\n')
	    data=csv.reader(fin)	

```
    found=False
    print("The Details are")
    for i in data:
        if int(i[2])>10000:
            found=True
            print(i[0],i[1],i[2])
        if found==False:
            print("Record not found")
        fin.close()
add()
print("Now displaying")
search()
```

33

i) a) MAN
b) LAN
ii) Switch/hub
iii) Optical Fibre
iv) Production Unit- It contains the maximum no.of computers

Star Topology

34

a)Ans :

Binary file:

- Extension is .dat
- Not human readable
- Stores data in the form of 0s and 1s

CSV file

· Extension is .csv

· Human readable

· Stores data like a text file Program

```
b)def update_data():
    rec={}
    fin=open("record.dat","rb")
```

2+3

```
  fout=open("temp.dat",”wb”)
  found=False
  eid=int(input("Enter employee id to update their salary :: "))
  while True:
       try:
           rec= pickle.load(fin)
           if rec["Employee id"]==eid:
               found=True
               rec["Salary"]=int(input("Enter new salary :: "))
               pickle. dump(rec,fout)

         else:
             pickle.dump(rec,fout)
     except:
          break
 if found==True:
       print("The salary of employee id ",eid," has been updated.")
 else:
       print("No employee with such id is not found")
fin.close()
fout.close()
```

or

i)Differentiate between seek( ) and tell( ) functions

Ans : The tell function is used to determine the current position of the file pointer, and the seek function is used to move the file pointer to the specified position of the file

ii)Anushka wants to increase the price by 10% for those items which are less than 250. The details of the items (itemno, iname and price) are stored in a file named ‘items.dat’. She wants to store the modified data in another file named items_copy.dat. Anushka has written a function named Price_Change() . Help Anushka to accomplish the task

```
import pickle
```

```
def Price_Change():
        f1=open(“items.dat”,’rb’)
        f2=open("items_copy.dat","wb")
        l=pickle.load(f1)
        for i in l:
            if i[2]<250:
                i[2]=i[2]+i[2]*10/100
        pickle.dump(I,f2)
        f2.close()
        f2=open("items_copy.dat","rb")
        l=pickle.load(f2)
        print(l)
        f1. close()
        f2.close()
```

35

i)In relational database terminology, a domain defines the permitted range of values for an attribute of an entity

ii) import mysql.connector as mysql

```
def sql_data():
  con=mysql.connect(host="localhost",user="root",password="tiger",
  database="vidyalaya")
 mycursor= con.cursor()
 print("Students with marks between 80 and 95 are : ")
 mycursor.execute("select * from vidyalaya where Marks between 80 and
95")
 data= mycursor.fetchall()
 for i in data:
      print(i)
```

or

```
con1=mysql.connect(host="localhost",user="root", passwd="abcd",
database="XIICS")
mycursor= con1.cursor()
rno=int(input("Enter Roll Number :: "))
name=input("Enter name :: ")
class=int(input("Enter class :: "))
```

	marks=int(input("Enter Marks :: "))
	s="insert into student values ({},'{}',{},{})".format(rno,name,class,marks)
	mycursor.execute(s)
	con1.commit()
	print("Data Added successfully")	

KENDRIYA VIDYALAYA SANGATHAN: ERNAKULAM REGION

SAMPLE QUESTION PAPER 2023-24

COMPUTER SCIENCE (083)

Maximum Marks: 70 **Time Allowed: 3 hours**

General Instructions:

1. ***This question paper contains five sections, Section A to E.***
3. Section A have 18 questions carrying 01 mark each.
4. Section B has 07 Very Short Answer type questions carrying 02 marks each.
5. Section C has 05 Short Answer type questions carrying 03 marks each.
6. Section D has 02 Long Answer type questions carrying 04 marks each.
7. Section E has 03 questions carrying 05 marks each.
8. All programming questions are to be answered using Python Language only.

SECTION A

1. **State True or False** **1**

A variable in python is defined only when some value is assigned to it.

2. …………………..command is used to remove primary key from the table in SQL.
a) delete b)remove c) alter d)drop

3. What will the following expression be evaluated to in Python? **1**
print(25 // 4 + 3**1**2 * 2)
a) 24 b) 18 c) 6 d) 12

4. What will be the output of the following Python code? **1**

```
string = "my name is x"
for i in string.split():
    print (i, end=", ")
```

a)m, y, , n, a, m, e, , i, s, , x, b)m, y , n, a, m, e , i, s , x,
c)my, name, is, x, d)error

5. Suppose a dictionary D is declared as `D={1:"John",2:"Dev",3:"Bob"}` which of the following is incorrect? State the reason. **1**
(a) print(D[3]) (b) D[[1,2,3]] = “Python”
(c) D[‘Bob’]=3 (d) print(D[3][1])

6. Observe the following table: **1**

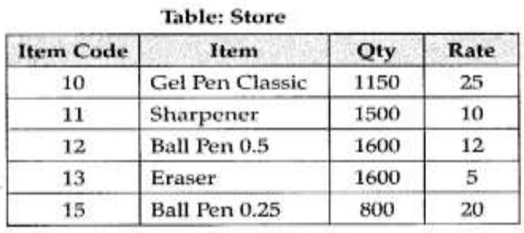


Table: Store

Item Code	Item	Qty	Rate
10	Gel Pen Classic	1150	25
11	Sharpener	1500	10
12	Ball Pen 0.5	1600	12
13	Eraser	1600	5
15	Ball Pen 0.25	800	20

What is the degree and cardinality of the above table?
a)4 and 5 b)5 and 4 c)4 and 6 d) 6 and 4

7. Fill in the blank: **1**

…………….. protocol is used to transmit an electronic mail over the internet.

a)FTP b)SMTP c)VOIP d)HTTP

8. Consider the statements given below and then choose the correct output from the given options: 1

```
example="snow world"
print(example[-2:1:-3])
```

a)lo b)lwn c)lwo d)lwn

9. Which of the following statement will return an error? T1 is a tuple 1

a)T1=T1+(23) b)T1=T1+[23] c)Both a and b d)None of the options

10. What possible outputs(s) are expected to be displayed on screen at the time of execution of the program from the following code? 1

```
import random
val=[20,30,40,50,60,70];
Low =random.randint(2,4)
High =random.randint(3,5)
for i in range(Low, High +1):
    print (val[i],end="#")
```

a) 20#30#40# b) 30#40#50# c) 50#60#70# d)60#70#

11. Which one of the following switches uses store and forward transmission? 1

a) Packet switching b)Circuit switching c)Both a and b d)None of the above

12. **Which of the following function headers is correct?** 1

a) def fun(a = 2, b = 3, c) b) def fun(a = 2, b, c = 3)
c) def fun(a , b =2, c=3) d) def fun(a, b, c = 3,d)

13. What will be the output of the following Python code? 1

```
def foo():
    try:
        print(1)
    finally:
        print(2)
foo()
```

a) 1
   2
b) 1 c) 2 d) none of the mentioned

14. Which of the following is not an integrity constraint? 1

a)Not null b) Composite key c) Primary key d) Check

15. **A device that forwards data packet from one network to another is called a……………** 1

a) Bridge b) Router c) Hub d) Gateway

16. 1

```
f = open("test.txt", "w+")
L = ["First Line\n", "Second Line\n", "Third Line"]
```

```
f.writelines(L)

f.flush()

f.seek(0)

O = f.readlines()

print(len(O))
```

a) 33 **b) 31** **c) 28** **d) 3**

Q17 and 18 are ASSERTION AND REASONING based questions. Mark the correct choice as

(a) Both A and R are true and R is the correct explanation for A.

(b) Both A and R are true and R is not the correct explanation for A.

(c) A is True but R is False.

(d) A is false but R is True

17. Assertion(A): The pop() method can be used to delete elements from a dictionary. **1**

Reason(R): The pop() method deletes the key-value pair and returns the value of deleted element

18. Assertion (A): Leading whitespace at the beginning of a statement in Python is called indentation. **1**

Reason (R): In Python, the same level of indentation associates statements into a different blocks of code.

SECTION B

19. (i) Expand the following terms: **1+1**

(a)NIC (b)SIM **(2)**

(ii)What is HTML? Where it is used?

OR

(i) Define the term bandwidth. Give any one unit of bandwidth.

(ii) Give one difference between FTP and HTTP.

20. Observe the following code carefully and rewrite it after rmoving all syntax and logical errors. Underline each correction done in the code. **2**

```
def execmain():
    x = input("Enter a number:")
    if (abs(x)=x):
        print ("You entered a positive number")
    else:
        x=*-1
        print("Number made positive:"x)
execmain()
```

21. Write a function frequencyCount( ), which takes the list (Marks) and empty dictionary (dict) as arguments. Update the dictionary, each marks as key and frequency of its occurrence as value and return the dictionary . **2**

Consider the following list **`Marks=[4,5,6,5,10,6,5,5,40,30,2,4] and dict`**

The function will return **`{4: 2, 5: 4, 6: 2, 10: 1, 40: 1, 30: 1, 2: 1}`**

OR

Write a function LeftShift(lst,x) in python which accepts number in a list (lst) and all the elements of the list should be shifted to left according to the value of x entered by the user.

Sample list `lst=[20,40,60,30,10,50,90,80,45,29]` value of x is **3**

Then lst will be `[30,10,50,90,80,45,29 20,40,60]`

22. Predict the output of the following Python code: **2**

Predict the output of the following Python code:

```
tup1 = ("George","Anderson","Mike","Luke","Amanda")
list1 =list(tup1)
list2 = []
for i in list1:
     if i[-1]=="e":
          list2.append(i)
tup2 = tuple(list2)
print(tup2)
```

23. Write the python statement for each of the following tasks using **BUILT-IN functions/methods only** **2**

Consider the list `L=[15,25,35,45,65,75,85]`

```
i) Insert value 55 between 45 and 65
ii)Remove all the elements from list
             OR
i)To add all the elements in L
ii)physically remove the list L
```

24. Differentiate between COUNT(*) and SUM( ) functions in SQL with appropriate example. **2**

(OR)

What do you understand by Candidate keys in a table? Give a suitable example of candidate keys from a table containing some meaningful data.

25. Predict the output of the following Python code: **2**

```
def Bigger(N1,N2):
  if N1>N2:
     return N1
  else:
     return N2


L=[32,10,21,54,43]
for c in range (4,0,-1):
   a=L[c]
   b=L[c-1]
   print(Bigger(a,b),'@', end=' ')
```

SECTION C

26. Predict the output of the Python code given below: **3**

```
st = "python programming"
count = 4
while True:
   if st[0]== "p":
      st = st[2:]
   elif st[-2]=="n":
      st = st[:4]
   else:
      count+=1
      break
print(st)
print(count)
```

27. Consider the table **EMPLOYEE** given below and write the output of the SQL queries that follow **3**

EmpId	Name	DoJ	Gender	City	Salary
101	Leonard	1995-06-06	M	London	9500
102	Priya	1993-05-07	F	Delhi	4500
103	Howard	1994-05-06	M	Paris	8500
104	Penny	1995-08-08	F	London	11000
105	Raj	1995-10-08	M	London	10000

106	Amy	1994-12-12	F	Paris	8500
107	Bernadette	1995-12-08	F	Lyon	8000
108	Sheldon	1995-06-12	M	Seattle	15000

i)SELECT COUNT(*), City FROM EMPLOYEE GROUP BY City HAVING COUNT(*) > 1;

ii)SELECT MIN(DoJ) FROM EMPLOYEE WHERE GENDER = ‘M’;

iii)SELECT AVG(Salary) from EMPLOYEE WHERE Name LIKE “%o%;

28. Write a function countline() in python to count the number lines in a text file ‘Country.txt’, which is starting with an alphabet ‘W’ or ‘H’. **3**

If the file contents are as follows:

Whose woods these are I think I know.

His house is in the village though;

He will not see me stopping here

To watch his woods fill up with snow.

The output of the function should be:

W or w : 1

H or h : 2

(OR)

Write a user defined function countwords( ) to display the total number of words present in the text file “Quotes.Txt”. If the text file has the following data written in it:

Living a life you can be proud of doing your best

Spending your time with people and activities that are important to you Standing up for things that are right even when it’s hard

Becoming the best version of you

The countwords() function should display the output as:

Total number of words : 40

29. Consider the table MobileMaster **3**

Table: **MobileMaster**

M_Id	M_Company	M_Name	M_Price	M_Mf_Date
MB001	Samsung	Galaxy	13000	2014-02-12

MB002	Nokia	N1100	7500	2011-04-15
MB003	Realme	C35	12000	2021-11-20
MB004	Oppo	SelfieEx	12500	2015-08-21

Based on the given table, write SQL queries for the following:

i) Reduce the Mobile Price of all mobiles by 10%

ii) Delete the mobiles with Price less than 12000

iii)Display Mobile company name and Model name separated by comma. in a new column name 'company&brand' and price ,in the ascending order of the Price

30. Vedika has created a dictionary containing names and marks as key-value pairs of 5 students. Write a program, with separate user-defined functions to perform the following operations: **3**

i)Push the keys (name of the student) of the dictionary into a stack, where the corresponding value (marks) is greater than 70.

ii)Pop and display the content of the stack.

The dictionary should be as follows:

```
d={"Ramesh":58, "Umesh":78, "Vishal":90, "Khushi":60, "Ishika":95}
```

Then the output will be: Umesh Vishal Ishika

SECTION D

31. a) Consider the table VEHICLE and TRAVEL given below: **4**

Table**: TRAVEL**

CNO	CNAME	TRAVELDATE	KM	VCODE	NOP
101	Arun	2015-12-13	200	V01	32
102	Balaji	2016-06-21	120	V03	45
103	Vignesh	2016-04-23	450	V02	42
104	Selva	2016-01-13	80	V02	40
105	Anupam	2015-02-10	65	V04	2
106	Tarun	2016-04-06	90	V05	4

Table: **VEHICLE**

VCODE	**VEHICLETYPE**	**PERKM**
V01	VOLVO BUS	150
V02	AC DELUXE BUS	125
V03	ORDINARY BUS	80

V04	CAR	18
V05	SUV	30

- PERKM is *Freight Charges per kilometre.*
- Km is *kilometres Travelled.*
- NOP is *number of passengers travelling in vehicle*.

i)Write the command to view the structure of the table TRAVEL.

ii)Display candidate name and Vehicle Type for each candidate.

iii)Display Customer Name and kilometres Travelled in descending order of kilometres Travelled.

iv) Display the number of customers in each vehicle Type(ie VCODE)

32. Genesis Infotech plans to develop a computerized system for payroll processing of all its employees for storing salary details, which stores data in **“employee.csv”.** **4**

The structure of the file is

[EMPNO,EMPNAME,SALARY]   Where

EMPNO is employee No (integer),EMPNAME employee Name(string),SALARY as salary(integer)

Genesis want develop the following user defined functions:

`AddEmp() -To accept a record from user and add to the file`

`ShowEmp() -Display the details of employees with column headings also display the number of employees who are drawing a salary of more than Rs 5000 per month.`

SECTION E

33. Oasis Public School, Coimbatore is setting up the network between its different wings of school campus. There are 4 wings namely SENIOR(S), JUNIOR (J), ADMIN (A) and HOSTEL (H). **5**

SENIOR
JUNIOR
ADMIN
HOSTEL

Distances between various wings are given below:

Wing A to Wing S	100m
Wing A to Wing J	200m
Wing A to Wing H	400m
Wing S to Wing J	300m

	Wing S to Wing H 100m Wing J to Wing H 450m **Number of Computers installed at various wings are as follows:** **Wings** **Number of Computers** Wing A 20 Wing S 150 Wing J 50 Wing H 25	
	i)Suggest the best-wired medium and mention the topology or layout to connect various wings of Oasis Public School, Coimbatore.	**1**
	ii)Name the most suitable wing to house the server. Justify your answer.	**1**
	iii)Suggest placement of HUB/SWITCH in the network of the School.	**1**
	iv)Suggest a device that can provide wireless Internet access to all smartphone /laptop users in the campus of Oasis Public School, Coimbatore.	**1**
	v)Suggest a protocol that shall be needed to provide file transfer within the School	**1**
34.	i)What is the difference between “w” and “a” modes in python? ii) Consider a file, University.dat, containing records of the following structure: `[Universityname ,collegename, noofstaff]` `Write a function CopyCollege(), that reads contents from the file` University.dat and copies the records with University name as Calicut to the file name Calicut.DAT. The function should return the total number of records copied to the file Calicut.DAT.**OR** i)What is the difference between `readline() and readlines()` functions? ii)Write a menu driven program `InsertRec() and Searchroll()` for adding and searching for a particular roll number using dictionary on student binary file `“student.DAT”`. Structure of dictionary is `{Rollno:rollno,”Name”:name,”Marks”:marks}`	**2+3** **(5)**
35.	i) Define the term Domain with respect to RDBMS. Give one example to support your answer ii) Swaraj wants to write a program in Python to insert the following record in the table named ‘Employee’ in MYSQL database, COMPANY: ● EID(Employee ID)- integer ● name(Name) - string ● DOB (Date of birth) – Date ● Salary – float	**5**

Note the following to establish connectivity between Python and MySQL:

- Username - root
- Password - tiger
- Host - localhost

The values of fields EID, name, DOB and Salary has to be accepted from the user. Help Swaraj to write the program in Python.

OR

i)what is the difference between primary key and foreign key?

ii) Nayana has created a table named ‘Employee’ in MYSQL database, COMPANY:

- EID(Employee ID)- integer
- name(Name) - string
- DOB (Date of birth) – Date
- Salary – float

Note the following to establish connectivity between Python and MySQL:

- Username - root
- Password - tiger
- Host - localhost

Nayana , now wants to display the records of employees whose salary is more than 25000. Help Nayana to write the program in python