TABLE OF CONTENTS

SNO	TOPIC	PAGES
1	SYLLABUS & MARKS DISTRIBUTIONS	01-02
2	PYTHON REVISION TOUR	03-17
3	WORKING WITH FUNCTIONS	18-29
4	EXCEPTION HANDLING	30-31
5	DATA FILE HANDLING	32-64
6	DATA STRUCTURE	65-73
7	COMPUTER NETWORKS	74-89
8	DATABASE MANAGEMENT	90-126
9	SOLVED CBSE QUESTION PAPER 2025-26	127-141
10	UNSOLVED SAMPLE PAPERS - (Set – 1,2 & 3)	142-169
11	REFERENCES	170

Computer Science (2026-27)
CLASS XII Code No. 083
Unit wise Syllabus

Unit 1: 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 2: 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 3: 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.

Distribution of Marks

Unit No.	Unit Name	Marks
1	Computational Thinking and Programming – 2	40
2	Computer Networks	10
3	Database Management	20
	Total	70

UNIT I:COMPUTATIONAL THINKING AND PROGRAMMING
PYTHON REVISION TOUR

About Python:

- Python programming language developed by Guido Van Rossum in February 1991.
- Python is based on two programming languages, ABC language and Modula-3
- It is a **case-sensitive** programming language.
- Python is one of the languages that are **not strict about data types**.
- Python works in to modes **– interactive and script**
- **Python is** Interpreted and Cross Platform Language, Free and Open Source, used for both procedural and object-oriented programming.

Basic terminologies:

- **TOKEN / LEXICAL UNIT:** Smallest individual unit in a Programming Language.
- **Types of Tokens** (Keywords, Identifiers, Literals, Operators, Punctuators)

NOTE: Mnemonic: "Kind Intelligent Lions Often Prey"

- **Keywords:** Reserved words having special meaning.

Literals	True, False, None, 22, "Hello", 3.14 etc…
Operators	not, and, or, is, in, +, -, *, / etc…
Keywords:	
Empty statement	pass
Conditional Statements	if, elif, else
Iterative statements	for, while, break, continue
Exception Handling	try, except, finally
Functions	def, return, global
File Handling	with
Others	del, import, from, as, lambda, nonlocal, yield, class

- **Identifiers:** These are the names given to **variables, objects, classes** or **functions** etc.

Identifier Naming Rules:

✓ Python identifiers **must begin** with a letter **(A-Z or a-z)** or an **underscore (_)**
✓ It can be followed by **letters, digits (0-9),** or **underscores**.
✓ However, Python keywords **cannot** be used as Identifiers
✓ Length of identifier is endless and is case sensitive.

- **Literals / Constants:** Data items that have a fixed value are called Literals.
  Example: 5 (integer), 5.9 (float), True (Boolean), "Hello" (string)

- **Operators:** Symbols that trigger some action.

Arithmetic Operators	+, -, *, /, //, %, **
Relational Operators	<, <=, >, >=, ==, !=
Logical Operators	not, and, or
Identity Operator	is, is not
Membership Operator	in, not in
Assignment Operator	=, +=, -=, *=, /=, //=, %=

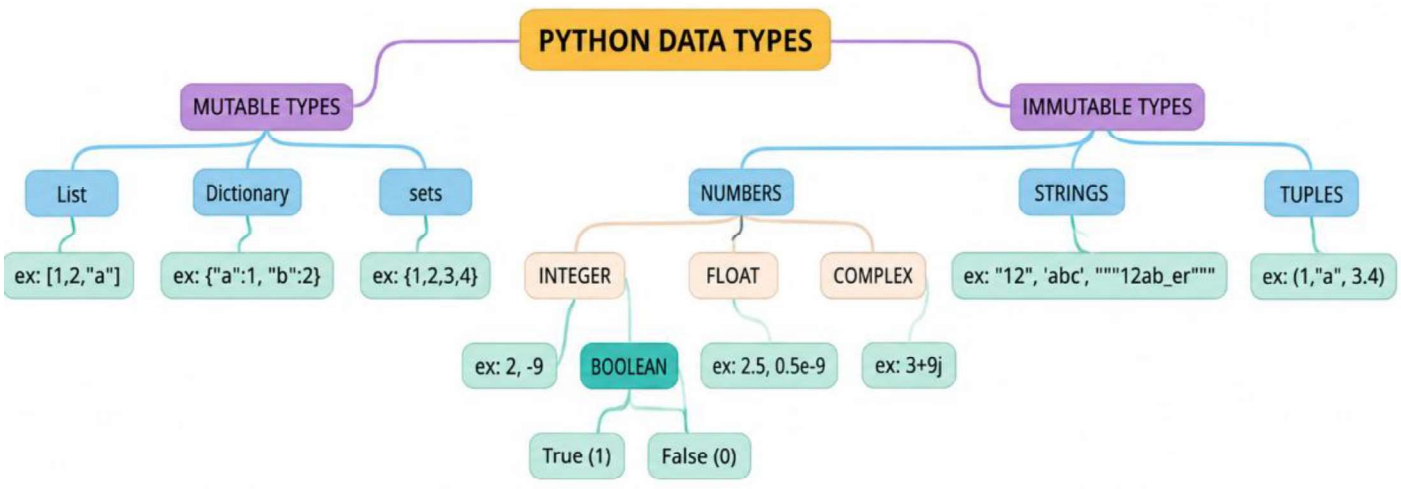
- **Operator Precedence:** Operator precedence defines the priority of operators in an expression.
- **Operator Associativity:** It define the direction in which operators of same precedence are evaluated either from left to right or right to left.

  Example : Evaluate 100 + 200 / 10 - 3 * 10
  Here, the operators /, * have similar precedence and +, - have same precedence, hence the expression will be evaluated as ((100 + (200 / 10)) – (3 * 10)), which equals to 90.0

- **Punctuators:** Symbols that are used to organize sentence structure. These are used to give syntactic and semantic meaning to the program statement.
  Some commonly used punctuators are:

‘	“	#	\	(	)	[	]	{	}	@	,	:	.	=	;

- **Data Types in Python**



Note: Python also offers a special literal (datatype) *None* which is used to create a variable without giving ut any significant value.

- **Variables:** Variables are containers for storing data values.
  Key points:
  - Variables do not need to be declared or defined in advance.

- A variable is created when you first assign a value to it.
- Python is dynamically typed language where data type of the variable is not required statically.
- **Example:**

```
# Initially assigning an integer value
x = 88                                  # x assigned as integer type
print ("x =", x, " Type:", type(x))
# Reassigning a string value to the same variable
x = "Dynamic Typing in Python"  # x assigned as string type
print ("x =", x, " Type:", type(x))
```

Note: Dynamic typing allows user to assign different values to same variable at different places in a program

- **Conversion of Data Types in Python**

Type Promotion (Implicit Type Conversion)	**Type Casting (Explicit Type Conversion):**
Python interpreter automatically converts/ promotes one data type to another higher type without any user involvement	The data type is manually changed by the user as per their requirement. With explicit type conversion, there is a risk of data loss since we are forcing an expression to be changed in some specific data type.
Example: x = 100 y = 10.6 z = x + y print(z) print ("z is of type:", type(z)) # z is implicitly converted to floating type	**Example:** a, b=5.88, '4' c=int(a) print("c=",c) #c=5 d=int(b) print("d=",d) #d=4

Python Modules:

- A **module** is a file containing Python code (functions, variables, classes) that can be imported and used in other Python programs.
- Modules help organize code and reuse functionality.

Random Number generation

random Module:

The random module in Python is a powerful tool for generating random numbers and performing random operations.

- To use the random module, **you must import it first**:

  import random
- It provides several functions to produce random values, simulate randomness, and shuffle data.

Name of method	**Method**	**Description**	**Example**
Random Float	random ( )	Generate a random float value between 0.0 and 1.0. (0.0 <= X < 1.0)	import random random. random ( )

Random Integer	randint ( )	Generate a random integer within a specified range (both values inclusive)	import random random. randint (1, 10)
Random Range	randrange( )	Generate a random number within a specified range with a step.	import random random. randrange (0, 100, 5) # Random number between 0 and 100 (does not include 100), step 5

STRING

- A string is a sequence of characters enclosed in **single (' ')**, **double (" ")**, or **triple (''' ''' / """ """)** quotes for multiline text.
- Strings are **immutable —** once created, they **cannot be changed**.
- Strings are **indexed**, starting from 0 (left to right) and -1 (right to left).

LIST

- A list is a collection of items (elements) which are ordered.
- Lists are mutable, meaning their contents can be changed after creation.
- Lists can contain elements of different data types: integers, strings, floats, even other lists.
- Lists are indexed, starting from 0 (left to right) and -1 (right to left).
- Defined using square brackets [ ].
- Example: my_list = [10, "Hello", 3.14, True]

TUPLE

- **Ordered –** Elements are stored in a specific order.
- **Immutable –** Once created, elements cannot be changed.
- **Allow duplicates –** Tuples can contain repeated values.
- **Can hold different data types –** Integers, strings, lists, etc.
- **Indexing –** Elements can be accessed using indexes.
- **Faster than lists –** Due to immutability.
- Tuples written in **parenthesis ( ).**

Multiple element tuple	Single element tuple	
t=(10,20,30) tup = 10, 20, 30	t = (10,) **Note:** t=10	# Comma is important #t is integer

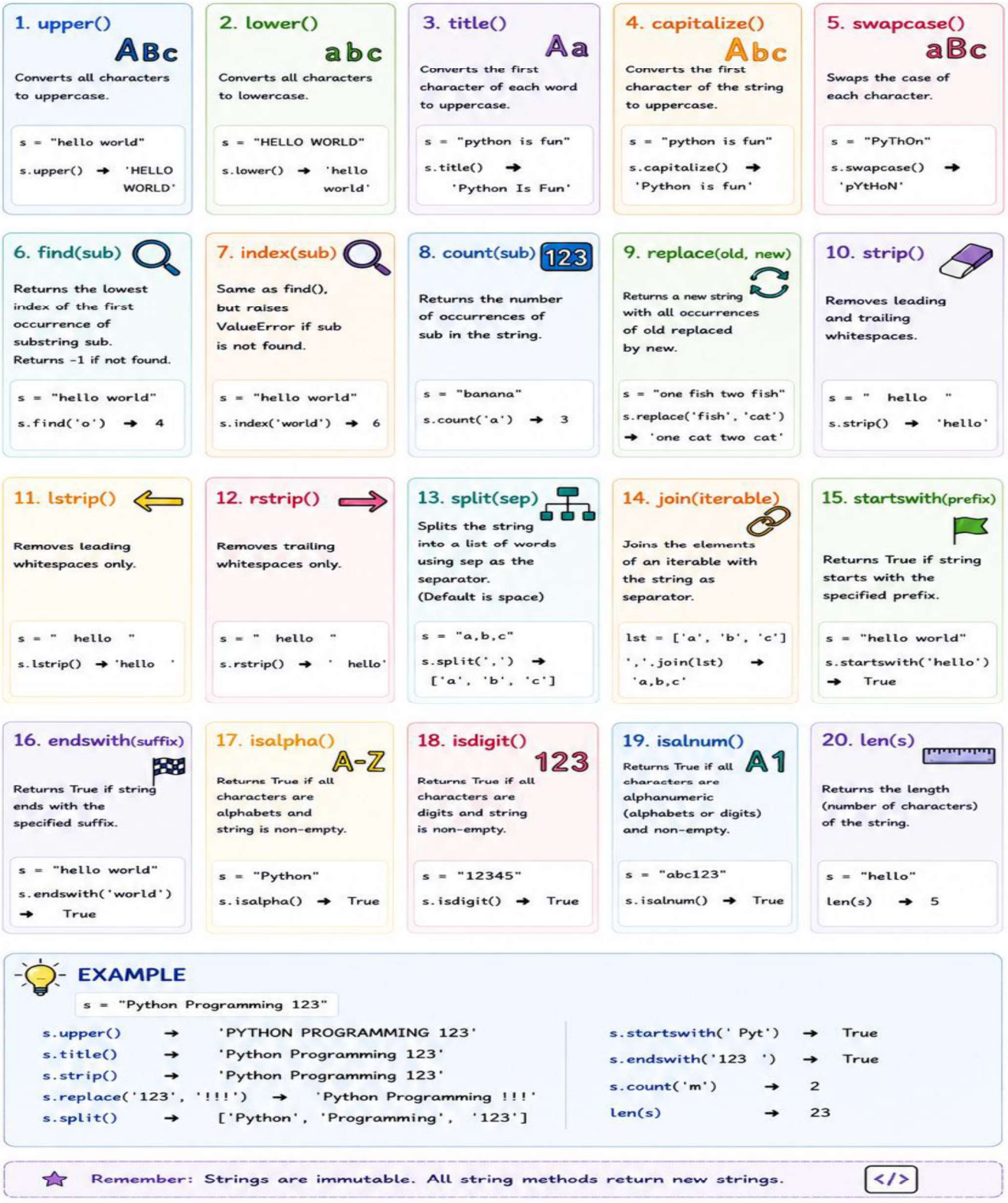
- **Packing Unpacking of Tuple**

```
tup = 10, 20, 30
a, b, c = tup
print(a,b,c)  # 10 20 30
```

DICTIONARY

- A **dictionary** in Python is an **unordered collection** of items.
- Each item is a **key-value pair**.
- Dictionaries are **mutable** (can be changed).
- Defined using **curly braces** {} with key:value pairs.
- **Key**: Must be **unique** and **immutable** (string, number, tuple)
- **Value**: Can be **any datatype** and can be **duplicate**
- Keys are used to access values.

String Functions and Methods



List Functions and Methods

LIST METHODS

No.	Method & Syntax	Description	Example	Result
1.	append(x) L.append(x)	Adds an element x at the end of the list.	L = [10, 20, 30] L.append(40)	L → [10, 20, 30, 40]
2.	insert(i, x) L.insert(i, x)	Inserts element x at index i.	L = [10, 20, 40] L.insert(2, 30)	L → [10, 20, 30, 40]
3.	remove(x) L.remove(x)	Removes the first occurrence of x.	L = [10, 20, 30, 20] L.remove(20)	L → [10, 30, 20]
4.	pop(i) L.pop(i)	Removes and returns the element at index i. (Default i = -1, last item)	L = [10, 20, 30] x = L.pop(1)	x → 20 L → [10, 30]
5.	index(x) L.index(x)	Returns the index of the first occurrence of x.	L = [10, 20, 30, 20] L.index(30)	2
6.	count(x) L.count(x)	Returns the number of occurrences of x.	L = [10, 20, 20, 30] L.count(20)	2
7.	reverse() L.reverse()	Reverses the order of elements in the list.	L = [10, 20, 30, 40] L.reverse()	L → [40, 30, 20, 10]
8.	sort() L.sort()	Sorts the list in ascending order (in place).	L = [30, 10, 40, 20] L.sort()	L → [10, 20, 30, 40]
9.	copy() L.copy()	Returns a shallow copy of the list.	L1 = [10, 20, 30] L2 = L1.copy()	L2 → [10, 20, 30] (L1 and L2 are different lists)
10.	extend(iterable) L.extend(iterable)	Extends the list by adding elements from iterable.	L = [10, 20] L.extend([30, 40])	L → [10, 20, 30, 40]
11.	clear() L.clear()	Removes all elements from the list.	L = [10, 20, 30] L.clear()	L → []

METHODS SUMMARY

Adding Elements	append(), insert(), extend()
Removing Elements	remove(), pop(), clear()
Searching / Info	index(), count()
Ordering	reverse(), sort()
Others	copy()

EXAMPLE PROGRAM

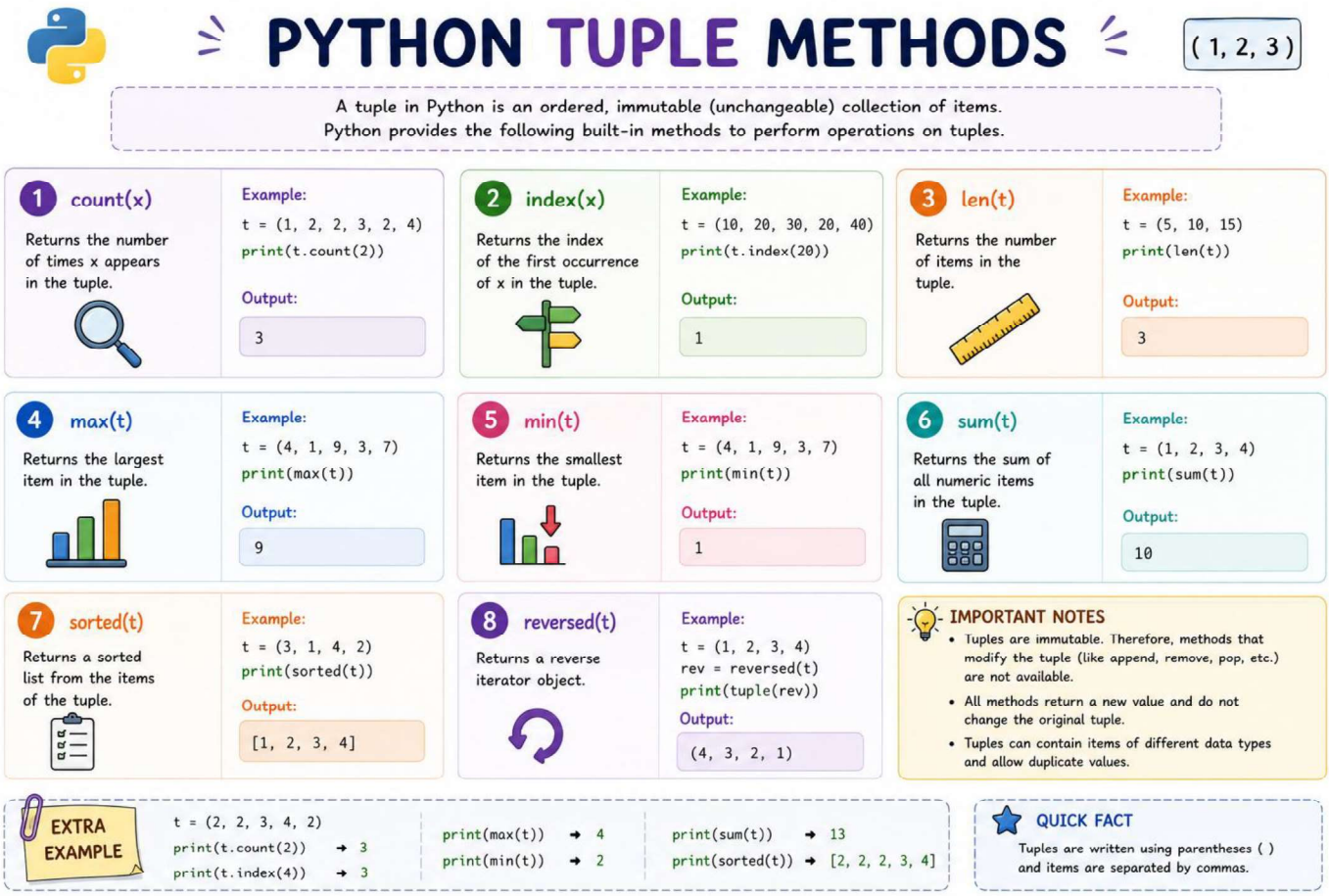
```
L = [10, 20, 30]
L.append(40)
L.insert(1, 15)
L.remove(30)
L.extend([50, 60])
L.sort()
print(L)
# Output:
[10, 15, 20, 40, 50, 60]
```

IMPORTANT NOTES

- sort() sorts the original list. Use sorted(L) to get a new sorted list without changing L.
- Most methods modify the list in place except index(), count(), copy() which return values.

Mastering list methods helps in efficient data handling and is widely used in solving real-life problems.

Tuple Functions and Methods



WORKING WITH DICTIONARIES

d = {'name': 'John', 'age': 16, 'grade': 'A'}

Operation	Code Example	Description	Output
Access a value	d['name']	Returns the value of the key	'John'
Add a new pair	D['city'] = 'Delhi'	Adds new key-value to the dictionary	{'city': 'Delhi'}
Modify a value	d['age'] = 17	Changes the value of existing key	{'age': 17}
Delete a pair	del d['grade']	Deletes key-value pair	Removes 'grade' key
Check key existence	'name' in d	Returns True if key exists	True
Loop through keys	for k in d:	Iterates over keys	'name', 'age', etc.
Loop through items	for k, v in d.items( ): print(k, v)	Iterates through key- value pairs	name John age 16 city Delhi

PYTHON DICTIONARY METHODS

A dictionary in Python stores data in key–value pairs.
Dictionary methods perform operations on dictionaries.

1 clear()



Removes all items from the dictionary.
The dictionary becomes empty.

Example:

```
d = {'a': 1, 'b': 2}
d.clear()
print(d)
```

Output:

```
{}
```

2 copy()



Returns a shallow copy of the dictionary.

Example:

```
d = {'a': 1, 'b': 2}
c = d.copy()
print(c)
```

Output:

```
{'a': 1, 'b': 2}
```

3 fromkeys()



Returns a new dictionary with the specified keys and value.

Example:

```
keys = ['a', 'b']
d = dict.fromkeys(keys, 0)
print(d)
```

Output:

```
{'a': 0, 'b': 0}
```

4 get(key, default)



Returns the value for the given key.
Returns default if key is not found.

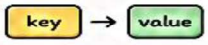
Example:

```
d = {'a': 1, 'b': 2}
print(d.get('a'))
print(d.get('c', 'Not found'))
```

Output:

```
1
Not found
```

5 items()



Returns a view object that displays a list of (dict) key–value pairs tuple.

Example:

```
d = {'a': 1, 'b': 2}
print(d.items())
```

Output:

```
dict_items([('a', 1),
 ('b', 2)])
```

6 keys()



Returns a view object that displays a list of all the keys.

Example:

```
d = {'a': 1, 'b': 2}
print(d.keys())
```

Output:

```
dict_keys(['a', 'b'])
```

7 pop(key, default)



Removes the item with the given key and returns its value.
Returns default if key is not found.

Example:

```
d = {'a': 1, 'b': 2}
print(d.pop('a'))
print(d)
```

Output:

```
1
{'b': 2}
```

8 popitem()



Removes and returns the last inserted key–value pair as a tuple.

Example:

```
d = {'a': 1, 'b': 2}
print(d.popitem())
print(d)
```

Output:

```
('b', 2)
{'a': 1}
```

9 setdefault(key, default)



Returns the value of the key if key is in dict.
If not, inserts key with default value and returns default.

Example:

```
d = {'a': 1}
print(d.setdefault('b', 0))
print(d)
```

Output:

```
0
{'a': 1, 'b': 0}
```

10 update([other])



Updates the dictionary with the key–value pairs from other dict or from an iterable.

Example:

```
d = {'a': 1}
d.update({'b': 2})
print(d)
```

Output:

```
{'a': 1, 'b': 2}
```

11 values()



Returns a view object that displays a list of all the values.

Example:

```
d = {'a': 1, 'b': 2}
print(d.values())
```

Output:

```
dict_values([1, 2])
```

NOTE:

- Dictionary keys must be immutable (like strings, numbers, tuples).
- Dictionary methods either modify the dictionary or return some useful information.



Multiple Choice Questions

1	What will be the output of the following code snippet? a = "10" b = "20" print (int (a + b) + int(a) * int(b)) a) 1220 b) 410 c) 620 d) 30
2	Which of the following code snippets correctly checks if a string s is a palindrome (case-insensitive)? a) s == s[::-1] b) s.lower( ) == s.upper( )[::-1] c) s.lower( ) == s.lower( )[::-1] d)s== s.reverse( )
3	What will be the output of the following code? x = 5 y = "2" print (x * int(y) + len(y + str(x))) a) 11 b) 12 c) 15 d) 10
4	Which of the following code snippets will raise an error for s = "Python"? a) print(s[10]) b) print(s[1:10]) c) print (s [-1: -3]) d) print(s[:2])
5	What will be the output of following python code: s = 'Programming' print(s.split("m")) a. ['Progra', ' ', 'ing'] b. ['Progra', 'ing']. c. ['Progra', 'm', 'ing'] d. [Progra', 'ming']
6	What will be the output of following python code: t1 = 1,2,3 t2 = (1,2,3) print(t1 is t2) a. True b. 1 c. False d. 0
7	State whether the following statement is True or False: An exception may be raised even if the program is syntactically correct.
8	Which of the following is not a type conversion function? a) int( ) b) str ( ) c) input ( ) d) float ( )
9	Select the correct output of the code: a = "foobar" a = a.partition("o") print(a) a. ["fo","","bar"] b. ["f","oo","bar") c. ["f","o","bar"] d. ("f","o","obar")
10	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
11	What will be the output of the following statement ? print(6+5/4**2//5+8) a.-14.0 b.14.0 c.-14 d. 1
12	Select the correct output of the code: S = "text#next" print(S.strip("t"))

	a. ext#nex b. ex#nex c. text#nex d. ext#next
13	Identify the valid Python identifier from the following : a. 2user b. user@2 c. user_2 d. user 2
14	Consider the statements given below and then choose the correct output from the given options : Game="World Cup 2023" print(Game[-6::-1]) a. CdrW b. ce o c. puC dlroW d. Error
15	Predict the output of the following Python statements : import statistics as s s.mode ([10, 20, 10, 30, 10, 20, 30]) a. 30 b. 20 c.10 d. 18.5
16	What will be the output of the following statement : print (16*5/4*2/5–8) a. – 3.33 b. 6.0 c. 0.0 d. –13.33
17	Identify the invalid Python statement from the following : a. d = dict( ) b. e = {} c. f = [ ] d. g = dict{}
18	Consider the statements given below and then choose the correct output from the given options : myStr = "MISSISSIPPI" print(myStr[:4]+"#"+myStr[–5:]) a. MISSI#SIPPI b. MISS#SIPPI c. MISS#IPPIS d. MISSI#IPPIS
19	Identify the statement from the following which will raise an error : a. print("A"*3) b. print(5*3) c. print("15" + 3) d. print("15" + "13")
20	What will be the output of the following code? L = ["India", "Incredible", "Bharat"] print(L[1][0] + L[2][-1]) a) IT b) it c) It d) iT
21	State True or False. “ Comments are not executed by interpreter.”
22	Given the following dictionary Day={1:"Monday", 2: "Tuesday", 3: "Wednesday"} Which statement will return "Tuesday"? a. Day.pop( ) b. Day.pop(2) c.Day.pop(1) d.Day.pop("Tuesday")
23	Consider the given expression : 7<4 or 6>3 and not 10==10 or 17>4 What will be the output of the above expression? a.True b. False c. NONE d. NULL
24	Select the correct output of the code : S="Amrit Mahotsav @ 75" A=S.split(" ",2) print(A) a.['Amrit', 'Mahotsav','@','75') b. ('Amrit', 'Mahotsav','@ 75) c. ['Amrit','Mahotsav','@ 75'] d.['Amrit','Mahotsav','@','75']
25	What will the following expression be evaluated to in Python ? print(6/3 + 4**3//8-4) (a) 6.5 (b) 4.0 (c) 6.0 (d) 4

26	State True or False: “A Python List must always contain all its elements of same data type.”
27	What will be the output of the following statement? print(14%3**2*4) a. 16 b. 64 c. 20 d. 256
28	Identify the correct output of the following code snippet: game="Olympic2024" print(game.index("C")) (A) 0 (B) 6 (C) -1 (D) ValueError
29	Which of the following is the correct identifier? (A) global (B) Break (C) def (D) with
30	Identify the invalid Python statement out of the following options: (A) print("A",10,end="*") (B) print("A",sep="*",10) (C) print("A",10,sep="*") (D) print("A"*10)
31	Consider the statements given below and then choose the correct output from the given options: L=['TIC', 'TAC'] print(L[::-1]) (A) ['CIT', 'CAT'] (B) ['TIC', 'TAC'] (C) ['CAT', 'CIT'] (D) ['TAC', 'TIC']
32	Which of the following operator evaluates to True if the variable on either side of the operator points towards the same memory location and False otherwise? (A) is (B) is not (C) and (D) or
33	Consider the statements given below and then choose the correct output from the given options: D={'S01':95, 'S02':96 } for I in D: print(I,end='#') (A) S01#S02# (B) 95#96# (C) S01,95#S02,96# (D) S01#95#S02#96#
34	What will be the output of the following Python code? my_dict = {"name": "Alicia", "age": 27, "city": "DELHI"} print(my_dict.get("profession", "Not Specified")) a) Alicia b)DELHI c)None d)Not Specified
35	Which of the following built-in function/method returns a dictionary ? (A) dict( ) (B) keys( ) (C) values( ) (D) items( )

ANSWERS									
1	A	**2**	C	**3**	b	**4**	A	**5**	a
6	a	**7**	True	**8**	c	**9**	D	**10**	B
11	B	**12**	A	**13**	C	**14**	C	**15**	C
16	C	**17**	D	**18**	B	**19**	C	**20**	c)
21	True	**22**	B	**23**	True	**24**	B	**25**	C
26	False	**27**	C	**28**	D	**29**	B	**30**	B
31	D	**32**	A	**33**	A	**34**	D	**35**	A

Assertion/Reasoning Type Questions

Directions: In the following questions, A statement of Assertion (A) is followed by a statement of Reason (R). Mark the correct choice as:

(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

1	Assertion (A): Lists can store only elements of the same data type. Reason (R): Python is a dynamically typed language
2	Assertion(A): List is an immutable data type. Reasoning(R): When an attempt is made to update the value of an immutable variable, the old variable is destroyed and a new variable is created by the same name in memory
3	Assertion. Dictionaries are mutable, hence its keys can be easily changed. Reason. Mutability means a value can be changed in place without having to create new storage for the changed value.
4	Assertion (A): You can add an element in a dictionary using key:value pair. Reasoning (R): A new (key:value) pair is added only when the same key doesn't exist in the dictionary. If the key is already present, then the existing key gets updated and the new entry will be made in the dictionary.
5	Assertion (A): Tuples hold a sequence of homogenous elements. Reason (R): Tuples are immutable
6	Assertion (A): The expression "Hello".sort( ) in Python will give an error. Reason (R): sort( ) does not exist as a method/function for strings in Python
7	Assertion(A): List is an immutable data type Reasoning(R): When an attempt is made to update the value of an immutable variable, the old variable is destroyed and a new variable is created by the same name in memory
8	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

ANSWERS ASSERTION REASONING

1	(D) A is false but R is true
2	(D) A is false but R is true
3	(D) A is false but R is true
4	(A) Both A and R are true and R is the correct explanation of A
5	(D) A is false but R is true
6	(A) Both A and R are true and R is the correct explanation of A
7	(D)A is false but R is True
8	(B) Both A and R are true but R is not the correct explanation for A

Short Answer Questions/Long Answer Questions

1 Write the output displayed on execution of the following Python code :

```
LS=["HIMALAYA","NILGIRI","ALASKA","ALPS"]
D={}
for S in LS :
   if len(S)%4 == 0:
         D[S] = len(S)
for K in D :
   print(K,D[K], sep = "#")
```

2	Write the Python statement for each of the following tasks using built-in functions/methods only : (i) To remove the item whose key is "NISHA" from a dictionary named Students. For example, if the dictionary Students contains {"ANITA":90, "NISHA":76, "ASHA":92}, then after removal the dictionary should contain{"ANITA":90,"ASHA":92} (ii) To display the number of occurrences of the substring "is" in a string named message. For example if the string message contains "This is his book", then the output will be 3.
3	A tuple named subject stores the names of different subjects. Write the Python commands to convert the given tuple to a list and thereafter delete the last element of the list.
4	Predict the output of the following code : d={"IND":"DEL","SRI":"COL","CHI":"BEI"} str1="" for i in d: str1=str1+str(d[i])+"@" str2=str1[:-1] print (str2)
5	Write the Python statement for each of the following tasks using BUILT-IN functions/methods only : (i) To delete an element 10 from the list lst. (ii) To replace the string "This" with "That" in the string str1.
6	A dictionary dict2 is copied into the dictionary dict1 such that the common key's value gets updated. Write the Python commands to do the task and after that empty the dictionary dict1.
7	Write difference between mutable and immutable data types? Write name of any two mutable data types.
8	**i.** Which operator in python has right to left associativity? **ii.** Evaluate the following expression and write output: 5//4**3%6+2
9	If str1= "foundation stone" and str2= "strong base", then Write the Python statements for each of the following tasks using Built-in functions/methods only: • To count the occurrence of the letter 'n' in str1. • To convert the string str2 in a list and each word of the string should become the element of the list
10	Explain the difference between explicit and implicit type conversion in Python with a suitable example.
11	How is a mutable object different from an immutable object in Python? Identify one mutable object and one immutable object from the following: (1,2), [1,2], {1:1,2:2}, '123'
12	if L1=[1,2,3,2,1,2,4,2, . . .], and L2=[10,20,30, . ..], then(Answer using built in functions only) I) A) Write a statement to count the occurrences of 4 in L1. B) Write a statement to sort the elements of list L1 in ascending order. (II) A) Write a statement to insert all the elements of L2 at the end of L1. B) Write a statement to reverse the elements of list L2.
13	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

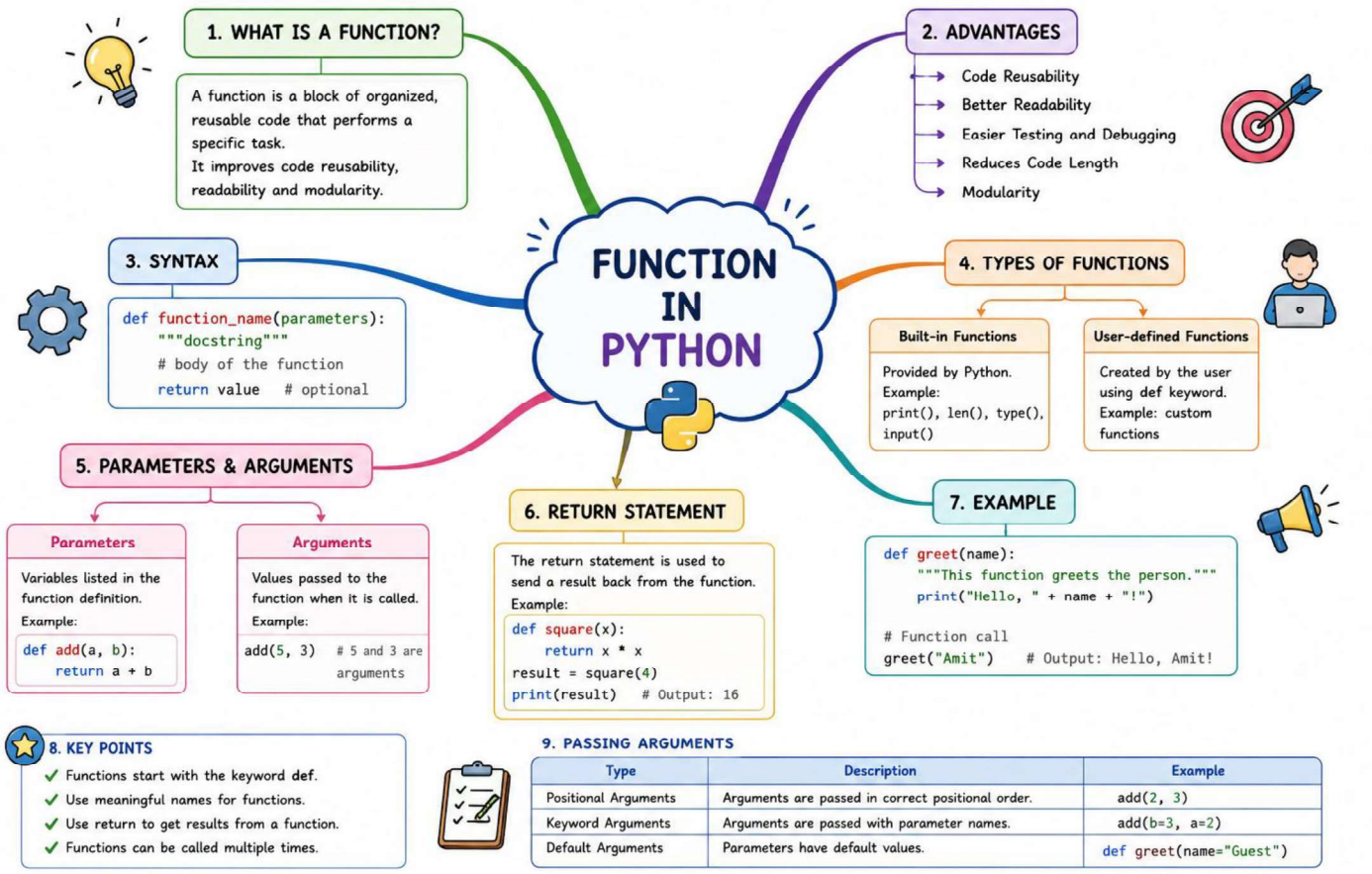
14	A list named studentAge stores age of students of a class. Write the Python command to import the required module and (using built-in function) to display the most common age value from the given list.
15	(a) Given is a Python string declaration: myexam="@@CBSE Examination 2022@@" Write the output of: print(myexam[::-2]) (b) Write the output of the code given below: my_dict = {"name": "Aman", "age": 26} my_dict['age'] = 27 my_dict['address'] = "Delhi" print(my_dict.items( ))
16	Predict the output of the Python code given below: tuple1 = (11, 22, 33, 44, 55 ,66) list1 =list(tuple1) new_list = [ ] for i in list1: if i%2==0: new_list.append(i) new_tuple = tuple(new_list) print(new_tuple)
17	Predict the output of the code given below: s="welcome2cs" n = len(s) m="" for i in range(0, n): if (s[i] >= 'a' and s[i] <= 'm'): m = m +s[i].upper( ) elif (s[i] >= 'n' and s[i] <= 'z'): m = m +s[i-1] elif (s[i].isupper( )): m = m + s[i].lower( ) else: m = m +'&' print(m)
18	Write a suitable Python statement for each of the following tasks using built-in functions/methods only: i To delete an element Mumbai:50 from Dictionary D. ii To display words in a string S in the form of a list

Answers

1	HIMALAYA#8 ALPS#4
2	i) Students.pop("NISHA") (ii) print(message.count("is")) message.count("is")
3	subject=list(subject) subject.pop( )
4	DEL@COL@BEI

<table>
<tr><td>5</td><td>(i) lst.remove(10)
ii) str1.replace("This","That")</td></tr>
<tr><td>6</td><td>dict1.update(dict2)
dict1.clear( )</td></tr>
<tr><td>7</td><td>
<table>
<tr><th>Mutable Data Type</th><th>Immutable Data Type</th></tr>
<tr><td>Value can be changed at particular index.</td><td>Value cannot be changed/assigned at particular index.</td></tr>
<tr><td>Examples:
i. list
ii. dictionary</td><td>Examples
i. Numbers
ii. strings</td></tr>
</table>
</td></tr>
<tr><td>8</td><td>i. Exponent operator (**)
ii. 2</td></tr>
<tr><td>9</td><td>A. print(str1.count('n'))
B. L=str2.split( )</td></tr>
<tr><td>10</td><td>A. Implicit Conversion: Python automatically converts one data type to another.
Example:
x = 10
y = 3.5
result = x + y # x is implicitly converted to float
Explicit Conversion: The user manually converts one data type to another using functions like int(), float().
Example:
x = "10"
y = int(x) # Explicit conversion from string to integer</td></tr>
<tr><td>11</td><td>A mutable object can be updated whereas an immutable object cannot be updated.
Mutable object: [1,2] or {1:1,2:2} (Any one)
Immutable object: (1,2) or '123' (Any one)</td></tr>
<tr><td>12</td><td>(I) A) L1.count(4)
(B) L1.sort( )
(II) A) L1.extend(L2)
(B) L2[::-1]</td></tr>
<tr><td>13</td><td>(i) L1.insert(2,200)
(ii) message.endswith('.')</td></tr>
<tr><td>14</td><td>import statistics
print(statistics.mode(studentAge))</td></tr>
<tr><td>15</td><td>a) @20 otnmx SC@
b) dict_items([('name', 'Aman'), ('age', 27), ('address', 'Delhi')])</td></tr>
<tr><td>16</td><td>(22,44,66)</td></tr>
<tr><td>17</td><td>sELCcME&Cc</td></tr>
<tr><td>18</td><td>i. del D['Mumbai']
ii. print(S.split( ))</td></tr>
</table>

WORKING WITH FUNCTIONS



- **FUNCTIONS:** Function can be defined as a named group of instructions that accomplish a specific task when it is invoked

- **FUNCTIONS CAN BE CATEGORIZED AS:**

a) **Built-in functions:** Built-in functions are the ready-made functions in Python that are frequently used in programs. We don't have to import any module to use these functions.

b) **Functions defined in module:** A module is a file containing Python definitions (i.e., functions) and statements. To use these modules in the program, a programmer needs to import the module by using either the import statement or the from statement.

- **import statement:** It is simplest and most common way to use modules in our code. 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 (.).

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. 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
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
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
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
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
Trigonometric functions sin(x), cos(x), tan(x)	Used to calculate trigonometric ratios such as sin(), cos(), tan() etc…	math.sin(math.pi/2) 1.0 math.cos(0) 1.0 math.tan(0) 0.0

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

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** at the end of the line.

c. Next follows the **block of statement(s)** that are the part of function, called body of the function.

Syntax:

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

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

Example:

```
def area_of_circle(radius ): # radius Formal Parameter
   print ("Area of circle = ",3.1416*radius*radius)

r=int (input ("Enter radius of circle"))
ar=area_of_circle(r)          # r - Actual Parameter or Argument
```

OUTPUT
Enter radius of circle 10
Area of circle = 314.16

TYPES OF ARGUMENTS / PARAMETERS

a) Positional Parameters/ Required Arguments: The function call statement must match the number and positional order of arguments as defined in the function definition, then it is called the positional parameters.

Example:

```
def Test(x,y,z):      # x,y,z are positional parameters
```

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
```

b) Keyword Arguments/Named Arguments: In Python, keyword arguments, also known as named arguments, are a way to pass arguments to a function by explicitly specifying the parameter name along with its value.

```
def my_function(name, age, city):
   print(name, age, city)
# Keyword arguments
my_function(age=25, city="London", name="Bob")
```

c) Default Parameters: The parameters which are assigned with a value in function header while defining the function are known as default parameters. If no value is provided in function call, then the default value will be taken by the parameter.

Example:

```
def SICal(amount, rate, time=10):    #time has default parameter
```

NOTE: Default parameters will be written in the end of the function header, which means positional parameter cannot appear to the right side of default parameter in function header.

FUNCTION RETURNING VALUE(S)

A function can returns 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.
If the function does not return any value, by default it returns None value to the function call.

SCOPE OF VARIABLES*:*

The part of the program where a variable is accessible can be defined as the scope of that variable.
There are two types of scope for variables:

1. Local Scope: A variable declared in a function-body is said to have local scope. It cannot be accessed outside the function.

2. Global Scope: A variable declared in top level segment (main) of a program is said to have a global scope.

Example:

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

***global* KEYWORD:**

The global keyword is used when a programmer wants to access and/or change the variables created in global space from a local space i.e. from the functions.

Example: First example below shows the use of global keyword in function, the secon example creates a local variable.

Flow of Execution of programs with functions:

Whenever the interpreter observes the def keyword during execution of the program, it omits the execution of the body of that function. The interpreter only executes the function body when it is called.
When the interpreter executes the function body after a function call, it executes the function until the return keyword is encountered or until the last statement of the function body is executed.
If the function returns any value, the execution will go to the statement of function call. If the function does not return any value, the function call goes to the statement next to the function call.

Examples:

1. def fun(x, y): 2. z = x + y 3. print(z) 4. return 5. a, b = 10, 20 6. fun(a, b) 7. print("Back to main") **Output:** 30 Back to main **Flow of execution:** 1 → 5 → 6 → 1 → 2 → 3 → 4 → 7	1. def fun(x, y): 2. z = x + y 3. print(z) 4. a, b = 10, 20 5. fun(a, b) 6. print("Back to main") **Output:** 30 Back to main **Flow of execution:** 1 → 4 → 5 → 1 → 2 → 3 → 6	1. def fun(x, y): 2. z = x + y 3. return z 4. a, b = 10, 20 5. c = fun(a, b) 6. print(c) 7. print("Back to main") **Output:** 30 Back to main **Flow of execution:** 1 → 4 → 5 → 1 → 2 → 3 → 5 → 6 → 7

When arguments are passed to a function in Python, the function receives a reference to the object. The behaviour depends on whether the object is mutable or immutable.

- Mutable data types can be changed after creation.
  Examples: list, dictionary
- Immutable data types cannot be changed after creation.
  Examples: int, float, str, tuple

If a mutable object is modified inside the function, the changes are reflected outside the function.
If an immutable object is modified, a new object is created inside the function and the original value remains unchanged.

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

Output:
10
10
15
15

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

Output:
10
10
15
10

Multiple Choice Questions

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	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
4	pow ( ) function belongs to which library? a) math b) string c) random d) maths
5	What will be the output of program statement math.sin(math.pi/2)? (assume math is imported) a) 0 b) 0.0 c) 1 d) 1.0
6	What is returned by int (math.pow(3, 2))? (assume math is imported) a) 6 b) 9.0 c) error, third argument required d) error, too many arguments
7	Observe the given Python code carefully a=20 def convert(a): b=20 a=a + b

	convert(10) print(a) Select the correct output from the given options: a. 10 b. 20 c. 30 d. Error
8	Identify the module to which the following function mode( ) belong to? a) math b) random c) statistics d) sys
9	How many argument(s) a function can receive? a) Only one b) 0 or many c) Only more than one d) At least one
10	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
11	It is optional to pass the values for default arguments in functions. a) True b) False
12	The variable declared inside the function is called a _____variable a) global b) local c) external d) none of the above
13	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
14	The_____of a variable is the area of the program where it may be referenced a) external b) global c) scope d) local
15	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
16	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 >5>6 >1> 2 > 3 >6 7 > 10>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
17	A_____can be skipped in the function call statements

	a) named parameter b) default parameter c) keyword parameters d) all of the above
18	Consider the statements given below and then choose the correct output from the given options: def Change(N): N=N+10 print(N,end='$$') N=15 Change(N) print(N) (A) 25$$15 (B) 15$$25 (C) 25$$25 (D)2525$$
19	What is wrong with the following function definition? def greet (name="Guest", age): print (name, "is", age, "years old")) a). The syntax of print statement is wrong b). Default parameter must follow required parameter c). Function name cannot be greet d). Nothing is wrong
20	Which of the following function definitions is INVALID? a). def func(a, b=2, c=3): pass. b). def func(a=1, b, c=2): pass c). def func(a, b, c=5): pass. d). def func(a=1, b=2, c=3): pass

Qno	Ans	Qno	Ans	Qno	Ans	Qno	Ans
1	A	6	B	11	A	16	A
2	D	7	B	12	B	17	B
3	B	8	C	13	A	18	A
4	A	9	B	14	C	19	B
5	D	10	A	15	C	20	B

ASSERTION AND REASONING QUESTIONS

(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): If the arguments in a function call match the number and order of arguments 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.
2	Assertion(A): The random module is a built-in module to generate the pseudo-random values. Reason(R): The randrange( ) function is used to generate a random number between the specified range in its parameter.
3	Assertion (A): Global variable is declared outside the all the functions.

	Reasoning (R): It is accessible through out all the functions.
4	Assertion (A): Built-in function is predefined in the language that are used directly. Reason (R): print ( ) and input ( ) are built-in functions
5	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.
6	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).
ANSWERS	
1	(c) A is True but R is False
2	(b) Both A and R are true and R is not the correct explanation for A
3	(a) Both A and R are true and R is the correct explanation for A
4	(a) Both A and R are true and R is the correct explanation for A
5	(a) Both A and R are true and R is the correct explanation for A
6	(c) A is true but R is false
Short Answer Questions/Long Answer Questions	
1	Rewrite the following Python program after removing all the syntactical errors (if any), underlining each correction: def checkval x = input ("Enter a number") if x % 2 =0: print (x, "is even") elseif x<0: print (x, "should be positive") else: print (x, "is odd")
2	Mani Ayyar, a python programmer, is working on a project which requires him to define a function with name CalculateInterest( ). He defines it as: def CalculateInterest (Principal, Rate=.06, Time): # Code But this code is not working; Can you help Mani Ayyar to identify the error in the above function and with the solution?
3	Predict the possible output(s) of the following code. Also specify the maximum and minimum value that can be assigned to the variable R when K is assigned value as 2. import random Signal=['stop', 'wait', 'go'] for K in range (2,0, -1): R=random.randrange(K) print (Signal[R], end=" #") a. Stop#wait#go# b. wait#stop# c. go#wait# d.go#stop#

4	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 6 3 12 5 b)12 5 6 3 5 5 c) 5 6 12 5 6 3 d) 5 5 7 7 6 6
5	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)
6.	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,"%")
7	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 @
8	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) 12 c) 0 d) None of these 12
9	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 $
10	Vivek has written a code to input a number and check whether it is even or odd number. His code is having errors. Rewrite the correct code and underline the corrections made. Def checkNumber(N): status = N%2 return #main-code num=int (input ("Enter a number to check :)) k=checkNumber(num) if k = 0: print ("This is EVEN number") else: print ("This is ODD number")
11	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

<table>
<tr><td></td><td>a=changer(a,b)
print(a,"$",b)
a=changer(a)
print(a,"$",b)</td></tr>
<tr><td>12</td><td>Write a function INDEX_LIST(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]</td></tr>
<tr><td>13</td><td>Write a program in Python named which takes the dictionary S as an argument. The dictionary, S contains Name:(Eng,Math,Science] as key:value pairs. The function displays the corresponding grade obtained by the students according to the following grading rules :
<table>
<tr><th>Average of Eng,Math,Science</th><th>Grade</th></tr>
<tr><td>>=90</td><td>A</td></tr>
<tr><td><90 but >=60</td><td>B</td></tr>
<tr><td><60</td><td>C</td></tr>
</table>
For example : Consider the following dictionary
S={"AMIT":[92,86,64],"NAGMA":[65,42,43],"DAVID":[92,90,88]}
The output should be :
AMIT – B
NAGMA – C
DAVID – A</td></tr>
<tr><td></td><td>ANSWERS</td></tr>
<tr><td>1</td><td>def checkval( ):
x = int (input ("Enter a number"))
if x % 2 == 0:
print (x, "is even")
elif x<0:
print (x, "should be positive")
else:
print (x, "is odd")</td></tr>
<tr><td>2</td><td>Non-default argument must appear before default argument
Correct definition: def CalculateInterest(Principal, Time ,Rate=.06,):</td></tr>
<tr><td>3</td><td>b) wait#stop#
Max value of R=1, Min value of R=0</td></tr>
<tr><td>4</td><td>Ans. a)
5 5
6 3
12 5</td></tr>
<tr><td>5</td><td>Ans.
0 *11 *12 *9 *
0 *11 *
0 *11 *12 *</td></tr>
<tr><td>6</td><td>PPW%RRlN%</td></tr>
</table>

7	Ans. Options i and iii i)40 @50 @ iii) 40 @50 @70 @90 @ Maximum value of L and U : L=2 ,U=5
8	Ans : a)12 12
9	Ans. iv) 7 $ Maximum value of x is 2 Minimum value of x is 1
10	**def** checkNumber(N): # Def should be def status = N%2 return status #main-code num=int(input(" Enter a number to check : **")) #** Message not enclosed within quotation mark k=checkNumber(num) if k **==** 0: print("This is EVEN number") **else:** print("This is ODD number")
11	Ans. 10.0 # 10.0 10.0 $ 20 1.0 # 1.0 1.0 $ 20
12	def INDEX_LIST(L): indexList= [] for i in range(len(L): if L[i]! =0: indexList.append(i) return indexList
13	def showGrades(S): for K, V in S. items( ): if sum(V)/3>=90: Grade="A" elif sum(V)/3>=60: Grade="B" else: Grade="C" print(K,"–",Grade) S={"AMIT":[92,86,64],"NAGMA":[65,42,43],"DAVID":[92,90,88]} showGrades(S)

EXCEPTION HANDLING IN PYTHON

Exception: An error that occurs **during program execution** is called an **exception**.
Exceptions interrupt the normal flow of a program.

Common examples of exceptions:

- Dividing by zero (ZeroDivisionError)
- Accessing a missing file (FileNotFoundError)
- Using an undefined variable (NameError)

Without handling exceptions, a program **crashes** when an error occurs.

Need for Exception Handling

- Prevents program from crashing abruptly.
- Helps in providing **user-friendly error messages**.
- Ensures **smooth program execution** even after an error.

Handling Exceptions using try-except-finally Blocks

Python provides a **structured way** to handle exceptions using three keywords:

Keyword	Purpose
try	Used to **write code** that may cause an exception.
except	Used to **catch and handle** the exception.
finally	Used to **execute code** whether an exception occurs or not (cleanup code).

```
try:
   # Code that might raise an exception
except [ExceptionType]:                    # it is optional to specify Exception
   # Code that runs if the exception occurs
except [Exception Type]:                   # can have multiple blocks
finally:
   # Code that will run no matter what (optional)
```

try block → Python runs the code inside try.
If an **exception occurs**:
The try block is exited immediately.
The control moves to the **except block**.
finally block always runs after the try/except.

Example 1: Handling a division by zero error

```
try:
   a = int(input("Enter numerator: "))
   b = int(input("Enter denominator: "))
   result = a / b
   print("Result =", result)
except ZeroDivisionError:
   print("Error: Cannot divide by zero!")
finally:
```

Example 2: Multiple except blocks

```
try:
   num = int(input("Enter a number: "))
   print("Reciprocal is", 1/num)
except ValueError:
   print("Invalid input! Please enter an integer.")
except ZeroDivisionError:
   print("Cannot find reciprocal of zero.")
finally:
```

print("Execution complete.")	print("Thank you!")
Output 1 (if denominator is 0): Enter numerator: 10 Enter denominator: 0 Error: Cannot divide by zero! Execution complete.	**Output 2 (normal input):** Enter numerator: 10 Enter denominator: 2 Result = 5.0 Execution complete.

Important Points

- try must be followed by at least one except, finally, or both.
- If no exception occurs, **except block is skipped**.
- finally block is **always executed**, even if:
    - An exception occurs
    - No exception occurs
    - return, break, or continue statements are used inside try/except

QUESTIONS:

1	Consider the statements given below and then choose the correct output from the given options: N='5' try: print('WORD' + N, end='#') except: print('ERROR',end='#') finally: print('OVER') (A) ERROR# (B) WORD5#OVER C) WORD5# (D) ERROR#OVER
2	State whether the following statement is True or False: While handling exceptions in Python, name of the exception has to be compulsorily added with **except** clause.
3	State whether the following statement is True or False: The finally block in Python is executed only if no exception occurs in the try block.
4	State whether the following statement is True or False: An exception may be raised even if the program is syntactically correct.
5	State True or False: In Python, if an exception is raised inside a try block and not handled, the program will terminate without executing any remaining code in the finally block.

ANSWERS

1	(B) WORD5#OVER
2	False
3	False
4	True
5	False

DATA FILE HANDLING
TEXT FILE



- A file is a collection of data stored on a storage device.
- Used to store data permanently for later use.
- Files can be read from and written to using programming languages like Python, C++, Java, etc.

Types of Files

1. Text Files

- Store data in readable characters (ASCII or Unicode).
- Extensions: .txt, .py, .html
- Can be opened and edited with any text editor.
- **CSV Files (Comma-Separated Values)**
  - Special type of text files that stores tabular data.
  - Each line represents a row; values separated by commas.
  - Extension: .csv
  - Easily used with spreadsheet tools and data processing libraries.

2. Binary Files

- Store data in binary (0s and 1s).
- Not human-readable.
- Extensions: .exe, .jpg, .dat
- Used for images, audio, video, and compiled programs.

NOTE: A text file can be understood as a sequence of characters consisting of alphabets, numbers and other special symbols. Files with extensions like .txt, .py, .csv, etc. are some examples of text files.

- ✓ While opening a text file, the text editor translates each ASCII value and shows us the equivalent character that is readable by the human being.
- ✓ Each line of a text file is terminated by a special character, called the End of Line (EOL). For example, the default EOL character in Python is the newline (\n).

Steps to Process Text Files:

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)

ex1. Fobj= open('abc.txt,'w')

ex2. Fobj= open('D:\\Computer\\abc.txt', 'w') #double slash should be used in file path

ex3. Fobj= open(r'D:\Computer\abc.txt, 'w') #if you don't want to use // then use r in front of file

Note: if file mode is not mentioned in open function, then default file mode, 'r' is used

File Open Modes

File Mode	Description	File Offset position
<r>	Opens the file in read-only mode.	Beginning of the file

<rb>	Opens the file in binary and read-only mode.	Beginning of the file
<r+>	Opens the file in both read and write mode.	Beginning of the file
<w>	Opens the file in write mode. If the file already exists, all the contents will be overwritten. If the file doesn't exist, then a new file will be created.	Beginning of the file
<wb>	Opens the file in binary mode for writting. If the file already exists, all its contents will be overwritten. If the file does not exist, a new file will be created.	Beginning of the file
<wb+>	Opens the file in read, write, and binary mode. If the file already exists, its contents will be overwritten. If the file does not exist, a new file will be created.	Beginning of the file
<a>	Opens the file in append mode. If the file doesn't exist, then a new file will be created.	End of the file
<a+>	Opens the file in append and read mode. If the file doesn't exist, then it will create a new file.	End of the file

Use of With Clause

Syntax:

with open ("file name", access mode) as file object:

example: with open("book.txt", 'r') as f:

NOTE: No need to close the file explicitly if we are using with clause

Closing a file

- Once done with the read/write operations on a file, close the file.
- Python provides a close( ) method to do so.
- While closing a file, the system frees the memory allocated to it.
- The syntax of close( ) is:

  file_object.close( )

Writing to a Text File

- For writing to a file, open it in write or append mode.
- If we open an existing file in write mode, the previous data will be erased, and the file object will be positioned at the beginning of the file.
- On the other hand, in append mode, new data will be added at the end of the previous data as the file object is at the end of the file.
- After opening the file, we can use the following methods to write data in the file.
  - **write** ( ) - for writing a single string
  - **writelines**( ) - for writing a sequence of strings

  Example 1:

```
with open ("sample.txt", "w") as f:
    f.write("Hello World\n")
    f.write("This is a second line.\n")
    #You need to manually add \n if you want line breaks.
```

Example 2:

```
lines = ["Line 1\n", "Line 2\n", "Line 3\n"]
with open ("sample_lines.txt", "w") as f:
    f.writelines(lines)
#It does not add newlines automatically – include \n in each string, if needed.
```

Reading from a Text File

- Before reading a file, we must make sure that the file is opened in "r", "r+", "w+" or "a+" mode.
- There are three ways to read the contents of a file:

read ( )	readline( )	readlines( )
Reads the **entire file** as **one string**	Reads **one line at a time**	Reads **all lines** into a **list of strings**
Use when you want to load the whole file at once	Call multiple times to read multiple lines	useful for looping through lines
f.read( ) **Output** "Hello\nWorld\nPython"	f.readline( ) **Output** "Hello\n"	f.readlines( ) **Output** ["Hello\n","World\n", Python"]
to read a specified number of bytes use f.read(n)	used to read a specified number (n) of bytes of data from a file but maximum up to the newline character (\n) f.readline(n)	using readlines( ) function, lines in the file become members of a list, where each list element ends with a newline character ('\n')

SETTING OFFSETS IN A FILE

• **The tell( ) method**	• **The seek( ) method**
This function returns an integer that specifies the current position of the file object in the file.	This method is used to position the file object at a particular position in a file.
The position specified is the byte position from the beginning of the file till the current position of the file object.	In the given 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
The syntax of using tell( ) is: file_object.tell( )	The syntax of using seek( ) is: file_object.seek(offset [, reference_point])

RELATIVE AND ABSOLUTE PATH

Absolute Path:

- It gives the complete path to a file or folder from the root directory.
- Always starts from the drive name (Windows) or root "\" (Linux/Mac).
- Used when the exact location of the file is known.

Example: f = open ("C:\Users\John\Documents\data.txt", "r")

Relative Path:

- It gives the **location relative to the current working directory** (where your Python script is).
- Useful for portability across systems.

Example:

f = open ("files\data.txt", "r")

Special Notations:

. → Current directory

.. → Parent directory

```
# Access file in parent directory
f = open ("..\data.txt", "r")
#To get current working directory
import os
print(os.getcwd( ))
```

Multiple Choice Questions

1.	What does the 'r' mode do when open a file in Python? a) Opens a file for writing only b) Opens a file for reading only c) Opens a file for appending d) Opens a file in binary mode
2.	Which of the following functions changes the position of file pointer? a. flush( ) b. tell( ) c. seek( ) d. offset( )
3.	Which of the following function returns a list datatype? a. d=f.read( ) b. d=f.read(10) c. d=f.readline( ) d. d=f.readlines( )
4.	What will file.write("Hello\nWorld") do in a text file? a. Write the string without a newline b. Raise an error c. Write "Hello World" on a single line d. Write "Hello" and "World" on separate lines
5.	The correct syntax of seek( ) is: a. file_object.seek(offset [, reference_point]) b. seek(offset [, reference_point]) c. seek(offset, file_object) d. seek.file_object(offset)
6.	What will be the output of the following statement in python? (fh is a file handle) fh.seek(-30,2) Options:- It will place the file pointer:- a. at 30th byte ahead of current file pointer position b. at 30 bytes behind from end-of file c. at 30th byte from the beginning of the file d. at 5 bytes behind from end-of file.
7.	Which Python function is used to open a text file for reading? a. open ("file.txt", "w") b. open ("file.txt", "r") c. read("file.txt") d. write("file.txt")
8.	Text file student.txt is stored in the storage device. Identify the correct option out of the following options to open the file in read mode. i. myfile = open('student.txt','rb') ii. myfile = open('student.txt','w') iii. myfile = open('student.txt','r') iv. myfile = open('student.txt') a. only i b. both i and iv c. both iii and iv d. both i and iii

9.	What is the correct way to ensure a file is automatically closed after reading? a) file.close( ) b) with open('file.txt') as file: c) open('file.txt').close( ) d) read(file.txt)
10.	State True or False. The writelines( ) method automatically adds newline characters (\n) after each line.

ANSWERS:

1	**b)** Opens a file for reading only	2	**c)** seek( )
3	**d)** f.readlines( )	4	**d)** Write "Hello" and "World" on separate lines
5	**a)** file_object.seek(offset [, reference_point])	6	**b)** at 30 bytes behind from end-of file
7	**b)** open("file.txt", "r")	8	**c)** both iii and iv
9	**b)** with open('file.txt') as file:	10	False

ASSERTION (A) and REASONING (R)

	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.
1.	**Assertion (A):** Opening a file in 'a' mode will delete its previous content. **Reason (R):** The 'a' mode appends new data at the end of the existing file content.
2.	**Assertion (A):** The with statement ensures a file is properly closed after its block finishes. **Reason (R):** Using with avoids the need to explicitly call close( ) on a file object.
3.	**Assertion (A):** Files must be closed using the close ( ) function to ensure data is saved properly. **Reason (R):** Not closing a file may result in data loss or corruption
4.	**Assertion (A):** Opening a file in write mode 'w' will delete its existing contents. **Reason (R):** In Python, the 'w' mode creates a new file if it doesn't exist, but preserves old content if the file exists.
5.	**Assertion (A):** The writelines( ) method adds newline characters automatically after each line. **Reason (R):** writelines( ) writes a list of strings.
6.	**Assertion (A):** File objects in Python have methods like read( ), write( ), and close( ). **Reason (R):** These methods are built-in for managing file operations in Python.
7.	**Assertion (A):** seek(0) moves the file pointer to the beginning of the file. **Reason (R):** The seek( ) function is used to move the file pointer to a specific location.
8.	**Assertion (A):** The with open( ) statement automatically closes the file after its block is executed. **Reason (R):** It helps prevent memory leaks and is a best practice in file handling.
9.	**Assertion (A):** The readline( ) function reads one line from a file. **Reason (R):** readline( ) can be used in a loop to read the entire file line by line.
10.	**Assertion (A):** The read( ) function reads the entire content of a file as a single string. **Reason (R):** The read( ) function returns a list of lines from the file.

Answers

1	**a**	**2**	**d**	**3**	**a**	**4**	**c**	**5**	**d**
6	**a**	**7**	**a**	**8**	**a**	**9**	**a**	**10**	**c**

Short Answer Type Questions

1.	Write a Python statement to open a text file named notes.txt in read mode

2.	Write a Python code snippet to count the number of lines in a text file named data.txt
3.	Difference between 'w' and 'a' file opening mode in text files.
4.	How is readlines( ) different from read( ) when used with a text file?
5.	Display only the words starting with a vowel from the text file poem.txt
6.	Display words longer than 5 characters from the text file para.txt
7.	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
8.	Write a Python program to count all the lines having 'a' as last character.
9.	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"
10.	What does the split( ) function do when used on a line read from a text file? Explain with example.

Answers

1

```
file = open("notes.txt", "r")  or file = open("notes.txt")
```

2

```
with open("data.txt", "r") as f:
        lines = f.readlines( )
        print("Number of lines:", len(lines))
```

3

w-Write mode	**a**-append mode
Opens the file for writing only.	Opens file for writing data at the end of the file i.e. appending.
If the file dot exits, a new file is created and the file pointer is placed at the beginning of the file.	If the file dot exits, a new file is created. File pointer placed at the beginning of the file.
If the file exists, its content is overwritten and file pointer is placed at the beginning of the file.	If file already exists, its is opened for writing and the file pointer is placed at the end of the file.

4

read( )	readlines( )
returns the entire file content as a single string.	returns a list of strings, where each string is a line from the file.

5

```
#Display only the words starting with a vowel from poem.txt
vowels = 'aeiouAEIOU'
f=open("poem.txt", "r")
content = f.read()
words = content.split()
for word in words:
    if word[0] in vowels:
        print(word)
```

6	#Display words longer than 5 characters from para.txt f=open("para.txt", "r") lines = f.readlines() for line in lines: for word in line.split(): if len(word) > 5: print(word)
7	def displayTheThis(): num=0 fin=open("sample.txt","r") N=fin.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)
8	#Number of lines having 'a' as last character count=0 f=open('fracture.txt',"r") data=f.readlines() for line in data: if line[-2] == 'a': #lines is terminated with '\n' hence -2 for last charcater of line count=count+1 print("Number of lines having 'a' as last character is/are : " ,count) f.close()
9	a) Text File b) File.write("ABC")
10	The split( ) function breaks a string into a list of words using whitespace as the default separator. line = "Hello world" print(line.split( )) **output** ['Hello', 'world']

Long Answer Type Questions

1.	Program to count number of "ME" or "MY" in a text file story.txt presnt in D:\xiic
Ans	#Program to count number of "ME" or "MY" in a text file story.txt def countwords(): f=open("D:\\xiic\\story.txt","r") count=0 lst=f.read() for i in lst.split(): if i.upper()=="Me".upper() or i.upper()=="My".upper(): count=count+1 print ("Number of me or my is", count) f.close()

	countwords()
2.	Program to count number of “A” or “M” in a text file story.txt
Ans	def countAM( ): f=open("D:\\xiic\\story.txt","r") countA=0 countM=0 CON=f.read() for i in CON: if i.upper()=='A': countA=countA+1 elif i.upper()== 'M': countM=countM+1 print ("Number of A is ", countA, "\nNumber of M is:", countM) f.close() countAM()
3	Program to count number of lines starting with “M” in the tet file D:\xiic\story.txt.
Ans	#Program to count number of lines starting with “M” def countlines( ): f=open("D:\\xiic\\story.txt","r") count1=0 LN=f.readlines( ) for i in LN: If i[0]=="M": count1+=1 print("Number of lines starting with M=", count1) f.close( ) countlines ( )
4	Program to print lines not staring with a vowel from a text file D:\XIIC\STORY.TXT
Ans	#Program to print lines not staring with a vowel from a text file TESTFILE.TXT def COUNTLINES( ): count=0 file =open ('D:\\XIIC\\STORY.TXT', 'r') lines=file.readlines ( ) for w in lines: if (w[0]).lower( ) not in 'aeoiu': count count + 1 print ("The number of lines not starting with any vowel: ", count) file.close( ) COUNTLINES( )
5	Write a function ETCount( ) in Python, which should read each character of a text file “D:\XIIC\TESTFILE.TXT” and then count and display the count of occurrence of alphabets E and T individually (including small cases e and t too)
Ans	#Write a function ETCount( ) in Python, which should read each character of a tex def ETCount(): file open ('D:\\XIIC\STORY.TXT', 'r')

	lines=file.read() countE=0 countT=0 for ch in lines: if ch in 'Ee': countE = countE +1 if ch in 'Tt': count = countT+1 print ("The number of E or e: ", countE) print ("The number of T or t : ", countT) file.close() ETCount()
6	Program to count the word “AND” in any case in a text file 'D:\XIIC\STORY.TXT'
Ans	#Program to count the word "AND" in a text file STORY.TXT. def COUNT_AND(): count=0 file=open('D:\\XIIC\\STORY.TXT', 'r') line=file.read() word=line.split() for w in word: if w.upper() == "AND": count=count+1 print ("Number of word AND is", count) file.close() COUNT_AND()
7	Write a function in Pyton to read lines from a text file D:\XIIC\story.txt, and display only those lines, which are starting with an alphabet 'P'.
Ans	#Write a function in Pyton to read lines from a text file story.txt, and #display only those lines, which are starting with an alphabet 'p'. def rdlines(): file= open('D:\\XIIC\\story.txt','r') lines=file.readlines() for line in lines: if line[0] == 'p': print(line) file.close() rdlines()
8	Program to remove lines starting with @ in d:\xiic\source.txt and write it into file d:\xiicanother file.
Ans	def filter(): fin=open("d:\\xiic\\source.txt","r") fout=open("d:\\xiic\\newfile2.txt", "w") text=fin.readlines() for i in text: if i[0]!="@":

	fout.write(i) fin.close() fout.close() def showcontent () : fin=open("d:\\xiic\\newfile2.txt","r"} text=fin.read() print(text) fin.close() filter() showcontent()
9	Write a function RevText( ) to read a text file " d:\xiic\story.txt " and print only the words starting with 'I' in reverse order. Example: If value in text file is: INDIA IS MY COUNTRY Output will be: AIDNI SI MY COUNTRY
Ans	def revtext(): f=open("D:\\xiic\\story.txt","r") S=" " while True: d=f.readline() if not d: break else: m=d.split( ) for i in m: if i[0]=='i' or i[0]=='I': S=S+" "+(i[::-1]) else: S=S+" "+i print (s) S="" revtext()
10	Read a text file and display the number of vowels/consonants/uppercase/lowercase. characters in the file.
Ans	with open("d:\\xiic\\story.txt","r") as f: str=f.read() vowelst=['a', 'e', 'i', 'o', 'u', 'A', 'E', 'I', '0', 'U'] vowel=0 consonants=0 uppercase=0 lowercase=0 for i in str: if i in vowelst: vowel+=1 elif i.isalpha(): consonants+=1 if i.isupper(): uppercase+=1 elif i.islower():

```
        lowercase+=1
print('vowel:',vowel,'\n','consonants:',consonants,'\n','uppercase:',uppercase,'\n','lowercase:',lowercase,'\n')
```

BINARY FILE

A binary file is a file whose content is in a binary format (0s and 1s). It stores data as a sequence of bytes (each byte = 8 bits). 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.

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 or UNICODE characters 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.
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.

Steps to process a binary file

- Opening a file
- Writing data into a file
- Reading data from 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>)

Closing a Binary file

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

The syntax of close( ) is as follows:
<file_object>.close( )

FILE MODES:

Mode	Description
Rb	Open file in binary mode for reading only. The file pointer stands at the beginning of the file. Gives error if file does not exist
rb+or r+b	Open file in binary mode for both reading and writing. The file pointer stands at the beginning of the file. Gives error if file does not exist
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+ or w+b	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+ or a+b	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.

Import pickle Module in Python

To write data to a binary file and read it subsequently, we need to use the Python module **pickle**.
The module pickle is used for **serializing** and **de-serializing** any Python object structure

Pickling/Serialization: The process of converting the structure (lists and dictionary etc.) into a byte stream just before writing to the file.
Unpickling/De-serialization: The reverse of pickling process where information from byte stream gets converted into object structure.



Methods of pickle module:

1. **pickle.dump( ):** Used to pickle the data object into byte stream and stores it in the binary file.
   Syntax
   pickle.dump(object , file_object) - used to write any object to the binary file.
2. **Pickle.load( ) :** Used to read data from a binary file. It converts the binary data back into the original python object (like a list, dictionary etc.).
   Syntax
   object=pickle.load(file_object) - used to read object from the binary file.

Writing data into a Binary File

Example:

```
import pickle
f=open("abc.dat","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( )
```

Reading data from a Binary file

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 handle runtime EOFError
4. Now load data into an object through the load( ) function
5. Print data as per need
6. Close the file

Example:

```
import pickle
f=open("abc.dat",'wb+')      # This will open the file in write and read mode
x=[1,2,3,4,5]
pickle.dump(x,f)             # This will write the object List into the file.
f.seek(0)                    #This will move the file pointer to the begining of the file
x=pickle.load(f)             #This will read the object from the file and store in variable.
print(x)
```

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:

```
import pickle
def bf_search():
  try:
    f=open("Sports.dat","rb")
    pc=int(input("Player 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:
```

```
            break
    except FileNotFoundError:
        print("The file Sports.dat does not exist")
    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():
    import pickle
    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():
    import pickle
    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()
```

Multiple choice Questions

1.	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+
2.	Nila wants to store a list of dictionaries into a binary file. Which of the following Python modules should she use? a) os b) csv c) pickle d) json
3.	Pick the correct syntax to read a binary file using pickle: a) pickle.read(file) b) pickle.load(file) c) pickle.open(file) d) pickle.input(file)
4.	What does the pickle.dump(obj, file) function do? a) Reads binary data from a file b) Writes text data to a file c) Converts a Python object into byte stream and writes it to a file d) Appends an object to a list
5.	Which file mode should be used to write a binary file in Python? a) 'w' b) 'r' c) 'wb' d) 'rb'
6.	What will happen if you try to read a binary file using 'r' mode instead of 'rb'? a) It will read data correctly. b) It will raise a SyntaxError. c) It will convert binary data into text d) It may raise an error or return incorrect data
7.	Which of the following is a key advantage of binary files over text files? a) Easy to read with any text editor b) Allows direct storage of complex Python objects b) Consumes more storage space d) Cannot be shared across system

Answers :

1	2	3	4	5	6	7
c	c	b	c	c	d	b

Assertion and Reasoning

	Mark the correct choice as: a) Both A and R are true, and R is the correct explanation of A. b) Both A and R are true, but R is not the correct explanation of A. c) A is true, but R is false. d) A is false, but R is true.
1.	**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
2.	**Assertion (A):** The pickle module in Python is used to store and retrieve Python objects in binary files. **Reason (R):** The pickle.dump( ) method reads data from a binary file and the pickle.load( ) method writes data to a binary file.
3.	**Assertion (A):** Binary files are used to store data in the same format as it is stored in memory. **Reason (R):** Binary files convert Python objects into byte streams which are machine-readable.
4.	**Assertion (A):** You can open a binary file using only 'r' or 'w' mode in Python. **Reason (R):** Binary files should be opened with 'rb' or 'wb' modes to ensure correct handling of byte streams.
5.	**Assertion (A):** Binary files are preferred when large volumes of data need to be stored with structure. **Reason (R):** Binary files are human-readable and can be opened using any text editor.
6.	**Assertion (A):** Binary files are preferred when large volumes of data need to be stored with structure. **Reason (R):** Binary files are human-readable and can be opened using any text editor.

Answers:

1	c) A is True and R is False
2	c) A is true, but R is false.
3	a) Both A and R are true, and R is the correct explanation of A
4	d) A is false, but R is true
5	c) A is true, but R is false
6	a) Both A and R are true, and R is the correct explanation of A

1	2	3	4	5	6
c	c	a	d	c	a

Short Answer Type Questions:

1	What is the purpose of the pickle module in Python?

2	State two advantages of using binary files over text files in Python.
3	How many times should pickle.load( ) be called while reading a binary file? Explain your answer with reference to the number of times pickle.dump( ) was used while writing the file.

Answers :

1	The pickle module is used for serializing and de-serializing Python objects into binary format. It allows saving complex data types like dictionaries, lists, etc., into a binary file and retrieving them later.
2	Binary files can store complex Python objects (like dictionaries, lists) directly. They are more secure and efficient in terms of storage space and speed.
3	The number of times pickle.load( ) should be called is **equal to the number of times** pickle.dump( ) **was called** when the file was written. If the file was written using pickle.dump( ) 3 times, then you must call pickle.load( ) 3 times to read all the data. Since we may not know the exact count, load( ) is usually called inside a loop until EOFError occurs.

Long Answer Type Questions:

1.	Karthik is a manager working in a recruitment agency. He needs to manage the records of various candidates. For this, he wants the following information of each candidate to be stored: Candidate_ID – integer Candidate_Name – string Designation – string Experience – float You, as a programmer of the company, have been assigned to do this job for Karthik. i. Write a function to input the data of a candidates and append it in a binary file ii. Write a function to update the data of candidates whose experience is less than 10 years and change their designation to "Assistant Manager". iii. Write a function to read the data from the binary file and display the data of all those candidates who are not "Assistant Manager".
Ans 1(i)	import pickle def input_candidates( ): f=open("candidate.dat","ab") n = int(input("Enter the number of candidates you want to add: ")) for i in range(n): candidate_id = Candidate ID: ")) candidate_name = input("Enter Candidate Name: ") designation = input("Enter Designation: ") experience = float(input("Enter Experience (in years):")): pickle.dump([candidate_id, candidate_name, designation, experience],f) print("Candidate data appended successfully. ")
(ii)	import pickle def update_assistant_manager(): updated_candidates=[] try:

	with open('candidates.dat', 'rb') as file: while True: try: Candidate= pickle.load(file) if candidate[3] < 10: # If experience <10 years candidate[2] = 'Assistant Manager' updated_candidates.append(candidate) except EOFError: break # End of file reached except FileNotFoundError: print("No candidate data found. Please add candidates first") with open('candidates.dat', 'wb') as file: for candidate in updated_candidates: pickle.dump(candidate, file)
1(iii)	import pickle def display_non_assistant_managers( ): try: with open('candidates.dat', 'rb') as file: while True: try: candidate pickle.load(file) if candidate[2] != 'Assistant Manager':# Check if not Assistant Manager print("Candidate ID: ",candidate[0]) print("Candidate Name:",candidate[1]) print("Designation:",candidate[2]) print("Experience:",candidate[3]) print("_____") except EOFError: break # End of file reached except FileNotFoundError: print("candidate data found. Please add candidates first.")
2	A school is maintaining student records in a binary file records.dat. Each record stores the following data: Roll Number (int) Name (str) Percentage Marks (float) Write Python functions to do the following i. Create a binary file by entering student data in the form of dictionaries. ii. Count and display the number of students who scored less than 40% (i.e., need academic improvement).
Ans 2(i)	import pickle def create_file( ): with open('records.dat', 'wb') as file: n = int(input("Enter number of students:")) for I in range(n): student = {} student['roll'] = int(input("Enter Roll No:")) student['name'] =input("Enter Name:")

	student['percentage'] = float(input("Enter Percentage:")) pickle.dump(student, file)
2(ii)	import pickle def count_low_score( ): count = 0 try: with open('records.dat', 'rb') as file: while True: try: student = pickle.load(file) if student['percentage'] < 40: Count+= 1 except EOFError: break print("Number of students with less than 40% marks:", count) except FileNotFoundError: print("File not found!")
3	A binary file books.dat contains records of books. Each record is stored as a dictionary with the following fields: bid- (Book ID – integer) title- (Book Title – string) price- (Book Price – float) The librarian wants to perform the following operations efficiently: i. Define a function add_books(n) that adds n book records to the binary file. ii. Define a function update_price(Bid, new_price) that updates the price of the book whose Book ID is Bid. iii. Define a function count_books_in_range(low, high) that returns the number of books whose price lies between low and high (inclusive).
3(i)	import pickle def add_books(n): with open("books.dat","ab") as f: for i in range(n): book ={} book['bid']=int(input("Enter Book ID:")) book['title']= input("Book Title:") book['price']= float(input("Enter Book Price: ")) pickle.dump(book, f)
(ii)	import pickle def update_price(Bid, new_price): updated = False temp_list =[] try: with open("books.dat", "rb") as f: while True: try: book = pickle.load(f) if book['bid'] == Bid: book['price'] = new_price

	updated = True temp_list.append(book) except EOFError: break with open("books.dat", "wb") as f: for book in temp_list: pickle.dump(book, f) if updated: print("Price updated successfully.") else: print("Book ID not found.") except FileNotFoundError: print("File not found.")
(iii)	import pickle def count_books_in_range(low,high): count = 0 try: with open("books.dat","rb") as f: while True: try: book = pickle.load(f) if low <=book['price'] <=high: count +=1 except EOFError: break print("Total books priced between",low,"and",high,":",count) except FileNotFoundError: print("File not found.")
4.	A binary file patient.dat stores records of patients in a hospital. Each record is stored as a dictionary with the following fields: pid- (Patient ID – integer) pname- (Patient Name – string) age- (integer) bill- (float – hospital bill amount) Perform the following tasks using three user-defined functions: i. Define a function add_patients(n) to add n new patient records to patient.dat. ii. Define a function update_bill(Pid, amount) that updates the bill of a patient by adding the given amount to the existing bill using their Patient ID. iii. Define a function list_senior_patients( ) to display details of all patients aged 60 and above.
Ans 4(i)	import pickle def add_patients(n): with open("patient.dat", "ab") as f: for i in range(n): patient = {} patient['pid'] = int(input("Enter Patient ID: ")) patient['pname'] = input("Enter Patient Name: ") patient['age'] = int(input("Enter Age: "))

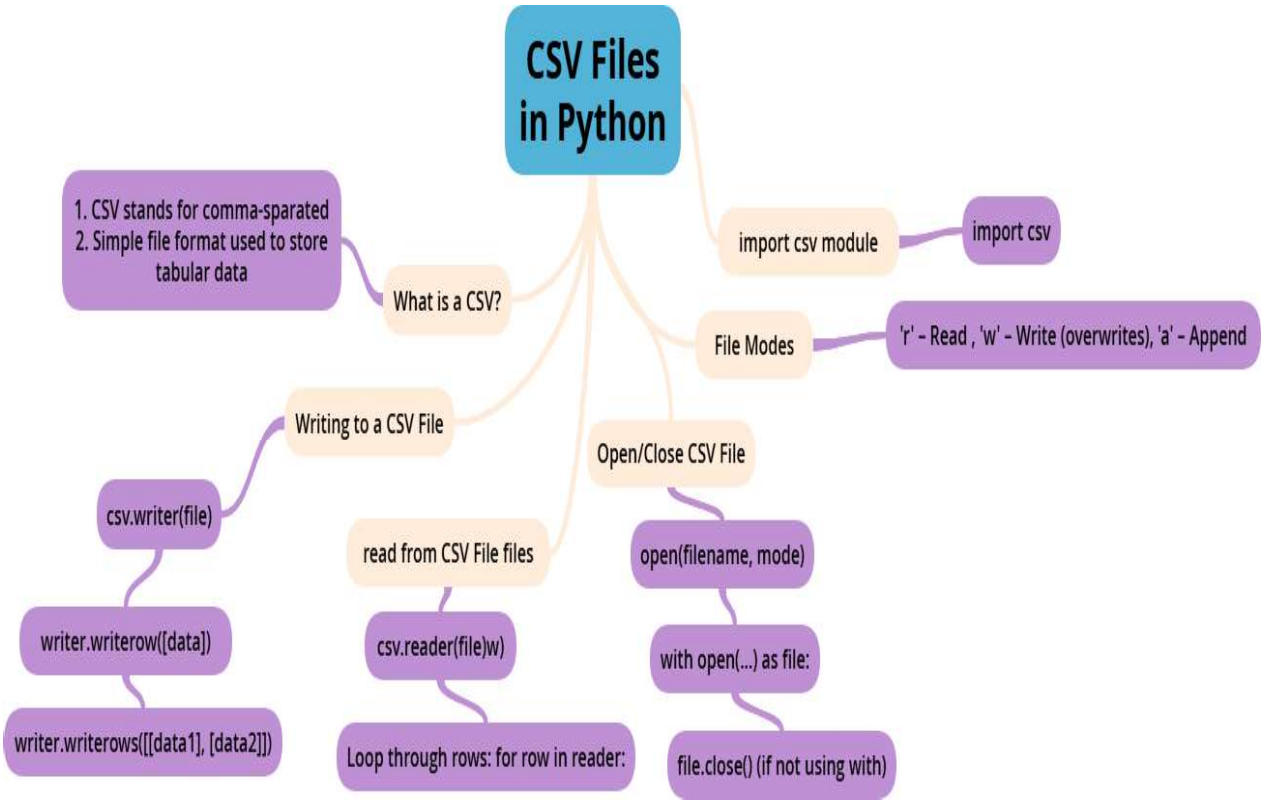
	patient['bill'] = float(input("Enter Bill Amount: ")) pickle.dump(patient, f)
4(ii)	import pickle def update_bill(Pid, amount): updated = False patients = [] try: with open("patient.dat", "rb") as f: while True: try: patient = pickle.load(f) if patient['pid'] == Pid: patient['bill'] += amount updated = True patients.append(patient) except EOFError: break with open("patient.dat", "wb") as f: for p in patients: pickle.dump(p, f) if updated: print("Bill updated successfully.") else: print("Patient ID not found.") except FileNotFoundError: print("File not found.")
4(iii)	import pickle def list_senior_patients(): try: with open("patient.dat", "rb") as f: print("Senior Patients (Age 60+):") found = False while True: try: patient = pickle.load(f) if patient['age'] >= 60: print(patient) found = True except EOFError: break if not found: print("No senior patients found.") except FileNotFoundError: print("File not found.")
5.	A binary file, **EMP.DAT** has the following structure : **[Emp_Id, Name, Salary]** where **Emp_Id :** Employee id **Name:** Employee Name **Salary**: Employee Salary

	Write a user defined function, **disp_Detail( )**, that would read the contents of the file **EMP.DAT** and display the details of those employees whose salary is below **25000**.
Ans	import pickle def Copy_new(): F2=open("new_items.dat", "wb") try: F1=open("items.dat","rb") Datal=pickle.load(F1) Data2={} for K,V in Datal.items(): if v[1]>1000: Data2[K]=V pickle.dump (Data2, F2) F2.close() except: print("File not found!") F1.close()
6.	Mr. Ashok is working on a Toy Shop project to manage toys records using Python. The toys data is stored in a binary file named **TOYDATA.DAT**. The binary file **TOYDATA.DAT** contains each record in given format: {"Toy_ID": T_ID, "TName":toy_name, "Price":price} Where ● **Toy_ID**: Toy ID (string) ● **TName**: Toy name (string) ● **Price**: Price of toy (integer) You as a programmer, help him to write following python functions: i. **ADD_Data( )** : To write **n** records in binary file TOYDATA.DAT by taking the values for each record from user. ii. **SHOW_Data( )** : Read all records from binary file and display them. iii. **Remove_Toy( )** : that deletes the record of a toy in the file **TOYDATA.DAT** based on the **Toy ID** provided by the user. If the Toy ID does not exist in the file, display an appropriate message.
Ans 6(i)	import pickle def ADD_Data(): f=open("TOYDATA.DAT", "wb") L=[] n=int(input("How many records you want to enter: ")) for i in range(n): T_ID=input("Enter Toy ID: ") toy_name=input("Enter Toy Name: ") price=int(input("Enter Price: ")) D={"Toy_ID":T ID, "TName": toy name, "Price":price} L.append(D) pickle.dump(Lf) f.close()
6(ii)	def SHOW_Data(): f=open("TOYDATA.DAT", "rb") L=pickle.load(f)

	print(L) f.close()
6(iii)	def Remove_Toy(): Toy_ID=input("Enter Toy ID: ") f=open("TOYDATA.DAT", "rb+") found=0 L=pickle.load(f) M=[] for D in L: if Toy ID not in D["Toy_ID"]: M.append(D) else: found=1 if found==1: f.seek(0) pickle.dump(M,f) print("Record Deleted successfully") else: print("Record not found") f.close()
7.	A file, PASSENGERS.DAT, stores the records of passengers using the following structure : [PNR, PName, BRDSTN, DESTN, FARE] where : PNR – Passenger Number (string type) PName – Passenger Name (string type) BRDSTN – Boarding Station Name (string type) DESTN – Destination Station Name (string type) FARE – Fare amount for the journey (float type) Write user defined functions in Python for the following tasks : (i) Create( ) – to input data for passengers and write it in the binary file PASSENGERS.DAT. (ii) SearchDestn(D) –to read contents from the file PASSENGERS.DAT and display the details of those Passengers whose DESTN matches with the value of D. (iii) UpdateFare( ) – to increase the fare of all passengers by 5% and rewrite the updated records into the file PASSENGERS.DAT.
Ans 7(i)	import pickle def Create(): F=open ("PASSENGERS.DAT", "wb") PNR=input("PNR No: ") PName=input("Name: ") BRDSTN=input ("Boarding at: ") DESTN=input("Destination: ") FARE=float(input("Fare: ")) Rec=[PNR, PName, BRDSTN, DESTN, FARE] pickle.dump (Rec, F) F.close()
7(ii)	def SearchDestn(D): try: F=open ("PASSENGERS.DAT", "rb")

	Rec=pickle.load(F) for R in Rec: if R[3]==D: print(R) F.close() except: print("File not found!")
7(iii)	def UpdateFare(): try: FR=open ("PASSENGERS.DAT", "rb+") Rec=pickle.load(FR) for I in range (len (Rec)): Rec [1] [4] += (Rec [I] [4] * 0.05) print("Updation Done!") FR.seek(0) pickle.dump (Rec, FR) FR.close() except: print("File not found!")
8.	Consider a binary file, items.dat, containing records stored in the given format : {item_id: [item_name,amount]} Write a function, Copy_new( ), that copies all records whose amount is greater than 1000 from items.dat to new_items.dat.
Ans	import pickle def Copy_new(): F2=open("new_items.dat", "wb") try: F1=open("items.dat","rb") Datal=pickle.load(F1) Data2={} for K,V in Datal.items(): if v[1]>1000: Data2 [K]=V pickle.dump (Data2, F2) F2.close() except: print("File not found!") F1.close()

COMMA SEPARATED VALUES(CSV) FILES



CSV stands for Comma Separated Values CSV file is a type of plain text file means data stored in form of ASCII or Unicode characters Each line is a row and in row each piece of data is separated by a comma It is common format for data interchange.

Steps to Process CSV Files:

1. First of all we have to **import csv module** for file operation.

2. For writing to CSV Files ,

i. We open file in 'w' writing or 'a' append mode using open( ) method

f=open('filename.csv','w',newline=' ')

By setting newline='', Python don't modify the newline characters at all. This is commonly used when working with the csv module to ensure that the csv.writer handles the newline characters correctly (i.e., without adding extra blank lines between rows).

i. Creating a writer object, associated with the file object. The writer object allows you to write rows of data into the file in CSV format. Through csv.writer( ) method ,we create it.

w_obj=csv.writer(f)

ii. Perform writing operation (read data from user and write on to csv file)

writerow( ) used for writing single row

writerows( )used for writing multiple rows

3. For Reading from CSV File

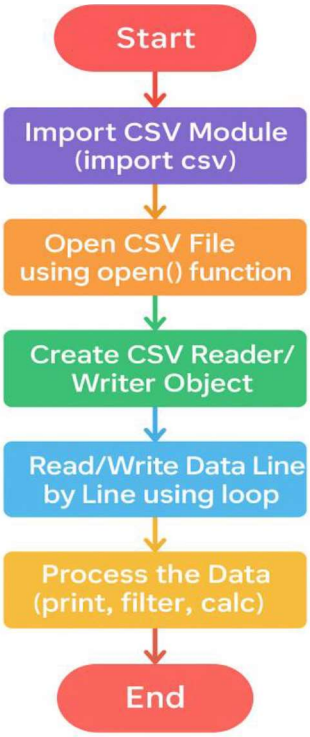
i. Creating a reader object, associated with the file object we use csv.reader( ) method. It Creates a CSV reader object that can iterate over the rows of the file object.

r_obj=csv.reader(f)

ii. Perform reading operation (read data from csv file and display using python)

4.Close the file using close( ) method

f.close( )



csv.reader( )	The csv.reader() function in Python is used to read data from a CSV (Comma Separated Values) file. It is part of Python's built-in csv module. csv.reader(file_object, delimiter=',') **Parameters:** • file_object: A file object opened using open( ). • delimiter: (Optional) Specifies the character used to separate fields. Default is comma (,). It returns an **iterator** that returns each row in the CSV file as a **list of strings**. **Example** import csv with open('data.csv', 'r') as file: reader = csv.reader(file) for row in reader: print(row)
csv.writer( )	The csv.writer( ) function in Python is used to **write data to a CSV file**. It is part of Python's built-in csv module and writes rows as comma-separated values. csv.writer(file_object, delimiter=',') **Parameters:** • file_object: A file object opened in **write ('w') or append ('a')** mode. • delimiter: (Optional) Character that separates values. Default is comma (,).

	• It returns a **writer object** that lets you write rows (lists or tuples) to the CSV file. **Common Methods:** • writer.writerow(row) → Writes a **single row**. • writer.writerows(list_of_rows) → Writes **multiple rows**. **Example:** import csv with open('output.csv', 'w', newline='') as file: writer = csv.writer(file) # Write header writer.writerow(['Name', 'Age', 'City']) # Write multiple data rows writer.writerows([['Alice', 23, 'Delhi'],['Bob', 30, 'Mumbai']])

Multiple Choice Questions

1.	Syntax for opening Student.csv file in write mode is myfile = open("Student.csv","w",newline=''). What is the importance of newline=''? a. A newline gets added to the file b. Empty string gets appended to the first line. c. Empty string gets appended to all lines. d. EOL translation is suppressed
2.	Which of the following is not a function / method of csv module in Python? a. read( ) b. reader( ) c. writer( ) d. writerow( )
3.	A CSV file a. is a text file b. can store images c. is a type of python program d. is a Computer Software Validation file
4.	What does the csv.reader( ) function do? a. Reads JSON files b. Reads CSV files row by row as lists c. Converts CSV to a dictionary d. Writes to CSV files
5.	What mode should be used to open a file for reading in CSV format? a. 'w' b. 'x' c. 'r' d. 'rw'
6.	What is the default delimiter in a CSV file? a. Semicolon (;) b. Tab (\t) c. Pipe (|) d. Comma (,)
7.	State True or False: CSV file is also known as delimited text file which can be opened in Notepad and Spreadsheet both.
8.	Which of the following is not a correct way to open a CSV file for writing? a. open('file.csv', 'r') b. open('file.csv', 'a') c. open('file.csv', 'w') d. open('file.csv', 'rw')
9.	Which method writes multiple rows at once in a CSV file? a. writerows( ) b. writeall( ) c. write( ) d. writemultiple( )
10.	When using the csv.reader( ) object to read from a CSV file, what type of data structure is each row of data represented as? a) String b) Dictionary c) List d) Tuple

ANSWERS:

1	2	3	4	5	6	7	8	9	10

d	**a**	**a**	**b**	**c**	**d**	**True**	**a**	**a**	**c**

ASSERTION REASONING 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
1	**Assertion (A) :** CSV file is a human readable text file where each line has a number of fields, separated by comma or some other delimiter. **Reason (R): writerow( )** method is used to write a single row in a CSV file.
2	**Assertion (A):** The csv.reader( ) object returns each row as a list of strings. **Reason (R):** The csv.reader( ) reads the entire file content at once into memory
3	**Assertion (A):** csv.writer( ) can be used to write data row by row into a CSV file. **Reason (R):** writerow( ) method of the csv.writer object writes a single row to the CSV file.
4	**Assertion (A):** The newline='' parameter is used while opening a CSV file in Python. **Reason (R):** It helps prevent insertion of blank lines between rows on Windows systems.
5	**Assertion (A):** csv.writer(file) returns an object that can be used to read data from a CSV file. **Reason (R):** csv.writer( ) is used for writing data into CSV files.

ANSWERS:

1	**2**	**3**	**4**	**5**
b	**c**	**a**	**a**	**d**

Long Answer Questions

1.	Write a Program in Python that defines and calls the following user defined functions: (i) ADD() – To accept and add data of an employee to a CSV file 'record.csv'. Each record consists of a list with field elements as empid, name and salary to store employee id, employee name and employee salary respectively. (ii) COUNTR() – To count the number of records present in the CSV file named 'record.csv'.
Ans 1(i)	import csv def ADD(): f=open('record.csv','a') wr=csv.writer(f) eid=int(input("Enter Employee ID")) name=input("Enter name of employee") sal=float(input("enter salary")) data=[eid,name,sal] wr.writerow(data) ADD()
1(ii)	import csv def COUNTR(): f=open('record.csv','r') c=0 data=csv.reader(f) for x in data: c=c+1

print(c)
COUNTR()

2. Create a function maxsalary( ) in Python to read all the records from an already existing file record.csv which stores the records of various employees working in a department. Data is stored under various fields as shown below:

E_code	E_name	Scale	Salary
A01	Bijesh Mehra	S4	65400
B02	Vikram Goel	S3	60000
C09	Suraj Mehta	S2	45300

Function should display the row where the salary is maximum.

Note: Assume that all employees have distinct salary.

Ans

```
import csv
def maxsalary():
   f=open('record.csv', 'r')
   reader=csv.reader(f)
   skip_header = True
   m= 0
   for row in reader:
      if skip_header:
         skip_header = False
      else:
         if(int(row[3])>m):
            m=int(row[3])
            rec=row
   print('Row with the highest salary: ', rec)
   f.close()
maxsalary()
```

3. A csv file "Happiness.csv" contains the data of a survey. Each record of the file contains the following data:

- Name of a country
- Population of the country
- Sample Size (Number of persons who participated in the survey in that country)
- Happy (Number of persons who accepted that they were Happy)

For example, a sample record of the file may be:
['Signiland', 5673000, 5000, 3426]

Write the following Python functions to perform the specified operations on this file:
(I) Read all the data from the file in the form of a list and display all those records for which the population is more than 5000000.
(II) Count the number of records in the file.

Ans 3(i)

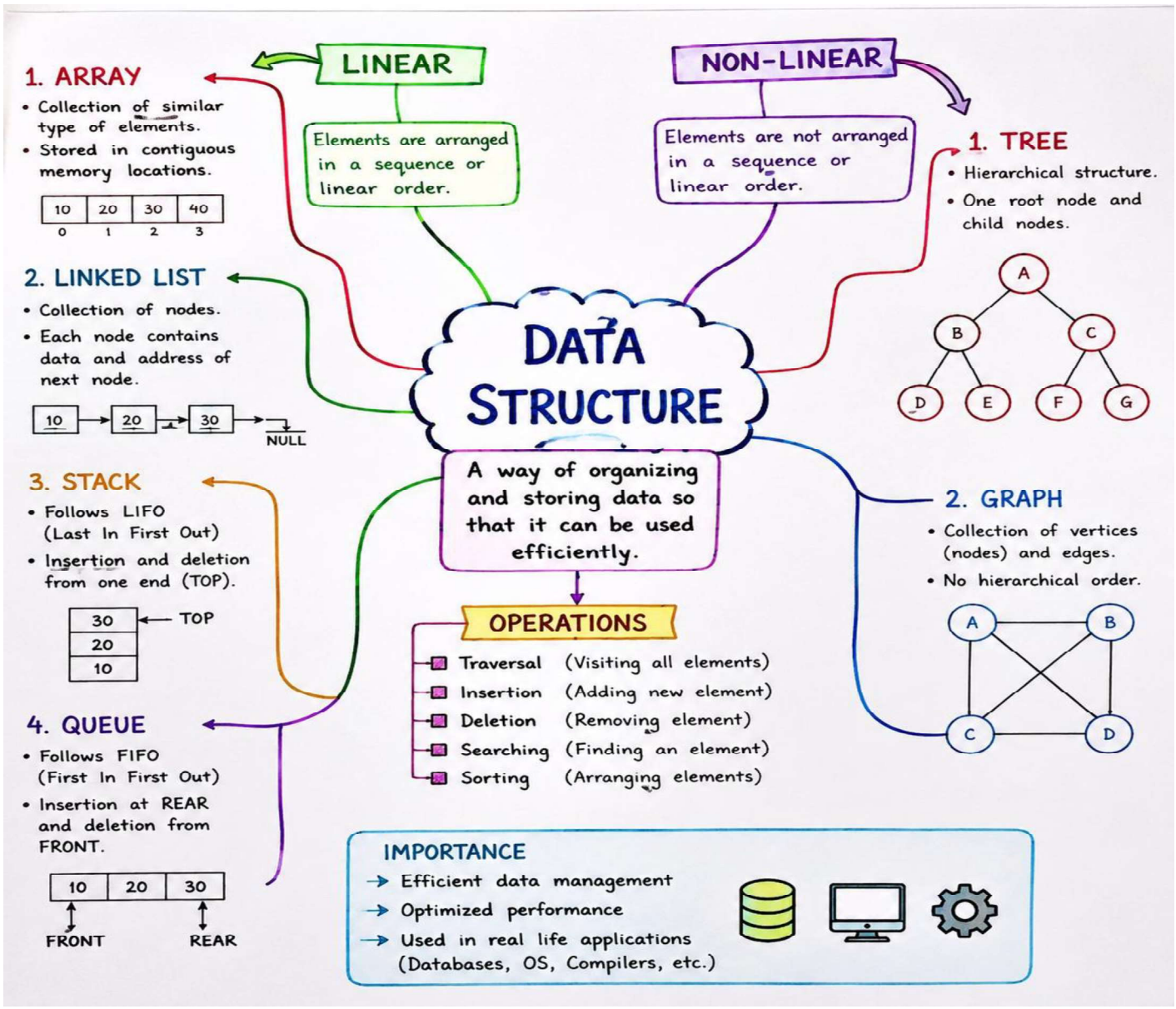
```
def show():
   import csv
   f=open("happiness.csv",'r')
   records=csv.reader(f)
```

	next(records, None) #To skip the Header row for i in records: if int(i[1])>5000000: print(i) f.close()
3(ii)	def Count_records( ): import csv f=open("happiness.csv",'r') records=csv.reader(f) next(records, None) #To skip the Header row count=0 for i in records: count+=1 print(count) f.close()
4	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.
Ans	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()
5	Mr. Mahesh is a Python Programmer working in a school. He has to maintain the records of the sports students. He has created a csv file named **sports.csv,** to store the details. The structure of **sports.csv** is : **[sport_id, competition, prize_won]** where **sport_id, is Sport id (integer)** **competition is competition name (string)** **prize_won is ("Gold", "Silver", "Bronze")** Mr. Mahesh wants to write the following user-defined functions : **Add_detail( ):** to accept the detail of a student and add to a csv file, **"sports.csv"**. **Count_Medal( ):** to display the name of competitions in which students have won **"Gold"** medal. Help him in writing the code of both the functions.
Ans	import csv def Add_detail(): F=open("sports.csv","a") W=csv.writer(F) sport_id=int(input("Sport id:")) competition=input("Competition:") prize_won=input("Prize won:") L=[sport_id,competition,prize_won] W.writerow(L) F.close( ) def Count_Medal( ): F=open("sports.csv","r") L=list(csv.reader(F)) for D in L: if D[2]=="Gold": print("Competition:",D[1]) F.close()
6	Write a program in Python that defines and calls the following user defined functions: (i) Add_Teacher( ) : It accepts the values from the user and inserts record of a teacher to a csv file 'Teacher.csv'. Each record consists of a list with field elements as T_id,Tname and desig to store teacher ID, teacher name and designation respectively. (ii) Search_Teacher( ) : To display the records of all the PGT (designation) teachers.
Ans	import csv def Add_Teacher(): fout=open("Teacher.csv","a",newline="\n") T_id=int(input("Enter Teacher id: "))

	Tname=input("Enter Teacher name: ") desig=input("Enter Designation: ") rec=[T_id,Tname,desig] csvw=csv.writer(fout) csvw.writerow(rec) fout.close() def Search_Teacher(): fin=open("Teacher.csv") csvr=csv.reader(fin) for record in csvr: if record[2]=="PGT": print(record) fin.close() Add_Teacher() Search_Teacher()
7.	Write a program in Python that defines and calls the following user defined functions: (i) COURIER_ADD( ) : It takes the values from the user and adds the details to a csv file 'courier.csv'. Each record consists of a list with field elements as cid, s_name, Source, destination to store Courier ID, Sender name, Source and destination address respectively. (ii) COURIER_SEARCH( ) : Takes the destination as the input and displays all the courier records going to that destination
Ans	import csv def COURIER_ADD() : f1=open("courier.csv","a",newline="\n") writ=csv.writer(f1) cid=int(input("Enter the Courier id")) s_name=input ("Enter the Sender Name") Source=input("Enter the Source Address") destination=input("Enter Destination Name") detail=[cid, s_name, Source, destination] writ.writerow (detail) f1.close() def COURIER_SEARCH(): f1=open("courier.csv", "r") # ignore newline detail=csv.reader(f1) name=input("Enter the Destination Name to be searched") for i in detail : if i[3]==name: print("Details of courier are: ",i) f1.close()

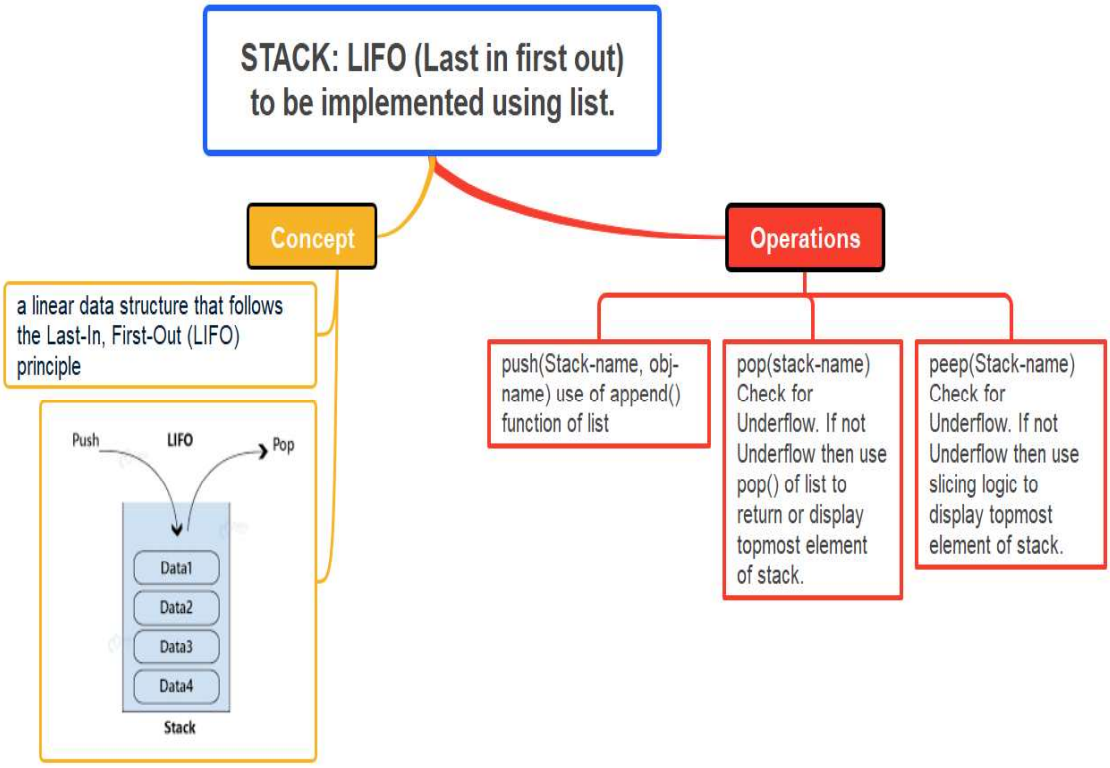
DATA STRUCTURES



Concept: A specialized format for organizing, processing, and storing data in a computer so it can be accessed and modified efficiently, so It is a way to store data in a structured way.

Need: provide a systematic way to **organize, manage, and store data** so it can be used effectively.
key in a database is an attribute or a set of attributes used to uniquely identify a record in a table and establish relationships between tables.

- Built-in data structures available in Python: List, Tuple, Dictionary and Set.
- User Defined data structures in Python: Stack, Queue, Tree, Linked List etc.



STACK:

- A Stack is a Linear data structure which works in LIFO (Last In First Out) manner (or we can say FILO i.e. First In Last Out manner
- Insertion and Deletion of elements will be done only from one end known as TOP.
- In Python, we can use List data structure to implement Stack.

Application of Stack:

1. Expression Evaluation
2. String Reversal
3. Function Call
4. Browser History
5. Undo/Redo Operations

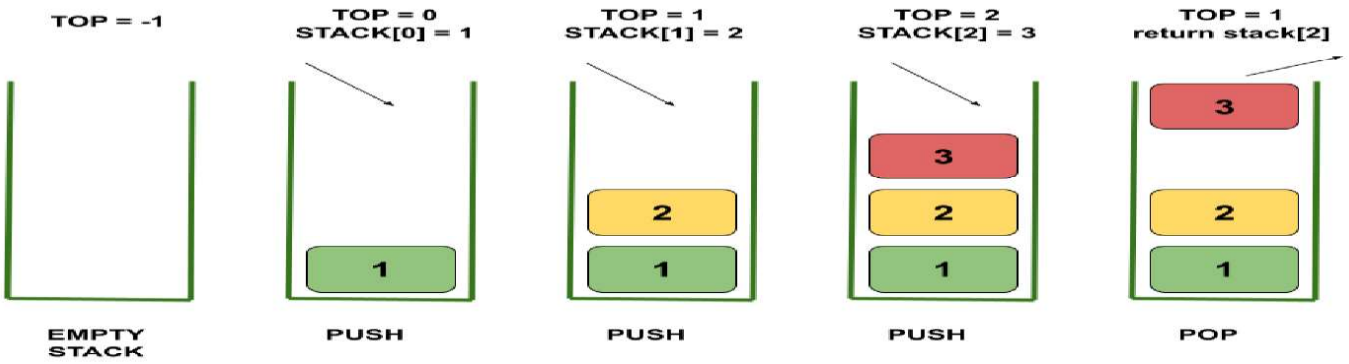
Operations on Stack:

The Stack supports following operations:

1. Push: It adds an element to the TOP of the Stack.
2. Pop: It removes an element from the TOP of the Stack.
3. Peek: It is used to know/display the value of TOP without removing it.
4. isEmpty: It is used to check whether Stack is empty.

OVERFLOW: It refers to the condition in which we try to PUSH an item in a Stack which is already FULL.

UNDERFLOW: It refers to the condition in which we are trying to POP an item from an empty Stack.



Stack Implementation in Python (Using List)

```
stack = [ ]
# Function to push element into the stack
def push( ):
   element = input("Enter element to push: ")
   stack.append(element)
   print("Element  pushed to stack.")
# Function to pop element from the stack
def pop_element( ):
   if not stack:
      print("Stack is empty!")
   else:
      element = stack.pop( )
      print("Element popped from stack.")
# Function to display stack
def display( ):
   if not stack:
      print("Stack is empty!")
   else:
      print("Stack elements (top to bottom):")
      for item in reversed(stack):
         print(item)

# Menu-driven program
while True:
   print("\nSTACK OPERATIONS")
   print("1. Push")
   print("2. Pop")
   print("3. Display")
   print("4. Exit")
   choice = int(input("Enter your choice (1-4): "))
   if choice == 1:
      push( )
   elif choice == 2:
      pop_element( )
   elif choice == 3:
      display( )
```

```
elif choice == 4:
    print("Exiting program...")
    break
else:
    print("Invalid choice! Please try again.")
```

Multiple Choice Questions

1.	What is the principle of a stack? a) FIFO – First In First Out b) LIFO – Last In First Out c) FILO – First In Last Out d) LILO – Last In Last Out
2.	Which Python list method is used to add an element to the stack? a) insert( ) b) add( ) c) append( ) d) push ( )
3.	Which method is used to remove the top element from a stack implemented using a list? a) remove( ) b) delete( ) c) pop( ) d) discard ( )
4.	What will be the output of the following code? stack = [10, 20, 30] stack.pop( ) print(stack) a) [10, 20, 30] b) [10, 20] c) [20, 30] d) Error
5.	What happens if you call pop ( ) on an empty stack? a) Returns None b) Raises IndexError c) Returns -1 d) Does nothing

Answers

1	2	3	4	5
b	c	c	b	b

ASSERTION REASONING 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.
1.	**Assertion (A):** In **Python**, a stack can be implemented using a list. **Reason (R)*:*** A stack is an ordered linear list of elements that works on the principle of First In First Out (FIFO).
2.	**Assertion (A):** A stack can be used to reverse the contents of a text file line by line. **Reason (R):** In a stack, the last element inserted is the first to be removed (LIFO).
3.	**Assertion (A):** Using a stack is an efficient method for checking matching brackets in a file containing Python code. **Reason (R):** Stack allows multiple ends for insertion and deletion of elements.

4.	**Assertion (A):** A stack can be implemented using a list in Python to read a file and store each word for later processing. **Reason (R):** Lists in Python do not support push and pop operations.

Answers

1	2	3	4
c	a	c	c

Long Answer Questions

1.	A dictionary, **d_city** contains the records in the following format: **{state:city}** Define the following functions with the given specifications: (i) **push_city(d_city):** It takes the dictionary as an argument and pushes all the cities in the stack **CITY** whose states are of more than 4 characters. (ii) **pop_city( ):** This function pops the cities and displays **"Stack empty"** when there are no more cities in the stack.
Ans 1(i)	CITY=[ ] def push_city(d_city): for c in d_city: if len(c) > 4: CITY.append(d_city[c])
1(ii)	def pop_city( ): while CITY: print(CITY.pop( )) else: print("Stack empty")
2.	Consider a list named Nums which contains random integers. Write the following user defined functions in Python and perform the specified operations on a stack named BigNums. (i) PushBig( ): It checks every number from the list Nums and pushes all such numbers which have 5 or more digits into the stack, BigNums. (ii) PopBig( ): It pops the numbers from the stack, BigNums and displays them. The function should also display "Stack Empty" when there are no more numbers left in the stack. For example: If the list Nums contains the following data: Nums = [213, 10025, 167, 254923, 14, 1297653, 31498, 386, 92765] Then on execution of PushBig( ), the stack BigNums should store: [10025, 254923, 1297653, 31498, 92765] And on execution of PopBig( ), the following output should be displayed: 92765 31498 1297653 254923 10025 Stack Empty

Ans	def PushBig(Nums,BigNums): for N in Nums: if len(str(N)) >= 5: BigNums.append(N) def PopBig(BigNums): while BigNums: print(BigNums.pop( )) else: print("Stack Empty")
3.	A list contains following record of course details for a University: **[Course_name, Fees, Duration]** Write the following user defined functions to perform given operations on the stack named **'Univ'** : (i) **Push_element( )** - To push an object containing the Course_name, Fees and Duration of a course, which has fees greater than 100000 to the stack. (ii) **Pop_element( )** - To pop the object from the stack and display it. Also, display "Underflow" when there is no element in the stack. For example: If the lists of courses details are: **["MCA", 200000, 3]** **["MBA", 500000, 2]** **["BA", 100000, 3]** The stack should contain **["MBA", 500000, 2]** **["MCA", 200000, 3]**
Ans	Univ=[ ] def Push_element(Course): for Rec in Course: if Rec[1]>100000: Univ.append(Rec) def Pop_element( ): while len(Univ)>0: print(Univ.pop( )) else: print("Underflow")
4.	Write the following user defined functions: (i) pushEven(N) - This function accepts a list of integers named N as parameter. It then pushes only even numbers into the stack named EVEN. (ii) popEven(EVEN) - This function pops each integer from the stack EVEN and displays the popped value. When the stack is empty, the message "Stack Empty" is displayed. For example:

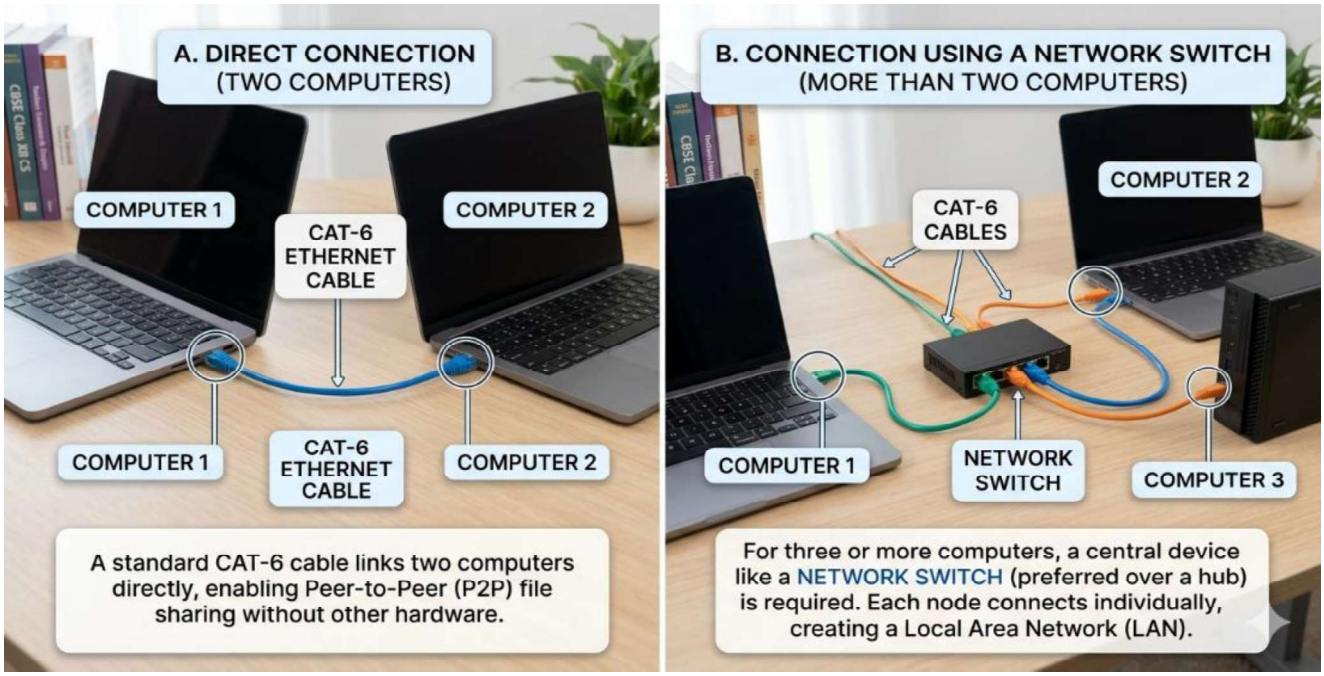
	If the list N contains: [10,5,3,8,15,4] Then the stack, EVEN should store [10,8,4] And the output should be 4 8 10 Stack Empty
Ans	EVEN=[ ] def pushEven(N): for z in N : if z%2==0 : EVEN.append(z) def popEven(EVEN): while EVEN : print(EVEN.pop( ), end=' ') else : print('Stack Empty')
5.	Write the definition of a user defined function PushNV(N) which accepts a list of strings in the parameter N and pushes all strings which have no vowels present in it, into a list named NoVowel. Write a program in Python to input 5 Words and push them one by one into a list named All. The program should that use the function PushNV( ) to create a stack of words in the list NoVowel so that it stores only those words which do not have any vowel present in it, from the list All. Thereafter, pop each word from the list NoVowel and display the popped word. When the stack is empty, display the message "EmptyStack". For example: If the Words accepted and pushed into the list All are ['DRY', 'LIKE', 'RHYTHM', 'WORK', 'GYM'] Then the stack NoVowel should store ['DRY', 'RHYTHM', 'GYM'] And the output should be displayed as GYM RHYTHM DRY EmptyStack
Ans	def PushNV(N): for W in N : flag = True for C in W: if C.upper() in 'AEIOU': flag = False break if flag: NoVowel.append(W) All=[ ] NoVowel=[ ]

	for i in range(5) : All.append(input('Enter a Word: ')) PushNV(All) while NoVowel : print(NoVowel.pop( ), end=' ') else : print('EmptyStack')
6.	Write the definition of a user defined function Push3_5(N) which accepts a list of integers in a parameter N and pushes all those integers which are divisible by 3 or divisible by 5 from the list N into a list named Only3_5. Write a program in Python to input 5 integers into a list named NUM. The program should then use the function Push3_5( ) to create the stack of the list Only3_5. Thereafter pop each integer from the list Only3_5 and display the popped value. When the list is empty, display the message "StackEmpty". For example: If the integers input into the list NUM are : [10,6,14,18,30] Then the stack Only3_5 should store [10,6,18,30] And the output should be displayed as 30 18 6 10 StackEmpty
Ans	NUM=[ ] def Push3_5(N): for i in N: if i%3==0 or i%5==0: Only3_5.append(i) Only3_5=[ ] for i in range(5): NUM.append(int(input('Enter an Integer: '))) Push3_5(NUM) while Only3_5: print (Only3_5.pop( ), end=' ') else: print('StackEmpty')
7.	A list contains following record of customer : [Customer_name, Room Type] Write the following user defined functions to perform given operations on the stack named 'Hotel' : (i) Push_Cust( ) – To Push customers' names of those customers who are staying in 'Delux' Room Type.

	(ii) Pop_Cust( ) – To Pop the names of customers from the stack and display them. Also, display "Underflow" when there are no customers in the stack. For example : If the lists with customer details are as follows : ["Siddarth", "Delux"] ["Rahul", "Standard"] ["Jerry", "Delux"] The stack should contain Jerry Siddharth The output should be: Jerry Siddharth Underflow
Ans	Hotel=[ ] Customer=[["Siddarth", "Delux"],["Rahul", "Standard"], ["Jerry", "Delux"]] def Push_Cust( ): for rec in Customer: if rec[1]=="Delux": Hotel.append(rec[0]) def Pop_Cust( ): while len(Hotel)>0: print(Hotel.pop( )) else: print("Underflow")
8.	Write a function in Python, Push (Vehicle) where, Vehicle is a dictionary containing details of vehicles – {Car_Name: Maker}. The function should push the name of car manufactured by 'TATA' (including all the possible cases like Tata, TaTa, etc.) to the stack. For example: If the dictionary contains the following data : Vehicle={"Santro":"Hyundai","Nexon":"TATA","Safari":"Tata"} The stack should contain Safari Nexon
Ans	stack=[ ] def Push(Vehicle) : for v_name in Vehicle: if Vehicle [v_name]. upper( ) == "TATA" : stack.append(v_name)

UNIT 2
COMPUTER NETWORKS

Concept Map:



EVOLUTION OF NETWORK: The evolution of networking refers to the development of communication technologies that allow computers and devices to share data and resources efficiently.

(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

(IV) 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.

Network: A group of two or more similar things or people interconnected with each other is called network. Examples are social network and mobile network

Computer Network: A Computer network is an interconnection among two or more computers or computing devices. The advantages of computer networks are:

- Resource Sharing

- Collaborative Interaction
- Cost Saving
- Increased storage
- Time Saving

DATA COMMUNICATION TERMINOLOGIES

DATA: Data means information in digital form such as text, audio, video which is stored processed and exchanged between digital devices like computer, mobile phones or laptop. Computers process the raw data into meaningful information. Information is processed data.

COMMUNICATION: The exchange of information between two or more networked or interconnected devices is called communication

COMPONENTS OF DATA COMMUNICATION

a) SENDER: Sender is a device which is capable of sending data over a communication network. In data communication Sender is also called Source.

b) RECEIVER: Receiver is a device which is capable of receiving data over a communication network. In data communication Receiver is also called Destination.

c) MESSAGE: message is the information being exchanged between a sender and a receiver over a communication network.

d) 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.

e) 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.

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. Example IP address: 24.171.248.170

SWITCHING TECHNIQUES

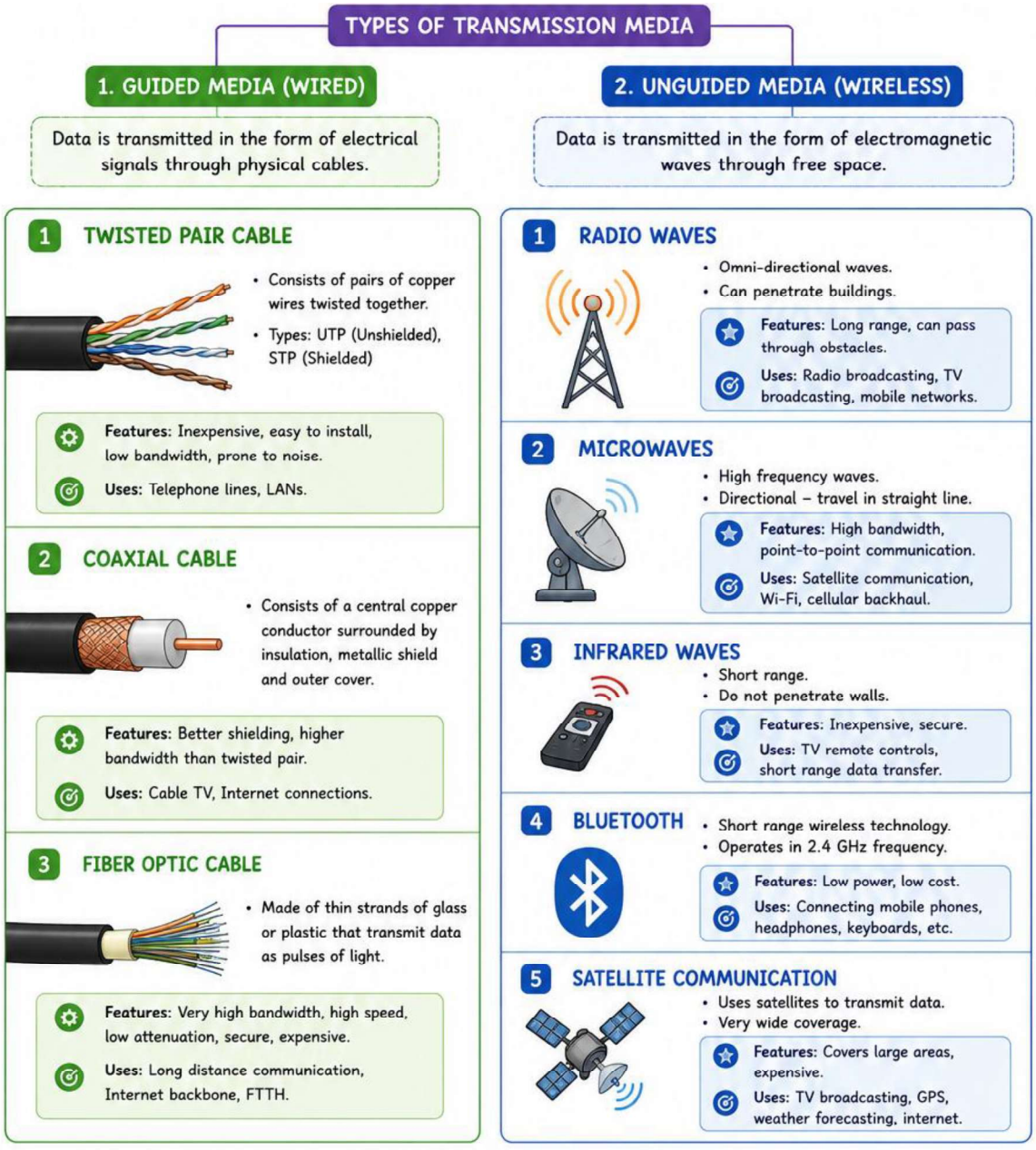
In large networks, there may be more than one paths for transmitting data from sender to receiver. The process of selecting a path of data out of the available paths is called switching. There are two popular switching techniques – circuit switching and packet switching.

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. Example: telephone network

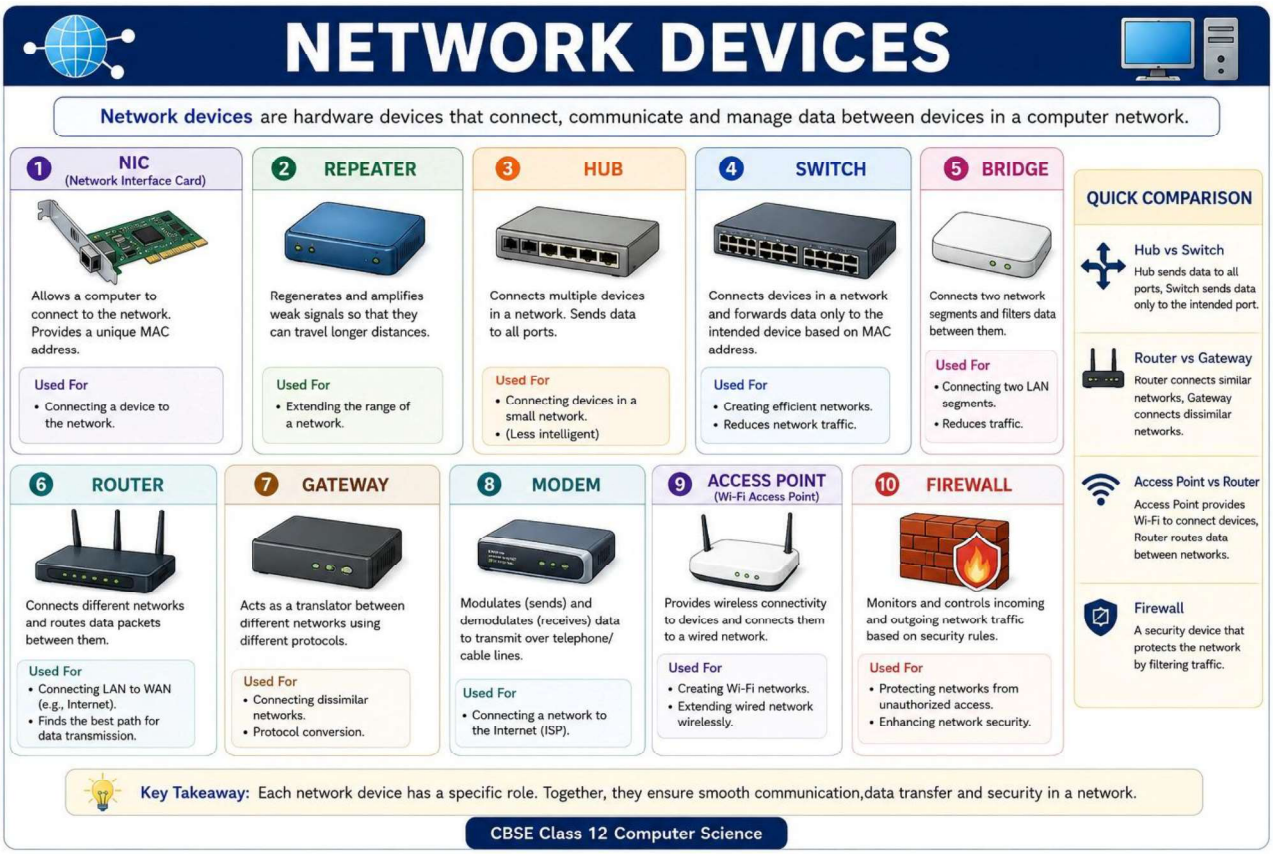
2. Packet Switching : In the packet switching technique, the whole message is split into small packets. Now, these packets are transmitted one by one from sender to the receiver through the intermediary switches in the network. The packets will take shortest path as possible.

Difference Between Radio Waves, Micro Waves, and Infrared Waves

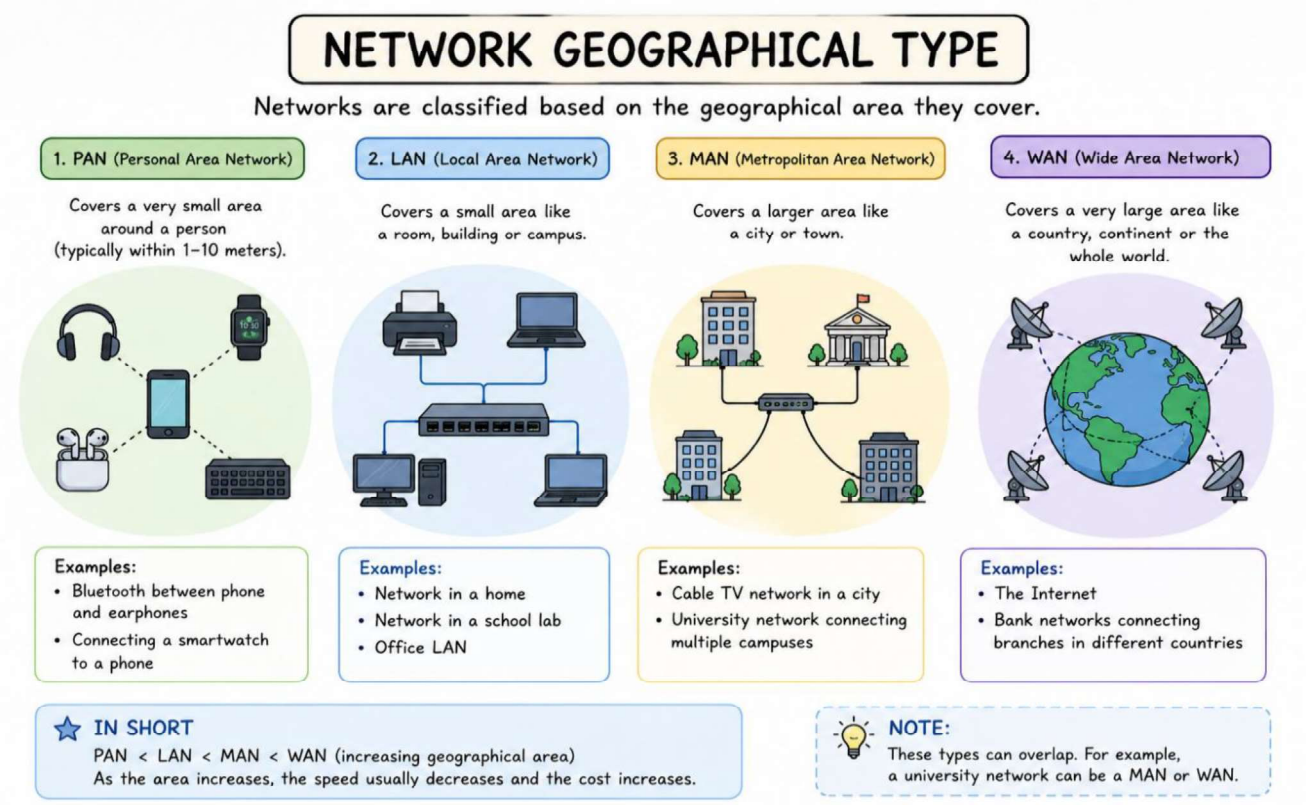
Transmission Media:



NETWORK DEVICES



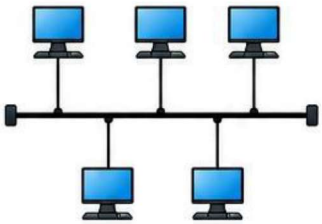
NETWORKING TOPOLOGIES



NETWORK TOPOLOGY

Network Topology is the physical or logical arrangement of nodes (computers, printers, servers, etc.) and links in a network.

1 BUS TOPOLOGY



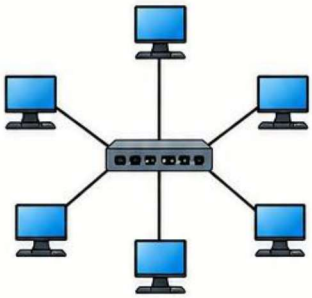
All nodes are connected to a single backbone cable. Data travels in both directions along the bus.

Advantages
- Easy to install
- Cost effective

Disadvantages
- Backbone failure stops the network
- Difficult to troubleshoot

Used In
- Small networks with less number of devices

2 STAR TOPOLOGY



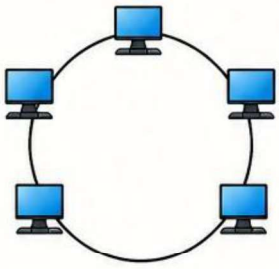
All nodes are connected to a central device (hub/switch). All data passes through the central device.

Advantages
- Easy to install and manage
- Failure of one link does not affect others

Disadvantages
- Central device failure stops the network
- Costlier than bus

Used In
- Offices, schools, homes, LANs

3 RING TOPOLOGY



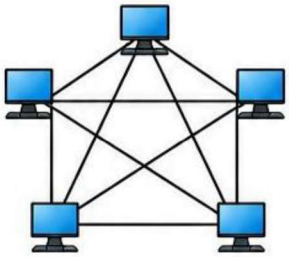
Each node is connected to two other nodes, forming a closed loop. Data travels in one direction around the ring.

Advantages
- No collision of data
- Performance is good under heavy load

Disadvantages
- Failure of one node can break the ring
- Difficult to add or remove nodes

Used In
- Token Ring networks, campus networks

4 MESH TOPOLOGY



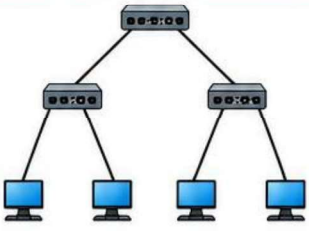
Each node is connected to many or all other nodes. Data can take multiple paths to reach the destination.

Advantages
- Highly reliable
- Failure of one link does not affect communication

Disadvantages
- Expensive
- Complex installation and management

Used In
- WANs, military networks, internet backbone

5 TREE TOPOLOGY



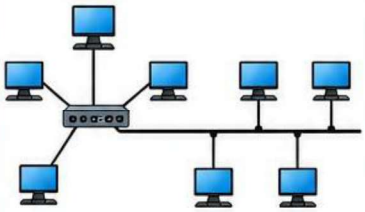
A combination of star and bus topologies arranged in a hierarchical (tree) structure.

Advantages
- Scalable
- Easy to manage larger networks

Disadvantages
- Backbone failure affects the entire network
- Costly

Used In
- Large organizations, ISPs, enterprise networks

6 HYBRID TOPOLOGY



A combination of two or more topologies (e.g., star–bus, star–ring) to suit specific network requirements.

Advantages
- Flexible
- Reliable and scalable

Disadvantages
- Complex design
- Costly

Used In
- Large organizations with diverse requirements

Choosing a topology depends on factors like cost, size, reliability, ease of installation and maintenance, and the purpose of the network.

CBSE Class 12 Computer Science

PROTOCOLS

A **protocol** is a set of rules and conventions that define how data is transmitted, received and interpreted between devices in a network.

CHARACTERISTICS OF A GOOD PROTOCOL

- **Standards** (Well defined rules)
- **Interoperability** (Works between different systems)
- **Reliable** (Error detection and correction)
- **Efficient** (Optimizes speed and resources)
- **Scalable** (Supports growth of networks)

1. APPLICATION LAYER PROTOCOLS

These protocols are used by applications to exchange data over the network.

- **HTTP** (HyperText Transfer Protocol) — Used to transfer web pages on the World Wide Web.
- **HTTPS** (HTTP Secure) — Secure version of HTTP. Uses SSL/TLS for encryption.
- **SMTP** (Simple Mail Transfer Protocol) — Used to send emails from sender's mail server to receiver's mail server.
- **POP3** (Post Office Protocol) — Used by email clients to receive emails from the mail server.
- **FTP** (File Transfer Protocol) — Used to transfer files between client and server.

2. TRANSPORT LAYER PROTOCOLS

These protocols provide end-to-end communication services.

TCP (Transmission Control Protocol)

Reliable, connection-oriented protocol. Ensures data delivery, error control, flow control and sequencing.

UDP (User Datagram Protocol)

Connectionless protocol. Faster but does not guarantee delivery or order of data.

3. NETWORK LAYER PROTOCOLS

These protocols handle logical addressing and routing.

- **IP** (Internet Protocol) — Provides logical addressing (IPv4 / IPv6) and routing of packets across networks.
- **ICMP** (Internet Control Message Protocol) — Used to send error messages and operational information (e.g., ping, traceroute).

4. INTERNET LAYER PROTOCOLS

These protocols are used for internetworking and packet delivery.

- **DNS** (Domain Name System) — Converts domain names (e.g., www.google.com) to IP addresses.
- **DHCP** (Dynamic Host Configuration Protocol) — Automatically assigns IP addresses and other network configuration to devices.

5. ROUTING PROTOCOLS

These protocols are used by routers to determine the best path for data transmission.

RIP (Routing Information Protocol)

Distance vector routing protocol. Uses hop count to find the best path.

OSPF (Open Shortest Path First)

Link state routing protocol. Uses Dijkstra's algorithm to find the shortest path.

6. OTHER IMPORTANT PROTOCOLS

These protocols are used for special purposes in networking.

ARP (Address Resolution Protocol)

Who has 192.168.1.5?

Maps IP addresses to MAC addresses in a local network.

SNMP (Simple Network Management Protocol)

Used to monitor and manage network devices.

Protocols work together in a layered manner (TCP/IP Model) to ensure successful communication across networks.

- Application Layer
- Transport Layer
- Internet Layer
- Network Access Layer

Data flows from top to bottom

Protocols make communication possible, reliable and meaningful!

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).

Servers: It is a remote computer which provides/transfers information to the client (in the form of web pages) or access to particular services.

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 a mark-up language that tells web browsers how to structure the web pages you visit. It has a variety of tags and attributes for defining the layout and structure of the web document. A HTML document has the extension .htm or .html. Hypertext is a text which is linked to another html document via clickable links known as hyperlinks.

XML (eXtensible Mark-up Language): XML is a mark-up language like HTML but it is designed to transport or store data. It does not have predefined tags but allows the programmer to use customized tags. An XML document has the extension .xml.

HTML v/s XML

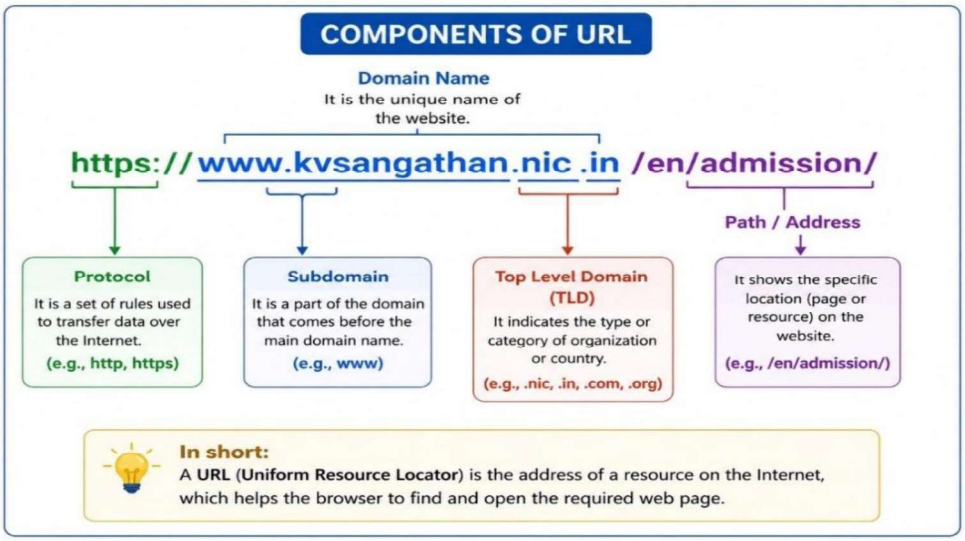
HTML	**XML**
HTML stands for Hyper Text Mark-up Language	XML stands for eXtensible Mark-up Language
HTML is a case insensitive.	XML is case sensitive.
Predefined tags (commands).	User defined tags (commands).
It is used for presentation of the Data.	It is used for transportation of the Data.
Small errors can be ignored.	Errors not allowed.
Closing tags are optional.	Compulsory to use closing tags.

Domain Names: Every device connected to the Internet has a numeric IP address which is very difficult to remember. Each computer server hosting a website or web resource is given a name known as Domain Name corresponding to unique IP addresses. 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

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).

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. The structure of a URL can be represented as follows:



The URL consists of four basic parts, namely, protocol, hostname, folder name and the filename. Each one of these has a specific function.

1) The "protocol" indicates the type of Protocol (http/https/ftp etc.) being used. The protocol 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.

Web page: Web page is an electronic document designed using HTM linked with hyperlinks.

Web Browser: Web browser is software program to navigate the web pages on the internet. E.g., Google Chrome, Mozilla Firefox, Internet Explorer, Safari, Opera etc.

Cookie: A cookie is a small text file that stores information stored on your computer. Cookies often store your settings for a website, such as your preferred language or location, pages visited.

Web Server: A web server is a computer or a group of computers hosting one or more websites. E.g., Apache, IIS etc.

Web Hosting: Web hosting is the process of uploading/saving the web content on a web server to make it available on WWW.

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 Full form of 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 store. 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 full form of MAN ? 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	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
13	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
14	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
15	Which of the following is a Web Browser? a) MS-office b) Notepad c) Firefox d) Word 2007
16	Which of the following is not a web browser? a) Chrome b) Microsoft Edge c) Safari d) Ubuntu
17	URL stands for (a) Uniform Research Limited (b) Uniform Resource Locator (c) Uniform Resource Labs (d) Uniform Research Locator
18	LinkedIn is an example for _____ website. (a) E-learning (b) E-commerce (c) Video conferencing (d) Social networking
19	Which of the following is not a web service? (a) Distance Learning (b) E-mailing (c) Video conferencing (d) Social networking
20	Web browsers are also called as _____ (a) Web Servers (b) Web Clients (c) Web Hosting (d) Web Designing
21	Working of WWW based on _____ architecture. (a) Peer-To-Peer architecture (b) Client-Client architecture (c) Client-Server architecture (d) Server-Server architecture
22	_____ is computer software capable of requesting, receiving & displaying information in the form of webpages. (a) Web Servers (b) Web Browser (c) Web Designers (d) Web Camera
23	_____ is a mark-up language that helps in developing web pages. (a) HTTP (b) HTML (c) XML (d) C++
24	_____ is a language used to transport data over internet. (a) HTTP (b) HTML (c) XML (d) C++
25	._____ is a set of rules for communication between two computers over a network. (a) Modem (b) Protocol (c) Switch (d) IP address

ANSWERS

1. **c**	2. **a**	3. **a**	4. **b**	5. **a**
6. **a**	7. **d**	8. **b**	9. **b**	10. **b**
11. **a**	12. d	13. **c**	14. **b**	15. **c**
16. d	17. b	18. d	19. **a**	20. **b**
21. c	22. **b**	23. **b**	24. **a**	25. **b**

Assertion Reason Questions

	(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):** A Repeater is a device that amplifies the network over geographical distance. **Reasoning (R):** A Hub is a device which is used to connect more than one device in the network.
2	**Assertion (A):** VoIP stands for Voice over Internet Protocol. **Reasoning (R)**: VolP is a technology that allows us to make voice calls using a broadband connection instead of a regular phone line.
3	**Assertion (A):** A protocol defines the set of rules that are applicable for data sharing and communication over a network. **Reasoning (R):** Local Area Network (LAN) is an example of protocol.
4	**Assertion (A):** A Gateway is a network device that connects dissimilar networks. **Reasoning (R):** A Gateway establishes a connection between local network and external network.
5	**Assertion(A):** On a computer network the users work on network nodes only. **Reason(B):** A server cannot act as a network node.

Assertion Reason Answers

1	(b)Both A and R are true but R is not the correct explanation of A.
2	(a) Both A and R are true and R is the correct explanation for A
3	(c) A is True but R is False
4	(a) Both A and R are true and R is the correct explanation for A
5	(c)A is True but R is False

Very Short Answer Type Questions

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

ANSWERS

1	Microwave / Radio wave
2	4 pairs of twisted copper wires
3	Router
4	World Wide Web
5	Interconnected Network
6	Tim Berners-Lee
7	Web browsers
8	Web servers
9	Web server or server
10	hyperlinks
11	HTTP- HyperText Transfer Protocol
12	HyperText Transfer Protocol
13	World Wide Web(WWW) or Web
14	
15	Chat
16	E-learning
17	Internet banking
18	E-reservation
19	Social networking websites

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.

20	HTTP- HyperText Transfer Protocol XML – eXtensible Mark-up Language HTTPS - HyperText Transfer Protocol Secure HTML - HyperText Mark-up Language VoIP-Voice over Internet Protocol
21	Google Chrome, Mozilla Firefox
22	Electronic mail
23	VoIP

Short Answer Type Questions

1	What are Protocols? Name the protocol used to transfer a file from one device to the other.
2	What is meant by an IP Address? Give an example for IP Address
3	Explain how an IP Address become helpful in investigating cyber-crimes
4	Why Protocols are needed in the case of Data Communication?
5	What is the difference between World Wide Web & Internet?
6	What is a protocol, give some examples?
7	What is the difference between E-mail and chat?
8	What are cookies?
9	What is the difference between domain name and IP address?
10	Give one suitable example of each URL and domain name?
11	Differentiate between XML and HTML.
12	Distinguish between website and web browser.
13	Differentiate between the terms Domain Name and URL in context of web services. Also write one example of each to illustrate the difference.
14	Differentiate between communication using Optical Fiber and Ethernet Cable in context of wired medium of communication technologies

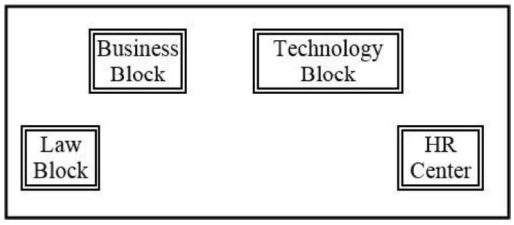
ANSWERS

1	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	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	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.

<table>
<tr><td>4</td><td>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</td></tr>
<tr><td>5</td><td>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.</td></tr>
<tr><td>6</td><td>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)</td></tr>
<tr><td>7</td><td>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 communicate with any email provider.</td></tr>
<tr><td>8</td><td>Cookies are temporary files stored on the user's computer by a web browser, which allow the www server to store persistent information associated with browsing user on user's system</td></tr>
<tr><td>9</td><td>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).</td></tr>
<tr><td>10</td><td>URL: https://kvsangathan.nic.in/hq-gyan-kosh
Domain name: kvsangathan.nic.in</td></tr>
<tr><td>11</td><td>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.</td></tr>
<tr><td>12</td><td>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</td></tr>
<tr><td>13</td><td>
<table>
<tr><td>Domain Name</td><td>URL</td></tr>
<tr><td>A domain name or website name is a human-friendly text form of the IP address.</td><td>URL is a string that represents the complete web address of any web page. It's used to locate a webpage.</td></tr>
<tr><td>It is the part of the URL that is more human friendly.</td><td>It is the string that represents a complete web address that contains the domain name.</td></tr>
<tr><td>Example: kvsangathan.nic.in</td><td>Example: https://kvsangathan.nic.in/contact-us</td></tr>
</table>
</td></tr>
<tr><td>14</td><td>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.</td></tr>
</table>

Long Answer Type Questions

1 MyPace University is setting up its academic blocks at Naya Raipurand 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)



distances between Block are as follows:

Law Block to business Block	40m
Law block to Technology Block	80m
Law Block to HR Block	105m
Business Block to technology Block	30m
Business Block to HR Block	35m
Technology block to HR Block	15m

No. of computers in each Block is:

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 wiredconnectivity

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

2 XYZNova Inc. is planning a new campus in Hyderabad while maintaining its headquarters in Bengaluru. The campus will have four buildings: HR, Finance, IT, and Logistics. As a network expert, you are tasked with proposing the best network solutions for their needs based on the following:

FROM	TO	DISTANCE
HR	Finance	50
HR	IT	175
HR	Logisic	90
Finance	IT	60
Finanace	Logisic	70
IT	Logisic	60

No. of computers in each Center is:

HR	60
Finance	40
IT	90
Logistic	35

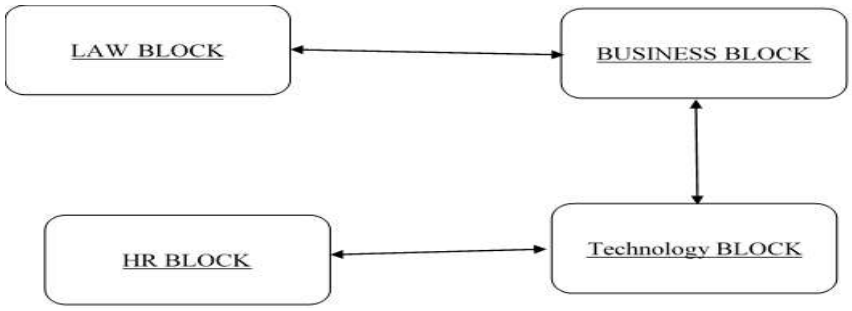
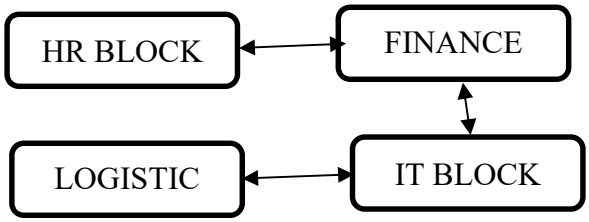
a) Suggest the best location for the server in the Hyderabad campus and explain your reasoning.

b) Suggest the placement of the following devices:
i) Repeater ii) Switch

c) Suggest and draw a cable layout of connections between the buildings inside the campus.

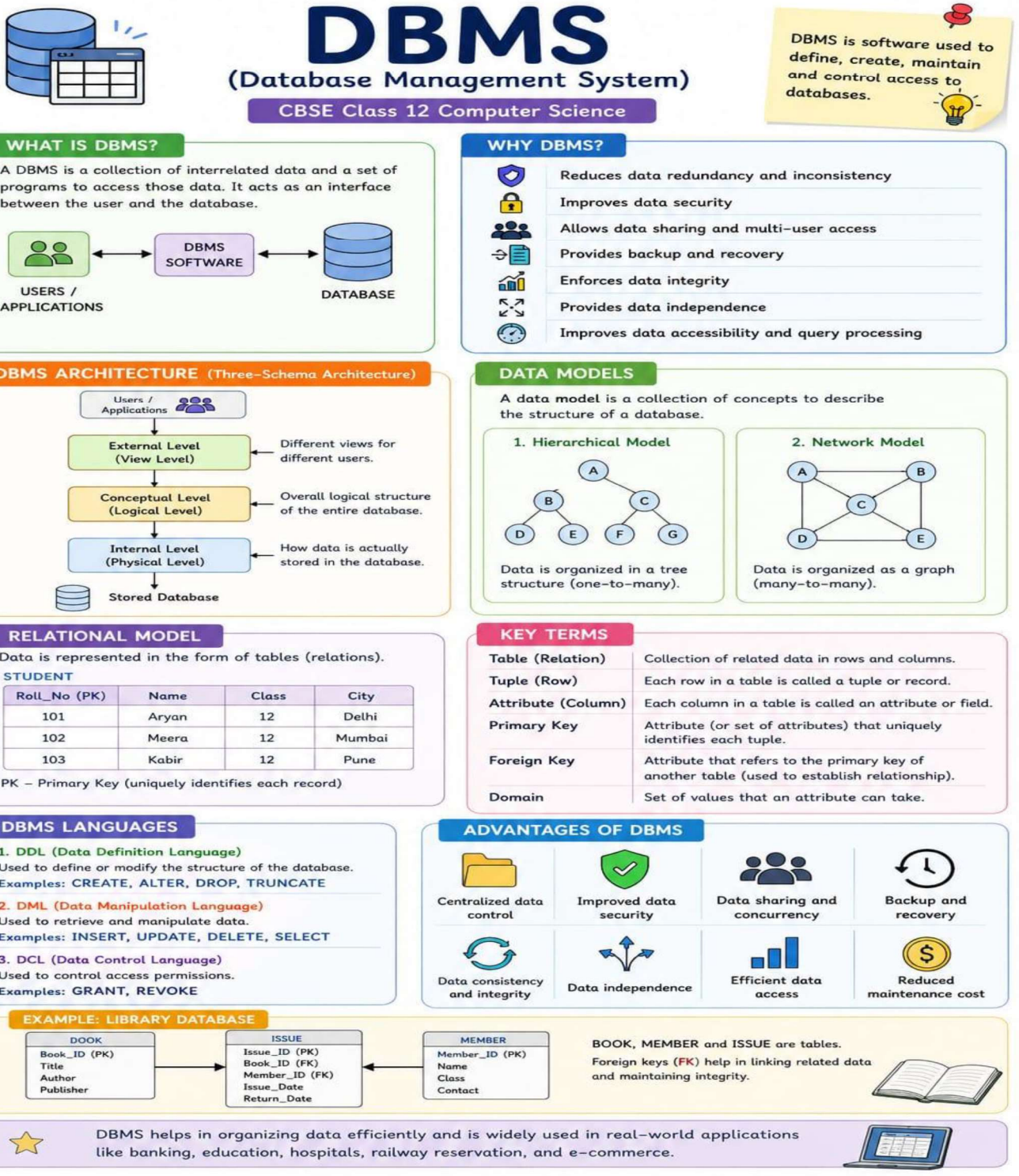
d) The organisation plans to provide a high-speed link with its head office using a wired connection. Which of the cables will be most suitable for this job?

e) What is the use of VoIP?

	OR Which type of network (PAN, LAN, MAN, or WAN) will be formed while connecting the Hyderabad campus to Bengaluru Headquarters?
	ANSWERS:
1.	a) HR center should house the server as it has **maximum** number of computers. b)  c) Switch is to be placed in each and every building d) Repeater is to be placed between Law as HR as distance between them is more than 100 metres. e) A WAN (Wide Area Network) will be formed. This is because the distance between the university and its admission office (over 1250 km) is the most appropriate option for such a vast geographical distance.
2.	a) Block IT should house the server as it has maximum number of computers. b) i) Repeater is to be placed between Block IT to Block HR as distance between them is more than 100 metres. ii) Switch is to be placed in each and every building. c)Draw the star topology cable layout.  d) Optical Fibre e) Voice over Internet Protocol (VoIP) is a technology that allows users to make phone calls and other communications over the Internet instead of a traditional phone line. OR WAN will be formed.

UNIT 3
DATABASE MANAGEMENT

A database is an organized collection of interrelated data that serves many applications. Its is generally a computer record keeping system. In a database we can not only store the data but we can also change the data as per the user requirement. These databases are generally managed by special software called DBMS (Database Management System)



Database Management System (DBMS):

It is software which is responsible for storing, manipulating, maintaining and utilizing the databases.

Database System:

A database along with the DBMS is referred to as database system.
Eg: Oracle, MySQL Server, MySQL, Sybase, SQLite, PostgreSQL, FoxPro, SAP , dBase

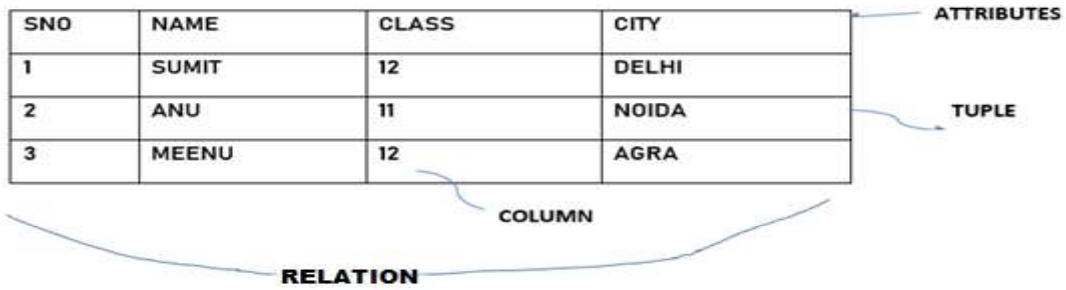
Need for DBMS:

- Databases reduce redundancy i.e. it removes the duplication of data.
- Database controls inconsistency i.e. when two copies of the same data do not agree to each other it is called inconsistency. By controlling redundancy, the inconsistency is also controlled
- Databases allows sharing of data
- Database ensures data security by the process of authentication and does not allow unauthorized access. Database Maintains integrity
- Database is maintained in a standard format which helps to interchange the data between two systems.

RELATIONAL DATABASE MODEL

In Relational Database Model, the data is stored in the form of tables i.e. rows and columns.

- In Relational Database Model a table is referred to as a Relation.
- In Relation Database Model a column is referred to as an attribute
- In relational database model a row is referred to as a tuple.



Relation: A Relation is logically related data organized in the form of tables.

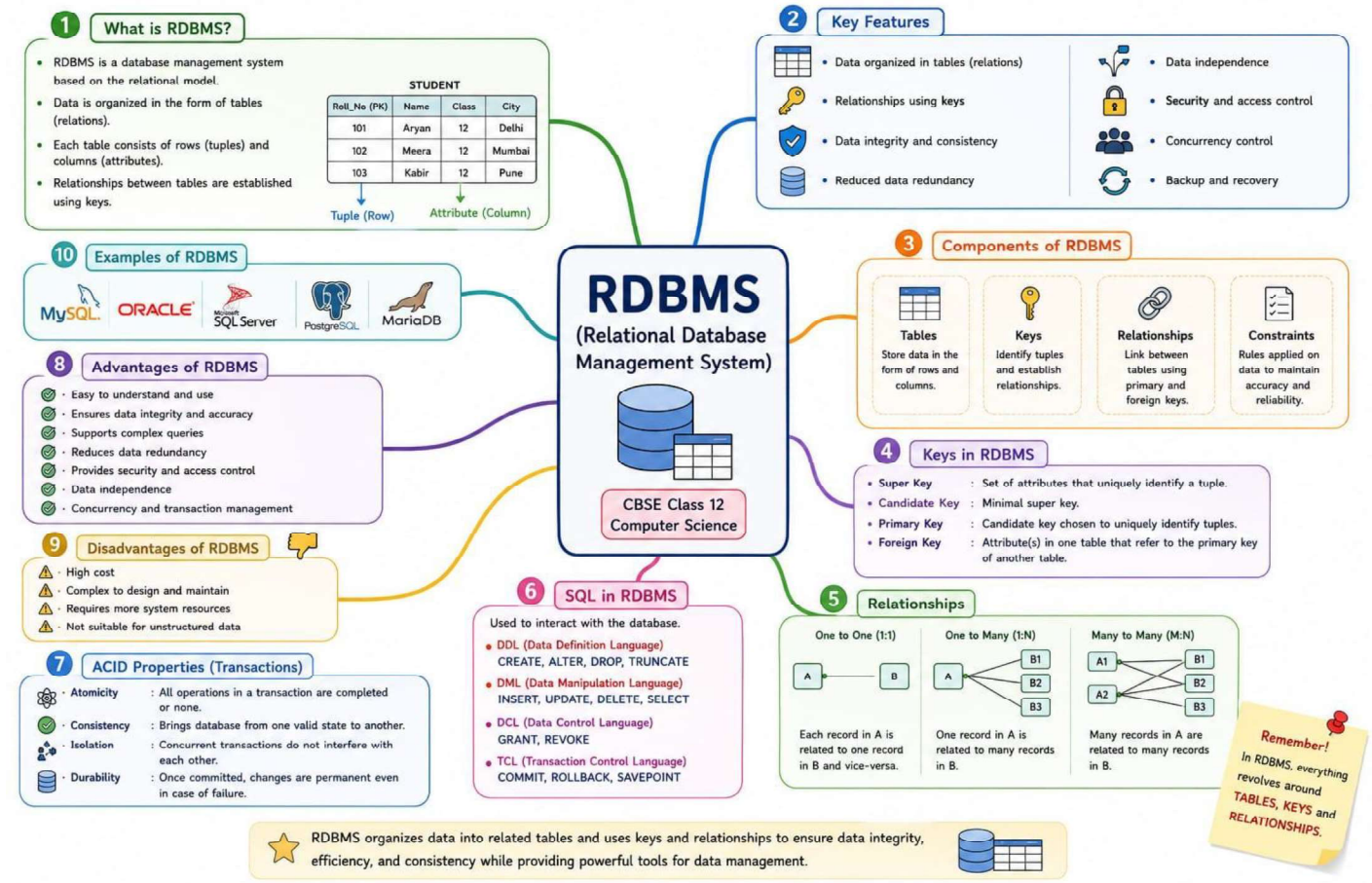
Attribute/ Field: Column of a table is called Attribute or Field.

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.



Keys:

In a relation each record should be unique i.e. no two records can be identical in a database. A key attribute identifies the record and must have unique values.

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.



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 b) 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 is called _____ a)Fields b) Records c) Keys d) Tables
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

ANSWERS:

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)

Very Short Answer Questions

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?

Very Short Answer Questions: 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

RNO	NAME	MARK
100	Tanay	30
101	Kiran	50
102	Manu	30

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)

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.

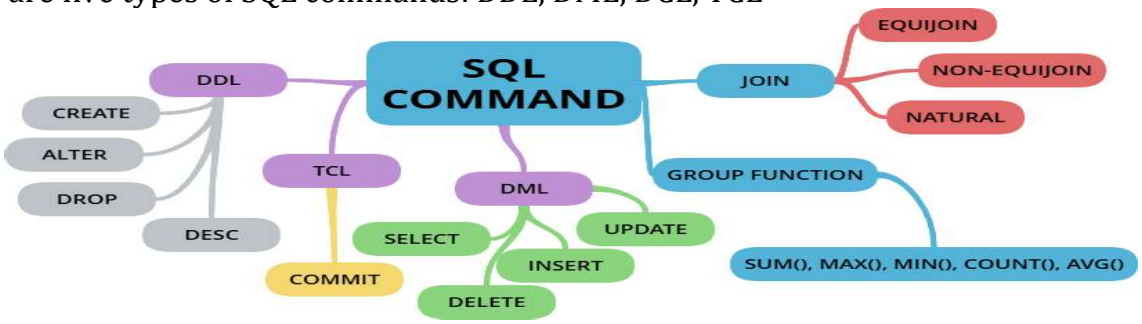
By using SQL commands, one can search for any data in the database and perform other functions like creating tables, adding records, modifying data, removing rows, dropping tables etc.

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.

Types of SQL Commands

- There are five types of SQL commands: DDL, DML, DCL, TCL



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.

DML (Data Manipulation Language):

The SQL commands that deal 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.

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

CONSTRAINTS

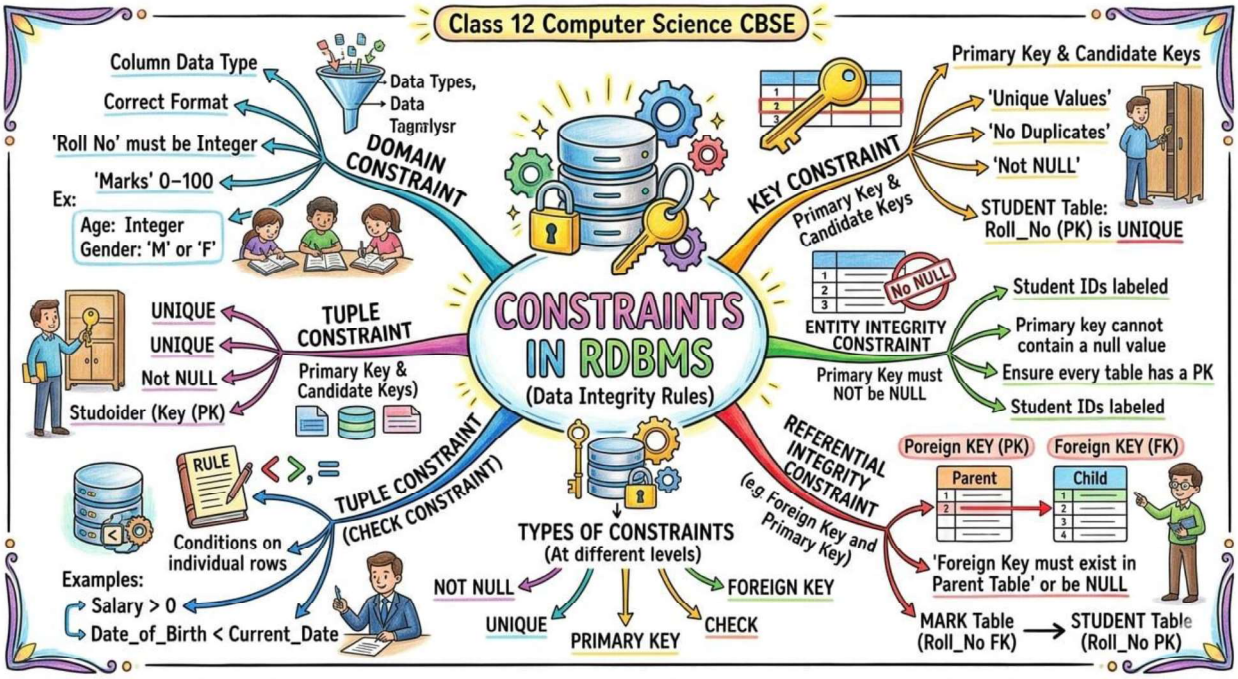
A Constraint is a condition or check applicable on a field or set of fields.

Types of Constraints:

- **Unique Constraint** :-This ensures that no rows have the same value in the specified column(s)
  Example
  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**
  Example:
  CREATE TABLE EMP(
  ecode integer Not null unique,
  ename char(20),
  sex char(1),
  grade char(2));



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

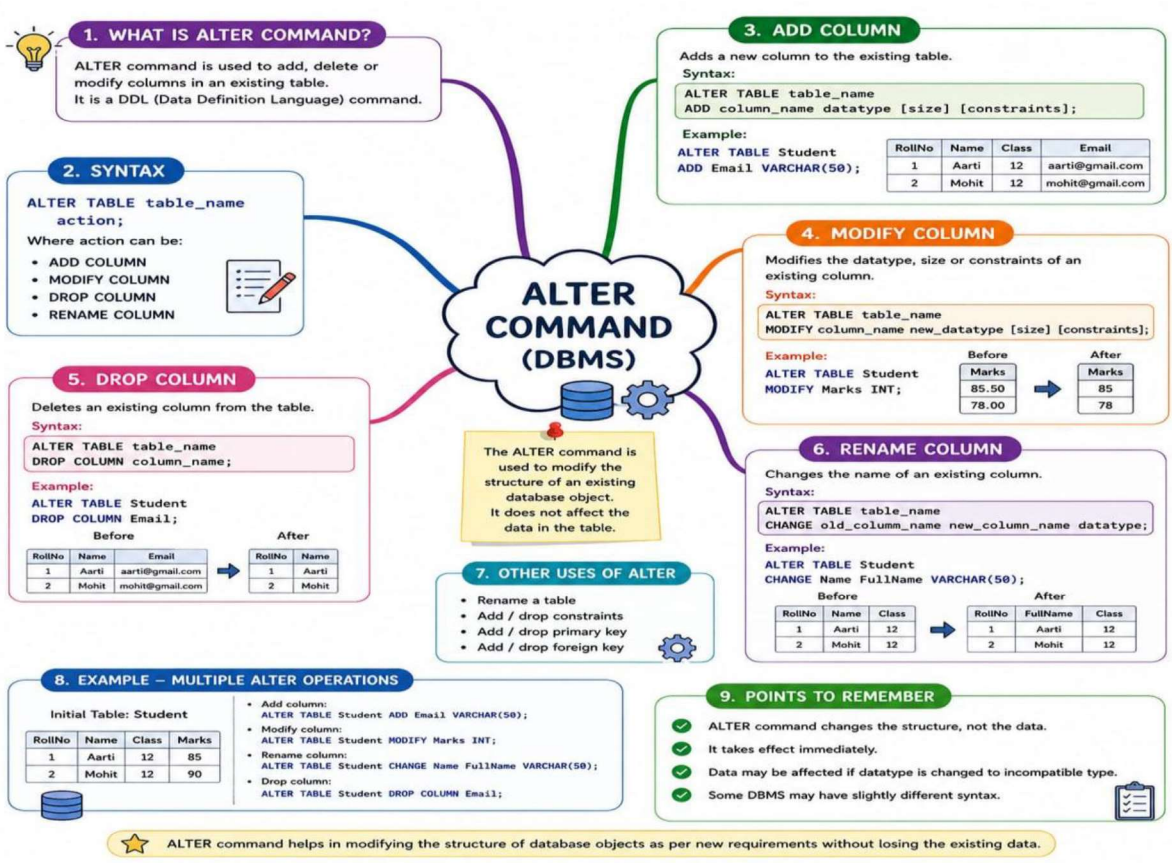
Use the **DESCRIBE** command to show the structure of the table, such as column names, constraints on column names, etc. The **DESC** command is a short form of the DESCRIBE command. Both DESCRIBE and DESC commands are equivalent.

Syntax The following are the syntax to display the table structure:

```
mysql>  DESCRIBE | DESC    table_name;
```

ALTER TABLE

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.



INSERT:

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*, ...);

DELETE:

The DELETE statement is used to delete existing records in a table.

DELETE Syntax:

DELETE FROM table_name;

Note : Delete or remove all records(tuples) from the table but Structure will be remains.

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;
```

UPDATE

The UPDATE statement is used to modify the existing records in a table.

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;

SELECT

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;
```

WHERE Clause:

The WHERE clause is used to filter records.

It is used to extract only those records that fulfill a specified condition.

Syntax: SELECT * FROM *table_name* WHERE *condition*;
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

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*;

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*, ...);

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;*

LIKE Operator

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 *column* 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"

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;*

DISTINCT

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.

Syntax:
SELECT DISTINCT (*column1)* 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;

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. 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;

ORDER BY

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;

AGGREGATE (GROUP) FUNCTIONS

- 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.

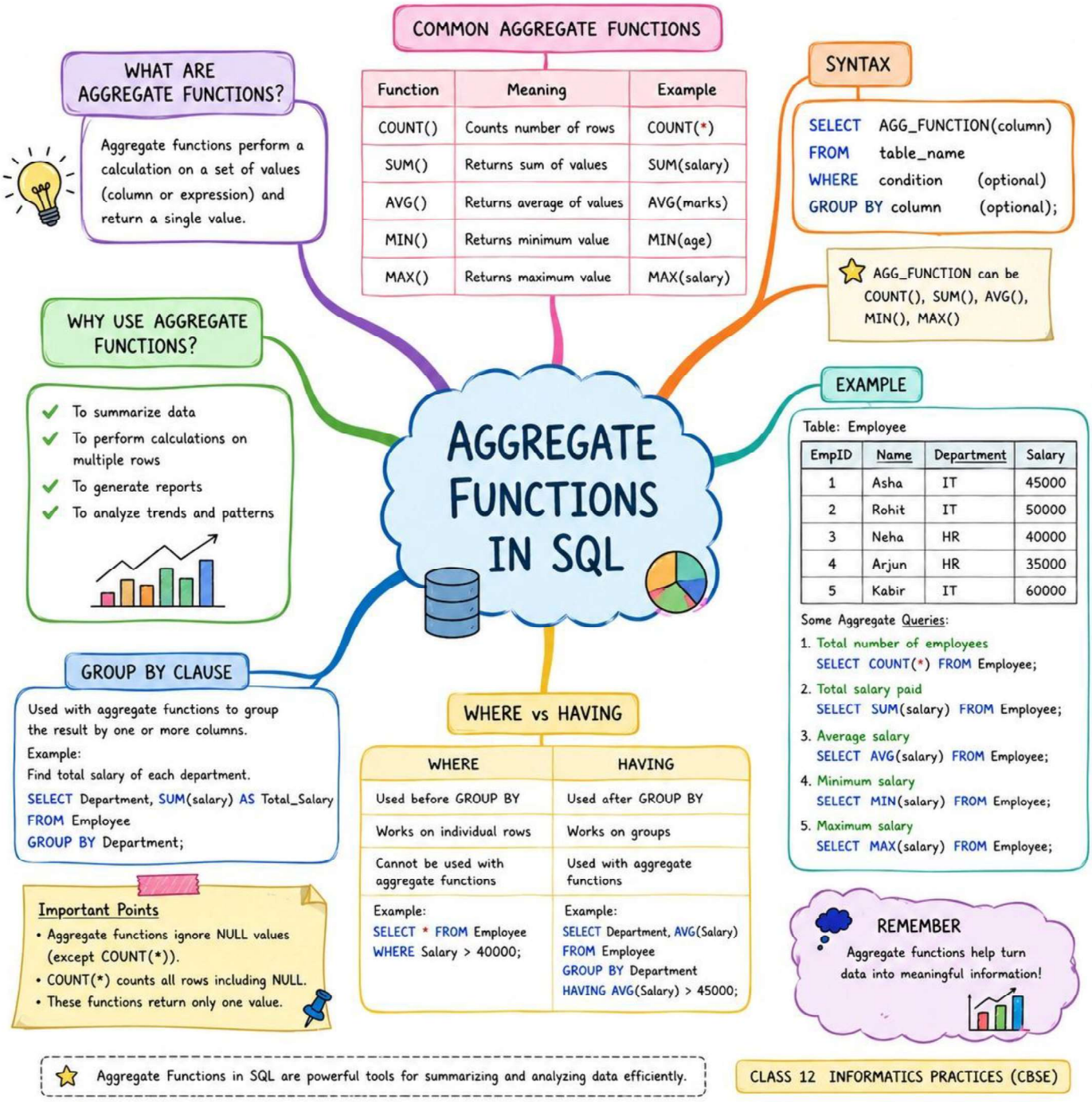


Table: employee

```
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)
```

sum( ), average( ), minimum( ), maximum( ) function

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 g
+--------+-------------+
| 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 joins:

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)
```

alias Name of the table

Note-

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.

MULTIPLE CHOICE QUESTIONS

1	Which of the following SQL commands is used to use/select a particular database? a. use b. select c. view d. project
2	Which SQL command is used to define and maintain physical structure or schema of table in database like creating, altering and deleting database object such as table and constraints? a. DDL b. DML c. DCL d. TCL
3	Which commands is used to show all table in current using database? a. display tables; b. show tables; c. view tables; d. select all tables;
4	Identify the MySQL Commands that belongs to DML category : a. ALTER b. DROP c. DELETE d. CREATE
5	Which command is used in where clause to search NULL values in a particular column? a. IS NULL b. IN NULL c. NOT NULL d. IS NOT NULL
6	Wild card operator (%,_) are used with? a. count b. max c. like d. min
7	Prapti is presently working in the database SUBJECT. She wants to change and go to the database RECORD. Choose the correct statement in MySQL to go to the database RECORD. a. GO TO DATABASE RECORD; b. USE DATABASE RECORD;

	c. CHANGE DATABASE RECORD; d. USE RECORD;
8	Which SQL clause is used in database table to eliminate duplicate rows from the query result? a. group by b. distinct c. describe d. duplicate
9	Which SQL function is used to count the entire number of row in database table? a. count b. count(*) c. max d. min
10	Which SQL function is used to determine the no. of row or non-null values? a. min b. max c. count d. sum

ANSWERS

1	a	2	a	3	b	4	c	5	a
6	c	7	b	8	b	9	b	10	c

ASSERTION AND REASONING QUESTIONS

Directions: In the following questions, A statement of Assertion (A) is followed by a statement of Reason (R). Mark the correct choice as:

(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

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.
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.
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.
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.
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.
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
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
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.

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.
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

ANSWERS

1	b	2	b	3	a	4	b	5	a
6	b	7	a	8	b	9	c	10	a

SHORT ANSWER TYPE QUESTIONS

1	Deepika wants to remove all rows from the table BANK. But she needs to keep the structure of the table. Which command is used to implement the same?
2	While creating table 'customer', Rahul forgot to add column 'price'. Which command is used to add new column in the table. Write the command to implement the same.
3	Mitali is a database programmer, She has to write the query from EMPLOYEE table to search for the employee who are not getting any commission, for this she has written the query as: SELECT * FROM EMPLOYEE WHERE commission=null; But the query is not producing the correct output, help her and correct the query so that she gets the desired output.
4	Which clause is used to eliminate the duplicate rows from output?
5	Which command is used to see information like name of columns, data type, size.
6	Differentiate between order by and group by clause in SQL with appropriate example.
7	Categorize the following commands as DDL or DML: INSERT, UPDATE, ALTER, DROP.
8	Muneer has created a database "school" and table "student". Now he wants to view all the databases present in his laptop. Help him to write SQL command for that , also to view the structure of the table he created
9	Ms. Minakshi 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
10	Meera working as database developer in Syntel Pvt Ltd Company Agra. She is designing as SQL table names A & B .If a MySQL table A has 5 columns and 6 rows and another table B has 3 columns and 4 rows, then what will be the degree and cardinality of the cartesian product of A and B?
11	Sunil decides to delete phoneno column from a table student . Write the SQL command to remove the column in the student table. Also mention the type of SQL command.
12	Write SQL command to remove the Primary Key constraint from a table, named M_ID is the primary key of the table MOBILE.
13	Write SQL command to make the column M_ID the Primary Key of an already existing table, named MOBILE.
14	What constraint should be applied on a table column so that duplicate values are not allowed in that column, but NULL is allowed.
15	What constraint should be applied on a table column so that NULL is not allowed in that column, but duplicate values are allowed.
16	Which command is used to open a database?
17	Mr.Murukesh is working as Database Administrator .He designs the following database table to store the information on candidates who register for examination.

<table>
<tr><td></td><td>Table:Candidates
<table><tr><th>Field Name</th><th>Data Type</th><th>Remarks</th></tr><tr><td>CNO</td><td>Char (8)</td><td>Candidate Number</td></tr><tr><td>CNAME</td><td>varchar (30)</td><td>Name of candidate</td></tr><tr><td>DOB</td><td>Date</td><td>Date of Birth of Candidate</td></tr></table>Write SQL command to create the above table. Also mention the advantage of using varchar data type over char data type for storing names.</td></tr>
<tr><td>18</td><td>A table, medicalstore has been created in a database with the following fields:
MedicineNo ,MedicineName ,MedCode ,Quantity. Give the SQL command to add a new field, PRICE (of type Integer) to the medicalstore table.</td></tr>
<tr><td>19</td><td>An educational institution EducationPoint is considering to maintain their inventory using SQL to store the data. As a database administer, Ajay has decided that :
Name of the database – EDUPOINT, Name of the table – STUDENT.
The attributes of student are as follows:
• STUDENTID – numeric, primary key
• STUDNAME – character of size 30
• SCODE – character of size 10
• MARKS – numeric
Help him to complete the task by suggesting appropriate SQL commands.</td></tr>
<tr><td>20</td><td>Which operator in SQL is used to compare a value to a specified list of values.</td></tr>
</table>

ANSWERS:

1	delete from BANK;
2	ALTER TABLE customer add price int;
3	SELECT * FROM EMPLOYEE WHERE commission is null;
4	distinct
5	DESCRIBE or DESC
6	ORDER BY is used to sort the result set based on one or more columns, either in ascending (ASC) or descending (DESC) order whereas group by is used to group rows that have the same values in specified columns into summary rows, often used with aggregate functions. Eg: SELECT name, salary FROM employees ORDER BY salary DESC; This will list all employees and their salaries, sorted by salary from highest to lowest. SELECT department, AVG(salary) AS average_salary FROM employees GROUP BY department; This will group employees by department and show the average salary for each department.
7	DDL –ALTER , DROP DML-INSERT , UPDATE
8	SHOW DATABASES; SHOW TABLES;
9	ALTER TABLE Staff ADD StaffId INT; ALTER TABLE Staff ADD CONSTRAINT pk_staff PRIMARY KEY (StaffId); INSERT INTO Staff (StaffId, Sname, Department, Salary) VALUES (111, 'Shalu', 'Marketing', 45000);
10	Degree = 8 Cardinality = 24

11	ALTER TABLE student DROP phoneno It's a DDL command.
12	ALTER TABLE mobile DROP PRIMARY KEY;
13	ALTER TABLE mobile ADD PRIMARY KEY (M_ID);
14	UNIQUE constraint makes sure that duplicate values are not allowed in that column, but NULL will be allowed.
15	NOT NULL
16	USE
17	CREATE TABLE Candidates (CNO CHAR(8), CNAME VARCHAR(30), DOB DATE); VARCHAR only uses as much space as needed for the actual string, while CHAR always uses the full defined length, padding with spaces if necessary.
18	ALTER TABLE medicalstore ADD price INTEGER;
19	CREATE DATABASE EDUPOINT; USE EDUPOINT; CREATE TABLE STUDENT (STUDENTID INT PRIMARY KEY, STUDNAME VARCHAR(30), SCODE CHAR(10), MARKS INT);
20	IN

SHORT ANSWER QUESTIONS II

1 Meenu has been entrusted with the management of NSE Database. She needs to access some information from STOCK and TRADERS tables for a survey analysis.
Help him extract the following information by writing the desired SQL queries as mentioned below.

Table Name: TRADERS

TCODE	TNAME	CITY
T01	RELIANCE DIGITAL	MUMBAI
T02	TATA DIGITAL	BHUBANESWAR
T03	BIRLA DIGITAL	NEW DELHI

Table name: STOCK

SCODE	SNAME	QTY	PRICE	BRAND	TCODE
1001	COMPUTER	90	45000	DELL	T01
1006	LCD PROJECTOR	40	42000	NEC	T02
1004	IPAD	100	55000	APPLE	T01
1003	DIGITAL CAMERA	160	15000	SAMSUNG	T02
1005	LAPTOP	600	35000	HP	T03

Write SQL queries for the following:
(i) Display the SNAME, QTY, PRICE, TCODE, and TNAME of all the stocks in the STOCK and TRADERS tables.
(ii) Display the details of all the stocks with a price >= 35000 and <=50000 (inclusive).
(iii) Display the SCODE, SNAME, QTY*PRICE as the "TOTAL PRICE" of BRAND "NEC" or "HP" in ascending order of QTY*PRICE.
iv) Display the number of stock items in each TCODE.
OR
To display the Cartesian Product of these two tables

2 Consider the following tables employees,empsalary.
Table : Employees

Empid	Firstname	Lastname	Address	City
010	Ravi	Kumar	Raj nagar	GZB
105	Harry	Waltor	Gandhi nagar	GZB
152	Sam	Tones	33 Elm St.	Paris
215	Sarah	Ackerman	440 U.S. 110	Upton
244	Manila	Sengupta	24 Friends street	New Delhi
300	Robert	Samuel	9 Fifth Cross	Washington
335	Ritu	Tondon	Shastri Nagar	GZB
400	Rachel	Lee	121 Harrison St.	New York
441	Peter	Thompson	11 Red Road	Paris

Table: EmpSalary

Empid	Salary	Benefits	Designation
010	75000	15000	Manager
105	65000	15000	Manager
152	80000	25000	Director
215	75000	12500	Manager
244	50000	12000	Clerk
300	45000	10000	Clerk
335	40000	10000	Clerk
400	32000	7500	Salesman
441	28000	7500	salesman

Write the SQL commands for the following:

a) To show first name, last name, address and city of all employees who lives in Paris.

b) To display the details of Employees table in descending order of First name.

c) To display the first name, last name and salary of all employees from the tables Employee and EmpSalary, who are working as Manager.

Give the Output of following SQL commands:

d) Select designation, sum(salary) from empsalary group by designation having count(*) > 2;

OR

d) Select sum(benefits) from empsalary where designation ='clerk';

3 Consider the following table **DOCTOR** given below and write the output of the SQL Queries that follows :

D_ID	D_NAME	D_DEPT	GENDER	EXPERIENCE
101	JOSEPH	ENT	MALE	10
104	GUPTA	MEDICINE	MALE	12
106	SUMAN	ORTHO	FEMALE	7
111	HANEEF	ENT	MALE	12
123	DEEPTI	CARDIOLOGY	FEMALE	6
132	VEENA	SKIN	FEMALE	12

a)SELECT D_NAME FROM DOCTOR WHERE GENDER=MALE AND EXPERIENCE=12 ;

b)SELECT DISTINCT(D_DEPT) FROM DOCTOR ;

c)SELECT D_NAME , EXPERIENCE FROM DOCTOR ORDER BY EXPERIENCE ;

4 **Consider the table PRODUCT and CLIENT given below:**

PR_ID	PR_NAME	MANUFACTURER	PRICE	QTY
BS101	BATH SOAP	PEARSE	45.00	25
SP210	SHAMPOO	SUN SILK	320.00	10
SP235	SHAMPOO	DOVE	455.00	15
BS120	BATH SOAP	SANTOOR	36.00	10
TB310	TOOTH BRUSH	COLGATE	48.00	15
FW422	FACE WASH	DETOL	66.00	10
BS145	BATH SOAP	DOVE	38.00	20

C_ID	C_NAME	CITY	PR_ID
01	DREAM MART	COCHIN	BS101
02	SHOPRIX	DELHI	TB310
03	BIG BAZAR	DELHI	SP235
04	LIVE LIFE	CHENNAI	FW422

Write SQL Queries for the following:

i) Display the details of those clients whose city is DELHI

ii) Increase the Price of all Bath soap by 10

iii) Display the details of Products having the highest price

iv) Display the product name, price, client name and city with their corresponding matching product Id.

5 **Write SQL Commands for (a) to (e) and write the outputs for (f) on the basis of table: FURNITURE**

NO	ITEM NAME	TYPE	DATEOFSTOCK	PRICE	DISCOUNT
1	White Lotus	Double Bed	2002-02-23	3000	25
2	Pink feathers	Baby Cot	2002-01-29	7000	20
3	Dolphin	Baby Cot	2002-02-19	9500	20
4	Decent	Office Table	2002-02-01	25000	30
5	Comfort zone	Double Bed	2002-02-12	25000	30
6	Donald	Baby cot	2002-02-24	6500	15
7	Royal Finish	Office Table	2002-02-20	18000	30
8	Royal tiger	Sofa	2002-02-22	31000	30
9	Econo sitting	Sofa	2001-12-13	9500	25
10	Eating Paradise	Dinning Table	2002-12-19	11500	25

1. To show all the information about the Baby cots from the furniture table.

2. To list the itemname which are priced at more than 15000 from the furniture table.

3. To list itemname and type of those items, in which dateofstock is after 2002-02-01 from the furniture table in descending order of itemname

4. To display itemname and dateofstock of those items, in which the discount percentage is more than 25 from the furniture table.

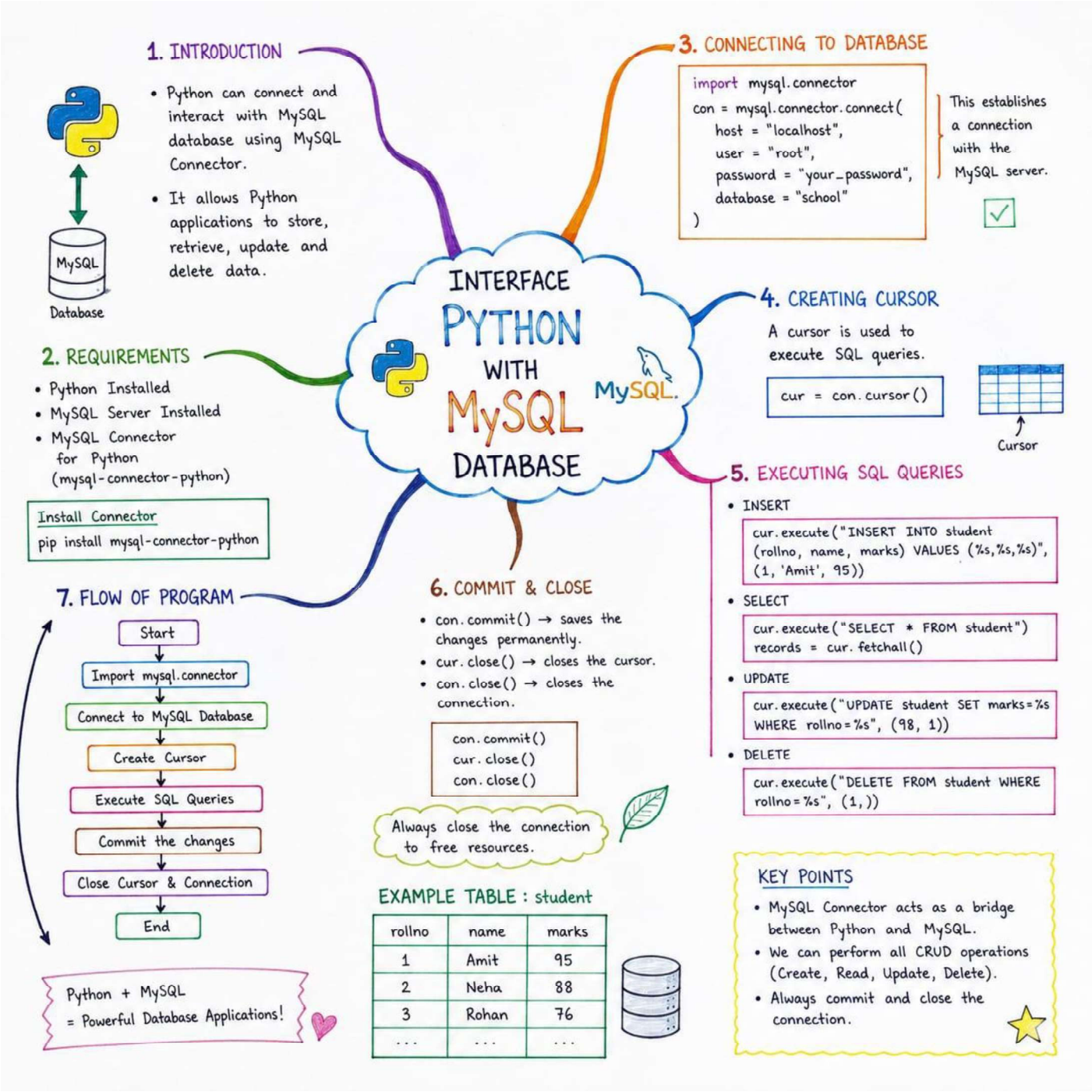
ANSWERS:

1

```
i)select sname, qty, price , stock.tcode, tname
from stock, traders where  stock.tcode = traders.tcode;
ii) select * from stock  where  price between 35000 and 50000;
iii) select scode, sname, qty * price as "total price"  from stock
     where brand in ('nec', 'hp')  order by qty * price asc;
```

	iv) select tcode, count(*) as stock_count from stock group by tcode;
2	i) select firstname, lastname, address, city from employees where city = 'paris'; ii) select * from employees order by firstname desc; iii) select e. firstname, e. lastname, s. salary from employees e join empsalary s on e.empid = s.empid where s.designation = 'manager'; iv) manager 215000 clerk 135000 or 32000
3	A) **D_NAME** GUPTA HANEEF B) **D_DEPT** ENT MEDICINE ORTHO CARDIOLOGY SKIN C) **D_NAME** **EXPERIENCE** DEEPTI 6 SUMAN 7 JOSEPH 10 GUPTA 12 HANEEF 12 VEENA 12
4	i) select * from client where city = 'delhi'; ii) update product set price = price + 10 where pr_name = 'bath soap'; iii) select * from product where price = (select max(price) from product); iv) select p.pr_name, p.price, c.c_name, c.city from product p join client c on p.pr_id = c.pr_id;
5	i) select * from furniture where lower(type) = 'baby cot'; ii) select item_name from furniture where price > 15000; iii) select item_name, type from furniture where dateofstock > '2002-02-01' order by item_name desc; iv) select item_name, dateofstock from furniture where discount > 25;

INTERFACE PYTHON WITH MySQL DATABASE



STATIC QUERIES

Queries which are formed without passing any python object/variable/data.

Now we can execute any MySQL query through Python. Below are few examples static queries.

a) TO CREATE A TABLE IN MYSQL USING PYTHON INTERFACE

```
import mysql.connector as m
con =m.connect(host="localhost",user="root", passwd="system", database="student")
cur=con.cursor( )
cur.execute("CREATE TABLE FEES (ROLLNO INT, NAME VARCHAR(20), AMOUNT INT);")
```

b) TO SHOW THE TABLES IN MYSQL USING PYTHON INTERFACE

```
import mysql.connector as m
con =m.connect(host="localhost",user="root", passwd="system", database="student")
cur=con.cursor( )
cur.execute("SHOW TABLES")
for data in cur:
    print(data)
```

c) TO DESCRIBE TABLE STRUCTURE USING PYTHON INTERFACE

```
import mysql.connector as m
con =m.connect(host="localhost",user="root", passwd="system", database="student")
cur=con.cursor( )
mycursor.execute("DESC STUDENT")
for data in cur:
print(data)
```

d) TO EXECUTE SELECT QUERY USING A PYTHON INTERFACE

```
import mysql.connector as m
con =m.connect(host="localhost",user="root", passwd="system", database="student")
cur=con.cursor( )
cur.execute("select * from student")
r=cur.fetchone( )
while r is not None:
    print(r)
    r=cur.fetchone( )
```

e) TO EXECUTE SELECT QUERY WITH WHERE CLAUSE USING A PYTHON INTERFACE

```
import mysql.connector as m
con =m.connect(host="localhost",user="root", passwd="system", database="student")
cur=con.cursor( )
cur.execute("select * from student where marks>90")
r=cur.fetchall( )
count=cur.rowcount
print("total no of rows:",count)
for row in r:
    print(row)
```

f) TO UPDATE A DATA IN A TABLE USING PYTHON INTERFACE

```
import mysql.connector as m
con =m.connect(host="localhost",user="root", passwd="system", database="student")
cur=con.cursor( )
cur.execute("UPDATE STUDENT SET MARKS=100 WHERE MARKS=40")
con.commit( )
print(cur.rowcount,"RECORD UPDATED")
```

g) TO DELETE A RECORD FROM THE TABLE USING PYTHON INTERFACE

```
import mysql.connector as m
con =m.connect(host="localhost",user="root", passwd="system", database="student")
cur=con.cursor( )
cur.execute("DELETE FROM STUDENT WHERE MARKS<50")
con.commit( )
print(cur.rowcount,"RECORD DELETED")
```

h) TO DROP AN ENTIRE TABLE FROM MYSQL DATABASE USING PYTHON INTERFACE

```
import mysql.connector as m
con =m.connect(host="localhost",user="root", passwd="system", database="student")
cur=con.cursor( )
cur.execute("DROP TABLE STUDENT")
```

i) TO ADD A COLUMN IN THE EXISTING TABLE USING PYTHON INTERFACE

```
import mysql.connector as m
con =m.connect(host="localhost",user="root", passwd="system", database="student")
cur=con.cursor( )
cur.execute("ALTER TABLE STUDENT ADD AGE NT”)
con.commit( )
```

j) TO DROP A COLUMN FROM THE TABLE USING PYTHON INTERFACE

```
import mysql.connector as m
con =m.connect(host="localhost",user="root", passwd="system", database="student")
cur=con.cursor( )
cur.execute("ALTER TABLE DROP AGE ”)
con.commit( )
```

k) TO ALTER THE DATATYPE OF A COLUMN IN A TABLE USING PYTHON INTERFACE

```
import mysql.connector as m
con =m.connect(host="localhost",user="root", passwd="system", database="student")
cur=con.cursor( )
cur.execute("ALTER TABLE STUDENT MODIFY GRADE CHAR(3)")
```

PARAMETERIZED QUERIES

Parameterize the query to add python variables/object into the string query to access values as per the user's input.

Example: to display record of a particular rollno.

There are two ways to use parameterized queries:

a) with { }.format pattern
b) with fstring & { } braces

Using {}.format pattern

(a)

```
rn = int(input('Enter Roll no. ')) # SUPPOSE USER ENTERS 10 AS ROLL NO
query  = "select * from student where rollno  = { }".format(rn)
cur.execute(query)
```

Note: "Here the format function will assign the value 10 in place of { } braces inside the string query. python will convert it to-> query = "select * from student where rollno = 10"

(b)

Suppose we want to display the data based on the column name & value given by user

```
col_name = eval(input('Enter Column Name ')) # User entered Rollno
col_value = eval(input('Enter Column Value ')) # User entered 12
query  = "select * from student where { }  = { }".format(col_name , col_value)
# python will convert it to ->>> query  = "select * from student where Rollno = 12 "
cur.execute(query)
```

(c)

Suppose we want to display the data from a specific class & section given by user

```
cls = eval(input('Enter Class '))          # User entered 12
sec = eval(input('Enter Section '))        # User entered A
query  =  "select * from student
          where class = { } and section  = '{ }' ".format(cls , sec)
# python will convert it to ->>> query  = "select * from student
                                  where class = 12 and section  = 'A' "
cur.execute(query)
```

Multiple Choice Questions

1	Which of the following command is used to connect Python with MySQL? a) mysql.connect( ) b) MySQL.connector.connect( ) c) mysql.connector.connect( ) d) connect.mysql( )
2	Which module needs to be imported to use MySQL in Python? a) mysql b) sqlite3 c) MySQL d) mysql.connector
3	What does the cursor( ) method do in MySQL Python connector? a) Executes the SQL commands directly b) Establishes a connection to the database c) Creates a cursor object to interact with the database d) Creates a database
4	Which method is used to execute SQL queries in Python? a) executeQuery( ) b) run( ) c) execute( ) d) query( )
5	Which method is used to retrieve all rows from the executed query? a) fetchone( ) b) fetchall( ) c) getrows( ) d) read( )
6	After performing an INSERT operation, which method is used to save the changes in the database? a) commit( ) b) save( ) c) flush( ) d) update( )
7	Which of the following is not a valid parameter in mysql.connector.connect( )? a) host b) username c) user d) password
8	What is the purpose of the close( ) method? a) To close the database file b) To shut down the MySQL server c) To close the cursor or connection d) To delete the database
9	Which of the following is correct for selecting a database after connection? a) cursor.select("mydb") b) connection.database("mydb") c) cursor.execute("USE mydb") d) connect.use("mydb")
10	What is the output type of fetchall( )? a) List of strings b) Tuple of strings c) List of tuples d) Dictionary
11	Which of the following is the correct syntax to connect to a MySQL database using mysql.connector? a) mysql.connector.connect(user='root', host='localhost') b) mysql.connect('localhost', 'root') c) connect.mysql(user='root') d) mysql.connector.connection(user='root')
12	Which of the following is NOT a valid connection parameter for `mysql.connector.connect( )`? a) `host` b) `user` c) `pwd` d) `database`

Answers

1	2	3	4	5	6	7	8	9	10	11	12
c	d	c	c	b	a	b	c	c	c	a	c

Assertion Reasoning Questions

Each question contains Assertion (A) and Reason (R). Choose the correct option from:
a) Both A and R are true and R is the correct explanation of A.
b) Both A and R are true but R is not the correct explanation of A.
c) A is true but R is false.
d) A is false but R is true.

1	Assertion (A): The cursor( ) method is used to execute SQL queries in Python. Reason (R): The cursor object allows interaction with the database.
2	Assertion (A): cursor.fetchall( ) is used to delete all records from a table. Reason (R): fetchall( ) retrieves all records returned by a SELECT query.
3	Assertion (A): mysql.connector.connect( ) requires parameters like host, user, password, and database to connect successfully. Reason (R): mysql.connector.connect( ) is used to establish a connection between Python and MySQL.
4	Assertion (A): commit( ) is used after every SELECT query to save the fetched results. Reason (R): commit( ) saves changes made by INSERT, UPDATE, and DELETE operations.
5	Assertion (A): The close( ) method is mandatory after the database operations are completed. Reason (R): Keeping database connections open unnecessarily may lead to resource leakage.
6	Assertion (A): fetchone( ) returns all rows from a query result. Reason (R): fetchone( ) returns only a single record from the result set.
7	Assertion (A): The execute( ) method can only be used with SELECT queries. Reason (R): execute( ) is a method of the cursor object used to run SQL commands.
8	Assertion (A): Without calling commit( ), changes made by an INSERT query might not be saved in the database. Reason (R): commit( ) finalizes the transaction and writes changes to the database.

Answers

1	2	3	4	5	6	7	8
a	d	a	d	a	d	c	a

VERY SHORT ANSWER QUESTIONS

1	Which command is use to install MySQL library in python?
2	Which method we use to establish the connection?
3	Which statement we use to access the MySQL module?
4	What is the Database Connector?
5	Which function is used to check the successful connection?

Answers

1	pip install MySQL. Connector
2	connect( ) method with connection object.
3	import mysql.connector
4	A database connector is a software that connects an application to any database.

5	.is_connected( ) method

SHORT ANSWER QUESTIONS

1	What are the steps for creating database connectivity applications?
2	What is a connection? What is its role?
3	What is a resultset?
4	What is a database cursor?
5	How to retrieve data from a table?

Answers

1	To create database connectivity, follow the given steps: Step 1: Start Python Step 2: Import mysql.connector Step 3: Open a connection to the database Step 4: Create a cursor instance Step 5: Execute a query Step 6: Extract data from result set Step 7. Clean up the environment
2	A connection (represented by the connection object) is the session between the application program and database. To do anything with database, one must have a connection object.
3	A result set refers to a logical set of records that are fetched from the database by executing a query and made available to the application program.
4	A database cursor is a special control structure that facilitates row by row processing of records in the result set, i.e., the set of records retrieved as per the query.
5	There are multiple ways to retrieve data: i. fetchall( ) – fetches all the remaining rows of a query result, current pointer position forwards ii. fetchone( ) – fetches the next row as a sequence; returns None when no more data iii. fetchmany(n) :It will return number of records as per value of n and by-default only one record.

LONG ANSWER QUESTIONS

1	Write a Python code to connect to a database
2	How to create a database in MySQL through Python ?
3	Write the Python code to display the present databases in MySQL
4	How to create a table in MySQL through Python ?
5	Write the Python code to insert data into student table of database kvs .
6	How to fetch data in Python from a student table of education database?
7	Write the Python code to update a record in the student table of education database.
8	Write the Python code to delete a record from the student table of education database
9	Mr.Harsh want to interface python with mysql and write some code help him to write the code import_____.connector as m #Line1 mydb=m._____(host="localhost",user="root", passwd="tiger",database=" choo ") # Line2 cursor=mydb._____( ) #Line3 cursor._____("select * from student") #Line4 data=cursor._____( ) #Line5 To retrieved all records

	count=cursor._____ #Line6
10	What is a cursor? Explain how to create and use the same.
11	What is the difference between fetchone( ), fetchmany( ), fetchall( )?

Answers

1	import mysql.connector as m Mycon=m.connect(host="localhost",user="root", password="tiger",database="project") print(mycon)
2	import mysql.connector as m mycon= m.connect(host="localhost",user="root",password="tiger") cursor=mycon.cursor( ) cursor.execute("create database education") mycon.commit( )
3	import mysql.connector as m mycon= m.connect(host="localhost",user="root",password="tiger") cursor=mycon.cursor( ) cursor.execute("show databases") for i in cursor: print(i)
4	import mysql.connector as m mycon= m.connect(host="localhost",user="root", password="tiger") cursor=mycon.cursor( ) query = "create table student(admn_no int primary key, sname varchar(30), gender char(2), DOB date, stream varchar(10), mark float" cursor.execute(query) mycon.commit( )
5	import mysql.connector as m mycon= m.connect(host="localhost",user="root",password="tiger") cursor=mycon.cursor( ) no=int(input("Enter admission no ")) n=input("Enter name ") g=input("Enter gender ") dob=input("Enter DOB ") st=input("Enter stream ") m=float(input("Enter m rk ")) query= "insert into student values({}, '{}' , '{}' , '{}' , '{}' , {})".format(no,n,g,dob, st,m) cursor.execute(query) mycon.commit( )
6	import mysql.connector as m mycon= m.connect(host="localhost",user="root",password="tiger", database = 'education') cursor=mycon.cursor( ) cursor.execute("select * from student") for row in cursor: print(row)
7	import mysql.connector as m mycon= m.connect(host="localhost",user="root",password="tiger", database = 'education') cursor=mycon.cursor( )

	cursor.execute("update student set marks =67 where admn_no=306") mycon.commit( )
8	import mysql.connector as m mycon= m.connect(host="localhost",user="root",password="tiger", database = 'education') cursor=mycon.cursor( ) cursor.execute("delete from student where admn_no=308 ") mycon.commit( )
9	Line1:-mysql, Line2:-connect, Line3:cursor, Line4: execute, Line5: fetchall, Line6: rowcount
10	A cursor is an object which helps to execute the query and fetch the records from the database. import mysql.connector as m mycon= m.connect(host="localhost",user="root",password="tiger", database = 'education') mycursor=mycon.cursor( ) mycursor.execute("MySQL Command") myconn.close( )
11	**fetchone( )** • **Returns:** A single row from the result set as a tuple (or None if no more rows). **fetchmany(size)** • **Returns:** A list of **up to size rows**, each as a tuple. **fetchall( )** • **Returns:** A list of **all remaining rows** as tuples in the result set.

LONG ANSWER QUESTIONS

1	A table, named STATIONERY, in ITEMDB database, has the following structure: Field Type itemNo int(11) itemName varchar(15) price float qty int(11) Assume the following for Python-Database connectivity: Host: localhost, User: root, Password: Pencil Write the following Python function to perform the specified operation: **Add( ):** To input details of an item and store it in the table STATIONERY. **Display( ):** The function should then retrieve and display all records from the STATIONERY table where the Price is greater than 120.
2	Sartaj has created a table named CLASS in MYSQL database, CLASS: rno(Roll number)- integer name(Name) - string DOB (Date of birth) – Date Fee – float Note the following to establish connectivity between Python and MySQL: Username – root, Password – tiger, Host – localhost, DB – SC Sartaj, now wants to display the records of students whose fee is more than 5000. Help Sartaj to write the program in Python.
3	Kabir wants to write a program in Python to insert the following record in the table named CLASS in MYSQL database SC, CLASS: rno(Roll number)- integer

	name (Name) - string DOB (Date of birth) – Date Fee – float Note the following to establish connectivity between Python and MySQL: Username – root, Password – tiger, Host – localhost, DB – SC The values of fields rno, name, DOB and fee has to be accepted from the user. Help Kabir to write the program in Python.
4	Rehaan wants to write a program in Python to insert the following record in the table named EMP in MYSQL database COMPANY: a. eno(Empno)- integer b. ename(Name) - string c. DOB (Date of birth) – Date d. Salary – float Note the following to establish connectivity between Python and MySQL: Username – root, Password – password, Host - localhost The values of fields eno, name, DOB and salary has to be accepted from the user. Help Rehaan to write the program in Python.
5	Gurmehar has created Product in MYSQL database COMM: Pno(Product Number)- integer Pname(Product Name) - string DOM (Date of Manufacture) – Date Price – float Note the following to establish connectivity between Python and MySQL: Username – root, Password – password, Host – localhost Gurmehar, now want to display the records of the product whose price is more than 10000. Help Gurmehar to write the program in python.

Answers:

1	import mysql.connector as m mycon= m.connect(host="localhost",user="root",password="Pencil") mycursor=mycon.cursor( ) def Add(mycursor): mycursor.execute('use ITEMDB') n = int(input('Enter no. of Items to be entered: ')) for i in range(n): ino = int(input('Enter Item no: ')) inm = input('Enter Item name: ') ipr = float(input('Enter Item price: ')) iq = int(input('Enter Item quantity: ')) query = f "insert into STATIONARY values({ino},'{inm}',{ipr},{iq})" mycursor.execute(query) mycon.commit( ) def Display(mycursor): mycursor.execute('use ITEMDB') query = 'Select * from STATIONARY where price > 120' mycursor.execute(query) data = mycursor.fetchall( ) for record in data:

	print('Item No = ' , record[0]) print('Item Name = ' , record[1]) print('Item Price = ' , record[2]) print('Item Qty = ' , record[3]) Add(mycursor) Display(mycursor) mycon.close( )
2	def Display(mycursor): query = 'Select * from CLASS where Fee > 5000' mycursor.execute(query) data = mycursor.fetchall( ) for record in data: print('Roll No = ' , record[0]) print('Name = ' , record[1]) print('DOB = ' , record[2]) print('Fee = ' , record[3]) import mysql.connector as m mycon=m.connect(host="localhost",user="root",password="tiger", database='SC') mycursor=mycon.cursor( ) Display(mycursor) mycon.close( )
3	def Add(mycursor): n = int(input('Enter no. of Student Records to be entered: ')) for i in range(n): rn = int(input('Enter Student Roll No: ')) nm = input('Enter Student name: ') dob = input('Enter Student DOB as 'YYYY-MM-DD' : ')) fee = float(input('Enter Student Fee: ')) query = f "insert into CLASS values({rn},'{nm}','{dob}',{fee})" mycursor.execute(query) mycon.commit( ) import mysql.connector as m mycon=m.connect(host="localhost",user="root",password="tiger", database='SC') mycursor=mycon.cursor( ) Add(mycursor) mycon.close( )
4	def Add(mycursor): mycursor.execute('use COMPANY') n = int(input('Enter no. of Records to be entered: ')) for i in range(n): en = int(input('Enter Student Roll No: ')) enm = input('Enter Student name: ') dob = input('Enter Student DOB as 'YYYY-MM-DD' : ')) sal = float(input('Enter Student Fee: ')) query = f "insert into EMP values({en},'{enm}','{dob}',{sal})" mycursor.execute(query) mycon.commit( )

	import mysql.connector as m mycon=m.connect(host="localhost",user="root",password="password") mycursor=mycon.cursor( ) Add(mycursor) mycon.close( )
5	def Display(mycursor): query = 'Select * from Product where Price > 10000' mycursor.execute(query) data = mycursor.fetchall( ) for record in data: print('Roll No = ' , record(0]) print('Name = ' , record(1]) print('DOB = ' , record(2]) print('Fee = ' , record(3]) import mysql.connector as m mycon=m.connect(host="localhost",user="root",password="password", database='COMM') mycursor=mycon.cursor( ) Display(mycursor) mycon.close( )

CBSE QUESTION PAPER 2025-26

Computer Science (083)

Subject: Computer Science (Theory) **Class: XII**
Time Allowed: 3:00 Hours **Max. Marks - 70**

General Instructions:

- This question paper contains 37 questions.
- All questions are **compulsory**. However, internal choices have been provided in some questions. Attempt only one of the choices in such questions
- The paper is divided into **5 Sections**- A, B, C, D and E.
- Section A consists of **21** questions (**1 to 21**). Each question carries **1 Mark**.
- Section B consists of **7** questions (**22 to 28**). Each question carries **2 Marks**.
- Section C consists of **3** questions (**29 to 31**). Each question carries **3 Marks**.
- Section D consists of **4** questions (**32 to 35**). Each question carries **4 Marks**.
- Section E consists of **2** questions (**36 to 37**). Each question carries **5 Marks**.
- All programming questions are to be answered using Python Language only.
- **In case of MCQ, text of the correct answer should also be written**.

Q. No.	SECTION-A (21 x 1 = 21 Marks)	Marks
1	State True or False : In Python, data type of 74 is same as the data type of 74.0.	1
2	Identify the output of the following code snippet : s = "the Truth" print(s.capitalize()) (A) The truth (B) THE TRUTH (C) The Truth (D) the Truth	1
3	Which of the following expressions in Python evaluates to True ? (A) 2>3 and 2<3 (B) 3>1 and 2 (C) 3>1 and 3>2 (D) 3>1 and 3<2	1
4	What is the output of the following code snippet ? s='War and Peace by Leo Tolstoy' print(s.partition("by")) (A) ('War and Peace ', 'by', ' Leo Tolstoy') (B) ['War and Peace ', 'by', ' Leo Tolstoy'] (C) ('War and Peace ', ' Leo Tolstoy') (D) ['War and Peace ', ' Leo Tolstoy']	1
5	What will be the output of the following statement ? print ("PythonProgram" [-1 : 2 :- 2])	1

6	What will be the output of the following code snippet ? t = tuple('tuple') t2 = t[2], t += t2 print (t) (A) ('tuple') (B) ('tuple', 'p') (C) ('t', 'u', 'p', '1', 'e', 'p') (D) ('t', 'u', 'p', '1', 'e')	1
7	Which of the following statements is true about dictionaries in Python ? (A) A dictionary is an example of sequence datatype. (B) A dictionary cannot have two elements with same key. (C) A dictionary cannot have two elements with same value. (D) The key and value of an element cannot be the same.	1
8	If L is a list with 6 elements, then which of the following statements will raise an exception ? (A) L.pop(1) (B) L.pop(6) (C) L.insert(1,6) (D) L.insert(6,1)	1
9	What will be the output of the following code ? def f1(a,b=1): print (a+b, end='-') c=f1(1,2) print (c, sep='*') (A) 3-2 (B) 3-2* (C) 3-None (D) 3*None-	1
10	Consider the statement given below : f1 = open("pqr.dat","_____") Which of the following is the correct file mode to open the file in read only mode ? (A) a (B) rb (C) r+ (D) rb+	1
11	State whether the following statement is True or False : In Python, Logical errors can be handled using try ..... except ..... finally statement.	1
12	A table has two candidate keys, one of which is chosen as the primary key. How many alternate keys does this table have ? (A) 0 (B) 1 (C) 2 (D) 3	1
13	Which of the following SQL command can change the degree of the existing relation ? (A) DROP TABLE (B) ALTER TABLE (C) UPDATE ... SET (D) DELETE	1

14	What will be the output of the query ? SELECT MACHINE_ID, MACHINE_NAME FROM INVENTORY WHERE QUANTITY <= 100; (A) All columns of INVENTORY table with quantity greater than 100 (B) ID and name of machines with quantity less than 100 from INVENTORY table (C) All columns of INVENTORY table with quantity greater than or equal to 100 (D) ID and name of machines with quantity less than or equal to 100 from INVENTORY table.	1
15	A relation in MySQL database consists of 2 tuples and 3 attributes. If 2 attributes are deleted and 4 tuples are added, what will be the cardinality of the relation ? (A) 4 (B) 5 (C) 6 (D) 7	1
16	Which aggregate function in SQL returns the smallest value from a column in a table ? (A) MIN () (B) MAX() (C) SMALL () (D) LOWER ()	1
17	With respect to computer networks, which of the following is the correct expanded form of RJ 45 ? (A) Radio Jockey 45 (B) Registered Jockey 45 (C) Radio Jack 45 (D) Registered Jack 45	1
18	Which network device serves as the entry and exit point of a network, as all data coming in or going out of a network must first pass through it in order to use routing paths ? (A) Modem (B) Gateway (C) Switch (D) Repeater	1
19	Expand the term XML.	1
	Q.20 and Q.21 are Assertion(A) and Reason(R) 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	
20	**Assertion (A)** : [1,2,3]+'123' is an invalid expression in Python. **Reason (R)**: In Python, a list cannot be concatenated with a string.	1
21	**Assertion (A):** The PRIMARY KEY constraint in SQL ensures that each value in the column(s) is unique and cannot be NULL. **Reason (R):** Candidate keys are not eligible to become a primary key.	1
	SECTION-B (7 x 2=14 Marks)	
22	What is the difference between default parameters and positional parameters in Python ? Also give an example of a function header which uses both.	2

23	Write a Python statement to perform the following tasks : (USE BUILT_IN FUNCTIONS METHODS ONLY) (i) To create a new list L1 containing the elements of list L arranged in ascending order, without modifying list L. (ii) A statement to check whether the given character, ch is an alphabet or a number.	2
24	Assuming that D1 is a dictionary in Python, (i) (a) Write a Python expression to check if the key, 'RNo' is present in D1. OR (b) Write a Python expression to check if any key in D1 has a value 12. (ii) (a) Write a single statement using a BUILT_IN function to add the key : value pair 'RNo' : 12, if the key 'RNo'is not present in D1. However, if the key 'RNo' is present, the function should return its value. OR (b) Write a single statement to delete all the elements from D1.	2
25	What possible output(s) from the given options will NOT be displayed when the following code is executed ? Also, mention, for how many iterations the for loop in the given code will run ? import random a = [1,2,3,4,5,6] for i in range (4) : j = random. randrange (i, 5) print (a[j],end='-') print () Options : (A) 3-4-5-4- (B) 2-2-4-5- (C) 4-3-3-5- (D) 5-1-2-4-	2
26	The function given below is written to accept a string s as a parameter and return the number of vowels appearing in the string. The code has certain errors. Observe the code carefully and rewrite it after removing all the logical and syntax errors. Underline all the corrections made. def CountVowels (s): c=0 for ch in range(s): if 'aeiouAEIOU' in ch: c=+1 return (ch)	2

27	Ms. Zoya is a Production Manager in a factory which packages mineral water. She decides to create a table in a database to keep track of the stock present in the factory. Each record of the table will have the following fields : W_Code - Code of the item (type - CHAR (5)) W_Description - Description of the item (type - VARCHAR (20)) B_Qty - Balance quantity of the item (type - INTEGER) U Price - Unit Price of the item (type - FLOAT) The name of the table is W_STOCK. (i) (a) Write an SQL command to create the above table (W_Code should be the primary key). OR (b) Can U_Price be the primary key of the above table? Justify your answer. (ii) (a) Assuming that the table W_STOCK is already created, write an SQL command to add an attribute E_Date (of DATE type) to the table. OR (b) Assuming that the table W_STOCK is already created, write an SQL command to remove the column B_Qty from the table.	2
28	(a) List one advantage and one disadvantage of Bus topology. OR (b) What is protocol in the context of computer networks? Which protocol is used to transmit hypertext across the web?	2
	SECTION-C (3 x 3 = 9 Marks)	
29	(A) Write a Python function that counts and returns the number of digits appearing in the text file "Space. txt". For example, if the file contains : *Space exploration has unlocked incredible advancements in technology and science since the first moon landing in 1969, space agencies have sent probes to Mars, Jupiter and beyond. The ISS, orbiting Earth at about 400 km, serves as a hub for research. With missions planned for 2030, humanity's cosmic journey continues!* **Then the function should return 11.** OR (B) Write a Python function that displays the words in which the lowercase letter 'e' appears at least twice in the text file 'space.txt'. For example, if the file contains : *Space exploration has unlocked incredible advancements in technology and science since the first moon landing in 1969, space agencies have sent probes to Mars, Jupiter and beyond. The ISS, orbiting Earth at about 400 km, serves as a hub for research. With missions planned for 2030, humanity's cosmic journey continues!* Then the function should display: **incredible advancements science. agencies serves research.**	3

30	A stack named FruitStack, implemented using list, contains records of some fruits. Each record is represented as a dictionary with keys 'Name', 'Origin', 'Price', and 'Expiry'. A sample record is given here : { 'Name' : 'Apple', 'Origin' : 'France' , ' Price' : 120, 'Expiry' : '12-08-2025' } Write the following user-defined functions in Python to perform the specified operations on FruitStack : (i) **push_fruit (FruitStack, Fruit)** : This function takes the stack FruitStack and a new record Fruit as arguments and pushes the record stored in Fruit onto FruitStack if the Price is less than 100. (ii) **pop_fruit(FruitStack):** This function pops the topmost record from the stack and returns it. If the stack is already empty, the function should display "UNDERFLOW". (iii) **display (FruitStack):** This function displays all the elements of the stack starting from the topmost element. If the stack is empty, the function should display 'EMPTY STACK'. OR Write a Python program to accept 10 integers from the user. If the entered number is a three-digit even integer, push it onto a stack. After all inputs are taken, pop all the three-digit even integers from the stack and display them. For example, if the user enters 12, 31, 320, 457, 6, 92, 924, 220, 1, 218, then the stack should contain : 320, 924, 220, 218 and the output of the program should be : 218 220 924 320	3
31	(a) Write the output of the following code : def Exam2026 (given) : new=[] for ch in given[1 :- 1] : if ch.isupper(): new.reverse () elif ch not in new: new. append (ch) elif ch in new: new.pop () print (new) Exam2026 ("Gold-24Medals") OR (b) Write the output of the following code : def Exam2026 (given) : new = 0	3

```
        while given:
            if new % 2:
                new += given % 10
            else:
                new += given % 5
        print(new, end='-')
        given //= 10
    Exam2026 (123456)
```

SECTION-D (4 x 4 = 16 Marks)

32 Abhishek has created a table, named STOCK, with a set of records to maintain the data of packaged milk in his shop. After creating the table, he entered the data and the table looked as follows - 4

Code	Type	Volume	Qty	Price
AF0.5	F	0.5	300	38.00
MF0.5	F	0.5	250	36.50
MT1.0	T	1.0	150	64.00
AT1.0	T	1.0	100	66.00
PD1.0	D	1.0	50	52.00
PT0.5	T	0.5	78	30.00

(a) Based on the data given above, write the SQL queries for the following tasks :

(i) To display Type and the maximum Price for each Type of milk.

(ii) For each record, increase the Price by 0.5 where Type is 'F'.

(iii) To display the total value of the stock (total of Qty x Price).

(iv) To display the details of all records where Code starts with 'A'.

OR

(b) Considering the table STOCK as given above, write the output on execution of the following queries :

(i) SELECT Volume, Qty, Price FROM STOCK WHERE Type IN ('F', 'D');

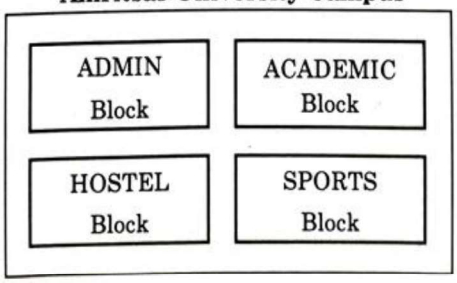
(ii) SELECT Code, Qty FROM STOCK WHERE Price BETWEEN 30 AND 50;

(iii) SELECT DISTINCT Type FROM STOCK;

(iv) SELECT Volume, count (*) FROM STOCK GROUP BY Volume;

33	A csv file "States. csv" contains some data about all the states of India. Each record of the file contains the following data : - Name of the State - Capital of the State - Population of the State - Official Language of the State **For example, a sample record in the file is :** ['Andhra Pradesh', 'Amaravati ', 52221000, 'Telugu'] Write a Python program which reads the data from this file and appends all those records where population is more than 10000000 into another csv file 'More. csv'. **Note : "States.csv" also contains the Header row. The Header row should NOT be copied to "More. csv".**	4
34	Assume that you are the Manager of the Loans department of a Finance House. To keep track of the loans you have created two tables : CUSTOMERS and LOANS. The sample data in these tables is given below : Table: CUSTOMERS C_ID | C_NAME | Phone 00001 | Raj Malhotra | 1234567890 00003 | David Xavier | 3456789012 00004 | Damini Iyer | 3156789012 00008 | Abdul | 2345678901 Table: LOANS Sno | C_ID | L_Amt | L_Date | Terms | RoI 1 | 00003 | 200000 | 2025-12-06 | 60 | 7.80 2 | 00008 | 2500000 | 2023-08-09 | 60 | 9.00 3 | 00001 | 500000 | 2025-08-13 | 48 | 6.00 4 | 00003 | 300000 | 2026-12-07 | 36 | 8.00 5 | 00004 | 600000 | 2026-12-07 | 60 | 6.00 Note : The tables may contain more records than shown here. The management of the Finance House needs certain reports from you. Write the queries to extract the following data to create the reports : (i) Number of records from LOANS table where Rate of Interest (RoI) is above 7.0. (ii) Names of the customers whose loan amount (L_Amt) is above 1000000. (iii) C_ID, C_Name and Terms of all those records where Loan Date (L_Date) is after 31st December, 2024. (iv) (a) Details of all the loans in the descending order of RoI. OR (b) C_ID and average term for each C_ID from the LOANS table.	4

35	Peter has created a table named Account in MySQL database, SCHOOL, having following structure : Stud_id - integer Sname - string Class - string Fees - float Help him in writing a Python program to display records of those students whose fees is less than 5000. Note the following to establish connectivity between Python and MySQL : - Username - admin - Password - root - Host – localhost	4
	SECTION-E (2 X 5 = 10 Marks)	
36	NextStep is an organization which has a pool of resource persons to conduct training workshops on various topics related to ICT. The data of all its Resource Persons is stored in a binary file RESOURCES. DAT using the following record structure (each record is a tuple) : (R_ID, R_Name, R_Expertise, Charges) where : R_ID - Resource Person's ID (An integer) R Name - Resource Person's Name (A string) R Expertise - Area of expertise of the Resource Person Charges - Charges (in rupees) per hour to conduct a workshop For example, a record in the file is : (12, 'P. Velusami', 'Machine Learning', 5000) In this context, write the following user defined functions in Python : (i) Append () - To input the data of a Resource Person and write it in the file RESOURCES. DAT. (ii) Update () - To increase the Charges of each resource person by 500.	5
37	Sanjeevani is a big group of educational institutions with its head office in Hyderabad. It is planning to set up a new University in Amritsar. The Amritsar University Campus will have four blocks/buildings - ADMIN, ACADEMIC, HOSTEL, SPORTS. You, as a network expert, need to suggest the best network-related solutions for them to resolve the issues/problems mentioned in points (i) to (v), keeping in mind the distances between various blocks / buildings and other given parameters.	5



Block to Block distances (in Mtrs.)

FROM	TO	DISTANCE
ADMIN	ACADEMIC	60 M
ADMIN	HOSTEL	160 M
ADMIN	SPORTS	80 M
ACADEMIC	HOSTEL	40 M
ACADEMIC	SPORTS	120 M
HOSTEL	SPORTS	150 M

Distance of Hyderabad Head Office from Amritsar University Campus = 2000 km.

Number of computers in each block is as follows :

ADMIN	**25**
ACADEMIC	**600**
HOSTEL	**120**
SPORTS	**50**

(i) Suggest the most appropriate location of the server inside the Amritsar University Campus. Justify your choice.

(ii)) Draw the cable layout to efficiently connect various blocks within the Amritsar University Campus.

(iii) Name any two wired media that can be used to connect various computers of a block inside Amritsar Campus.

(iv) For the academic purpose, the University will provide its own 24 x 7 FM channel within the University Campus. Which communication medium, out of the following, is used by FM ?

(A) Radio Waves
(B) Micro Waves
(C) Infrared Waves

V) (A) The students will be attending a lot of online academic sessions and workshops. These will involve audio-visual communication. Write the full name of the protocol which will be used for such a communication through the internet.

OR

(B) Where should a repeater be installed in Amritsar University campus to boost the signal between blocks ? Justify your answer.

CBSE: 2025-26
MARKING SCHEME: 2025-26
Computer Science (083)

Subject: Computer Science (Theory) **Class: XII**

	SECTION A (21x1=21)	
1.	False	1
2.	(A) The truth	1
3.	(C) 3>1 and 3>2 (B) 3>1 and 2	1
4.	(A) ('War and Peace ', 'by', ' Leo Tolstoy')	1
5.	mroPo	1
6.	(C) ('t', 'u', 'p', 'l', 'e', 'p')	1
7.	(B) A dictionary cannot have two elements with same key.	1
8.	(B) L.pop(6)	1
9.	(C) 3–None	1
10.	(B) rb	1
11.	False	1
12.	(B) 1	1
13.	(B) ALTER TABLE	1
14.	(D) ID and name of machines with quantity less than or equal to 100 from INVENTORY table.	1
15.	(C) 6	1
16.	(A) MIN()	1
17.	(D) Registered Jack 45	1
18.	(B) Gateway	1
19.	Extensible Markup Language	1
20.	(A) Both Assertion (A) and Reason (R) are true and Reason (R) is the correct explanation for Assertion (A).	1
21.	(C) Assertion (A) is true, but Reason (R) is false.	1
	SECTION B (7x2=14)	
22.	• Default parameters have default values whereas positional parameters are assigned values to them based on their position in the function call. • In a function header, default parameters are always written after positional parameters(if any). Example def Q22(X, Y=10):	2
23.	(i) L1=sorted(L) OR L1=list(L) L1.sort()	2

	(ii) ch.isalnum() OR ch.isdigit() or ch.isalpha()	
24.	(i) (a) 'RNo' in D1 OR 'RNo' in D1.keys() (b) 12 in D1.values() (ii) (a) D1.setdefault('RNo',12) OR if D1.get('RNo'): print(D1['RNo']) # return D1['RNo'] else: D1['RNo']=12 (b) D1.clear() OR D1={} OR D1=dict()	2
25.	(D) 5–1–2–4– The for loop will run for 4 iterations.	2
26.	def CountVowels(s): c=0 for ch in **s**: #Correction-1 if **ch in 'aeiouAEIOU'**: #Correction-2 c**+=**1 #OR c=c+1 #Correction-3 **return(c)** #Correction-4	2
27.	(i) (a) CREATE TABLE W_STOCK (W_Code CHAR(5) PRIMARY KEY, W_Description VARCHAR (20), B_Qty INTEGER, U_Price FLOAT); (b) No, U_Price cannot be the primary key because it may contain duplicate and Null values. ii) (a) ALTER TABLE W_STOCK ADD COLUMN E_Date DATE; OR ALTER TABLE W_STOCK ADD E_Date DATE; (b) ALTER TABLE W_STOCK DROP COLUMN B_Qty; OR ALTER TABLE W_STOCK DROP B_Qty;	2
28.	(a) Advantages : • Cost-effective: As it requires less cable. • Easy to install • Easy to extend for small networks. Disadvantages : • Failure of the main backbone brings down the entire network. • Fault isolation is difficult. • Not suitable for large networks • Data collisions are frequent when many devices are connected. • Network speed slows down with heavy traffic. • Security is low as all devices can access transmitted data. (b) A protocol is a set of rules that need to be followed by the communicating devices in order to have successful and reliable data communication. The protocol used to transmit hypertext across the web is Hypertext Transfer Protocol (HTTP).	2

SECTION C (3x3=9)

29. (a)

```
def CountDigits():
    with open("Space.txt",'r') as F:
        Data=F.read()
        C=0
        for CH in Data:
            if CH.isdigit(): # OR if CH in '0123456789'
                C+=1          # OR if CH >= '0' and CH <= '9'
    return C
```

(b)

```
def showWords():
    with open("Space.txt",'r') as F:
        All=F.read()
        Words=All.split()
        for word in Words:
            if word.count('e')>=2:
                print(word, end=' ')
```

30. (a)

(i)

```
def push_fruit(FruitStack, Fruit):
    if Fruit['Price']<100:
        FruitStack.append(Fruit)
```

(ii)

```
def pop_fruit(FruitStack):
    if FruitStack==[]: # OR if len(FruitStack)==0:
        print('UNDERFLOW')
    else:
        return FruitStack.pop()
```

(iii)

```
def display(FruitStack):
    if FruitStack==[]:  # OR if not FruitStack: # OR if len(FruitStack)==0:
        print('EMPTY STACK')
    else:
        for ele in FruitStack[::-1]:
            print(ele)
```

(b)

```
Stack=[] #Stack
for C in range(10):
    Num=int(input("Integer:"))
    if Num>=100 and Num<=999 and Num%2==0: #
        if len(str(Num))==3 and Num%2==0:
            Stack.append(Num)
while Stack:
    print(Stack.pop(), end=' ')
```

31. (a) ['4', '2', '-', 'd', 'l', 'o']

(b) 1-6-10-13-15-16-

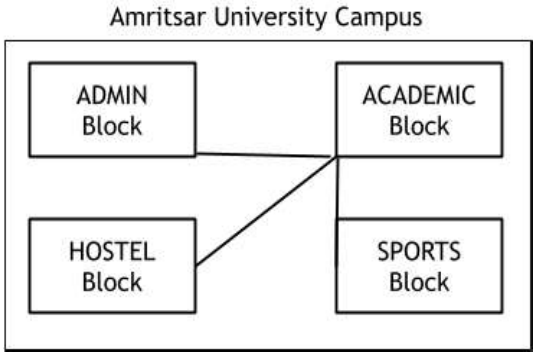
SECTION D (4x4=16)

32. (a)

(i) SELECT TYPE, MAX(PRICE) FROM STOCK GROUP BY TYPE;

(ii) UPDATE STOCK SET PRICE=PRICE+0.5 WHERE TYPE='F';

	(iii) SELECT SUM(QTY*PRICE) FROM STOCK; (iv) SELECT * FROM STOCK WHERE CODE LIKE 'A%'; (b) (i) VOLUME QTY PRICE 0.5 300 38.00 0.5 250 36.50 1.0 50 52.00 (ii) CODE QTY AF0.5 300 MF0.5 250 PT0.5 78 (iii) DISTINCT TYPE F T D (iv) VOLUME COUNT(*) 0.5 3 1.0 3
33.	import csv with open('States.csv') as F1 : with open('More.csv','a',newline='') as F2: # file mode 'w' is also acceptable R=csv.reader(F1) W=csv.writer(F2) RECS=list(R) for rec in RECS[1:]:#skipping the Header row if int(rec[2])>10000000: W.writerow(rec)
34.	(i) SELECT COUNT(*) FROM LOANS WHERE RoI > 7.0; (ii) SELECT C_Name FROM CUSTOMERS JOIN LOANS ON CUSTOMERS.C_ID = LOANS.C_ID WHERE L_Amt > 1000000; *(iii) SELECT L.C_ID, C_NAME, TERMS FROM CUSTOMERS C, LOANS L* *WHERE C. C_ID = L.C_ID AND L_DATE > '2024-12-31';* *(iv)(a) SELECT * FROM LOANS ORDER BY RoI DESC;* *OR* *(iv) (b) SELECT C_ID, AVG(TERMS) FROM LOANS GROUP BY C_ID;*
35.	*import pymysql as pm # OR import ysql.connector as pm* *DB = pm.connect(host='localhost',user='admin', password='root', database = 'SCHOOL'))* *CUR = DB.cursor()* *SQL = "SELECT * FROM Account WHERE Fees<5000"* *CUR.execute(SQL)* *Data=CUR.fetchall( )* *for D in Data:*

	print(D) *DB.close()*
	SECTION E (2x5=10)
36.	(i) import pickle def Append(): #Assuming independent tuples in file F=open("RESOURCES.DAT","ab") R_ID = int(input("Resource Person ID:")) R_Name = input("Resource Person Name:") R_Expertise = input("Area of Expertise:") Charges = int(input("Hourly Charges:")) Rec=(R_ID, R_Name, R_Expertise, Charges) pickle.dump (Rec, F) F.close()
	(ii) import pickle def Update(): #Assuming independent tuples in file F=open("RESOURCES.DAT","rb+") try: while True: Pos=F.tell() Rec=pickle.load(F) Newrec=list(Rec) Newrec[3]+=500 F.seek(Pos) pickle.dump(tuple(Newrec),F) except: F.close()
37.	(i) Academic Block as it has the maximum number of computers (ii)  (iii) Twisted pair cable, Ethernet cable, Coaxial Cable, Optical Fiber Cable (any two) (iv) (A) Radio waves (v) (a) Voice over Internet Protocol (b) Between Academic and Sports block because distance is more than 100m.

KENDRIYA VIDYALAYA SANGATHAN
UNSOLVED SAMPLE QUESTION PAPER-1

Subject: Computer Science (Theory) **Class: XII**
Time Allowed: 3:00 Hours **Max. Marks - 70**

General Instructions:

- This question paper contains 37 questions.
- All questions are compulsory. However, internal choices have been provided in some questions. Attempt only one of the choices in such questions
- The paper is divided into 5 Sections- A, B, C, D and E.
- Section A consists of 21 questions (1 to 21). Each question carries 1 Mark.
- Section B consists of 7 questions (22 to 28). Each question carries 2 Marks.
- Section C consists of 3 questions (29 to 31). Each question carries 3 Marks.
- Section D consists of 4 questions (32 to 35). Each question carries 4 Marks.
- Section E consists of 2 questions (36 to 37). Each question carries 5 Marks.
- All programming questions are to be answered using Python Language only.
- In case of MCQ, text of the correct answer should also be written.

Q. No.	SECTION-A (21 x 1 = 21 Marks)	Marks
1.	State True or False: "In a nested loop, a break statement terminates all the nested loops in one go."	1
2.	Identify the output of the following code snippet: line="successful entrepreneurs" r=line.partition("en") print(r) (A) 'successful ', 'en', 'trepreneurs' (B) ('successful', 'en', 'trepreneurs') (C) ['successful ', 'en', 'trepreneurs'] (D) ('successful ', 'en', 'trepreneurs')	1
3.	Which of the following expressions evaluates to Non-zero value? (A) 3 and 0 (B) True==1 and 6-6 (C) 0==1 and 0 or 5 (D) 2**(2**0)+5%7-50//7	1
4.	What is the output of the given expression? remark = "NLP-Natural Language Processing" note = remark[2:18].split( ) print(note) (A) ['P-Natural', 'Languag'] (B) ['LP-Natural', 'Langua'] (C) ['P-Natural', 'Langua'] (D) ['LP-Natural', 'Languag']	1
5.	Which among the following errors will occur at the time of execution of given python code: >>> Oceans=('pacific', 'arctic', 'Atlantic', 'southern') >>> print(Oceans[4])	1

	(A) ValueError (B) IndexError (C) NameError (D) TypeError	
6.	What will be the output of the following code? Bparts=['ear','nose','eye'] print('2'.join(Bparts)) (A) ear2nose2eye (B) ear2nose2eye2 (C) ['ear','nose','eye', 'ear','nose','eye'] (D) 2ear2nose2eye2	**1**
7.	Which of the following statement(s) would give an error after executing the following code ? Stud= { "Kiran": 70, "Jaya": 95} # Statement 1 print (Stud[95]) # Statement 2 Stud ["Arvind"]=60 # Statement 3 print(Stud.pop( )) # Statement 4 print(Stud) # Statement 5 (A) Statement- (B) Statement-3 (C) Statement-4 (D) Statement-2 and 4	**1**
8.	Which of the following is not correct? (A) del deletes the list or tuple from the memory (B) remove deletes the list or tuple from the memory (C) pop is used to delete an element at a certain position (D) pop(<index>) and remove(<element>) performs the same operation	**1**
9.	If a table has one Primary key, two alternate keys and one foreign key. How many Candidate keys will this table have? (A) 1 (B) 2 (C) 3 (D) 4	**1**
10.	The correct syntax of dump( ) is: (A) file_object=dump(data_object) (B) file_object.dump(data_object) (C) pickle.dump(data_object,file_object) (D) pickle.dump(file_object,data_object)	**1**
11.	State True or False: "The keys of a dictionary must be of immutable types."	**1**
12.	What will be the output of the following code? v = 50 def Change(n): global v v, n = n, v print(v, n, sep = "#", end = "@") Change(20) print(v) (A) 20#50@20 (B) 50@20#50 (C) 50#50#50 (D) 20@50#20	**1**

13.	Which SQL command can decrease cardinality of a relation?	**1**
14.	Which aggregate function in SQL displays the number of values in the specified column ignoring the NULL values? (A) len( ) (B) count( ) (C) number( ) (D) num( )	**1**
15.	In MySQL, Which type of value should not be enclosed within quotation marks? (A) Date (B) Varchar (C) Float (D) Char	**1**
16.	State True or False: If table A has 6 rows and 3 columns, and a table B has 5 rows and 2 columns, the Cartesian product of A and B will have 30 rows and 5 columns.	**1**
17.	Which device is primarily used to amplify and regenerate signals in a network, allowing data to travel longer distances? (A) Gateway (B) Repeater (C) Router (D) Switch	**1**
18.	_____ is used for point-to-point communication or unicast communication such as radar and satellite. (A) Infrared (B) Bluetooth (C) Microwaves (D)Radio waves	**1**
19.	In _____ switching technique, data is divided into chunks of packets and travels through different paths and finally reach the destination.	**1**
	Q.20 and Q.21 are **Assertion(A)** and **Reason(R)** 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	
20.	**Assertion (A):** The return statement in a Python function is optional. **Reason (R):** If no return statement is used, the function returns None by default.	**1**
21.	**Assertion (A):** In SQL, the GROUP BY clause is used to arrange identical data into groups. **Reason (R):** The GROUP BY clause is mandatory when using aggregate functions like SUM( ) or AVG( ).	**1**
	SECTION-B (7 x 2=14 Marks)	
22.	Write difference between mutable and immutable data types? Write name of any two mutable data types.	**2**
23.	i. Which operator in python has right to left associativity? ii. Evaluate the following expression and write output: >>> 5//4**3%6+2	**2**
24.	Consider T=(10,20,30) and L=[60,50,40] and answer the question I and II. **(I) A.** Write command(s) to add tuple **T** in list **L**.	**2**

	OR B. Write command to find and delete element 20 from tuple **T** **(II) A.** Write command to add **50** in **L** at position **2**. **OR** B. Write command to delete the variable **T**.	
25.	Look at the following python code and find the possible output(s) from the options (i) to (iv) following it. Also, write the highest and lowest values that can be pointed by label **p**. `import random` `text = "NewYear"` `p=random.randint(0,2)` `q=random.randint(1,2)` `for j in range(0, p+q):` `  print(text[j], end='@')` **i.** e@w@Y@ **ii.** N@ **iii.** N@e@w@Y@ **iv.** N@e@w@Y@e@	**2**
26.	The code provided below is intended to check a number whether it is perfect number or not. However, there are **syntax and logical errors** in the code. Rewrite the code in python after removing all error(s). Underline each correction done in the code. `def Perfect_Number( ):` `  n=input(int("Enter a number: "))` `  sum=0` `  for i in range(1,n):` `  if n%i==0:` `  sum=+i` `  if N==sum:` `  print("Perfect number")` `  else:` `  print("Not a perfect number")` `Perfect_Number( ):`	**2**
27.	**i.** A. What constraint should be applied on a table column so that **NULL** is not allowed in that column, but duplicate values are allowed. **OR** B. Categorize the following commands as DDL and DML: INSERT, UPDATE, ALTER, DROP **ii.** A. Write the SQL command to change the size of the attribute **Salary** in the table **Employee** to allow the maximum value of 99999.99 to be stored in it. **OR** B. Write an SQL command to change a table name from **Employee** to **Emp**.	**2**
28.	Write two points of difference between **XML** and **HTML**. **OR**	**2**

	Write two points of difference between **Circuit Switching** and **Packet switching**.	
	SECTION-C (3 x 3 = 9 Marks)	
29.	Mr. Ashutosh has written a poem in a text file named as **"POEM.TXT".** Now he wants to count number of lines which start with the character 'W' (including uppercase and lowercase) in the text file "POEM.TXT". Help him to write a user defined function **CountLineW( )** in python to solve the problem. Example: If the "POEM.TXT" contents are as follows: A tiny ant goes marching by, With little legs and wings to fly. Carrying crumbs, oh so small, Working together, one and all. The output of the function should be: No. of lines start with character W are: 2 **OR** Write a function **CountYouMe( )** in python which reads the content of a text file "BIOGRAPHY.TXT" and counts the words 'You' and 'Me' separately. (Not case sensitive). *Example:* If the contents in "BIOGRAPHY.TXT" are as follows: You are a hero for me you gifted a book to me which helped me a lot to learn the programming It gave me a chance to develop business software The output of the function should be: Count of You in file: 2 Count of Me in file: 4	**3**
30.	You have a stack named **MovieStack** that contains records of movies. Each movie record is represented as a list containing **movie_title**, **director_name**, and **release_year**. Write the following user-defined functions in Python to perform the specified operations on the stack **MovieStack**: i. **add_movie(MovieStack, new_movie):** This function takes the stack MovieStack and a new movie record new_movie as arguments and pushes the new movie record onto the stack and display the stack. ii. **remove_movie(MovieStack)**: This function removes the topmost movie record from the stack and returns it. If the stack is already empty, the function should display "Underflow". iii. **view_top(MovieStack)**: This function displays the topmost element of the stack without deleting it. If the stack is empty, the function should display 'None'. **OR** Stationary_Item is a dictionary containing the details of stationary items.	**3**

	Write a user defined function **PUSH_DATA(Stationery_Item),** to push the name of those items into the stack named as **PriceStack** which have price **more than 75** and display stack. Also display the number of elements pushed into the stack. For example: If the dictionary contains the following data: Stationery_Item={"Pen":106,"Pencil":59,"Notebook":80,"Eraser":25} The stack should contain ['Pen', 'Notebook'] The output should be: Number of elements in stack: 2	
31.	Predict the output of the following code: d={ } V="programs" for x in V: if x in d.keys( ): d[x]=d[x]+1 else: d[x]=1 print(d) **OR** Predict the output of the following code: V="interpreter" L=list(V) L1="" for x in L: if x in ['e','r']: L1=L1+x print(L1)	**3**
	SECTION-D (4 x 4 = 16 Marks)	
32.	Consider a table **TRAIN** as given below:	**4**

TNo	TName	Train_Type	Source	Destination	Fare
12001	Rajdhani Express	Superfast	Delhi	Mumbai	3000
12625	Shatabdi Express	Superfast	Delhi	NULL	1200
12501	North East Express	Express	Guwahati	Delhi	1500
16159	Kanyakumari Exp	Express	Kanyakumari	Delhi	2500
20814	Jodhpur-Puri Exp	Express	Jodhpur	Puri	2800
19412	Sabarmati Express	Express	Sabarmati	NULL	1100

A) Write the following queries:

i. Display type of train and total fare of each type of train.

ii. Display Train Number, Train Name and Fare of those trains whose name starts with the alphabet letter **'S'**.

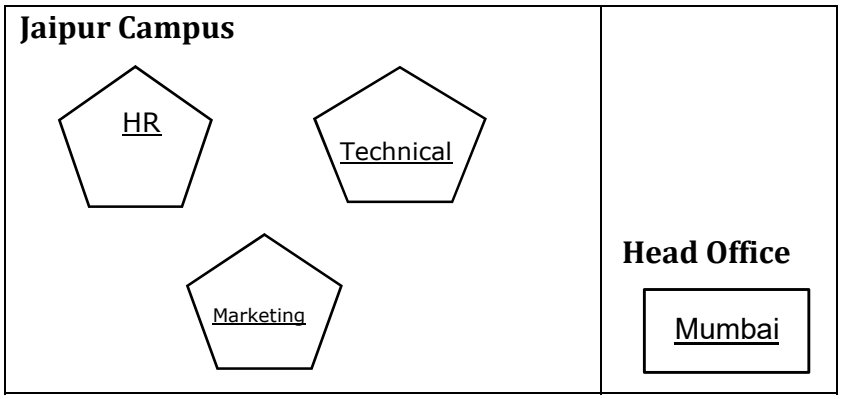
iii. Display train number, train name and source of those trains whose destination is **NULL**.

	iv. Count the number of trains which have train type as superfast and source is '**Delhi**'. **OR** **B) Write the output:** **i.** Select TNAME, Fare from TRAIN Where Fare<2000 order by TName; **ii.** Select avg(fare) from TRAIN group by Train_Type; **iii.** Select TName, Destination, Fare from TRAIN where fare <>2500 and destination IS NULL; **iv.** Select min(fare) from TRAIN where Train_Type= "Express";	
33.	Ms. Pallavi is a python programmer working in a software company. She has to develop a simple inventory management system of all books in a library. She has created a csv file named **bookdata.csv**, to store the details of books. The structure of **bookdata.csv** is: **[Book_Number, Book_Name, Author, Price]** Ms. Pallavi wants to write a Program in Python that defines and calls the following user defined functions: **i.** **Insert_Books( ) –** To accept and add data of **n** number of books to a CSV file **'bookdata.csv'.** **ii.** **Display( ) –** To show the record of those books which have the price more than 400 in the CSV file named **'bookdata.csv'**.	**4**
34.	Ms. Nishi has been entrusted with the bank Database. She needs to access some information from **LOAN** and **BORROWER** tables for a survey analysis. Help her to extract the following information by writing the desired SQL queries as mentioned below. **Table: LOAN** **Table : BORROWER** **i.** To display customer name and branch name of those customers who have taken loan from Delhi branch. **ii.** To display loan number, customer name and amount of those customers who have taken loan more than 40000.	**4**

loan_number	branch_name	Amount
L123	Nagpur	45000
L456	Pune	60000
L347	Delhi	80000
L987	Delhi	25000
L901	Pune	45000

customer_name	loan_number
Ajit Das	L456
Rohan Yadav	L901
Suman Verma	L123
Ayesha Tiwari	L987
Saurav	L347

	iii. To display branch name and average amount of that branch which has given average loan amount more than 50000. **iv.** A. To display customer name and amount in descending order of amount. **OR** B. What will be degree of resultant table after performing natural join of these two tables.	
35.	Consider a database named 'DB' containing a table named '**Vehicle**' with the following structure **Field** | **Type** Model | char(10) Make_year | Int(4) Qty | Int(3) Price | Number(8,2) Write the following Python functions to perform the following operation as mentioned: **i. Add_Vehicle( )** - which takes input of data and store it to the table **ii. Search_vehicle( ) –** which can search a model given by user and show it on screen * Assume the following for Python – Database connectivity: **Host:** localhost, **User:** root, **Password:** work	**4**
	SECTION-E (2 X 5 = 10 Marks)	
36.	Mr. Ashok is working on a Toy Shop project to manage toys records using Python. The toys data is stored in a binary file named **TOYDATA.DAT**. The binary file **TOYDATA.DAT** contains each record in given format: {"Toy_ID": T_ID, "TName":toy_name, "Price":price} Where • **Toy_ID**: Toy ID (string) • **TName**: Toy name (string) • **Price**: Price of toy (integer) You as a programmer, help him to write following python functions: **i. ADD_Data( )** : To write **n** records in binary file TOYDATA.DAT by taking the values for each record from user. **ii. SHOW_Data( )** : Read all records from binary file and display them. **iii. Remove_Toy( )** : that deletes the record of a toy in the file **TOYDATA.DAT** based on the **Toy ID** provided by the user. If the Toy ID does not exist in the file, display an appropriate message.	**5**
37.	**"MyTech Services"** is planning to set up its India campus at Jaipur with its Head Office at Mumbai. The Jaipur campus has 3-main blocks-**HR, Technical** and **Marketing**. You as a network expert have to suggest the best network related solutions for their problems raised in **(i)** to **(v)**.	**5**



Distance between various building blocks:

HR BLOCK to TECHNICAL BLOCK	45 m
HR BLOCK to MARKETING BLOCK	98 m
TECHNICAL BLOCK to MARKETING BLOCK	107 m
Head Office to JAIPUR Campus	1275 KM

Number of computers in each Block:

HR BLOCK	10
TECHNICAL BLOCK	105
MARKETING BLOCK	45

i. Suggest the most appropriate location of the server inside the JAIPUR campus (out of the 3 blocks), to get the best connectivity for maximum number of computers. Justify your answer.

ii. Which among the following devices will you suggest to be procured by the company for connecting all the computers within each of their offices?
- Switch/Hub
- Modem
- Bridge

iii. Suggest network type (out of LAN, MAN, WAN) for connecting each of the following set of their offices:
   a. HR and Marketing Block
   b. Head Office and Jaipur office

iv. Which of the following communication medium, you will suggest to be procured by the company for connecting their local offices in Jaipur for very effective and fast communication?
- Telephone cable
- Optical fiber
- Ethernet cable

v. A. In JAIPUR Campus, in between which offices repeater should be installed? Justify the answer.

OR

B. Suggest and draw the cable layout to efficiently connect various blocks within the JAIPUR campus for connecting the computers.

KENDRIYA VIDYALAYA SANGATHAN
UNSOLVED SAMPLE QUESTION PAPER-2

Subject: Computer Science (Theory) **Class: XII**
Time Allowed: 3:00 Hours **Max. Marks - 70**

General Instructions:

- This question paper contains 37 questions.
- All questions are compulsory. However, internal choices have been provided in some questions. Attempt only one of the choices in such questions
- The paper is divided into 5 Sections- A, B, C, D and E.
- Section A consists of 21 questions (1 to 21). Each question carries 1 Mark.
- Section B consists of 7 questions (22 to 28). Each question carries 2 Marks.
- Section C consists of 3 questions (29 to 31). Each question carries 3 Marks.
- Section D consists of 4 questions (32 to 35). Each question carries 4 Marks.
- Section E consists of 2 questions (36 to 37). Each question carries 5 Marks.
- All programming questions are to be answered using Python Language only.
- In case of MCQ, text of the correct answer should also be written.

Q. No.	SECTION-A (21 x 1 = 21 Marks)	Marks
1.	State True or False: "A dictionary is mutable data type. Both key and value are also mutable."	**1**
2.	Identify the output of the following code snippet: str = "The ball is in your court" str=str.replace('ou','*') print(str) (A) The ball is in y*r court (B) The ball is in y*r c*rt (C) The ball is in y**r c**rt (D) The ball is in your court	**1**
3.	What will be output of the following statement: print(3-4//5+3**2//3) (A) 2 (B) 6 (C) 6.0 (D) 4	**1**
4.	What will be the output of the given expression? v="red colour" print(v.strip("er")) (A) "d colour" (B) "ed colou" (C) "red colour" (D) "d colou"	**1**
5.	Identify the invalid python statement from the following: (A) k=(90) (B) k=tuple( ) (C) k=(90,) (D) None of these	**1**
6.	Select the correct output of the following code: s="cutting corners" print(s.partition("s")) (A) ('cutting corner', 's', '') (B) ('cutting corner', '', '') (C) ('cutting corner', 's') (D) ('cutting corners', '', '')	**1**

7.	Which of the following operator evaluates to False if the variable on either side of the operator points towards the same memory location and False otherwise? (A) is (B) is not (C) and (D) or	**1**
8.	Identify the output of the following code snippet: >>>H=[45, 60, ["news", "paper"], [50,20]] >>>H[-1:1:-1][0] (A) [“news”, “paper”] (B) [[“news”, “paper”]] (C) [50,20] (D) [[50,20]]	**1**
9.	A relation in a database can have ____ number of primary key(s)? (A) 1 (B) 2 (C) 3 (D) 4	**1**
10.	Write the missing statement to complete the following code: f = open("Essay.txt", "r") data = f.read(80) cur_pos = _____ # Get the current position of the file pointer print(“Current position is: ”, cur_pos) f.close( )	**1**
11.	Write the name of block / command(s) that can be used to handle the error/exception in Python.	**1**
12.	What will be the output of the following code? p=6 def Demo( ): global p p**=2 print(p, end='+') Demo( ) print(p, end='@') (A) 6+36@ (B) 72@ (C) 36+36@ (D) 36+6@	**1**
13.	Which type of command is used to delete the structure of the relation? (A) DDL (B) DML (C) Select (D) Cannot delete structure	**1**
14.	What will be the output of the query: SELECT Ename FROM Employee WHERE Ename like “%an%”; (A) Display name of employees whose name ends with ‘an’. (B) Display details of employees whose name ends with ‘an’. (C) Display name of employees whose name has ‘an’ anywhere in name. (D) Display name of employees whose name starts with ‘an’.	**1**
15.	A result set is extracted from the database using a cursor object by giving the following statement: **records=mycursor.fetchall( )** What will be the data type of **records,** after the execution of above statement?	**1**

	(A) tuple (B) string (C) dictionary (D) list	
16.	The degree and cardinality of a table named **SONG** are **2** and **4**, respectively. The degree and cardinality of another table named SINGER are **3** and **5**, respectively. There is one common field in both tables. After performing the Cartesian product of both tables, what will be the new degree and cardinality of the resultant table? (A) 4 and 20 (B) 5 and 20 (C) 5 and 9 (D) 4 and 9	**1**
17.	Which of the following protocol helps in e-mail services? (A) FTP (B) PPP (C) SMTP (D) VoIP	**1**
18.	Which among the following network devices, is used to connect two dissimilar networks: (A) Modem (B) Gateway (C) Switch (D) Repeater	**1**
19.	Which network device converts digital data from a computer into analog signals for transmission over phone lines?	**1**
	Q.20 and Q.21 are **Assertion(A)** and **Reason(R)** 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	
20.	**Assertion(A):** In a relation of RDBMS, redundancy can be reduced. **Reasoning (R):** This can be done with the help of join operations in between relation.	**1**
21.	**Assertion (A):** A Python function can return multiple values using a tuple. **Reason (R):** In Python, when a function returns multiple values, they are automatically packed into a tuple and returned as a single entity.	**1**
	SECTION-B (7 x 2=14 Marks)	
22.	Define dynamic data typing in python. Write an example to illustrate your answer.	**2**
23.	Write one example of each of the following in Python: i. Logical Error ii. Data Type casting	**2**
24.	If **str1= "foundation stone"** and **str2= "strong base",** then Write the Python statements for each of the following tasks using **Built-in functions/methods** only: **i.** A. To count the occurrence of the letter **'n'** in **str1**. **OR** B. To convert the string **str2** in a list and each word of the string should become the element of the list.	**2**

	ii. A. To convert the string **str2** in capital letters. **OR** B. To convert the first character of each word to uppercase of **str1**.	
25.	Identify the correct output(s) of the following code. Also write the minimum and the maximum possible values of the variable b. import random a="Wisdom" b=random.randint(1,6) for i in range(0,b,2): print(a[i],end='#') (A) W# (B) W#i# (C) W#s# (D) W#i#s#	**2**
26.	The code given below accepts N as an integer argument and returns the sum of all integers from 1 to N. Observe the following code carefully and rewrite if after removing all syntax and logical errors. Underline all the corrections made. def Sum(N) for I in range(N): S=S+I return S print(Sum(10)	**2**
27.	**(I)** **A)** Benjamin a database administrator created a table with few columns. He wants to stop duplicating the data in the table. Suggest how he can do so. **OR** **B)** Consider two tables student (rno, name, class) and marks (rno, mrk_obt, percent). You as a database administrator how will your stop redundancy of data in the table students and how the tables students and marks can be connected with each other . **(II)** A) Write an SQL command to change the data type of a column named price to number (10,2) in a table named stationary **OR** B) Write an SQL command to change the values of all the rows of the column price of table stationary to Null	**2**
28.	**A)** List one advantage and one disadvantage of star topology. **OR** **B)** Expand the term SMTP. What is the use of SMTP?	**2**
	SECTION-C (3 x 3 = 9 Marks)	
29.	Write a function **Show_Words( )** in python to read the content of a text file	**3**

	"NOTES.TXT" and display those words in capital letters which start with any vowel. Example, if the file contains: "Comparing apples to oranges" Then the function should display the output as: APPLES ORANGES **OR** Write a function **count_my( )** in python to read the text file **"DATA.TXT"** and count the number of times the word **"my"** (Including uppercase and lowercase) occurs in the file. For example, if the file **"DATA.TXT"** contains: "This is MY website. I have displayed my preferences in the CHOICE section" The **count_my( )** function should display the output as : The word my occurs :2 times	
30.	**A.** Ms. Aarti has a stack named **ToysStack** that contains records of toys. Each toy record is represented as a list containing toy_name, brand_name and year_of_manufacture. You as a python programmer help her to write the following user-defined functions in Python to perform the specified operations on the stack **ToysStack**: i. **PUSH_TOY(ToysStack, new_toy):** This function takes the stack ToysStack and a new toy record new_toy as arguments and pushes the new toy record onto the stack. ii. **POP_TOY(ToysStack):** This function pops the topmost toy record from the stack and returns it. If the stack is already empty, the function should display "Underflow". iii. **PEAK(ToysStack):** This function displays the topmost element of the stack without deleting it. If the stack is empty, the function should display 'None'. **OR** **B.** Consider a list named Space which contains the terms related to space. Write the following user defined functions in python and perform the specified operations on a stack named **MyStack**. i. **Push_Term( ):** It checks every term from the list Space and pushes all such terms which have 5 or more characters into the stack, MyStack. ii. **Pop_Term( ):** It pops the terms from stack, MyStack and displays them. The function should also display **"Stack is Empty"** when there are no more elements left in the stack.	**3**
31.	Predict the output of the following code k=4 def DataWork(a, b=5): global k	**3**

```
   k=b+a**2
   print(k, end="@")
p=10
q=15
DataWork(p,q)
DataWork(b=3, a=8)
```

OR

Write the output of the code given below:

```
a =30
def call (x) :
    global a
    if a%2==0:
        x+=a
        x=5
    else:
        x-=a
    return x
print(call(35),end="#")
print(call(40),end= "@")
```

SECTION-D (4 x 4 = 16 Marks)

32. Consider a table **Food** as given below**:** **4**

Food_ID	Food_name	Category	Quantity	Expiry_date
201	Apple	Fruit	100	2026-01-31
202	Banana	Fruit	150	2026-01-25
206	Carrot	Vegetable	200	2026-02-17
209	Milk	Dairy	50	2026-02-22
204	Butter	Dairy	100	2026-02-20
203	Rice	Grain	60	2026-09-20
210	Bread	Bakery	120	2026-01-23
208	Tomato	Vegetable	110	2026-01-22
207	Chicken	Meat	20	2026-03-15
205	Orange	Fruit	80	2026-02-05

A) Write the SQL queries:

i. Display the Category and average quantity of each category of food.

ii. Display Food_ID, Food Name and Expiry date of those foods which has alphabet **'g'** as second last character in food name.

iii. Display Food name, Category and Quantity of those foods whose food category is not NULL.

iv. Display name of each food category in alphabetic order without repetition of category.

OR

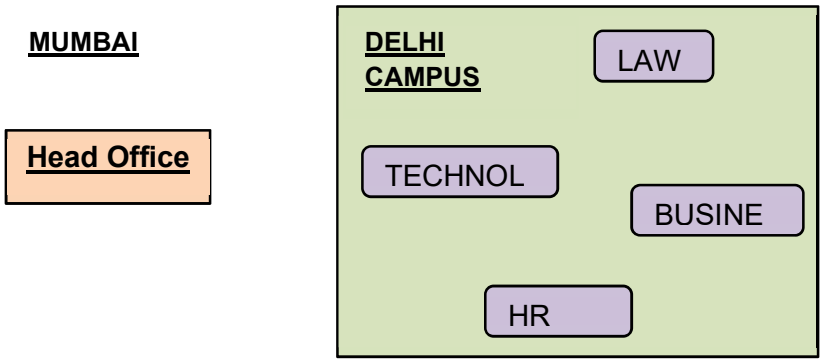
B) Write the output:

i. Select Food_ID, Food_Name, Quantity from FOOD Where Expiry_date< "2026-01-31" order by Quantity desc;

	ii. Select Category, avg(Quantity) from Food group by Category; **iii.** Select Food_Name, Expiry_date from Food where Category= "Fruit" and Quantity<110; **iv.** Select min(Expiry_date) from Food where Category= "Vegetable";	
33.	A csv file **"candidates.csv"** contains the data collected from an online application form for selection of candidates for different posts, with the following data • Candidate Name • Qualification • Percent_XII • Percent_Qualification E.g. ['Kartik, 'MCA', 80, 76] Write the following Python functions to perform the specified operations on this file: **a) READ( )** function which can read all the data from the file and display only records with Percent_XII more than 75. **b) IDENTIFY( )** function which can find and print the number of such records which are having Percent_XII not more than 75.	**4**
34.	Aman has been entrusted with the management of Law University Database. He needs to access some information from **FACULTY** and **COURSES** tables for a survey analysis. Help him extract the following information by writing the desired SQL queries as mentioned below. **Table: FACULTY** **F_ID** | **FName** | **LName** | **Hire_Date** | **Salary** 102 | Amit | Mishra | 12-10-1998 | 12000 103 | Nitin | Vyas | 24-12-1994 | 8000 104 | Rakshit | Soni | 18-5-2001 | 14000 105 | Rashmi | Malhotra | 11-9-2004 | 11000 106 | Sulekha | Srivastava | 5-6-2006 | 10000 **Table: COURSES** **C_ID** | **F_ID** | **CName** | **Fees** C21 | 102 | Grid Computing | 40000 C22 | 106 | System Design | 16000 C23 | 104 | Computer Security | 8000 C24 | 106 | Human Biology | 15000 C25 | 102 | Computer Network | 20000 C26 | 105 | Visual Basic | 6000 **i.** To display complete details (from both the tables) of those Faculties whose salary is less than 12000.	**4**

	ii. To display the details of courses whose fees is in the range of 20000 to 50000 (both values included). **iii.** To increase the fees of all courses by 500 which have "Computer" in their Course names. **iv.** **(A)** To display names (FName and LName) of faculty taking System Design. **OR** **(B)** To display the Cartesian Product of these two tables.	
35.	Mr. Sanjeev wants to create a table named **SHOP** in the **MARKET** database, which should have the following structure: **Field** | **Type** | **Remarks** Shop_Number | int | Primary Key Shop_Name | varchar(50) | NOT NULL Start_Date | Date | Category | Varchar(30) | TurnOver | float(11,2) | Write the following Python function to perform the specified operation: ● **Shop_Details( )** : To create a table **SHOP** as per details given above and after creation of the table assume that you have inserted 20 records in it. Now, display the details of those shops which have the Category as "**Footwear**". Consider the following for Python-Database connectivity: host: localhost, user: root, password: dbwork	**4**
	SECTION-E (2 X 5 = 10 Marks)	
36.	Mr. Mohit is working on a school project to manage student records using Python. The student data is stored in a binary file named **STUDENT.DAT**. The binary file **STUDENT.DAT** contains each record in given format: {"Admn_No":admn, "SName":name, "Marks":marks} Where • **Admn_No**: Admission Number (integer) • **SName**: Student Name (string) • **Marks**: Marks (integer) You as a programmer, help him to write following python functions: **i.** **ADD_Data( )** : To write 7 records in binary file STUDENT.DAT by taking the values for each record from user. **ii.** **Display_Data( )** : Read all records from binary file and display them. **iii.** **Modify_Marks( )** : that updates the marks of a student in the file **STUDENT.DAT** based on the admission number provided by the user. If the admission number does not exist in the file, display an appropriate message.	**5**
37.	"Deccan Tech Services" is planning to set up its India campus at Delhi with its Head Office at Mumbai. The Delhi campus has 4-main blocks-BUSINESS, TECHNOLOGY,	**5**

LAW and HR. You as a network expert have to suggest the best network related solutions for their problems raised in (i) to (v).



Distance between various building blocks:

LAW BLOCK to BUSINESS BLOCK	40 m
LAW BLOCK to TECHNOLOGY BLOCK	80 m
LAW BLOCK to HR BLOCK	105 m
BUSINESS BLOCK to TECHNOLOGY BLOCK	30 m
BUSINESS BLOCK to HR BLOCK	35 m
TECHNOLOGY BLOCK to HR BLOCK	15 m
Head Office to Delhi Campus	1485 KM

Number of computers in each of the Blocks:

LAW BLOCK	15
TECHNOLOGY BLOCK	45
HR BLOCK	100
BUSINESS BLOCK	30

i. Suggest the most appropriate location of the server inside the DELHI campus (out of the 4 blocks), to get the best connectivity for maximum number of computers. Justify your answer.

ii. Suggest and draw the cable layout to efficiently connect various blocks within the DELHI campus for connecting the computers.

iii. Which hardware device will you suggest to be procured by the company to be installed to protect and control the internet uses within the campus?

iv. Suggest network type (out of LAN, MAN, WAN) for connecting each of the following set of their offices:
- Law & Technology Department
- Head Office and Delhi Office

v.

A. Company wants to connect its local offices situated in Delhi. Company can compromise with speed but cost of connectivity should be economic. Which of the following communication medium, you will

	suggest to be procured by the company: a) Telephone cable b) Twisted Pair Cable c) Optical Fiber Cable **OR** B. Suggest the placement of the following devices with appropriate reasons: a) Switch / Hub b) Repeater	

KENDRIYA VIDYALAYA SANGATHAN
UNSOLVED SAMPLE QUESTION PAPER-3

Subject: Computer Science (Theory) **Class: XII**
Time Allowed: 3:00 Hours **Max. Marks - 70**

General Instructions:

- This question paper contains 37 questions.
- All questions are compulsory. However, internal choices have been provided in some questions. Attempt only one of the choices in such questions
- The paper is divided into 5 Sections- A, B, C, D and E.
- Section A consists of 21 questions (1 to 21). Each question carries 1 Mark.
- Section B consists of 7 questions (22 to 28). Each question carries 2 Marks.
- Section C consists of 3 questions (29 to 31). Each question carries 3 Marks.
- Section D consists of 4 questions (32 to 35). Each question carries 4 Marks.
- Section E consists of 2 questions (36 to 37). Each question carries 5 Marks.
- All programming questions are to be answered using Python Language only.
- In case of MCQ, text of the correct answer should also be written.

Q. No.	SECTION-A (21 x 1 = 21 Marks)	Marks
1.	State True or False: "The **is** operator checks the identity of two variables, not the equality of their values."	**1**
2.	Which of the following will return False: (A) not (True and False) (B) True or False (C) not (True or False) (D) not (False and False)	**1**
3.	Identify the correct output of the following code snippet: s= "Student@2025" print(s.index("E") (A) 0 (B) 4 (C) -1 (D) ValueError	**1**
4.	Which of the following is a valid keyword in python? (A) true (B) return (C) non_local (D) none	**1**
5.	What will be correct output for the following: R = (4.5, 5), (5,4), (8.2,7), (8,7,3) print(max(R)) (A) Error (B) (8.2, 7) (C) (8, 7, 3) (D) (4.5, 5, 8.2, 8)	**1**
6.	Select the correct output of the following code: person={"name":"Noyal", "stream":"science"} R="KV".join(person) print(R) (A) NoyalKVscience (B) nameKVstream (C) KVNoyalscience (D) nameKVstreamKV	**1**
7.	If **"dict"** is a dictionary as defined below, then which of the following statements	**1**

	will raise an exception? dict = {'Rose': 10, 'Lily': 20, 'Sunflower': 30} (A) dict.get('Sunflower') (B) print(dict['Rose', 'Lily']) (C) dict['Rose']=40 (D) print(str(dict))	
8.	What does the list.append(x) method do in Python? (A) Inserts the element x at the beginning of the list (B) Inserts the element x at the end of the list (C) Inserts the element x in between two elements in the list (D) It merges the two lists and creates another list	**1**
9.	The primary key is selected from the set of _____ . (A) composite keys (B) alternate keys (C) candidate keys (D) foreign keys	**1**
10.	What is the value of 'p' and how many characters will be there in the variable 'data' in the following statement with open ("lists.txt","r",encoding="utf-8") as F: data = F.read(100) p=F.seek(10,0) print(p) (A) 10, 100 (B) 100, 10 (C) 10, 110 (D) 110, 10	**1**
11.	State True or False: We can concatenate two tuples using the **+** operator in Python.	**1**
12.	What will be the output of the following code? def add( ): c=1 d=1 while(c<9): c=c+2 d=d*c print(d, end="$") return c print(add( ), end="*") (A) 945$9* (B) 945$9 (C) 9*945$ (D) 9$945*	**1**
13.	Fill in the blanks: _____ clause is used in SQL query to display data in a sorted form with respect to a specified column. (A) WHERE (B) ORDER BY (C) HAVING (D) DISTINCT	**1**
14.	Which function in SQL is used to count the total number of records regardless of NULL from table in a database? (A) sum(*) (B) total(*) (C) count(*) (D) count( )	**1**
15.	Which of the following statements about the CHAR and VARCHAR datatypes in SQL is False?	**1**

	(A) CHAR is a fixed-length datatype, and it pads extra spaces to match the specified length. (B) VARCHAR is a variable-length datatype and does not pad extra spaces. (C) The maximum length of a VARCHAR column is always less than that of a CHAR column. (D) CHAR is generally used for storing data of a known, fixed length.	
16.	Which of the following is not an aggregate function? (A) max( ) (B) count( ) (C) sum( ) (D) upper( )	**1**
17.	Which of the following protocols is used for remote login: (A) VoIP (B) HTTP (C) IMAP (D) TELNET	**1**
18.	Which of the following options is the correct protocol used for phone calls over the internet? (A) PPP (B) FTP (C) HTTP (D) VoIP	**1**
19.	Expand the following terms: **i.** URL **ii.** IP	**1**
	Q.20 and Q.21 are **Assertion(A)** and **Reason(R)** 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	
20.	**Assertion (A):** We can retrieve records from more than one table in MYSQL. **Reason (R):** Foreign key is used to establish a relationship between two tables.	**1**
21.	**Assertion (A):** A function in Python can have any number of arguments. **Reasoning(R):** variable length parameter can be used to deal with such number of arguments.	**1**
Q. No	**SECTION-B (7 x 2=14 Marks)**	
22.	Your Vidyalaya decided to conduct Solo singing competition. CCA in charge wants to store the admission numbers of the participants. Help your CCA in charge in choosing the correct/suitable data structure (data type) in Python for the following. i. To store all the admission numbers of the registered candidates. May get changed any time till completion of registration process. ii. To store the admission numbers of all the winners which never gets changed.	**2**
23.	i. Define Operator Associativity. ii. Write the following operators in descending order (Higher precedence to lower precedence in order of operation) of their operator precedence: **or, **, and, +, *, ==**	**2**

24.	If M1=[60,25,30,……] and M2=[3,6,9,12, …….], then Write the Python statements for each of the following tasks using **Built-in functions/methods** only: **i.** **A.** To delete an element 25 from the list **M1**. **OR** **B.** Write a statement to add an element 85 in the list **M2** between the elements 9 and 12. **ii.** **A.** Write a statement to sort the elements of list **M1** in descending order. **OR** **B.** Write the statement to delete the last element of list **M2**.	**2**
25.	Look at the following python code and find the possible output(s) from the options (i) to (iv) following it. Also, write the highest and lowest values that can be pointed by label VALUE. import random for y in range(4): VALUE = random.randrange(4,11) + y print(VALUE, "#", end=" ") **i.** 6 # 7 # 12 # 13 # **ii.** 5 # 11 # 8 # 11 # **iii.** 4 # 7 # 12 # 14 # **iv.** 9 # 15 # 8 # 6 #	**2**
26.	The code provided below is intended to find the factorial of an integer number. However, there are **syntax and logical errors** in the code. Rewrite the code in python after removing all error(s). Underline each correction done in the code. num=int(input("Enter a number: ")) if num<0 print("Factorial not defined for negative numbers") Elif num==0: fact=1 print(fact) else: fact=1 for i in range(1, num): fact=*i print(fact)	**2**
27.	**i.** **A.** Which SQL command is used to save changes permanent in the database after executing data manipulation operations like INSERT, UPDATE or DELETE? **OR**	**2**

	B. What constraint should be applied on an attribute in a relation so that the values under that attribute must not have NULL value and duplicate values? **ii.** **A.** Write the SQL command to list the details (all column names, data type, size, constraint) of a table named as "MUSIC". **OR** **B.** Write an SQL command to remove a primary key which is applied on the attribute "Book_ID" of integer data type in existing table "BOOKS".	
28.	Write one advantage and one disadvantage of using Optical Fiber cable. **OR** Expand the term PPP. What is the use of PPP?	**2**
	SECTION-C (3 x 3 = 9 Marks)	
29.	Write a function **Show_Words( )** in python to read the content of a text file **"NOTES.TXT"** and display those words in capital letters which start with any vowel. Example, if the file contains: "Comparing apples to oranges" Then the function should display the output as: APPLES ORANGES **OR** Write a function **count_my( )** in python to read the text file **"DATA.TXT"** and count the number of times the word **"my"** (Including uppercase and lowercase) occurs in the file. For example, if the file **"DATA.TXT"** contains: "This is MY website. I have displayed my preferences in the CHOICE section" The **count_my( )** function should display the output as : The word my occurs :2 times	**3**
30.	You have a stack named **MovieStack** that contains records of movies. Each movie record is represented as a list containing **movie_title**, **director_name**, and **release_year**. Write the following user-defined functions in Python to perform the specified operations on the stack **MovieStack**: i. **add_movie(MovieStack, new_movie):** This function takes the stack MovieStack and a new movie record new_movie as arguments and pushes the new movie record onto the stack and display the stack. ii. **remove_movie(MovieStack)**: This function removes the topmost movie record from the stack and returns it. If the stack is already empty, the function should display "Underflow". iii. **view_top(MovieStack)**: This function displays the topmost element of the stack without deleting it. If the stack is empty, the function should display 'None'. **OR**	**3**

	Stationery_Item is a dictionary containing the details of stationary items. Write a user defined function **PUSH_DATA(Stationery_Item),** to push the name of those items into the stack named as **PriceStack** which have price **more than 75** and display stack. Also display the number of elements pushed into the stack. For example: If the dictionary contains the following data: Stationery_Item={"Pen":106,"Pencil":59,"Notebook":80,"Eraser":25} The stack should contain ['Pen', 'Notebook'] The output should be: Number of elements in stack: 2	
31.	Predict the output of the following code s="science" L=[1,3,4,8,7,18] d={} for i in range(len(L)//2): if i%2==0: d[L.pop( )]=s[i] else: d[L.pop( )]=i*2 for k,v in d.items( ): print(k,v,sep="-") **OR** Write the output of the code given below: def FindOutput(p, q=2, r=40): x=p**2*4 y=x+r print(x, "@", y) return y c=FindOutput(q=5, r=7, p=4) a,b=4,3 c=FindOutput(b,a,c) print(a,"@",b,"@",c)	**3**
Q. No	**SECTION-D (4 x 4 = 16 Marks)**	
32.	Consider the table ORDERS as given below:	**4**

O_Id	C_Name	Product	Quantity	Amount
1001	Jitendra	Laptop	NULL	12000
1002	Mustafa	Smartphone	2	10000
1003	Dhwani	Headphone	1	1500
1008	Sonali	Laptop	3	40000
1025	Jayra	Headphone	NULL	22000
1231	Varun	Laptop	5	90000

A) Write the following queries:

i To display the total Quantity for each Product, excluding Products with total Quantity less than 5.

ii To display the orders table sorted by total amount in descending order.

iii To display the distinct customer names from the Orders table.

iv Display the sum of amount of all the orders for which the quantity is null.

OR

B) Write the output:

i Select c_name, sum(quantity) as total_quantity from orders group by c_name;

ii Select * from orders where product like '%phone%';

iii Select o_id, c_name, product, quantity, price from orders where price between 1500 and 12000;

iv Select max(price) from orders;

33. Mr. Ankit is a python programmer working in a software company. He has to develop a simple inventory management system of all employees working in an educational institute. He has created a csv file named **record.csv**, to store the details of employees. The structure of **record.csv** is: **[employee_id, emp_name, salary]**

Mr. Ankit wants to write a Program in Python that defines and calls the following user defined functions: **4**

i. **ADD( ) –** To accept and add data of 6 employees to a CSV file **'record.csv'.**

ii. **COUNTR( ) –** To count the number of records present in the CSV file named **'record.csv'**.

34. Ms. Pooja has been entrusted with the online shopping Database. She needs to access some information from **CUSTOMER** and **ORDER** tables for a survey analysis. Help her to extract the following information by writing the desired SQL queries as mentioned below. **4**

Table: CUSTOMER

CUST_ID	Customer_name	Gender	City
C011	Hemali	F	Pune
C025	Gaurang	M	Mumbai
C031	Kiran	F	Nagpur
C045	Payal	F	Pune
C050	Manish	M	Mumbai

Table : ORDERS

ORDER_ID	Product_Name	Amount	CUST_ID
OD01	TV	75000	C031
OD02	Laptop	80000	C050
OD03	Mobile	25000	C011
OD04	Mobile	40000	C031
OD05	TV	30000	C045

i. To display customer name and city of those customers who have placed order for TV.

	ii. To display **C**ustomer name, Product name and amount of those orders which have the amount more than 60000. **iii.** To display Order ID, Product name and customer name of those orders which have been placed by any male customer. **iv.** **A.** To display customer name, city and amount in descending order of order amount. **OR** B. What will be cardinality of resultant table after performing cartesian product of CUSTOMER table and ORDERS table.	
35.	A table, named STATIONERY, in ITEMDB database, has the following structure: **Field** / **Type**: itemNo / int(11); itemName / varchar(15); price / float; qty / int(11) Write the following Python function to perform the specified operation: **AddAndDisplay( )** : To input details of an item and store it in the table STATIONERY. The function should then retrieve and display all records from the STATIONERY table where the Price is greater than 120. Assume the following for Python-Database connectivity: Host: localhost, User: root, Password: Pencil	**4**
Q. No	**SECTION-E (2 X 5 = 10 Marks)**	
36.	Surya is a manager working in a recruitment agency. He needs to manage the records of various candidates. For this, he wants the following information of each candidate to be stored: - - Candidate_ID – integer - Candidate_Name – string - Designation – string - Experience – float You, as a programmer of the company, have been assigned to do this job for Surya. i. Write a function to input the data of a candidate and append it in a binary file. ii. Write a function to update the data of candidates whose experience is more than 10 years and change their designation to "Senior Manager". iii. Write a function to read the data from the binary file and display the data of all those candidates who are not "Senior Manager".	**5**
37.	**"Tech_AI Services"** is planning to set up its India campus at Pune with its Head Office at Delhi. The Pune campus has 3-main blocks-**App Development, Generative AI** and **Web Designing**. You as a network expert have to suggest the best network related solutions for their problems raised in **(i)** to **(v)**. ***Distance between various building blocks:***	**5**

App Development to Generative AI BLOCK	90 m
App Development to Web Designing BLOCK	110 m
Generative AI to Web Designing BLOCK	40 m
Head Office to Pune Campus	1390 KM

Number of computers in each Block:

App Development BLOCK	50
Generative AI BLOCK	100
Web Designing BLOCK	60

i. Suggest the most appropriate location of the server inside the Pune campus (out of the 3 blocks), to get the best connectivity for maximum number of computers. Justify your answer.

ii. Suggest the placement of following devices in various blocks at Pune campus:
   a. Switch/Hub
   b. Repeater

iii. What would be your recommendation for enabling live visual communication between the Generative AI Block at Pune campus and Delhi Head Office from the following options:
   - Video Conferencing
   - Telephony
   - E-Mail
   - Instant Messaging

iv. Which of the following communication medium, you will suggest to be procured by the company for connecting their local offices in Pune for very effective and fast communication?
   - Telephone cable
   - Optical fiber
   - Ethernet cable

v.

A. What type of network (out of LAN, MAN, WAN, PAN) would be setup among the computers within the Pune campus.

OR

B. Suggest and draw the cable layout to efficiently connect various blocks within the Pune campus for connecting the computers.

**********END**********

REFERENCES

1. **NCERT Textbook for Computer Science – Class XII**
   National Council of Educational Research and Training (NCERT)
   https://ncert.nic.in

2. **CBSE Curriculum and Guidelines for Class XII Computer Science (083)**
   Central Board of Secondary Education
   https://cbseacademic.nic.in

3. **Python Official Documentation**
   Python Software Foundation
   https://docs.python.org/3/