ANSWERS

SECTION — A

OBJECTIVE TYPE QUESTIONS

1. **Option (a)** per%marks **1**
   Explanation: Python identifiers cannot carry special characters such as '%','/' , '*' etc.
2. **Option (b)** list.append(element) **1**
   Explanation: The append() function adds an element at the end of a list.
3. **Option (b)** lcomme T **1**
   Explanation: This is an example of string slicing . In the first case the stringg is sliced from position 2 to 5 and in the second case the string is sliced from position 5 to 8. Both these slices are concatenated.
4. **Option (b)** One block of except statement cannot handle multiple exceptions **1**
   Explanation: In python exception handling an except block can have code for multiple exceptions.
5. **Option (c)** Statement 3 **1**
   Explanation: In statement 3 a tuple is tried to be modified. Tuples are immutable structures. Hence this statement will have error.
6. **Option (d)** dump **1**
   Explanation: The dump() method of pickle module is used to write data to a binary file.
7. **Option (d)** dict_student.update(dict_marks) **1**
   Explanation: The update() method of a dictionary updates the contents of a dictionary with another. Since the keys here are different , the new contents will be added.
8. **Option (b)** mean() **1**
   Explanation: The mean() method belongs to statistics module and not math module.
9. **Option (c)** 13.5 **1**
   Explanation: The len() function returns the number of elements in the list L. So len(L)/2 returns 4/2=2.0. Len(0) returns 3. Therefor 3*2.0=6.0.
10. PPP – Point to Point Protocol
    VoIP - Voice Over Internet Protocol **1**
    Explanation: PPP stands for Point to point protocol, a TCP/IP protocol used to connect one computer

system with another.
VOIP : Voice over internet protocol is a protocol used for audio and video communication in tele and video conferencing applications.

11. **Option (b)** LIKE operator 1
Explanation: The Like operator is used to match patterns in SQL. It uses the % and _ characters for matching patterns.

12. **Option (d)** fetchone 1
Explanation: The fetchone() method fetches one record from the resultset, that is the data extracted from the back end database.

13. **Option (b)** r 1
Explanation: The 'r' mode opens a file in read mode. Hence if the file does not exist , an error is generated.

14. **Option (a)** file_object.seek(offset [, reference_ point]) 1
Explanation: The seek() function takes 2 parameters. The byte offset and the reference (beginning or eend or current position).

15. **Option (d)** Interlinking of collection of webpages is called Internet. 1
Explanation: Collection of interlinked web pages is a web site. Internet is a worldwide network of networks.

16. **Option (c)** TelNet 1
Explanation: The Telnet is a protocol that allows to resources in remote systems.

17. **Option (a)** Both A and R are true and R is the correct explanation for A 1
Explanation: Global variables are globally accessible in all functions. These variables must be declared outside all functions or declared using the keyword global.

18. **Option (c)** A is True but R is False 1
Explanation: A binary file stores data in binary record format The data in them can be handled using the pickle module functions.
These files cannot be read using text editors.

19. **Advantages:**
1) A dedicated communication channel increases the quality of communication.
2) Suitable for long continuous communication.
Disadvantages:
1) Resources are not utilized fully.
2) The time required to establish the physical link between the two stations is too long.
½ mark for each advantage and disadvantage 1

OR

Web browser
Purpose: Receives and displays web content.
Function: Initiates requests to web servers, and receives and displays content for users.
Web server
Purpose: Delivers web content to clients.
Function: Listens to incoming requests, processes them, and sends requested content to the client.
Name of Web browsers: Google Chrome, Mozilla Firefox
1 mark for any one correct difference and 1/2 mark for each two correct examples

20. num1, num2 = 10, 45 while num1 % num2 == 0: num1+= 20 num2+= 30 else: print('hello') 1
½ mark for while
½ mark for :
½ mark for correct indentation (inside the block of while) ½ mark for else

21. def dispBook(BOOKS):
    for key in BOOKS:
        if BOOKS[key][0] not in "AEIOUaeiou":
            print(BOOKS[key].upper())
BOOKS = {1:"Python",2:"Internet Fundamentals ",3:"Networking ",4:"Oracle sets",5:"Understanding HTML"} dispBook(BOOKS)
½ mark for for loop
1 mark for if condition
½ mark for display in upper case
Explanation: The code picks each key of the dictionary using the for loop and checks the 0th or start character for a consonant using the "in" operator and accordingly prints the value.

OR

def FindWord(STRING,SEARCH):
    return (STRING . count (SEARCH))
str = input('Enter String : ')
word = input('Enter word to search : ')
print('The word', word, 'occurs', FindWord(str,word), 'times')
½ mark for input
½ mark for print statement
1 mark for counting the word and returning the value

SECTION — B

22. 9$14$19$5$ 2
½ mark for 9$
½ mark for 14$
½ mark for 19$
½ mark for 5$
Explanation: The variables L and G are both global. The for loop adds the value of the global variable to the elements of the list.

23. i. del D['Mumbai'] 2
1 mark for correct answer
ii. print(S.split())
1 mark for correct answer

OR

my_str = "Computer Science"
alternate_chars = my_str[::2]
print(alternate_chars)
1.5 mark for logic of alternate characters
½ mark for printing alternate characters
Explanation: The del method takes input a key and removes the key:value pair from a dictionary.
The split function parses words from a string and returns a list of such words.

24. % (Percentage): 2
- Matches any sequence of characters (including empty sequence).
- Example: LIKE 'T%' matches all those strings starting with the letter 'T'. The string with just 1 character 'T' will also be considered.

_(Underscore):

- Matches a single character.
- Example: LIKE '_ _T' on the other hand will search for a three letter string, whose 3rd letter is 'T'. At first two places any two character can appear.

1 mark for one correct difference. 1/2 mark each for correct example of each.

OR

DROP is a DDL command in SQL and can be used to remove tables (or database).

Example: 'DROP TABLE STUDENT;' will remove the table STUDENT from the database.

DELETE is a DML command used to remove or delete rows/records from a table.

Example: 'DELETE FROM STUDENT WHERE PER < 33;' will remove all those records from the table STUDENT where the percentage is less than 33.

1 mark for one correct difference. 1/2 mark each for correct example of each.

25. • COUNT(*) returns the count of all rows in the table, whereas COUNT() is used with Column_Name passed as an argument and counts the number of non-NULL values in a column that is given as an argument. Hence the result may differ.
- The SQL command with COUNT(*) may have higher value as it count all rows in the table.

1 mark for suitable reason

1 mark for mentioning correct command

SECTION — C

26. (a) 2

CODE	NAME	TYPE	MNO	MNAME	ISSUE-DATE
L102	Easy Python	Pro-gram-ming	M101	SNEH SINHA	2022-10-13
F102	Untold Story	Fiction	M103	SARTHAK	2021-02-23
C101	Juman Ji	Thriller	M102	SARA KHAN	2022-06-12

Explanation: In a natural join all the columns from both tables are listed and the joining condition is an equality. Also the common field is not repeated.

1 mark for correct answeras a new class in rural areas.

(b)

(i) The order by clause is used to arrange records in ascending or descending order.

NAME	PROJECT
Satyansh	P04
Ranjan	P01
Muneera	P01
Alex	P02
Akhtar	P04

½ mark for correct output

(ii) Name like "A%" means names starting with "A"

NAME	SALARY
Akhtar	125000
Alex	75000

½ mark for correct output

(iii) Between clause is used to match a range of values.

NAME	DOJ
Ranjan	2015-01-21
Akhtar	2015-02-01
Muneera	2018-08-19

½ mark for correct output

(iv)

Eid	Name	DOB	DOJ	Salary	Project
E01	Rannja	1990-07-12	2015-01-21	150000	P01
E03	Muneera	1996-11-15	2018-08-19	135000	P01

½ mark for correct output

27. (a)

(i) Group by clause is used to group records on common values of a column and get aggregate results.

FID	MIN(FEES)	MAX(FEES)
F01	12000	40000
F04	15000	17000
F03	8000	8000
F05	NULL	NULL

½ mark for correct answer

(ii) The avg() function finds the average of a set of values.

'%a' means fnames ending with 'a'.

AVG(SALARY)
29500

½ mark for correct answer

(iii) The query brings corresponding records from the two tables by joining them on the foreign key relationship FID.

FNAME	CNAME
Neha	Python
Neha	Computer Network

½ mark for correct answer

(iv) The query joins the two tables on FID and brings corresponding data of FNAME, CNAME and FEES.

FNAME	CNAME	FEES
Anishma	Grid Computing	40000
Neha	Python	17000

½ mark for correct answer

(b)

DESC or DESCRIBE command

1 mark for correct answer

28.
```
def Count():
    F=open('Gratitude.txt')
    T=F.readlines()
    X=1
    for i in T:
        print('Line',X,':',i.count('e'))
        X=X+1
    F.close()
Count()
```

1

½ mark for function header

½ mark for opening and closing the file

½ mark for reading lines

½ mark for loop

½ mark for count function/or any other alternate correct statement(s)

½ mark for counter

In the code the readlines() function reads each line from the file and the count() function counts the number of instances of 'e'

OR

```
def Start_with_I():
    F=open('Gratitude.txt')
    T=F.readlines()
    for i in T:
        if i[0] in 'Ii':
            print(i,end='')
    F.close()
Start_with_I()
```

½ mark for function header

½ mark for opening and closing the file

½ mark for reading lines

½ mark for loop

½ mark for if condition

½ mark for print statment

In the code the readlines() function reads each line from the file and the for loop picks the 1st letters of each line.

29. (i) Candidate Keys : ADMNO, ROLLNO 3

1 mark for correctly writing both names of candidate keys. OR ½ mark for specifying any one candidate key correctly.

All the columns of a table that are eligible for becoming the primary key are called candidate keys.

Number of columns is called Degree

Number of rows is called Cardinality

(ii) Degree-8, Cardinality=4

½ mark for degree and ½ mark for cardinality

(iii) Update result set SEM2=SEM2+.03*SEM2 where SEM2 between 70 and 100;

½ mark for writing Update result set part correctly

½ mark for writing SEM2=SEM2+.03*SEM2 where SEM2 between 70 and 100; correctly.

30.
```
Stu_dict={5:(87,68,89), 10:(57,54,61), 12:(71,67,90),
14:(66,81,80), 18:(80,48,91)}
Stu_Stk=[]
def Push_elements(Stu_Stk, Stu_dict):
    for Stu_ID, marks in Stu_dict.items():
        if marks[2]>=80:
            Stu_Stk.append(Stu_ID)
def Pop_elements(Stu_Stk):
    while len(Stu_Stk)>0:
        print(Stu_Stk.pop())
    if not Stu_Stk:
        print('Stack Empty')
Push_elements(Stu_Stk, Stu_dict)
Pop_elements(Stu_Stk)
```

3

1.5 marks for correct implementation of Push_elements()

1.5 marks for correct implementation of Pop_elements()

The append() function adds an element to the end or to the top of the list implemented stack.

The pop() function removes an element from the top of the stack.

SECTION — D

31.
```
import csv
def maxsalary():
    f=open('record.csv', 'r')
    reader=csv.reader(f)
    skip_header = True
    max= 0
    for row in reader:
        if skip_header:
            skip_header = False
        else:
            if(int(row[3])>max):
                max=int(row[3])
                rec=row
    print('Row with the highest salary : ', rec)
    f.close()
maxsalary()
```

4

½ mark for importing module

½ mark for function definition

½ mark for opening and closing file

½ for reader object

½ for skipping first row (i.e. header)

1 mark for calculating maximum salary

½ mark for displaying record having maximum salary

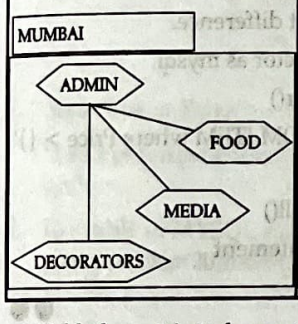
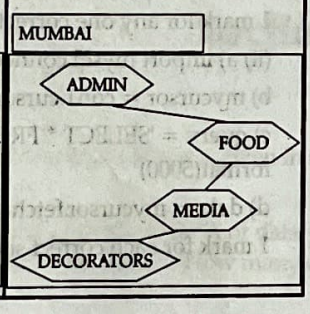
32.
```
import pickle
def expensiveProducts():
    with open('INVENTORY.DAT', 'rb') as file:
```

```
expensive_count = 0
while True:
    try:
        product_data = pickle.load(file)
        product_id, product_name, quantity,
        price = product_data
        if price > 1000:
            print("Product ID:", product_id)
            expensive_count += 1
    except EOFError:
        break
print("Total expensive products: ", expensive_
count)
```

expensiveProducts()
½ mark for function definition
½ mark for opening and closing file
½ mark for correct try and except block
1.5 mark identifying and displaying details of expensive products
1 mark for displaying count of expensive products
pickle module is imported for binary file functions.
load() function reads each record from the file.
The list of data is unpacked into variables.
The price is checked and matching records are printed.

SECTION — E

33. i. The most appropriate location of the server inside the MUMBAI campus is ADMIN building due to the maximum number of computers in it.
Server is to be placed in the campus with highest number of computers.
½ mark for mentioning the branch and ½ mark for proper justification
ii. Cable Layout

Star Topology (Based on server location)	Bus Topology (Based on minimum distance between branches)
	

Cable layout is to be made keeping in mind the least three distances. or connecting other campuses with the campus with maximum computers.
1 mark for drawing any valid cable layout
iii. Switch or Hub
Switch/Hub can connect multiple computers in a LAN
1mark for suggesting the correct device
iv. c. Video Conferencing
1 mark for correct answer
v.
(a) WAN
(b) LAN
½ mark for mentioning WAN and ½ mark for mentioning LAN
a) Mumbai to Delhi is a large distance that ismore than a city , hence a WAN will be formed.
b) Computers in Mumbai campus are with a distance of few metres. Hence a LAN will be formed.

34. i. 5

	Seek()	Tell()
Purpose	Repositions the file pointer to a specific location within a file	Returns the current position of the file pointer
Syntax	seek(offset [,reference point])	tell()
Parameters	Requires specifying the offset and an optional reference point	Requires no parameters

2 marks for mentioning two correct differences.
OR
1 marks for mentioning only one correct differences.
ii. import pickle

```
def COPY_REC():
    In_file = open('FLIGHT.DAT','rb')
    out_file = open('RECORD.DAT','wb')
    try:
        while True: data = pickle.load(In_file)
        if data[3] == 'DELHI' and data[4] == '
    MUMBAI':
            pickle.dump(data,out_file)
    except:
        In_file.close()
        out_file.close()
COPY_REC()
```

½ mark for function definion
½ mark for correctly opening and closing file
½ mark for correct try and except block
1.5 marks for writing required data in RECORD.DAT
OR

i.

Binary	CSV
1. pickle module to be used	1. csv module is used
2. Data is stored in binary format(0s and 1s) and is not in human readable form using any plain text editor.	2. Data is stored in tabular fashion and comma separated by default. The file can be read by any spreadsheet software or text editor.
3. File extension .dat/. pdf/.exe etc.	3. File extension .csv

2 marks for mentioning two correct differences.

OR

1 marks for mentioning only one correct differences.

ii.

```
import pickle
def findBook(price):
    with open('BOOK.DAT', 'rb') as file:
        while True:
            try:
                book_record = pickle.load(file)
                for item in book_record:
                    book_price = book_
                    record[item][2]
                    if book_price >= price:
                        print(item, book_
                        record[item])
        except EOFError:
            break
findBook(50)
```

½ mark for function definion

½ mark for correctly opening and closing file

½ mark for correct try and except block

1.5 marks for displying required records

35. i. SQL constraints are used to specify rules for the data in a table. Constraints are used to limit the type of data that can go into a table.

Constraints –

NOT NULL - Ensures that a column cannot have a NULL value

UNIQUE - Ensures that all values in a column are different

PRIMARY KEY - A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table

½ mark for correct definition, ½ mark for correct example (anyone)

(ii)

a) password='tiger'

b) mycursor = con1.cursor()

c) query = 'delete from ITEM where Iname = "{}" '.format.(item_name)

d) con1.commit()

1 mark for each correct statement

OR

(i) Candidate Key: A candidate key is a set of attributes in a relation that can uniquely identify each tuple (row). A relation can have multiple candidate keys, but only one of them is chosen as the primary key.

Alternate Key: An alternate key is a candidate key that is not selected as the primary key.

1 mark for any one correct difference.

(ii) a) import mysql.connector as mysql

b) mycursor = con1.cursor()

c) query = 'SELECT * FROM ITEM where Price > {}'.format(5000)

d) data = mycursor.fetchall()

1 mark for each correct statement