CBSE Additional Practice Question Paper
Class: XII Session: 2023-24
Computer Science (083)

Time allowed: 3 Hours **Maximum Marks: 70**

General Instructions:

- Please check this question paper contains 35 questions.
- The paper is divided into 5 Sections- A, B, C, D and E.
- Section A, consists of 18 questions (1 to 18). Each question carries 1 Mark.
- Section B, consists of 7 questions (19 to 25). Each question carries 2 Marks.
- Section C, consists of 5 questions (26 to 30). Each question carries 3 Marks.
- Section D, consists of 2 questions (31 to 32). Each question carries 4 Marks.
- Section E, consists of 3 questions (33 to 35). Each question carries 5 Marks.
- All programming questions are to be answered using Python Language only.

Q No.	Questions Section-A (18 Marks)	Marks
1	Which of the following is an invalid identifier to be used in Python? **a.** per%marks **b.** _for **c.** While **d.** true	1
2	What is the correct way to add an element to the end of a list in Python? **a.** list.add(element) **b.** list.append(element) **c.** list.insert(element) **d.** list.extend(element)	1
3	What will be the output of `print("Welcome To My Blog"[2:6] + "Welcome To My Blog"[5:9])` **a.** Lcomme **b.** lcomme T **c.** lcomme To **d.** lcomme	1
4	Which of the following statements is false? **a.** A try-except block can have more than one except statement **b.** One block of except statement cannot handle multiple exceptions **c.** The finally block is always executed **d.** When 1 == "1" is executed, no exception is raised	1
5	Which of the following statement(s) would give an error during the execution of the following code? R = {'pno':52,'pname':'Virat', 'expert':['Badminton','Tennis'] ,'score':(77,44)} print(R) #Statement 1	1

	R['expert'][0]='Cricket' #Statement 2 R['score'][0]=50 #Statement 3 R['pno']=50 #Statement 4 **a.** Statement 1 **b.** Statement 2 **c.** Statement 3 **d.** Statement 4	
6	Which pickle module method is used to write a Python object to a binary file? **a.** save() **b.** serialize() **c.** store() **d.** dump()	1
7	Given the following dictionaries `dict_student = {"rno" : "53", "name" : 'Rajveer Singh'}` `dict_marks = {"Accts" : 87, "English" : 65}` Which statement will append the contents of dict_marks in dict_student? **a.** dict_student + dict_marks **b.** dict_student.add(dict_marks) **c.** dict_student.merge(dict_marks) **d.** dict_student.update(dict_marks)	1
8	Which of the following is not a component of the math module in Python? **a.** ceil() **b.** mean() **c.** fabs() **d.** pi	1
9	What will be the output of the following code? `L=["One , Two", "Three", "Four"]` `print(len(L)/2*len(L[0]))` **a.** 6.5 **b.** 13 **c.** 13.5 **d.** 6.0	1
10	Expand the following terms: (i) PPP (ii) VoIP	1
11	Which SQL operator performs pattern matching? **a.** BETWEEN operator **b.** LIKE operator **c.** EXISTS operator **d.** =	1

12	Which Python function is used for displaying only one result set from SQL table in a database? **a.** fetch1() **b.** fetchno() **c.** fetchall() **d.** fetchone()	1
13	Which of the following file opening mode in Python, generates an error if the file does not exist? **a.** a **b.** r **c.** w **d.** w+	1
14	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)	1
15	Which of the following statements is false? **a.** SMTP and POP protocols are used in email communication. **b.** URL of a page is not always the same as its domain name. **c.** HTTPS is safer than HTTP. **d.** Interlinking of collection of webpages is called Internet.	1
16	Fill in the blank: _____ protocol provides access to services hosted on a remote computer. **a.** FTP **b.** PPP **c.** Telnet **d.** SMTP	1
	Q17 and 18 are ASSERTION AND REASONING based questions. Mark the correct choice as (a) Both A and R are true and R is the correct explanation for A (b) Both A and R are true and R is not the correct explanation for A (c) A is True but R is False (d) A is false but R is True	
17	**Assertion (A):** For changes made to a variable defined within a function to be visible outside the function, it should be declared as global. **Reasoning (R):** Variables defined within a function are local to that function by default, unless explicitly specified with the global keyword.	1
18	**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.	1

	Reasoning (A): Binary files are just like normal text files and can be read using a text editor like Notepad.	
Q No.	**Questions** **Section-B (14 Marks)**	**Marks**
19	Write two advantages and two disadvantages of circuit switching. **OR** Differentiate between Web server and web browser. Write the names of any two web browsers.	2
20	Rewrite the following code in Python after removing all the syntax errors. Underline each correction done in the code. `num1, num2 = 10, 45` `While num1 % num2 == 0` `num1+= 20` `num2+= 30` `Else:` `    print('hello')`	2
21	Write a function `dispBook(BOOKS)` in Python, that takes a dictionary `BOOKS` as an argument and displays the names in uppercase of those books whose name starts with a consonant. For example, Consider the following dictionary `BOOKS = {1:"Python", 2:"Internet Fundamentals ", 3:"Networking ", 4:"Oracle sets", 5:"Understanding HTML"}` The output should be: `PYTHON` `NETWORKING` **OR** Write a Python Program containing a function `FindWord(STRING, SEARCH)`, that accepts two arguments : `STRING` and `SEARCH`, and prints the count of occurrence of `SEARCH` in `STRING`. Write appropriate statements to call the function. For example, if `STRING` = "Learning history helps to know about history with interest in history" and `SEARCH` = 'history', the function should display `The word history occurs 3 times.`	2
22	What will be the output of the following code? `L = [5,10,15,1]` `G = 4` `def Change(X):` `    global G` `    N=len(X)` `    for i in range(N):` `        X[i] += G` `Change(L)` `for i in L:` `    print(i,end='$')`	2

23 — 2

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

OR

Write a Python Program to display alternate characters of a string `my_str`.

For example, if `my_str` = "Computer Science"

The output should be `Cmue cec`

24 — 2

Differentiate between % (percentage) and _(underscore) characters used with the LIKE operator in SQL with appropriate examples.

OR

Differentiate between DROP and DELETE commands in SQL with appropriate examples.

25 — 2

Consider the following two commands with reference to a table, named `Employee` having a column named *`Department`*:

(a) `Select count(Department) from Employee;`

(b) `Select count(*) from Employee;`

If these two commands are producing different results,

(i) What may be the possible reason?

(ii) Which command (a) or (b) might be giving a higher value?

Q No	**Questions** **Section-C (15 Marks)**	**Marks**

26 — 3

(a) Consider the table, `BOOK` and `MEMBER` given below:

`TABLE : BOOK`

CODE	BNAME	TYPE
F101	The priest	Fiction
L102	Easy Python	Programming
C101	Juman Ji	Thriller
F102	Untold Story	Fiction
C102	War Stories	Comic

`Table: MEMBER`

MNO	MNAME	CODE	ISSUEDATE
M101	SNEH SINHA	L102	2022-10-13
M103	SARTHAK	F102	2021-02-23
M102	SARA KHAN	C101	2022-06-12

What will be the output of the following statement?

```
SELECT * FROM BOOK NATURAL JOIN MEMBER;
```

(b) Write the output of the queries (i) to (iv) based on the table

`Table: Employee`

EID	Name	DOB	DOJ	Salary	Project
E01	Ranjan	1990-07-12	2015-01-21	150000	P01
E02	Akhtar	1992-06-21	2015-02-01	125000	P04
E03	Muneera	1996-11-15	2018-08-19	135000	P01
E04	Alex	1991-10-25	2018-10-19	75000	P02
E05	Satyansh	1993-12-16	2018-10-19	85000	P04

i
```
SELECT NAME, PROJECT FROM EMPLOYEE ORDER BY NAME DESC;
```
ii
```
SELECT NAME, SALARY FROM EMPLOYEE WHERE NAME LIKE 'A%';
```
iii
```
SELECT NAME, DOJ FROM EMPLOYEE WHERE SALARY BETWEEN
100000 AND 200000;
```
iv
```
SELECT * FROM EMPLOYEE WHERE PROJECT = 'P01';
```

27 (a) Consider the following tables – FACULTY and COURSES : 3

Table: FACULTY

FID	FNAME	LNAME	JOINDATE	SALARY
F01	Anishma	Garg	2000-12-14	32000
F03	Bhumi	Goel	2001-08-10	15000
F04	Neha	Verma	2000-05-17	27000
F05	Meenu	Sharma	2006-07-11	30000

Table: COURSES

C_ID	FID	CNAME	FEES
C11	F01	Grid Computing	40000
C12	F04	Python	17000
C13	F03	C++	8000
C14	F04	Computer Network	15000
C15	F01	HTML	12000
C16	F05	Data Science	NULL

What will be the output of the following statement?

i
```
SELECT FID, MIN(FEES), MAX(FEES) FROM COURSES GROUP BY
FID;
```
ii
```
SELECT AVG(SALARY) FROM FACULTY WHERE FNAME LIKE '%a';
```
iii
```
SELECT FNAME, CNAME FROM FACULTY F, COURSES C WHERE
F.FID=C.FID AND COURSES.FID='F04';
```
iv
```
SELECT FNAME, CNAME , FEES FROM FACULTY F , COURSES C
WHERE F.FID = C.FID AND FEE>15000;
```

(b) Write the name of the command to display the structure of a table in a database.

28 Write a function COUNT*()* in Python to read from a text file 'Gratitude.txt' and display the count of the letter 'e' in each line 3

Example: If the file content is as follows:

```
Gratitude is a humble heart's radiant glow,
A timeless gift that nurtures and bestows.
It's the appreciation for the love we're shown,
In moments big and small, it's truly known.
```

The COUNT() function should display the output as:

```
Line 1 : 3
Line 2 : 4
Line 3 : 6
Line 4 : 1
```

OR

Write a function Start_with_I() in Python, which should read a text file 'Gratitude.txt' and then display lines starting with 'I'.

Example: If the file content is as follows:

```
Gratitude is a humble heart's radiant glow,
A timeless gift that nurtures and bestows.
It's the appreciation for the love we're shown,
In moments big and small, it's truly known.
```

Then the output should be

```
It's the appreciation for the love we're shown,
In moments big and small, it's truly known.
```

29 (3)

Navdeep creates a table `RESULT` with a set of records to maintain the marks secured by students in Sem1, Sem2, Sem3, and their divisions. After the creation of the table, he entered data of 7 students in the table.

ADNO	ROLLNO	SNAME	SEM1	SEM2	DIVISION
123	101	KARAN	366	410	I
245	102	NAMAN	300	350	I
128	103	ISHA	400	410	I
129	104	RENU	350	357	I
234	105	ARPIT	100	75	IV
187	106	SABINA	100	205	II
181	107	NEELAM	470	450	I

Based on the data given above answer the following questions:

i Identify the columns which can be considered as candidate keys?

ii If 2 more columns are added and 3 rows are deleted from the table result, what will be the new degree and cardinality of the above table?

iii Write a statement to increase the SEM2 marks by 3% for the students securing marks between 70 to 100.

30 (3)

Given a Dictionary `Stu_dict` containing marks of students for three test-series in the form `Stu_ID`*:(TS1, TS2, TS3)* as key-value pairs.
Write a Python program with the following user-defined functions to perform the specified operations on a stack named `Stu_Stk`

(i) `Push_elements(Stu_Stk, Stu_dict)` : It allows pushing IDs of those students, from the dictionary `Stu_dict` into the stack `Stu_Stk`, who have scored more than or equal to 80 marks in the TS3 Test.
(ii) `Pop_elements(Stu_Stk):` It removes all elements present inside the stack in LIFO order and prints them. Also, the function displays 'Stack Empty' when there are no elements in the stack.
Call both functions to execute queries.

For example:
If the dictionary `Stu_dict` contains the following data:

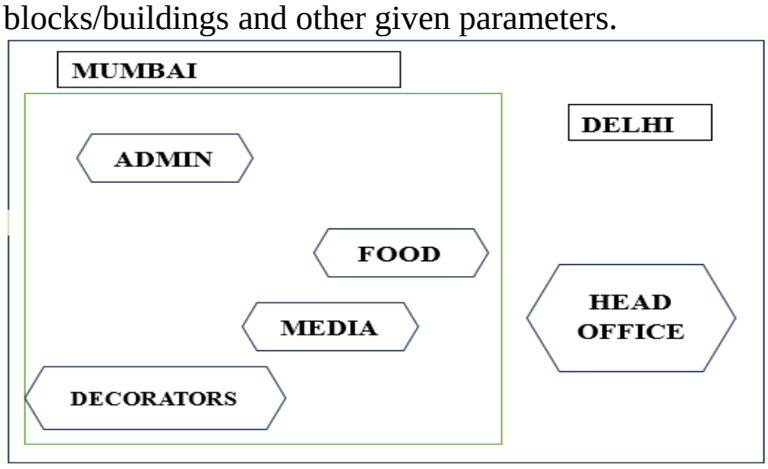
```
Stu_dict ={5:(87,68,89), 10:(57,54,61), 12:(71,67,90), 14:(66,81,80), 18:(80,48,91)}
```

After executing `Push_elements()`, `Stk_ID` should contain [5,12,14,18]

After executing `Pop_elements()`, The output should be:

```
18
14
12
5
Stack Empty
```

<table>
<tr><th>Q No.</th><th>Questions
Section-D (8 Marks)</th><th>Marks</th></tr>
<tr><td>31</td><td>Create a function <code>maxsalary()</code> in Python to read all the records from an already existing file <code>record.csv</code> which stores the records of various employees working in a department. Data is stored under various fields as shown below:
<table>
<tr><th>E_code</th><th>E_name</th><th>Scale</th><th>Salary</th></tr>
<tr><td>A01</td><td>Bijesh Mehra</td><td>S4</td><td>65400</td></tr>
<tr><td>B02</td><td>Vikram Goel</td><td>S3</td><td>60000</td></tr>
<tr><td>C09</td><td>Suraj Mehta</td><td>S2</td><td>45300</td></tr>
<tr><td>……</td><td>……</td><td>……</td><td>……</td></tr>
</table>
Function should display the row where the salary is maximum.
Note: Assume that all employees have distinct salary.</td><td>4</td></tr>
<tr><td>32</td><td>Consider a binary file <code>'INVENTORY.DAT'</code> that stores information about products using tuple with the structure <code>(ProductID, ProductName, Quantity, Price).</code> Write a Python function <code>expensiveProducts()</code> to read the contents of <code>'INVENTORY.DAT'</code> and display details of products with a price higher than Rs. 1000. Additionally, calculate and display the total count of such expensive products.

For example: If the file stores the following data in binary format
(1, 'ABC', 100, 5000)
(2, 'DEF', 250, 1000)
(3, 'GHI', 300, 2000)
then the function should display
<code>Product ID: 1</code>
<code>Product ID: 3</code>
<code>Total expensive products: 2</code></td><td>4</td></tr>
<tr><th>Q No.</th><th>Questions
Section-E (15 Marks)</th><th>Marks</th></tr>
<tr><td>33</td><td>Fun Media Services Ltd is an event planning organization. It is planning to set up its India campus in Mumbai with its head office in Delhi. The Mumbai campus will have four blocks/buildings - ADMIN, DECORATORS, FOOD, and MEDIA.
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.
 
</td><td>5</td></tr>
</table>

Shortest distance between various buildings:

FROM – TO	DISTANCE
ADMIN TO DECORATORS	90 meters
ADMIN TO MEDIA	75 meters
ADMIN TO FOOD	50 meters
DECORATORS TO FOOD	65 meters
DECORATORS TO MEDIA	50 meters
FOOD TO MEDIA	45 meters
DELHI Head Office to MUMBAI Campus	1475 KM

The number of computers at various buildings is as follows:

BUILDING	NUMBER OF COMPUTERS
ADMIN	110
DECORATORS	75
MEDIA	12
FOOD	20

i. Suggest the most appropriate location of the server inside the MUMBAI campus (out of the 4 buildings). Justify your answer.
ii. Draw the cable layout to efficiently connect various buildings within the MUMBAI campus.
iii. Which hardware device will you suggest to connect all the computers within each building?
iv. Which of the following will you suggest to establish online face-to-face communication between the people in the Admin Office of the MUMBAI campus and the DELHI Head Office?
a. Cable TV
b. Email
c. Video Conferencing
d. Text Chat
v. What type of network (out of PAN, LAN, MAN, WAN) will be set up in each of the following cases?
a. The Mumbai campus gets connected with the Head Quarter in Delhi
b. The computers connected in the MUMBAI campus

34

i. Mention any two differences between seek() and tell(). 2+3=5
ii. Consider a file `FLIGHT.DAT` containing multiple records. The structure of each record is as shown below:
`[Fno, FName, Fare, Source, Destination]`
Write a function `COPY_REC()` in Python that copies all those records from FLIGHT.DAT where the source is `DELHI` and the destination is `MUMBAI`, into a new file `RECORD.DAT`

OR

i. Mention any two differences between binary files and csv files?
ii. Consider a Binary file `BOOK.DAT` containing a dictionary having multiple elements. Each element is in the form `BNO:[BNAME,BTYPE,PRICE]` as key:value pair
where
`BNO` – Book Number
`BNAME` – Book Name
`BTYPE` - Book Type
`PRICE` – Book price

	Write a user-defined function, `findBook(price)`, that accepts price as parameter and displays all those records from the binary file `BOOK.DAT` which has a book price more than or equal to the price value passed as a parameter.	
35	i. Define the term constraint with respect to RDBMS. Give a suitable example. ii. Sameera maintains a database named `STORE` which contains a table named `ITEM` with the structure given below: • `Ino`(Item number)- integer • `Iname`(Item Name) – string • `Price` (Item Price) – float • `Discount` (Discount) – float Note the following to establish connectivity between Python and MySQL: • Username - root • Password - tiger • Host - localhost Help her to remove the record from the table ITEM for a particular value of item name input by the user. `import mysql.connector as mysql` `con1= mysql.connect(host='localhost', user='root', password= '__', database='STORE')  #Statement-1` `mycursor = ________  #Statement-2` `item_name = input("Enter the Item name to remove the record : ")` `query = _____  #Statement-3` `mycursor.execute(query)` `con1.____  #Statement-4` `print('Data Deleted successfully')` `con1.close()` With reference to the above code, answer the following questions a) Complete statement 1 to establish the connection with the database. b) Write statement 2 to create the cursor object. c) Complete statement 3 to remove the record from the table ITEM based on the item name entered by the user d) Complete statement 4 to save the changes in the table. **OR** i. Write one difference between the alternate key and the candidate key. ii. A table named `ITEM` is created in a database `STORE`. The table contains multiple columns whose details are as shown below: • `Ino`(Item number)- integer • `Iname`(Item Name) – string • `Price` (Item Price) – float • `Discount` (Discount) – float Note the following to establish connectivity between Python and MySQL: • Username - root • Password - tiger • Host - localhost However, the table is to be interfaced with Python to perform certain tasks. The incomplete code is given below:	5

```
______  #Line 1
con1= mysql.connect(host='localhost', user = 'root', password =
'tiger', database='STORE')
mycursor = con1.____     #Line 2
query = 'SELECT * FROM ITEM where Price > {}'.format(___) #Line3
mycursor.execute(query)
data = mycursor.___  #Line 4
for rec in data:
      print(rec)
con1.close()
```

i. Complete line 1 to import the appropriate module.

ii. Complete Line 2 to create the cursor object

iii. Complete the query given in Line 3 to display details of all such items from the table ITEMS whose price is more than 5000.

iv. Complete Line 4 to extract all the records.

CBSE Additional Practice Question Paper
Class: XII Session: 2023-24
Computer Science (083)
Marking Scheme

Q No.	Answer	Total Marks
1	a. per%marks	1
2	b. list.append(element)	1
3	b. lcomme T	1
4	b. One block of except statement cannot handle multiple exceptions	1
5	c. Statement 3	1
6	d. dump	1
7	d. dict_student.update(dict_marks)	1
8	b. mean()	1
9	c. 13.5	1
10	PPP – Point to Point Protocol VoIP - Voice Over Internet Protocol	1
11	b. LIKE operator	1
12	d. fetchone	1
13	b. r	1
14	a. file_object.seek(offset [, reference_point])	1
15	d. Interlinking of collection of webpages is called Internet.	1
16	c. TelNet	1
17	a. Both A and R are true and R is the correct explanation for A	1
18	c. A is True but R is False	1
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* **OR** **Web browser** Purpose: Receives and displays web content.	2

	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')` *½ mark for while* *½ mark for :* *½ mark for correct indentation (inside the block of while)* *½ mark for else*	2
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* **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*	2
22	9$14$19$5$ *½ mark for 9$* *½ mark for 14$* *½ mark for 19$* *½ mark for 5$*	2
23	i. del D['Mumbai'] *1 mark for correct answer*	2

	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*	
24	% (Percentage): • 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.*	2
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*	2
26	(a) CODE | BNAME | TYPE | MNO | MNAME | ISSUEDATE L102 | Easy Python | Programming | 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 *1 mark for correct answer*	3

(b)

(i)

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

½ mark for correct output

(ii)

NAME	SALARY
Akhtar	125000
Alex	75000

½ mark for correct output

(iii)

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) 3

(i)

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

½ mark for correct answer

(ii)

AVG(SALARY)
29500

½ mark for correct answer

(iii)

FNAME	CNAME
Neha	Python
Neha	Computer Network

½ mark for correct answer

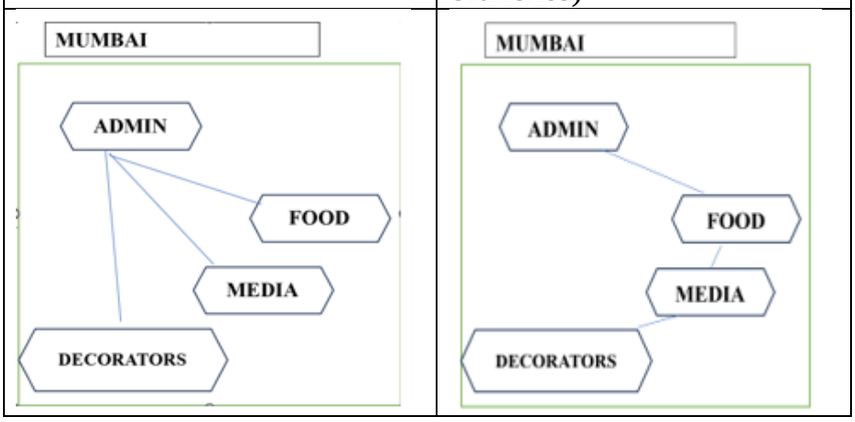
(iv)

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() ``` *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* **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*	3
29	(i) Candidate Keys : ADMNO, ROLLNO *1 mark for correctly writing both names of candidate keys. OR ½ mark for specifying any one candidate key correctly.* (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.*	3
30	``` Stu_dict={5:(87,68,89), 10:(57,54,61), 12:(71,67,90), 14:(66,81,80), 18:(80,48,91)} ```	3

	`` 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) ``` *1.5 marks for correct implementation of Push_elements()* *1.5 marks for correct implementation of Pop_elements()*	
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() ``` *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*	4
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*	4

	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*	
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. *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)  *1 mark for drawing any valid cable layout* iii. Switch or Hub *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*	5
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)	5

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