



**COMMON PRE-BOARD EXAMINATION**  
**COMPUTER SCIENCE - Code No. 083**  
**Class-XII-(2025-26)**



**SET: 3**

**Time allowed: 3 Hrs.**

**Maximum Marks: 70**

**MARKING SCHEME**

General Instructions:

1. In Python, string content is accepted within a pair of single quotes ' ' or within a pair of double quotes " ".
2. In MySQL, CHAR/VARCHAR/DATE type content is accepted within a pair of single quotes ' ' or within a pair of double quotes " ".
3. In MySQL commands, lowercase/UPPERCASE both are correct.
4. All answers/codes are suggestive, any other alternative correct answers to be accepted.

Q.No.	Questions	Marks
	<b>Section-A (21 x 1 = 21 Marks)</b>	
1.	State if the following statement is True or False: Using the statistics module in Python, the output of the below statements will be 30: <pre>import statistics print(statistics.median([10, 20, 35, 40, 50]))</pre>	<b>1</b>
	<b>Answer:</b> False <i>(1 mark for correct answer)</i>	
2.	What will be the output of the following Python code? <pre>text = "DeepLearning@2025" print(text.find("E"))</pre> A) False    B) 0    C) ValueError    D) -1	<b>1</b>
	<b>Answer:</b> D) -1 <i>(1 mark for correct answer)</i>	

3.	Fill in the blank with the Boolean value (True/False) such that the following expression evaluates to False:  (True and _____) and (not False or True)	<b>1</b>
	<b>Answer:</b>  False <i>(1 mark for correct answer)</i>	
4.	In SQL, which clause is used to filter records based on a condition?	<b>1</b>
	<b>Answer:</b>  Where clause <i>(1 mark for correct answer)</i>	
5.	What will be the output of the following Python code? <pre>s = "Data#Science#Python" s = s.split('#') result = s[2] + "@" + s[1] + "@" + s[0] print(result)</pre> A) Science@Data@Python                      B) Data@Science@Python C) Python@Science@Data                    D) Python@Data@Science	<b>1</b>
	<b>Answer:</b>  C) Python@Science@Data <i>(1 mark for correct answer)</i>	
6.	Write the output of the following Python code: <pre>L = [10, 20, (30, 40), 50] L[2] = (30, 400) print(L)</pre>	<b>1</b>
	<b>Answer:</b>  [10, 20, (30, 400), 50] <i>(1 mark for correct answer)</i>	
7.	What will be the output of the following Python statement? <pre>print(100 - 2**3**1 + 40/4)</pre>	<b>1</b>
	<b>Answer:</b>  102.0 <i>(1 mark for correct answer)</i>	
8.	Consider the given SQL Query: SELECT course, AVG(fees) AS avg_fees FROM institution ORDER BY avg_fees DESC GROUP BY course;	<b>1</b>

	Anju is executing the above query, but facing an error. Identify and write the corrected SQL query.	
	<p><b>Answer:</b></p> <p>SELECT course, AVG(fees) AS avg_fees FROM institution GROUP BY course ORDER BY avg_fees DESC; <i>(1 mark for correct answer)</i></p>	
9.	<p>What will be the output of the following Python code?</p> <pre>def check():     try:         x = 10 / 5         return "Success"     except TypeError:         return "Error"     finally:         return "End" print(check())</pre> <p>A) Success      B) Error      C) TypeError      D) End</p>	<b>1</b>
	<p><b>Answer:</b></p> <p>C) End <i>(1 mark for correct answer)</i></p>	
10.	<p>What will be the output of the following Python code?</p> <pre>stock = {"laptop": 150, "mouse": 220} print(stock.popitem( ), end=' ') print(stock)</pre> <p>A) ('laptop ', 150) {'mouse ': 220}      B) ('mouse ', 220) { 'laptop ': 150} C) {'mouse ': 220} (' laptop ', 150)      D) {'mouse ': 220} {'laptop ', 150}</p>	<b>1</b>
	<p><b>Answer:</b></p> <p>B) ('mouse', 220) {'laptop': 150} <i>(1 mark for correct answer)</i></p>	
11.	<p>What possible output is not expected to be displayed on the screen at the time of execution of the Python program from the following code?</p> <pre>import random num = random.randint(1, 4) Technologies = ["Java", "Python", "Ruby", "C", "C++"] for i in range(0, num):     print(Technologies[i], end='**') print()</pre> <p>A) Java**      B) Java**Python**Ruby**C**C++** C) Java**Python**Ruby**      D) Java**Python**Ruby**C**</p>	<b>1</b>

	<p><b>Answer:</b></p> <p>B) Java**Python**Ruby**C**C++**  <i>(1 mark for correct answer)</i></p>	
12.	<p>What will be the output of the following Python code?</p> <pre> p = 10 q = 5 def compute(y):     global q     p = y + 2     q = q + p     print(p, q, end='\$\$') compute(3) print(p, q) </pre> <p>A) 5 10\$\$10 10    B) 5 10\$\$10 5    C) 3 8\$\$10 8    D) 5 12\$\$10 12</p>	<b>1</b>
	<p><b>Answer:</b></p> <p>A) 5 10\$\$10 10  <i>(1 mark for correct answer)</i></p>	
13.	<p>Which SQL function is used to return the number of rows in a result set?</p> <p>A) TOTAL()    B) SUM()    C) ADD()    D) COUNT()</p>	<b>1</b>
	<p><b>Answer:</b></p> <p>D) COUNT()  <i>(1 mark for correct answer)</i></p>	
14.	<p>Give the output of the given Python code?</p> <pre> phrase = "DEEP learning IN neural NETWORKS" print(phrase.count("NE")) </pre> <p>A) 4    B) 3    C) 2    D) 1</p>	<b>1</b>
	<p><b>Answer:</b></p> <p>D) 1  <i>(1 mark for correct answer)</i></p>	
15.	<p>Consider a table STUDENT with 12 records and 5 attributes, and another table COURSE with 8 records and 4 attributes. What would be the degree and cardinality of the resultant table after performing a NATURAL JOIN operation, assuming one common attribute with 6 matching records?</p>	<b>1</b>
	<p><b>Answer:</b></p> <p>Degree = 8, Cardinality = 6  <i>(1/2 mark for correct degree and 1/2 mark for correct cardinality answer)</i></p>	

16.	Which SQL query will display all records without duplicates from the table EMPLOYEE? A) SELECT ALL * FROM EMPLOYEE; B) SELECT UNIQUE * FROM EMPLOYEE; C) SELECT DISTINCT * FROM EMPLOYEE; D) SELECT * FROM EMPLOYEE WHERE UNIQUE;	<b>1</b>
	<b>Answer:</b> C) SELECT DISTINCT * FROM EMPLOYEE; (1 mark for correct answer)	
17.	Which network topology connects all devices to a central device? A) Bus    B) Ring    C) Star    D) Mesh	<b>1</b>
	<b>Answer:</b> C) Star (1 mark for correct answer)	
18.	The modem at the receiver's computer end acts as a _____. A) Model    B) Modulator    C) Demodulator    D) Convertor	<b>1</b>
	<b>Answer:</b> C) Demodulator (1 mark for correct answer)	
19.	In _____ switching, a dedicated communication path is established between sender and receiver for the entire duration of communication. A) Circuit    B) Packet    C) Message    D) Virtual	<b>1</b>
	<b>Answer:</b> B) Packet Switching (1 mark for correct answer)	
	Q20 and Q21 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 Python, a function cannot return multiple values at the same time. Reason (R): Python allows functions to return multiple values as a tuple by separating them with commas.	<b>1</b>
	<b>Answer:</b> D) A is False but R is True. (1 mark for correct answer)	

21.	Assertion (A): A primary key uniquely identifies each record in a table. Reason (R): A primary key can contain NULL values.	<b>1</b>
	<b>Answer:</b> C) A is True but R is False. <i>(1 mark for correct answer)</i>	
	<b>Section-B (7 x 2=14 Marks)</b>	
22.	A) Differentiate between Differentiate between syntax error and runtime error with suitable example of each. <p style="text-align: center;"><b>OR</b></p> B) Explain the difference between mutable and immutable data types in Python with suitable examples.	<b>2</b>
	<b>Answer:</b> A) Syntax Error: Occurs when code violates the rules of Python syntax, detected during parsing. Example: print("Hello" (missing closing parenthesis) Runtime Error: Occurs during program execution when a valid statement causes an error. Example: x = 10/0 (division by zero) <i>(1 mark for correct difference)</i> <i>(1/2 mark for each correct example)</i> <p style="text-align: center;"><b>OR</b></p> B) Mutable: Objects whose value can be changed after creation. Example: Lists, Dictionaries, Sets Immutable: Objects whose value cannot be changed after creation. Example: Strings, Tuples, Numbers <i>(1 mark for correct difference)</i> <i>(1/2 mark for each correct example)</i>	
23.	The code given below has been programmed to accept a number as an argument and check whether the given number is Perfect Number or not. Observe the following code carefully and rewrite it after removing all syntax and logical errors. Underline all the corrections made. <pre> def perfectNum(num):     sum = 0     for i in range(1, num):         if num / i == 0:             s = sum + i     if sum == num:         print("The number is a Perfect number")     else         print("The number is not a Perfect number") </pre>	<b>2</b>

	<pre>num =input("Enter the number") perfectNum(num)</pre>	
	<p><b>Answer:</b></p> <pre>def perfectNum(num):     sum = 0     for i in range(1, num):         if <u>num % i == 0:</u>             <u>sum = sum + i</u>     if sum == num:         print("The number is a Perfect number")     <u>else:</u>         print("The number is not a Perfect number") <u>num =int(input("Enter the number"))</u> perfectNum(num)</pre> <p><i>(1/2 mark each for correcting 4 mistakes)</i></p>	
24.	<p>A) Consider the following List, Colors = ['Red', 'Blue', 'Green', 'Yellow', 'Orange'] (Answer using Python built-in methods/functions only):</p> <p>I) Write a Python statement to insert the element "Purple" at index 2 in the List 'Colors'.</p> <p>II) Write a Python statement to reverse the List 'Colors'.</p> <p style="text-align: center;"><b>OR</b></p> <p>B) Predict the Output for following Python code:</p> <pre>def Process(x, y):     if x &lt; y:         print('smaller', end=' ')         return x, y     else:         print('larger', end=' ')         return y, x numbers = [5, 15, 8, 25, 12] for i in range(0, 4, 2):     print(Process(numbers[i], numbers[i+1]))</pre>	<b>2</b>
	<p><b>Answer:</b></p> <p>A) I) Colors.insert(2, "Purple") II) Colors.reverse() <i>(1 mark for each correct answer)</i></p> <p>B) smaller (5, 15) smaller (8, 25) <i>(1 mark for each correct answer)</i></p>	

25.	<p>A) Write a function CUBE_LIST(L), where L is the List of elements passed as argument to the function. The function returns another list named 'CList' that stores the Cubes of all Even Elements of L. For example: If L contains [2, 5, 8, 3, 6, 7, 10] The CList will have [8, 512, 216, 1000]</p> <p style="text-align: center;"><b>OR</b></p> <p>B) Write a function MAX_MARKS() that creates a Dictionary of students and their marks and returns the name of the student with the maximum marks. For example: If Dictionary students contains {"Rahul": 85, "Priya": 92, "Amit": 78} Then output is: Top Student - Priya with 92 marks</p>	<b>2</b>
	<p><b>Answer:</b></p> <p>A)</p> <pre>def CUBE_LIST(L):     CList = []     for element in L:         if element % 2 == 0:             CList.append(element ** 3)     return CList L = [2, 5, 8, 3, 6, 7, 10] result = CUBE_LIST(L) print("Original List:", L) print("Cube List:", result)</pre> <p><i>(1/2 mark for correct function header)</i>  <i>(1/2 mark for correct loop)</i>  <i>(1/2 mark for correct if statement)</i>  <i>(1/2 mark for return statement)</i>  <b>Note: Any other relevant and correct code may be marked</b></p> <p style="text-align: center;"><b>OR</b></p> <p>B)</p> <pre>def MAX_MARKS():     students = {"Rahul": 85, "Priya": 92, "Amit": 78}     max_name = ""     max_marks = 0     for name, marks in students.items():         if marks &gt; max_marks:             max_marks = marks             max_name = name     print("Top Student - ", max_name, "with ", max_marks, "marks")     return max_name MAX_MARKS()</pre> <p><i>(1/2 mark for correct function header)</i></p>	

	<p><i>(1 mark for calculation of maximum marks and getting the name)</i>  <i>(½ mark for return statement)</i>  <b>Note: Any other relevant and correct code may be marked</b></p>	
26.	<p>Predict the output of the Python code given below:</p> <pre>products = [("Laptop", 45000), ("Mouse", 500)] inventory = { } for item in products:     if item[1] &gt; 1000:         inventory[item[0]] = "Expensive-" + str(item[1])     else:         inventory[item[0]] = "Cheap-" + str(item[1]) print(inventory)</pre>	<b>2</b>
	<p><b>Answer:</b>  {'Laptop': 'Expensive-45000', 'Mouse': 'Cheap-500'}  <i>(1 mark for each correct output item of Dictionary)</i></p>	
27.	<p>A) Write suitable commands to do the following in MySQL.</p> <p>I) Display all tables present in the current database.  II) Remove a column named 'phone' from a table named EMPLOYEES.</p> <p style="text-align: center;"><b>OR</b></p> <p>B) Differentiate between INSERT and UPDATE query in SQL with suitable examples.</p>	<b>2</b>
	<p><b>Answer:</b>  A)  I) SHOW TABLES;  II) ALTER TABLE EMPLOYEES DROP COLUMN phone;  <i>(1 mark for each correct answer)</i></p> <p style="text-align: center;"><b>OR</b></p> <p>B) INSERT is used to add new records into a table.  Example: INSERT INTO STUDENT VALUES (101, 'Amit', 85);  UPDATE is used to modify existing records in a table.  Example: UPDATE STUDENT SET Marks = 90 WHERE RollNo = 101;  <i>(1 mark for correct difference)</i>  <i>(½ mark for each correct example)</i></p>	
28.	<p>A) Define the following terms:</p> <p>I) Protocol  II) Gateway</p> <p style="text-align: center;"><b>OR</b></p> <p>B)</p> <p>I) Expand the following terms: HTTP and SMTP  II) Bring out the difference between Circuit Switching and Packet Switching.</p>	<b>2</b>
	<p><b>Answer:</b></p> <p>A)  I) A protocol is a set of rules and standards that govern the communication between devices in a network.</p>	

	<p><i>(1 mark for correct answer)</i></p> <p>II) A gateway is a network device that acts as an entry/exit point between two different networks using different protocols.</p> <p><i>(1 mark for correct answer)</i></p> <p style="text-align: center;"><b>OR</b></p> <p>B)</p> <p>I) HyperText Transfer Protocol Simple Mail Transfer Protocol</p> <p><i>(1/2 mark for each correct expansion)</i></p> <p>II) Circuit Switching - A dedicated communication path is established between sender and receiver before data transmission begins Packet Switching - No dedicated path is established; data is divided into packets that travel independently</p> <p><i>(1 mark for correct point of difference)</i></p>	
<b>Section-C (3 x 3 = 9 Marks)</b>		
29.	<p>A) Write a function VC_COUNT() in Python, which should read each character of a Text File “THEORY.TXT” and then count and display all the vowels and consonants separately (including upper cases and small cases. Also, digits and other special characters should not be counted).</p> <p>For example: If the file “THEORY.TXT” content is as follows: Artificial intelligence is a constellation of more than 100 to 200 different technologies working together to enable machines to sense, comprehend, act, and learn with human-like levels of intelligence. The VC_COUNT() function should display the output as: Total vowels are: 65 Total consonants are: 98</p> <p style="text-align: center;"><b>OR</b></p> <p>B) Write a Python function REVERSELINE( ) that copies the reverse of each line from the Text File “Input.txt” to another Text File “Output.txt”.</p> <p>For example: If the content of Input.txt is: The plates will still shift and the clouds will still spew. The sun will slowly rise and the moon will follow too. The content of “Output.txt” should be: tfihs llits lliw setalp ehT .weps llits lliw sduolc eht dna esir ylwols lliw nus ehT .oot wollof lliw noom eht dna</p>	<b>3</b>
	<b>Answer:</b>	

	<p>A)</p> <pre>def VC_COUNT():     obj=open("THEORY.TXT","r")     text=obj.read()     VCOUNT=0     CCOUNT=0     for i in text:         if( i.isalpha()):             if i in ('a','e','i','o','u') or i in ('A','E','I','O','U'):                 VCOUNT+=1             else:                 CCOUNT+=1     print("Total vowels are: ",VCOUNT)     print("Total consonants are: ",CCOUNT)     obj.close() VC_COUNT()</pre> <p>OR Any other correct variation of the code</p> <p><i>(1/2 mark for correctly opening the Theory.txt file)</i>  <i>(1/2 mark for correct reading from file)</i>  <i>(1/2 mark for correct looping statement and alphabet checking)</i>  <i>(1/2 mark for correct if statement for vowel checking and counting)</i>  <i>(1/2 mark for correct if statement for consonant checking and counting)</i>  <i>(1/2 mark for correct display statements)</i></p> <p style="text-align: center;"><b>OR</b></p> <p>B)</p> <pre>def REVERSELINE():     IN=open("Input.txt","r")     OUT=open("Output.txt","w")     L=IN.readlines()     for i in L:         OUT.write(i[::-1])     IN.close()     OUT.close() REVERSELINE()</pre> <p>OR Any other correct variation of the code</p> <p><i>(1/2 mark for correctly opening the Input.txt file)</i>  <i>(1/2 mark for correctly opening the Output.txt file)</i>  <i>(1/2 mark for correct readlines from Input.txt)</i>  <i>(1/2 mark for correct looping statement)</i>  <i>(1/2 mark for correctly writing to Output.txt file)</i>  <i>(1/2 mark for correctly closing the Input.txt and Output.txt file)</i></p>	
30.	<p>A List containing records of Gadgets as:</p> <pre>L = [("SmartWatch", 5000), ("Earbuds", 2500), ("Charger", 800), ("PowerBank", 3500)]</pre> <p>Write the following User-Defined Functions to perform operations on a Stack named Product to:</p>	<b>3</b>

	<p>I) Push_gadget() – To push an item containing gadget name and price of gadgets costing more than 2000 into the Stack. Output: [('SmartWatch', 5000), ('Earbuds', 2500), ('PowerBank', 3500)]</p> <p>II) Pop_gadget() – To pop the items from the Stack and display them. Also, display "Stack Empty" when there are no elements in the Stack. Output: (('PowerBank', 3500) (('Earbuds', 2500) (('SmartWatch', 5000) Stack Empty</p>	
	<p><b>Answer:</b></p> <p>I) L = [("SmartWatch", 5000), ("Earbuds", 2500), ("Charger", 800), ("PowerBank", 3500)] product = [] def Push_gadget(L):     for i in L:         if i[1] &gt; 2000:             product.append(i)     print(product)</p> <p>II) def Pop_gadget(product):     while product:         print(product.pop())     else:         print("Stack Empty")</p> <p><i>(1½ marks for each correct part)</i></p>	
31.	<p>A) Predict the output of the following Python code: grades = {"Math": 85, "Science": 92, "English": 78} display = [ ] for subject, marks in grades.items():     display.append(subject.lower() + " : " + str(marks + 5) + " marks\n") final = "#".join(display) print(final)</p> <p style="text-align: center;"><b>OR</b></p> <p>B) Predict the output of the following Python code: value = 100 def process(word, factor=2):     global value     for letter in word:         if letter in 'BCDFGbcdfg':             value = value - 5             print(letter.lower(), '&amp;', factor + value)</p>	<b>3</b>

```

else:
    value *= 2
    print(letter.upper(), '*', factor * value)
text = 'CS'
process(text, 3)
print(value, '%', text)

```

**Answer:**

A) math : 90 marks  
 #science : 97 marks  
 #english : 83 marks  
*(1 mark for each correct line)*

**OR**

B) c & 98  
 S \* 570  
 190 % CS  
*(1 mark for each correct line)*

**Section-D (4 x 4 = 16 Marks)**

32. Consider the table ADMISSIONS as given below:

**4**

**Table: ADMISSIONS**

AdmissionID	PatientID	AdmissionDate	Department	TotalBill
1	101	2023-01-15	Surgery	1500.00
2	102	2023-02-01	Neurology	2000.00
3	103	2023-02-15	Physiotherapy	1000.00
4	104	2023-03-01	Surgery	3000.00
5	105	2023-03-15	Neurology	2500.00
6	106	2023-04-01	Neurology	1800.00
7	107	2023-04-15	Surgery	NULL
8	108	2023-05-01	Neurology	2200.00
9	109	2023-05-15	Physiotherapy	1700.00
10	110	2023-06-01	Surgery	2800.00

A) Write the following SQL queries:

- I) To display the patients' admissions details, sorted by their total bill in descending order.
- II) To display names of the department, having an average total bill amount greater than ₹2000.
- III) To display the admission details of patients who do not belong to the Surgery department.

IV) To display the ID of the patient, total bill and tax amount (10% of the TotalBill and column name should be TAX) for each patient.

**OR**

B) Write the output for the SQL queries:

I) SELECT ADMISSIONID, PATIENTID, ADMISSIONDATE FROM ADMISSIONS WHERE ADMISSIONDATE LIKE '2023-03%';

II) SELECT DEPARTMENT, COUNT(TOTALBILL) BILLCOUNT FROM ADMISSIONS GROUP BY DEPARTMENT;

III) SELECT ADMISSIONID, PATIENTID, ADMISSIONDATE FROM ADMISSIONS WHERE DEPARTMENT = 'NEUROLOGY' AND TOTALBILL > 2000;

IV) SELECT PATIENTID, ADMISSIONDATE FROM ADMISSIONS WHERE DEPARTMENT NOT IN ('NEUROLOGY', 'SURGERY');

**Answer:**

A)

I) SELECT \* FROM ADMISSIONS ORDER BY TOTALBILL DESC;

II) SELECT DEPARTMENT FROM ADMISSIONS GROUP BY DEPARTMENT HAVING AVG(TOTALBILL) > 2000;

III) SELECT \* FROM ADMISSIONS WHERE DEPARTMENT <> 'SURGERY';

IV) SELECT PATIENTID, TOTALBILL, TOTALBILL \* 0.1 AS TAX FROM ADMISSIONS;

*(4 x 1 mark for each correct query)*

**OR**

B)

I)

AdmissionID	PatientID	AdmissionDate
4	104	2023-03-01
5	105	2023-03-15

II)

Department	BillCount
Surgery	3
Neurology	4
Physiotherapy	2

III)

AdmissionID	PatientID	AdmissionDate
5	105	2023-03-15
8	108	2023-05-01

IV)

PatientID	AdmissionDate
103	2023-02-15
109	2023-05-15

*(4 x 1mark for each correct output)*

33.

Mr. Jaisan is a sports coordinator for a college who maintains data about athletes in a CSV file named Athletes.csv, which stores details of each athlete.

The columns of the CSV file are: Athlete\_ID, Name, Sport and MedalCount

Help him efficiently manage the data by writing User-Defined Functions in Python to:

- I) AddAthlete() – to accept athlete details from the user and add them to the file.
- II) DisplayTopPerformers() – to display the names and sports of athletes who have won more than 3 medals.

**4**

**Answer:**

I)

```
import csv
def AddAthlete():
    f = open('Athletes.csv', 'a', newline='')
    w = csv.writer(f)
    aid = input("Enter Athlete ID: ")
    name = input("Enter Name: ")
    sport = input("Enter Sport: ")
    medalc = int(input("Enter Medals: "))
    w.writerow([aid, name, sport, medalc])
    f.close()
    print("Record added successfully.")
```

*(1/2 mark for correctly taking user input)*  
*(1/2 mark for opening the file in append mode)*  
*(1/2 mark for correctly creating the writer object)*  
*(1/2 mark for correctly using writerow() of writer object)*

II)

```
import csv
def DisplayTopPerformers():
    f = open('Athletes.csv', 'r')
    r = csv.reader(f)
    print("Top Performers (Medals > 3):")
    for rec in r:
        if rec and rec[3].isdigit() and int(rec[3]) > 3:
            print(rec[1], "-", rec[2])
    f.close()
```

*(1/2 mark for opening in the file in right mode)*  
*(1/2 mark for correctly creating the reader object)*  
*(1/2 mark for correctly checking the condition)*  
*(1/2 mark for correctly displaying the names and events)*

*Note (for both parts (I) and (II)): Ignore import csv as it may be considered the part of the complete program.*

34.

Ms. Sheethal is the trusted employee at a highly reputed Airport Agency. She needs to access some information from AIRCRAFT and FLIGHT tables for a board meeting scheduled for performance analysis. Help her extract the following information by writing the desired SQL queries as mentioned below:

4

**Table: AIRCRAFT**

AircraftID	Manufacturer	Model	TotalSeats	ManufactureYear	MCost
VT-ANL	Boeing	787-8	256	2023-01-15	200000
VT-TTA	Airbus	A320neo3	180	2024-09-20	150000
VT-IFP	Airbus	A321	222	2024-01-25	210000
VT-SXA	Boeing	737 MAX	189	2023-05-18	160000
VT-WGC	Airbus	A320neo	180	2024-07-22	155000

**Table: FLIGHT**

FlightID	FlightNumber	AircraftID	Airline
AI107	107	VT-TTA	Air India
UK910	910	VT-SXA	Vistara
6E505	505	VT-ANL	IndiGo
G8202	202	VT-SXA	Go First

- I) To display the FlightID and Airline name having seat capacity less than 200.
- II) To display the details of AIRCRAFT that have the string 'neo' anywhere in the name of the model.
- III) To delete details of the flight from the table FLIGHT flying in 'IndiGo' airlines.
- IV) A) To display the count of distinct manufactures from the table AIRCRAFT.

**OR**

- B) To display each manufacturer's, maximum and minimum manufacturing costs of aircraft.

**Answer:**

- I) SELECT FLIGHTID, AIRLINE FROM AIRCRAFT, FLIGHT WHERE AIRCRAFT.AIRCRAFTID=FLIGHT.AIRCRAFTID AND TOTALSEATS<200;
- II) SELECT \* FROM AIRCRAFT WHERE MODEL LIKE '%NEO%';
- III) DELETE FROM FLIGHT WHERE AIRLINE='INDIGO';
- IV) A) SELECT COUNT(DISTINCT MANUFACTURER) FROM AIRCRAFT;

	<b>OR</b>	
	<p>B) SELECT MANUFACTURER, MIN(MCOST), MAX(MCOST) FROM AIRCRAFT GROUP BY MANUFACTURER;</p> <p><i>(4 x 1 mark for each correct query)</i></p>	
35.	<p>Mr. Jithesh has created a table named RESTAURANTS in MySQL. The fields of the table are:</p> <ul style="list-style-type: none"> <li>Rest_ID – ID of the Restaurant (String)</li> <li>Name – Name of the Restaurant (String)</li> <li>City – City of the Restaurant (String)</li> <li>Average_Cost – Average cost for two people (Float)</li> </ul> <p>Consider the following details to establish Python-MySQL connectivity:</p> <ul style="list-style-type: none"> <li>UserName – admin</li> <li>Password – admin@123</li> <li>Host – localhost</li> </ul> <p>The table RESTAURANTS exists in a MySQL database named FOOD_GUIDE. Rahul wants to display all records of the RESTAURANTS table where City is 'Delhi' and Average_Cost is less than 800. Help Rahul to write the program in Python.</p>	<b>4</b>
	<p><b>Answer:</b></p> <pre>import mysql.connector as DB def Restaurant_Database():     con = DB.connect(hostlocalhost,                     user="admin",                     password="admin@123",                     database="FOOD_GUIDE")      cursor = con.cursor()     print("Restaurants in Delhi with Average Cost &lt; 800.0:")     cursor.execute("SELECT * FROM RESTAURANTS WHERE City = 'Delhi' AND Average_Cost &lt; 800")     records = cursor.fetchall()      for rec in records:         print(rec)     con.close() Restaurant_Database()</pre> <p><i>(1/2 mark for correctly importing the connector object)</i>  <i>(1/2 mark for correctly creating the connection object)</i>  <i>(1/2 mark for correctly creating the cursor object)</i>  <i>(1 mark for correct creation of select query)</i>  <i>(1 mark for correctly fetching the records and displaying the result)</i>  <i>(1/2 mark for correctly closing the connection)</i></p>	
	<b>Section-E (2 X 5 = 10 Marks)</b>	
36.	<p>Ms. Pooja is working as an Inventory Manager at a Sports Equipment Store. She needs to manage the records of various sports items at the shop. For this, she wants the following information of each sports item to be stored:</p>	<b>2+3</b>

- Item\_ID – Integer
- Sport\_Type – String (e.g., 'Cricket', 'Football', 'Tennis', 'Badminton')
- Item\_Cost – Float
- Quantity – Integer

You, as a programmer of the shop, have been assigned to do this job for Pooja on a Binary file named SPORTS.DAT

- I) Write a function DISP\_ITEMS() to read the data from the Binary file and display the Item IDs of all 'Cricket' and 'Football' type sports items.
- II) Write a function UPDATE\_QUANTITY() to update the quantity by adding 10 units to all items whose current quantity is less than 5 in the Binary file.

**Answer:**

I)

```
import pickle
def DISP_ITEMS():
    file1 = open("SPORTS.DAT", "rb")
    flag = 0
    print("\nItem IDs of 'Cricket' and 'Football' Type Items:")
    try:
        while True:
            data = pickle.load(file1)
            if data[1] == "Cricket" or data[1] == "Football":
                print("Item ID:", data[0])
                flag = 1
    except EOFError:
        file1.close()
    if flag == 0:
        print("Sorry! No Cricket or Football items found")
DISP_ITEMS()
```

OR Any other correct variation of the code

*(½ Mark for correctly opening file in read and loading the data correctly)*

*(½ Mark for loading the file)*

*(½ Mark for checking the condition)*

*(½ Mark for printing the data correctly)*

II)

```
import pickle
import os

def UPDATE_QUANTITY():
    file1 = open("SPORTS.DAT", "rb")
    file2 = open("temp.dat", "wb")
    flag = 0
    try:
        while True:
            data = pickle.load(file1)
```

```

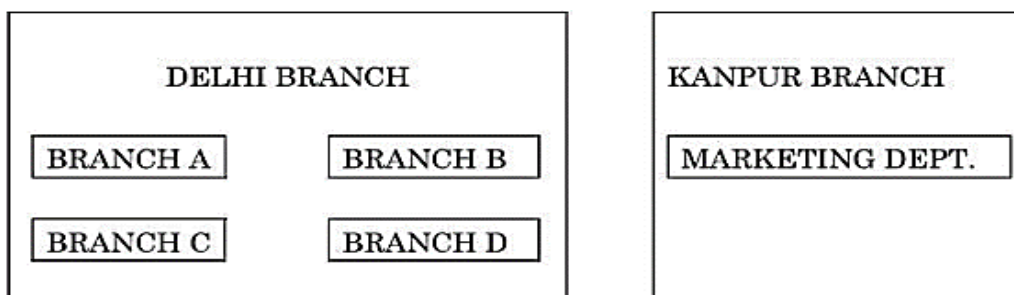
if data[3] < 5:
    data[3] = data[3] + 10
    flag = 1
    pickle.dump(data, file2)
except EOFError:
    file1.close()
    file2.close()
os.remove("SPORTS.DAT")
os.rename("temp.dat", "SPORTS.DAT")
if flag == 0:
    print("No items with quantity less than 5")
else:
    print("Quantities updated successfully for low-stock items")
UPDATE_QUANTITY()

```

OR Any other correct variation of the code  
*(1/2 mark for correctly defining the function header)*  
*(1/2 mark for correctly opening the files)*  
*(1 mark for using load() with while loop and try-except block)*  
*(1 mark for checking the condition and deleting the value)*

**Note (for both parts (I) and (II)): (i) Ignore import pickle as it may be considered the part of the complete program.**

37. Tiska Inc., an IT-based firm, located in Delhi is planning to setup a network for its four branches – BRANCH A, BRANCH B, BRANCH C AND BRANCH D within a city with its Marketing department in Kanpur. 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.



Distance between various branches is as follows:

Branch	Distance
Branch A to Branch B	40 m
Branch A to Branch C	80 m
Branch A to Branch D	65 m
Branch B to Branch C	30 m
Branch B to Branch D	35 m
Branch C to Branch D	15 m
Delhi Branch to Kanpur	300 km

5

Number of computers in each of the blocks/Center is as follows:

Branch	Number of Computers
Branch A	15
Branch B	25
Branch C	40
Branch D	115

- I) Suggest the most suitable place to install the server for the Delhi branch with a suitable reason.
- II) Suggest and draw an ideal cable layout for connecting all these branches within Delhi.
- III) Which device will you suggest that should be placed in each of these branches to efficiently connect all the computers within these branches?
- IV) Is there a requirement of a repeater in the given cable layout? Why/ Why not?
- V) A) Delhi firm is planning to connect to its Marketing department in Kanpur which is approximately 300 km away. Which type of network of LAN, MAN or WAN will be formed?

**OR**

- B) Suggest a protocol that shall be needed to provide help for transferring files between the Delhi and Kanpur branches.

**Answer:**

I)

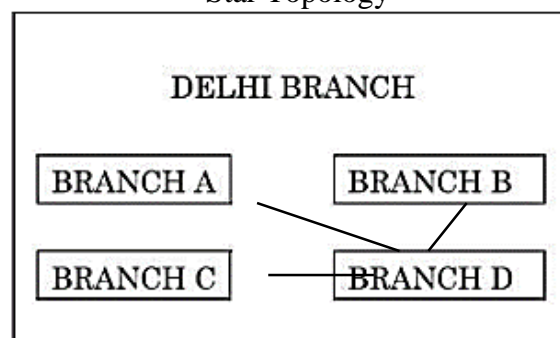
Suitable place to install server is Branch D, as it has the maximum number of Computers.

*(1/2 Mark for the correct answer)*

*(1/2 Mark for the valid justification)*

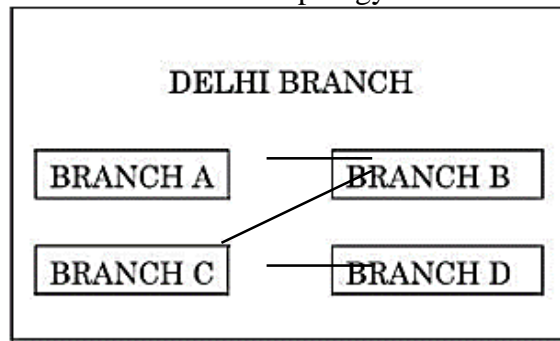
II)

Star Topology



**OR**

Bus Topology



*(½ Mark for suggesting the correct topology)*  
*(½ Mark for drawing the correct cable layout)*

III)

Hub OR Switch  
*(1 Mark for correct answer)*

IV)

There is no requirement of the Repeat in the given cable layout.  
Reason: Repeaters are placed to regenerate the signal periodically and hence extends the distance over which data can safely travel. Here the distance between the building is within the optimal meters.  
OR Any valid reason  
*(½ Mark for correct answer)*  
*(½ Mark for correct reason)*

V)

A) WAN  
*(1 Mark for correct answer)*

**OR**

B) FTP – File Transfer Protocol  
*(1 Mark for correct answer)*

*(5 x 1 mark for each correct part of the answer)*