

COMPUTER SCIENCE – Code No. 083
MARKING SCHEME
Class - XII - (2025-26)

Time Allowed: 3 Hrs.

Maximum 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	Answer: True (1 mark for correct answer)	1
2	Answer: c) It (1 mark for correct answer)	1
3	Answer: b) False (1 mark for correct answer)	1
4	Answer: Equi-Join or Cartesian Join (1 mark for correct answer)	1
5	Answer: a) ISf (1 mark for correct answer)	1
6	Answer: Error as unsupported operand type(s) for +: 'int' and 'str' (1 mark for the correct answer)	1
7	Answer: -59.0 (1 mark for the correct answer)	1

8	<p>Answer:</p> <p>SELECT department, COUNT(*) FROM employees GROUP BY department HAVING COUNT(*) > 5;</p> <p>(1 mark for correct answer)</p>	1
9	<p>Answer:</p> <p>b) Some other error!</p> <p>(1 mark for correct answer)</p>	1
10	<p>Answer:</p> <p>d) Not Specified</p> <p>(1 mark for correct answer)</p>	1
11	<p>Answer:</p> <p>a) 50@70@</p> <p>(1 mark for correct answer)</p>	1
12	<p>Answer:</p> <p>c) 5@@12##12</p> <p>(1 mark for correct answer)</p>	1
13	<p>Answer:</p> <p>c) Both a) & b)</p> <p>(1 mark for correct answer)</p>	1
14	<p>Answer:</p> <p>b) ['Watersk', ", 'ng ', 's thr', 'll', 'ng!']</p> <p>(1 mark for correct answer)</p>	1
15	<p>Answer:</p> <p>a) Degree: 7</p> <p>(1 mark for correct answer)</p>	1
16	<p>Answer:</p> <p>b) ALTER</p> <p>(1 mark for correct answer)</p>	1
17	<p>Answer:</p> <p>c) POP3</p> <p>(1 mark for correct answer)</p>	1
18	<p>Answer:</p> <p>a) A hub sends data to all devices in a network, while a switch sends data to the specific device</p> <p>(1 mark for correct answer)</p>	1

19	Answer: b) HTML (1 mark for correct answer)	1
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	Answer: d) A is False but R is True (1 mark for correct answer)	1
21	Answer: a) Both A and R are true and R is the correct explanation for A (1 mark for correct answer)	1
Q No.	Section-B (7 x 2=14 Marks)	Marks
22	Answer: 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 (1 mark for correct difference) (1/2 mark for each correct example) <p style="text-align: center;">OR</p> B. Break exits the loop entirely, while continue skips the current iteration and moves to the next one. Example of break: for i in range(5): if i == 2: break # Exits the loop print(i) Output: 1 Example of continue: for i in range(5): if i == 2:	2

	<pre> continue # Skips printing 2 print(i) Output: 1 3 4 5 </pre> <p>(1 mark for correct difference) (1/2 mark for each correct example)</p>	
23	<p>Answer:</p> <pre> def remove_first_last(str): if len(str) < 2: return str new_str = str[1:-1] return new_str result = remove_first_last("Hello") print("Resulting string: ", result) </pre> <p>(1/2 mark each for correcting 4 mistakes)</p>	2
24	<p>Answer:</p> <p>A.</p> <ol style="list-style-type: none"> I. index = review.find("good") II. L1.sort(reverse=True) <p style="text-align: center;">OR</p> <p>B. ('Learn Python ', 'with', ' fun and practice')</p> <p style="text-align: center;">3</p> <p>(1 mark for correct answer) (1 mark for correct answer) (1 mark for correct answer) (1 mark for correct answer)</p>	2
25	<p>Answer:</p> <p>A.</p> <pre> def remove_element(L, n): if n in L: L.remove(n) print(L) else: print("Element not found") </pre> <p style="text-align: center;">OR</p> <p>B.</p> <pre> def add_contact(phone_book, name, number): if name in phone_book: print("Contact already exists") else: phone_book[name] = number print("Contact added successfully") </pre> <p>(1/2 mark for function definition) (1½ marks for the correct/similar logic)</p>	2
26	<p>Answer:</p> <pre> ['Arv', 'Ria'] </pre> <p>(2 marks for the correct output)</p>	2

27	<p>Answer:</p> <p>A.</p> <ol style="list-style-type: none"> I. Desc table_name; or describe table_name; II. Create database SQP; <p>(1 mark for each correct answer.)</p> <p style="text-align: center;">OR</p> <p>B. The DELETE query removes all the records or specific records from a table, preserving the table structure. Example: DELETE FROM Employees WHERE EmployeeID = 5; The DROP query removes the entire table or database along with its data. Example: DROP TABLE Employees;</p> <p>(1 mark for correct difference) (1/2 mark for each correct example)</p>	2
28	<p>Answer:</p> <p>A.</p> <ol style="list-style-type: none"> I. A modem is a device that helps connect your computer or other devices to the internet. It converts digital signals from your device into analog signals that can travel through phone lines or other networks, and vice versa. II. A gateway is a device that connects two different networks and helps them communicate with each other. It translates the data between different network types, allowing them to work together. <p>(1 mark for each correct definition)</p> <p style="text-align: center;">OR</p> <p>B.</p> <ol style="list-style-type: none"> I. HTTP: Hypertext Transfer Protocol and FTP: File Transfer Protocol (1/2 mark for each correct expansion.) II. A web server stores and delivers web pages to users over the internet. A web browser requests and displays these web pages on the user's device. <p>(1 mark for correct point of difference)</p>	2
Q No.	Section-C (3 x 3 = 9 Marks)	Marks
29	<p>Answer:</p> <p>A.</p> <pre>def count_python(): count = 0 with open("Prog.txt", 'r') as file: text = file.read() words = text.split() for word in words: if word.lower() == "python": count += 1 print("The word Python appears", count, "times.")</pre> <p>(1/2 mark for correct function header) (1/2 mark for correctly opening the file) (1/2 mark for correctly reading from the file) (1/2 mark for splitting the text into words) (1/2 mark for correct use of counter variable)</p>	3

	<p>(1/2 mark for displaying the result)</p> <p style="text-align: center;">OR</p> <p>B.</p> <pre>def display_non_vowel_lines(): with open("STORIES.TXT", "r") as file: print("Lines that don't start with a vowel:") lines = file.readlines() for line in lines: if line[0].lower() not in 'aeiou': print(line) display_non_vowel_lines()</pre> <p>(1/2 mark for correct function header) (1/2 mark for correctly opening the file) (1/2 mark for correctly reading from the file) (1 mark for correctly displaying the desired lines) (1/2 mark for correctly calling the function)</p>	
30	<p>Answer:</p> <p>I.</p> <pre>L = [("Laptop", 90000), ("Mobile", 30000), ("Pen", 50), ("Headphones", 1500)] product = [] def Push_element(L): for i in L: if i[1] > 50: product.append(i) print(product)</pre> <p>II.</p> <pre>def Pop_element(product): while product: print(product.pop()) else: print("Stack Empty")</pre> <p>(1½ marks for each correct part)</p>	3
31	<p>Answer:</p> <p>A. QP-¹⁴</p> <p>(3 marks for the correct output)</p> <p style="text-align: center;">OR</p> <p>B. ['K', 'R']</p> <p>(3 marks for the correct output)</p>	3
Q No.	Section-D (4 x 4 = 16 Marks)	Marks
32	<p>Answer:</p> <p>A.</p> <ol style="list-style-type: none"> I. SELECT Product, SUM(Quantity_Sold) FROM SALES GROUP BY Product HAVING SUM(Quantity_sold) > 12; II. SELECT * FROM SALES ORDER BY Product DESC; III. SELECT DISTINCT Product FROM SALES; IV. SELECT * from SALES where Customer_Name like "%e"; 	4

(4 x 1 mark for each correct query)

OR

B.

I.

sales_id	customer_name	product	quantity_sold	price
S003	Michael Lee	Tablet	3	15000
S007	Mark	Tablet	5	34000

II.

sales_id	customer_name
S002	Jane Smith
S005	Emily Davis
S006	David

III.

COUNT(*)
3

IV.

AVG(price)
24500.0000

(4 x 1 mark for each correct query)

33

Answer:

I.

```
import csv
def Accept():
    product_id = input("Enter Product ID: ")
    product_name = input("Enter Product Name: ")
    quantity_sold = int(input("Enter Quantity Sold: "))
    price_per_unit = float(input("Enter Price Per Unit: "))
    with open('Sales.csv', 'a', newline='') as file:
        writer = csv.writer(file)
        writer.writerow([product_id, product_name, quantity_sold, price_per_unit])
    print("Sales 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)

4

	<p>(1/2 mark for correctly using writerow() of writer object)</p> <p>II.</p> <pre>def CalculateTotalSales(): total_sales = 0.0 with open('Sales.csv', 'r') as file: reader = csv.reader(file) for row in reader: total_sales += int(row[2]) * float(row[3]) print("Total Sales is:", total_sales)</pre> <p>(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 total sales)</p> <p>Note (for both parts (I) and (II)): Ignore import csv as it may be considered the part of the complete program.</p>	
34	<p>Answer:</p> <p>I. SELECT Customer_Name FROM Hotels, Bookings WHERE Hotels.H_ID = Bookings.H_ID AND City = 'Delhi';</p> <p>II. SELECT Bookings.* FROM Hotels, Bookings WHERE Hotels.H_ID = Bookings.H_ID AND City IN ('Mumbai', 'Chennai', 'Kolkata');</p> <p>III. DELETE FROM Bookings WHERE Check_In < '2024-12-03';</p> <p>IV. A. SELECT * FROM Hotels, Bookings;</p> <p style="text-align: center;">OR</p> <p>B. SELECT Customer_Name, Hotel_Name FROM Hotels, Bookings WHERE Hotels.H_ID = Bookings.H_ID;</p> <p>(4 x 1 mark for each correct query)</p>	4
35	<p>Answer:</p> <pre>import mysql.connector connection = mysql.connector.connect(host='localhost',user='admin_user',password='warehouse 2024',database='WarehouseDB') cursor = connection.cursor() update_query = "UPDATE product_inventory SET Quantity = 91 WHERE Item_code = 208" cursor.execute(update_query) connection.commit() print("Data updated successfully.") cursor.close() connection.close()</pre> <p>(1/2 mark for correctly importing the connector object) (1/2 mark for correctly creating the connection object) (1/2 mark for correctly creating the cursor object)</p>	4

	<p>(1 mark for correct creation of update query) (1 mark for correctly executing the query with commit) (1/2 mark for correctly closing the connection)</p>	
Q No.	Section-E (2 X 5 = 10 Marks)	Marks
36	<p>Answer:</p> <p>I. import pickle def append_data(): with open("emp.dat", 'ab') as file: employee_id = int(input("Enter Employee ID: ")) employee_name = input("Enter Employee Name: ") department = input("Enter Department: ") salary = float(input("Enter Salary: ")) pickle.dump([employee_id, employee_name, department, salary], file) print("Employee data appended successfully.")</p> <p>(1/2 mark for correctly defining the function header) (1/2 mark for correctly opening the file in append mode) (1/2 mark for correctly taking user input) (1/2 mark for using dump() method of the pickle module)</p> <p>II. def update_data(): updated = False employees = [] with open("emp.dat", 'rb') as file: try: while True: employee = pickle.load(file) if employee[2] == "IT": employee[3] = 200000 updated = True employees.append(employee) except EOFError: pass with open("emp.dat", 'wb') as file: for employee in employees: pickle.dump(employee, file) if updated: print("Salaries updated for IT department.") else: print("No employee found in the IT department.")</p> <p>(1/2 mark for correctly defining the function header) (1/2 mark for correctly opening the file) (1 mark for using load() with while loop and try-except block) (1 mark for checking the condition and updating the value)</p>	2+3

	<p>Note: Note (for both parts (I) and (II)): (i) Ignore import pickle as it may be considered the part of the complete program.</p>	
37	<p>Answer:</p> <ol style="list-style-type: none"> I. Block IT should house the server as it has maximum number of computers. II. <ol style="list-style-type: none"> a) Repeater is to be placed between Block IT to Block HR as distance between them is more than 100 metres. b) Switch is to be placed in each and every building. III. Draw the star topology cable layout. <div style="text-align: center; margin: 10px 0;"> <pre> graph TD IT[IT] --- HR[HR] IT --- Finance[Finance] IT --- Logistics[Logistics] </pre> </div> <ol style="list-style-type: none"> IV. Optical Fibre V. <ol style="list-style-type: none"> A. 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. <p style="text-align: center;">OR</p> <ol style="list-style-type: none"> B. WAN will be formed. <p>(5 x 1 mark for each correct part)</p>	5