

Central Board of Secondary Education

SAMPLE PAPER

Class 12
2024

Solutions



Marking Scheme

Class XII

Computer Science (083)

Time Allowed: 3 hours

MM: 70

<u>Ques No</u>	Question and Answers	Distribution of Marks	Total Marks
<u>SECTION A</u>			
1	False	1 mark for correct answer	1
2	Option b 6,20	1 mark for correct answer	1
3	Option c -244.0	1 mark for correct answer	1
4	PYTHON-is-Fun	1 mark for correct answer	1
5	Option b 8,15	1 mark for correct answer	1
6	Option a PAN	1 mark for correct answer	1
7	Option b <code>del D1["Red"]</code>	1 mark for correct answer	1
8	Option b	1 mark for correct answer	1

	ceieP0		
9	Option d Statement 4	1 mark for correct answer	1
10	Option b YELLOW* WHITE* BLACK* RED*	1 mark for correct answer	1
11	Option b Modulator	1 mark for correct answer	1
12	Option c global b	1 mark for correct answer	1
13	True	1 mark for correct answer	1
14	Option c A candidate key that is not a primary key is a foreign key.	1 mark for correct answer	1
15	circuit	1 mark for correct answer	1
16	Option c seek ()	1 mark for correct answer	1

	<pre> rem = num % 10 rev = rev * 10 + rem num = num // 10 <u>return rev</u> print (revNumber (1234)) </pre>	correction made	
21	<pre> PLACES={1:"Delhi",2:"London",3:"Paris",4:"New York",5:"Dubai"} def countNow(PLACES): for place in PLACES.values(): if len(place)>5: print(place.upper()) countNow(PLACES) OR def lenWords (STRING) : T= () L=STRING.split() for word in L: length=len(word) T=T+(length,) return T </pre> <u>Note: Any other correct logic may be marked</u>	½ mark for correct function header ½ mark for correct loop ½ mark for correct if statement ½ mark for displaying the output ½ mark for correct function header ½ mark for using split() ½ mark for adding to tuple ½ mark for return statement	2

22	4*L 33*4 21*S 10*6	½ mark for each correct line of output	2
23	(i) L1.insert(2,200) (ii) message.endswith('.')	1 mark for each correct statement	1+1=2
24	SQL Command to add primary key: ALTER TABLE Employee ADD EmpId INTEGER PRIMARY KEY; As the primary key is added as the last field, the command for inserting data will be: INSERT INTO Employee VALUES("Shweta","Production",26900,999); OR INSERT INTO Employee(EmpId,Ename,Department,Salary) VALUES(999,"Shweta","Production",26900);	1 mark for correct ALTER TABLE command 1 mark for correct INSERT command	2
25	10.0\$20 10.0\$2.0###	1 mark for each correct line of output	2
<u>SECTION C</u>			
26	ND-*34	½ mark for each correct character	3
27			

	(i) <table border="1"><tr><td>COUNT (DISTINCT SPORTS)</td></tr><tr><td>4</td></tr></table> (ii) <table border="1"><tr><td>CNAME</td><td>SPORTS</td></tr><tr><td>AMINA</td><td>CHESS</td></tr></table> (iii) <table border="1"><tr><td>CNAME</td><td>AGE</td><td>PAY</td></tr><tr><td>AMRIT</td><td>28</td><td>1000</td></tr><tr><td>VIRAT</td><td>35</td><td>1050</td></tr></table>	COUNT (DISTINCT SPORTS)	4	CNAME	SPORTS	AMINA	CHESS	CNAME	AGE	PAY	AMRIT	28	1000	VIRAT	35	1050	1 mark for each correct output	1*3=3
COUNT (DISTINCT SPORTS)																		
4																		
CNAME	SPORTS																	
AMINA	CHESS																	
CNAME	AGE	PAY																
AMRIT	28	1000																
VIRAT	35	1050																
28	<pre>def test(): fObj1 = open("Alpha.txt", "r") data = fObj1.readlines() for line in data: L=line.split() if L[0]=="You": print(line) fObj1.close()</pre> OR	1 mark for correctly opening and closing files ½ mark for correctly reading data 1 mark for correct loop and if statement ½ mark for displaying data	3															

	<pre> def vowelCount(): fObj = open("Alpha.txt", "r") data = str(fObj.read()) cnt=0 for ch in data: if ch in "aeiouAEIOU": cnt=cnt+1 print(cnt) fObj.close() </pre> <u>Note: Any other correct logic may be marked</u>	1 mark for correctly opening and closing the files ½ mark for correctly reading data 1 mark for correct loop and if statement ½ mark for displaying the output.	
29	(i) <pre> UPDATE Personal SET Salary=Salary*0.5 WHERE Allowance IS NOT NULL; </pre> (ii) <pre> SELECT Name, Salary+Allowance AS "Total Salary" FROM Personal; </pre> (iii) <pre> DELETE FROM Personal WHERE Salary>25000 </pre>	1 mark for each correct query	1*3=3

30	<pre> travel = [] def Push_element(NList): for L in NList: if L[1] != "India" and L[2]<3500: travel.append([L[0],L[1]]) def Pop_element(): while len(travel): print(travel.pop()) else: print("Stack Empty") </pre>	1 ½ marks for each function	3
----	---	-----------------------------	---

SECTION D

31	a) Bus Topology b) Switch c) Admin block, as it has maximum number of computers. d) Microwave e) Firewall	1 mark for each correct answer	1*5=5
32	(i) r+ mode: • Primary function is reading • File pointer is at beginning of file • if the file does not exist, it results in an error w+ mode:	1 mark for each correct difference (minimum two differences should be given)	2+3=5

	• primary function is writing • if the file does not exist, it creates a new file. • If the file exists, previous data is overwritten • File pointer is at the beginning of file (ii) <pre>def copyData(): fObj = open("SPORT.DAT", "rb") fObj1 = open("BASKET.DAT", "wb") cnt=0 try: while True: data = pickle.load(fObj) print(data) if data[0] == "Basket Ball": pickle.dump(data, fObj1) cnt+=1 except: fObj.close() fObj1.close() return cnt</pre> OR (Only for option ii) <pre>def Searchtype(mtype): fObj = open("CINEMA.DAT", "rb") try: while True: data = pickle.load(fObj) if data[2] == mtype: print("Movie number:", data[0]) print("Movie Name:", data[1]) print("Movie Type:", data[2]) except EOFError: fObj.close()</pre>	½ mark for correctly opening and closing files ½ mark for correct try and except block ½ mark for correct loop 1 mark for correctly copying data ½ mark for correct return statement	
		½ mark for correctly opening and closing files ½ mark for correct try and except block ½ mark for correct loop	

	<u>Note: Any other correct logic may be marked</u>	½ mark for correct if statement 1 mark for correctly displaying data	
33	(i) Domain is a set of values from which an attribute can take value in each row. For example, roll no field can have only integer values and so its domain is a set of integer values (ii) <pre>import mysql.connector as mysql con1 = mysql.connect(host="localhost",user="root", password="tiger", database="sample2023") mycursor=con1.cursor() rno = int(input("Enter Roll Number:: ")) name = input("Enter the name:: ") DOB = input("Enter date of birth:: ") fee= float(input("Enter Fee:: ")) query = "INSERT into student values({},'{}','{}',{})".format(rno,name,DOB,fee) mycursor.execute(query) con1.commit() print("Data added successfully") con1.close()</pre> <u>Note: Any other correct logic may be marked</u>	½ mark for correct definition ½ mark for correct example ½ mark for importing correct module 1 mark for correct connect() ½ mark for correctly accepting the input 1 ½ mark for correctly executing the query ½ mark for correctly using commit()	1+4=5

SECTION E

34	(i) <pre>SELECT PName, BName FROM PRODUCT P, BRAND B WHERE P.BID=B.BID;</pre> (ii) <pre>DESC PRODUCT;</pre> (iii) <pre>SELECT BName, AVG(Rating) FROM PRODUCT P, BRAND B WHERE P.BID=B.BID GROUP BY BName HAVING BName='Medimix' OR BName='Dove';</pre> (iv) <pre>SELECT PName, UPrice, Rating FROM PRODUCT ORDER BY Rating DESC;</pre>	1 mark for each correct query	1*4=4
35	<pre>def Accept(): sid=int(input("Enter Student ID ")) sname=input("Enter Student Name ") game= input("Enter name of game ") res=input("Enter Result") headings=["Student ID","Student Name"," Game Name", "Result"] data=[sid,sname,game,res] f=open('Result.csv','a',newline='') csvwriter=csv.writer(f) csvwriter.writerow(headings) csvwriter.writerow(data) f.close()</pre>	½ mark for accepting data correctly ½ mark for opening and closing file ½ mark for writing headings ½ mark for writing row	4

	<pre> def wonCount(): f=open('Result.csv','r') csvreader=csv.reader(f, delimiter=',') head=list(csvreader) print(head[0]) for x in head: if x[3]=="WON": print(x) f.close() </pre>	½ mark for opening and closing file ½ mark for reader object ½ mark for print heading ½ mark for printing data	
--	--	--	--