

PREBOARD EXAMINATION 1-(2024-25)

CLASS: XII

COMPUTER SCIENCE (083)

SET-B(Marking Scheme)

Time allowed: 3 Hours

Maximum Marks: 70

Q No.	Section-A (21 x 1 = 21 Marks)	Marks
	Q1 to Q21 (1 mark for each correct answer)	
1.	True	(1)
2.	c)9.5	(1)
3.	d)myfile.readlines()	(1)
4.	b)23	(1)
5.	Identity	(1)
6.	d)(40,60)	(1)
7.	False	(1)
8.	b)top	(1)
9.	b)It is Case Sensitive	(1)
10.	a)fp.seek(offset,0)	(1)
11.	d) more than zero	(1)
12.	c) invalid code	(1)
13.	c) (23, 56)	(1)
14.	Comma	(1)
15.	b)_(underscore) and %	(1)
16.	c)4<2<1<3	(1)
17.	b) Bus	(1)
18.	b) Uninterrupted Power Supply (UPS).	(1)
19.	b) transmission control protocol	(1)
20.	(c) A is True but R is False	(1)
21.	(a)Both A and R are true and R is the correct explanation for A	(1)
	SECTION B (7 X 2 =14)	
22.	<pre>def vowelConso(sen): vowel=[] conso=[] for w in sen.split(): if w[-1] not in 'aeiou': conso.append(w) elif w[-1] in 'aeiou': vowel.append(w) print("List of words ending with vowels", vowel) print("List of words ending with consonants", conso) sent=input("enter the sentence") vowelConso(sent) (½ mark for correct function header) (½ mark for splitting the text into words) (1 mark for applying correct)</pre>	(2)
23.	A foreign key is a column or columns in a table that references the unique data values in another table. <pre>CREATE TABLE Orders (OrderID int NOT NULL, OrderNumber int NOT NULL, PersonID int, PRIMARY KEY (OrderID), FOREIGN KEY (PersonID) REFERENCES Persons(PersonID));</pre> (1 mark for correct definition & 1 mark for example)	(2)

24.	((1, 2, 3), (4, 5, 6)) OR {1: 1, 2: 4, 3: 1, 4: 1} (2 mark for correct output)	(2)						
25.	(i) bbcc (ii) cc 4 4 (1mark for each correct output)	(2)						
26.	<u>def</u> Update(x=10): x+=15 print(('x=' , x) <u>x=20</u> Update() print('x=',x) (½ mark for each correction made)	(2)						
27.	I)a) The UPDATE command is used to update existing rows or data in a table. Eg :- UPDATE Customers SET ContactName = 'Alfred Schmidt', City= 'Frankfurt' WHERE CustomerID = 1; (1 mark for correct definition and 1 mark for example) OR b) (i) SQL is easy to learn. (ii) SQL is used to access data from relational database management systems. (1 mark for each correct characteristics 2marks allotted if other valid characteristics defined) II)a) COUNT(*) returns the count of all rows in the table, whereas COUNT () is used with Column_Name passed as argument and counts the number of non-NULL values in a column that is given as argument. (1marks for each correct difference) OR b) <table><tr><td>NATURAL JOIN</td><td>CROSS JOIN</td></tr><tr><td>Natural Join joins two tables based on same attribute name and datatypes.</td><td>Cross Join will produce cross or cartesian product of two tables .</td></tr><tr><td>In Natural Join, The resulting table will contain all the attributes of both the tables but keep only one copy of each common column</td><td>In Cross Join, The resulting table will contain all the attribute of both the tables including duplicate columns also</td></tr></table> (1marks for each correct difference)	NATURAL JOIN	CROSS JOIN	Natural Join joins two tables based on same attribute name and datatypes.	Cross Join will produce cross or cartesian product of two tables .	In Natural Join, The resulting table will contain all the attributes of both the tables but keep only one copy of each common column	In Cross Join, The resulting table will contain all the attribute of both the tables including duplicate columns also	(2)
NATURAL JOIN	CROSS JOIN							
Natural Join joins two tables based on same attribute name and datatypes.	Cross Join will produce cross or cartesian product of two tables .							
In Natural Join, The resulting table will contain all the attributes of both the tables but keep only one copy of each common column	In Cross Join, The resulting table will contain all the attribute of both the tables including duplicate columns also							
28.	(i) Satellite Communication (1 mark for correct answer) (ii) Bluetooth or infrared whichever is supported by the phone. (1 mark for correct answer) OR (i)IP Address 59.177.134.72 (1 mark for correct answer) (ii) URL http://www.ccc.nic.in/welcome.htm (1 mark for correct answer)	(2)						
SECTION C (3 X 3 = 9)								
29.	myfile= open("Answer.txt","r") newfile= open("nswer.txt","w") line= " " while line: line= myfile.readLine()	(3)						

	<pre> if 'a' not in line: newfile.write(line) myfile.close() newfile.close() print("Newly created file contains") print("-----") newfile=open("nswer.txt","r") line="" while line: line= newfile.readLine() print(line) newfile.close() (½ mark for correctly opening the file) (½ mark for correctly reading from the file) (1 mark for valid conditions) (1 mark for correctly displaying the desired text in new file) </pre> OR <pre> myfile=open("Answer.txt","r") ch="" lcount=0 ucount=0 while ch: ch=myfile.read(1) if ch.isupper()==True: ucount = ucount +1 else: lcount=lcount+1 print("Uppercase letters in the file: ", ucount) print("Lowercase letters in the file: ", lcount) myfile.close() (½ mark for correctly opening the file) (½ mark for correctly reading from the file) (1 mark for valid conditions) (1 mark for correctly displaying the desired text) </pre>	
30.	<pre> def Push(stk,item): stk.append(item) top = len(stk)-1 def Display(stk): if stk == []: print("stack empty") else: top =len(stk)-1 print(stk[top],"<-top") for a in range(top-1,-1,-1): print(stk[a]) Stack =[] top = None while True: print("Stack Operations") print("1. Push") print("2. Display Stack") print("3. Exit") ch = int(input("Enter your choice(1-3):")) </pre>	(3)

	<pre> if ch==1: bno = int(input("enter book no:")) bname= input("Enter Book Name:") item = [bno,bname] Push(Stack,item) elif ch==2: Display(Stack) elif ch ==3: break else: print("Invalid Choices") </pre> (1 mark for correctly adding data to stack) (½ mark for correctly popping data on the stack and ½ mark for checking condition) (½ mark for correctly displaying the data) (½ mark for function call statements) OR <pre> Status=[] def pushelement(cust): If cust[2]=='Goa': L1=[cust[0],cust[1]] Status.append(L1) def popelement(): num=len(Status) while len(Status)!=0: dele=Status.pop() print(dele) num=num-1 else: print("Stack empty") </pre> (1 mark for correctly adding data to stack) (1 mark for correctly popping data on the stack and ½ mark for checking condition) (½ mark for showing stack empty)	
31.	<pre> {'Apple': 1} {'Apple': 1, 'Banana': 1} {'Apple': 1, 'Banana': 1, 'apple': 1} {'Apple': 1, 'Banana': 2, 'apple': 1} 3 </pre> (½ mark for each correct line of output & 1 mark for displaying correct length) OR <pre> U * i * e * Nations a * i * n * </pre> (2 marks for displaying correct character & string at correct position) (1 mark for displaying correct * at correct position)	(3)
	SECTION D (4 X 4 = 16)	
32.	(a) SELECT PCode, PName, UPrice FROM Product ORDER BY PName DESC (½ mark for select -from part and ½ mark for where part) (b) ALTER TABLE Product ADD COLUMN Discount float; (½ mark for Alter table part and ½ marks for adding column) (c) UPDATE Product SET Discount = IF(UPrice >100, (UPrice *(10/100))+UPrice,0); (½ mark for Update -Set part and ½ marks for where) (d) SELECT Manufacturer, COUNT(*) AS TotalProducts FROM Product GROUP BY Manufacturer; (½ mark for select -from part and ½ mark for group part)	(4)

	OR (a)<table><tr><td>PName</td><td>avg(UPrice)</td></tr><tr><td>WASHING POWDER</td><td>120.0000</td></tr><tr><td>TOOTHPASTE</td><td>59.5000</td></tr><tr><td>SOAP</td><td>31.5000</td></tr><tr><td>SHAMPOO</td><td>245.0000</td></tr></table> (b) Manufacturer SURF COLGATE LUX PEPSODENT DOVE (c) COUNT(DISTINCT PName) 4 (d)<table><tr><td>PName</td><td>MAX(UPrice)</td><td>MIN(UPrice)</td></tr><tr><td>WASHING POWDER</td><td>120</td><td>120</td></tr><tr><td>TOOTHPASTE</td><td>65</td><td>54</td></tr></table> (1mark for each correct output)	PName	avg(UPrice)	WASHING POWDER	120.0000	TOOTHPASTE	59.5000	SOAP	31.5000	SHAMPOO	245.0000	PName	MAX(UPrice)	MIN(UPrice)	WASHING POWDER	120	120	TOOTHPASTE	65	54	
PName	avg(UPrice)																				
WASHING POWDER	120.0000																				
TOOTHPASTE	59.5000																				
SOAP	31.5000																				
SHAMPOO	245.0000																				
PName	MAX(UPrice)	MIN(UPrice)																			
WASHING POWDER	120	120																			
TOOTHPASTE	65	54																			
33.	import csv f = open("Sports.csv","w") wobj = csv.writer(f, delimiter = '\t') wobj.writerow(['Sport', 'Competitions', 'Prizes Won']) ans = 'y' i = 1 while ans == 'y': print("Record :", i) sport = input("Sport Name :") comp = int(input("No. of competitions participated :")) prize = int(input("Prizes won:")) record = [sport,comp,prize] wobj.writerow(record) i += 1 ans = input("Do u want to continue ? (y/n) :") f.close() (½ mark for opening in the file in right mode) (1mark for correctly creating write object) (1mark for correctly checking the condition) (1mark for writing records in csv) (½ mark for closing file)	(4)																			
34.	i)Select * from TRANSACT where TYPE= 'Withdraw'; (½ mark for select -from part and ½ mark for where part) ii)Select ANO, AMOUNT from TRANSACT where DOT like '%-05-%'; (½ mark for select -from part and ½ mark for where part) iii)Select ANAME from ACCOUNT where Address in (Chennai,Chandigarh); (½ mark for select -from part and ½ mark for where part) iv)a)Select ANO,ANAME, AMOUNT ,DOT from ACCOUNT A, TRANSACT T where A.ANO=T.ANO and AMOUNT<=3000; (½ mark for select -from part and ½ mark for where part) OR b)8,25 (1 mark for correct answer)																				
35.	Fetchone() returns the next row from the result set as tuple. If there are no more to retrieve, None is returned	(4)																			

	For e.g. <pre>import mysql.connector as mysql def sql_data(): con1=mysql.connect(host="localhost",user="root", password="tiger", database="PythonDB") mycursor=con1.cursor() try: mycursor.execute("select name, rollno, address, percentage from student") display=mycursor.fetchone() print (display) except: con1.rollback() con1.Close() (1 mark for correct definition) (½ mark for correctly importing the connector object) (½ mark for correctly creating the connection object) (½ mark for correctly creating the cursor object) (½mark for correct query) (½ mark for correctly executing the query) (½ mark for correctly displaying the data)</pre>	
	SECTION E (2 X 5 = 10)	
36.	<pre>Import pickle emp={} found= false fin=open("emp.dat",rb+) fin.seek(0) try: while True : rpos=fin.tell emp=pickle.load(fin) if emp['empno']==1251: emp['salary']=emp['salary']+2000 fin.seek(rpos) pickle.dump(emp,fin) print(emp) found=True except EOFError: if found==False: print("sorry, no matching record") else: print("Record successfully updated") fin.close() (½ mark of import pickle) (½ mark for opening file in correct mode and 1mark for using dump &load) (1 mark for using tell & seek) (1 mark for checking the condition and updating the value) (1 mark for checking the condition and displaying data correctly)</pre>	(5)
37.	(a) The most suitable topology is star topology to connect the computers of all wings. (b) The server should be installed in Wing A because Wing A has the highest number of computers (150), and this will help reduce network traffic. (c) The Hub/Switch or should be placed in all the wings. (d) The economic way to provide internet(e accessibility to all the wings is to use the proxy server at Wing A and connect to the internet through a dial-up network. (e) Two popular Open Source Software Operating Systems are Linux and Open Solaris. OR Malicious software that can harm a system if not detected early. (1mark for each correct answer)	(5)

