CBSE AISSCE 2024 Marking Scheme for Computer Science

(Series &RQPS Sub Code: 083 Q.P. Code 91) SET-4

Marking Scheme

Strictly Confidential

(For Internal and Restricted use only)

Senior Secondary School Certificate Examination, 2024

Subject Name: Computer Science (Q.P. CODE 91)

1	You are aware that evaluation is the most important process in the actual and correct assessment of the candidates. A small mistake in evaluation may lead to serious problems which may affect the future of the candidates, education system and teaching profession. To avoid mistakes, it is requested that before starting evaluation, you must read and understand the spot evaluation guidelines carefully.
2	"Evaluation policy is a confidential policy as it is related to the confidentiality of the examinations conducted, Evaluation done and several other aspects. Its' leakage to public in any manner could lead to derailment of the examination system and affect the life and future of millions of candidates. Sharing this policy/document to anyone, publishing in any magazine and printing in News Paper/Website etc. may invite action under various rules of the Board and IPC."
3	Evaluation is to be done as per instructions provided in the Marking Scheme. It should not be done according to one's own interpretation or any other consideration. Marking Scheme should be strictly adhered to and religiously followed. However, while evaluating, answers which are based on latest information or knowledge and/or are innovative, they may be assessed for their correctness otherwise and due marks be awarded to them. In class-XII, while evaluating two competency-based questions, please try to understand given answer and even if reply is not from marking scheme but correct competency is enumerated by the candidate, due marks should be awarded.
4	The Marking scheme carries only suggested value points for the answers These are in the nature of Guidelines only and do not constitute the complete answer. The students can have their own expression and if the expression is correct, the due marks should be awarded accordingly.
5	The Head-Examiner must go through the first five answer books evaluated by each evaluator on the first day, to ensure that evaluation has been carried out as per the instructions given in the Marking Scheme. If there is any variation, the same should be zero after deliberation and discussion. The remaining answer books meant for evaluation shall be given only after ensuring that there is no significant variation in the marking of individual evaluators.
6	Evaluators will mark(√) wherever answer is correct. For wrong answer CROSS 'X' be marked. Evaluators will not put right (✓)while evaluating which gives an impression that answer is correct and no marks are awarded. This is most common mistake which evaluators are committing.
7	If a question has parts, please award marks on the right-hand side for each part. Marks awarded for different parts of the question should then be totaled up and written in the left-hand margin and encircled. This may be followed strictly.
8	If a question does not have any parts, marks must be awarded in the left-hand margin and encircled. This may also be followed strictly.
9	If a student has attempted an extra question, answer of the question deserving more marks should be retained and the other answer scored out with a note "Extra Question".
10	No marks to be deducted for the cumulative effect of an error. It should be penalized only once.
11	A full scale of marks **70 marks** as given in Question Paper has to be used. Please do not hesitate to award full marks if the answer deserves it.

12	Every examiner has to necessarily do evaluation work for full working hours i.e., 8 hours every day and evaluate 20 answer books per day in main subjects and 25 answer books per day in other subjects (Details are given in Spot Guidelines).This is in view of the reduced syllabus and number of questions in question paper.
13	Ensure that you do not make the following common types of errors committed by the Examiner in the past:- ● Leaving answer or part thereof unassessed in an answer book. ● Giving more marks for an answer than assigned to it. ● Wrong totaling of marks awarded on an answer. ● Wrong transfer of marks from the inside pages of the answer book to the title page. ● Wrong question wise totaling on the title page. ● Wrong totaling of marks of the two columns on the title page. ● Wrong grand total. ● Marks in words and figures not tallying/not same. ● Wrong transfer of marks from the answer book to online award list. ● Answers marked as correct, but marks not awarded. (Ensure that the right tick mark is correctly and clearly indicated. It should merely be a line. Same is with the X for incorrect answer.) ● Half or a part of answer marked correct and the rest as wrong, but no marks awarded.
14	While evaluating the answer books if the answer is found to be totally incorrect, it should be marked as cross (X) and awarded zero (0)Marks.
15	Any unassessed portion, non-carrying over of marks to the title page, or totaling error detected by the candidate shall damage the prestige of all the personnel engaged in the evaluation work as also of the Board. Hence, in order to uphold the prestige of all concerned, it is again reiterated that the instructions be followed meticulously and judiciously.
16	The Examiners should acquaint themselves with the guidelines given in the “Guidelines for Spot Evaluation” before starting the actual evaluation.
17	Every Examiner shall also ensure that all the answers are evaluated, marks carried over to the title page, correctly totaled and written in figures and words.
18	The candidates are entitled to obtain photocopy of the Answer Book on request on payment of the prescribed processing fee. All Examiners/Additional Head Examiners/Head Examiners are once again reminded that they must ensure that evaluation is carried out strictly as per value points for each answer as given in the Marking Scheme.

SPECIFIC INSTRUCTIONS FOR COMPUTER SCIENCE ONLY

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	**In MySQL output questions, column headings to be ignored.**
5	**In MySQL output questions, alignment (left/right) of content to be ignored.**
6	**All answers/codes are suggestive, any other alternative correct answers to be accepted.**

CBSE AISSCE 2024 Marking Scheme for Computer Science

(Series &RQPS Sub Code: 083 Q.P. Code 91) SET-4

General Instructions:

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

SECTION-A

1.	State True or False :				1
	While defining a function in Python, the positional parameters in the function header must always be written after the default parameters.				
Ans	False				
	(1 Mark for the correct answer)				
2.	The **SELECT** statement when combined with ___ clause, returns records without repetition.				1
	(a)	**DISTINCT**	(b)	**DESCRIBE**	
	(c)	**UNIQUE**	(d)	**NULL**	
Ans	(a)	**DISTINCT**			
	(1 Mark for the correct answer)				
3.	What will be the output of the following statement :				1
	`print (16*5/4*2/5-8)`				
	(a)	`- 3.33`	(b)	`6.0`	
	(c)	`0.0`	(d)	`-13.33`	
Ans	(c)	`0.0`			
	(1 Mark for the correct answer)				
4.	What possible output from the given options is expected to be displayed when the following Python code is executed ?				1

	`import random`
	`Signal=['RED','YELLOW','GREEN']`
	`for K in range (2, 0, - 1) :`
	`  R = random.randrange(K)`
	`  print (Signal[R], end = '#')`				

	(a)	`YELLOW # RED #`	(b)	`RED # GREEN #`	
	(c)	`GREEN # RED #`	(d)	`YELLOW # GREEN #`	
Ans	(a)	`YELLOW # RED #`			
	(1 Mark for the correct answer)				
5.	In SQL, the aggregate function which will display the cardinality of the table is _____.				1
	(a)	`sum()`	(b)	`count(*)`	
	(c)	`avg()`	(d)	`sum(*)`	
Ans	(b)	`count(*)`			
	(1 Mark for the correct answer)				
6.	Which protocol out of the following is used to send and receive emails over a computer network ?				1
	(a)	`PPP`	(b)	`HTTP`	
	(c)	`FTP`	(d)	`SMTP`	
Ans	(d)	`SMTP`			
	(1 Mark for the correct answer)				
7.	Identify the invalid Python statement from the following :				1
	(a)	`d = dict()`	(b)	`e = {}`	
	(c)	`f = []`	(d)	`g = dict{}`	
Ans	(d)	`g = dict{}`			
	(1 Mark for the correct answer)				

8.	Consider the statements given below and then choose the correct output from the given options :
	`myStr = "MISSISSIPPI"`
	`print(myStr[:4]+"#"+myStr[-5:])`
					1

	(a)	`MISSI#SIPPI`	(b)	`MISS#SIPPI`	
	(c)	`MISS#IPPIS`	(d)	`MISSI#IPPIS`	
Ans	(b)	`MISS#SIPPI`			
	(1 Mark for the correct answer)				
9.	Identify the statement from the following which will raise an error :				1
	(a)	`print("A"*3)`	(b)	`print(5*3)`	
	(c)	`print("15" + 3)`	(d)	`print("15" + "13")`	

Ans	(c)	`print("15" + 3)`			
	(1 Mark for the correct answer)				

10.	Select the correct output of the following code :
	
	`event="G20 Presidency@2023"`
	`L=event.split(' ')`
	`print(L[::-2])`
					1

	(a)	`'G20'`	(b)	`['Presidency@2023']`	
	(c)	`['G20']`	(d)	`'Presidency@2023'`	
Ans	(b)	`['Presidency@2023']`			
	(1 Mark for the correct answer)				
11.	Which of the following options is the correct unit of measurement for network bandwidth ?				1
	(a)	`KB`	(b)	`Bit`	
	(c)	`Hz`	(d)	`Km`	
Ans	(c)	`Hz`			
	(1 Mark for the correct answer)				

12.	Observe the given Python code carefully :
	
	`a=20`
	`def convert(a):`
	`  b=20`
	`  a=a+b`
	
	`convert(10)`
	`print(a)`
	
	Select the correct output from the given options :
					1

	(a)	`10`	(b)	`20`	
	(c)	`30`	(d)	`Error`	
Ans	(b)	`20`			
	(1 Mark for the correct answer)				
13.	State whether the following statement is True or False: While handling exceptions in Python, name of the exception has to be compulsorily added with **except** clause.				1
Ans	`False`				
	(1 Mark for the correct answer)				

14.	Which of the following is not a DDL command in SQL ?				1
	(a)	`DROP`	(b)	`CREATE`	
	(c)	`UPDATE`	(d)	`ALTER`	
Ans	(c)	`UPDATE`			
	(1 Mark for the correct answer)				
15.	Fill in the blank : _____ is a set of rules that needs to be followed by the communicating parties in order to have a successful and reliable data communication over a network.				1
Ans	Protocol OR Name of any protocol				
	(1 Mark for the correct answer as protocol or name of any protocol)				
16.	Consider the following Python statement : `F=open('CONTENT.TXT')` Which of the following is an invalid statement in Python ?				1
	(a)	`F.seek(1,0)`	(b)	`F.seek(0,1)`	
	(c)	`F.seek(0,-1)`	(d)	`F.seek(0,2)`	
Ans	(c)	`F.seek (0,−1)`			
	(1 Mark for the correct answer)				
Q. **17** and **18** are ASSERTION (A) and REASONING (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.			
17	**Assertion (A) :** CSV file is a human readable text file where each line has a number of fields, separated by comma or some other delimiter. **Reason (R):** `writerow()` method is used to write a single row in a CSV file.				1
Ans	**(b)**	Both (A) and (R) are true and (R) is not the correct explanation for (A).			
		(1 Mark for the correct answer)			
18	**Assertion (A):** The expression `"Hello".sort()` in Python will give an error. **Reason (R):** `sort()` does not exist as a method/function for strings in Python.				1

Ans	(a)	**Both (A) and (R) are true and (R) is the correct explanation for (A).**	
		(1 Mark for the correct answer)	

SECTION-B

19	(A)	(i) Expand the following terms: **XML ,PPP** (ii) Give one difference between circuit switching and packet switching. **OR**	2
	(B)	(i) Define the term web hosting. (ii) Name any two web browsers.	
Ans	(A)	(i) eXtensible Markup Language Point-to-Point Protocol	
		(½ Mark for writing correct expansion of XML) ***(½ Mark for writing correct expansion of PPP)***	
Ans		(ii)	

Circuit Switching	Packet Switching
A dedicated path is established between the sender and the receiver before starting data transmission. Entire data is transmitted in one go.	Data to be transmitted is divided into small packets which are transmitted via nearest service provider till all packets reach the recipient where the packets are reassembled.

		(½ Mark for writing correct technique for Circuit Switching) ***(½ Mark for writing correct technique for Packet Switching)***	
		OR	
Ans	(B)	(i) Web hosting is a service that allows users to put a website or a webpage onto the internet, and make it a part of the World Wide Web.	
		(1 Mark for writing the correct definition of Web hosting)	
		(ii) Google Chrome,Microsoft Edge, Safari, Mozilla Firefox, Opera etc.	
		(1 mark for writing names of any two web browsers)	
20	The code given below accepts five numbers and displays whether they are even or odd: Observe the following code carefully and rewrite it after removing all syntax and logical errors : Underline all the corrections made.		2

```
def EvenOdd()
  for i in range(5) :
     num=int(input("Enter a number")
     if num/2==0:
        print("Even")
     else:
     print("Odd")
EvenOdd()
```

Ans

```
def EvenOdd():                          # Error 1
  for i in range(5) :
     num=int(input("Enter a number"))   # Error 2
     if num%2==0:                       # Error 3
       print("Even")
     else:
       print("Odd")                     # Error 4
EvenOdd()
```

(½ Mark for each correction made)

21. (A) Write a user defined function in Python named **showGrades(S)** which takes the dictionary S as an argument. The dictionary, S contains **Name:[Eng,Math,Science]** as **key:value** pairs. The function displays the corresponding grade obtained by the students according to the following grading rules : 2

Average of Eng,Math,Science	Grade
>=90	A
<90 but >=60	B
<60	C

For example : Consider the following dictionary

```
S={"AMIT":[92,86,64],"NAGMA":[65,42,43],"DAVID":[92,90,88]}
```

The output should be :

```
AMIT - B
NAGMA - C
DAVID - A
```

Ans

```
def showGrades(S):
  for K, V in S. items():
    if sum(V)/3>=90:
      Grade="A"
    elif sum(V)/3>=60:
      Grade="B"
    else:
      Grade="C"
    print(K,"-",Grade)
S={"AMIT":[92,86,64],"NAGMA":[65,42,43],"DAVID":[92,90,88]}
showGrades(S)
```

OR

```
def showGrades(S):
  for K in S:
    Sum=0
    for i in range(3):
      Sum+=S[K][i]
    if Sum/3>=90:
      Grade="A"
    elif Sum/3>=60:
      Grade="B"
    else:
      Grade="C"
    print(K,"-",Grade)
S={"AMIT":[92,86,64],"NAGMA":[65,42,43],"DAVID":[92,90,88]}
showGrades(S)
```

OR
Any other correct variation of the code

(½ Mark for the loop to process individual students from the dictionary)
(1 Mark for calculating grades)
(½ Mark for displaying grades)

OR

(B) Write a user defined function in Python named **`Puzzle(W,N)`** which takes the argument **`W`** as an English word and **`N`** as an integer and returns the string where every Nth alphabet of the word **`W`** is replaced with an underscore (**`"_"`**).

For example : if **`W`** contains the word **`"TELEVISION"`** and **`N`** is **`3`**, then the function should return the string **`"TE_EV_SI_N"`**. Likewise for the word **`"TELEVISION"`** if **`N`** is **`4`**, then the function should return **`"TEL_VIS_ON"`**.

Ans

```
Word="TELEVISION"
def Puzzle(W, N):
    NW=""
    Count=1
    for Ch in W:
        if Count!=N:
            NW+=Ch
            Count+=1
        else:
            NW+="_"
            Count=1
    return NW
print(Puzzle(Word,3))
```

OR

```
def Puzzle(W,N):
  W1=""
  for i in range(len(W)):
    if (i+1)%N==0:
      W1=W1+"_"
    else:
      W1=W1+W[i]
  return W1
print(Puzzle("TELEVISION",4))
```

OR
Any other correct variation of the code

(½ Mark for the loop to process individual (or Nth) characters)
(1 Mark for changing/replacing the required characters)
(½ Mark for returning the new word)

22. Write the output displayed on execution of the following Python code : 2

```
LS=["HIMALAYA","NILGIRI","ALASKA","ALPS"]
D={}
for S in LS :
    if len(S)%4 == 0:
            D[S] = len(S)
for K in D :
    print(K,D[K], sep = "#")
```

Ans		**HIMALAYA#8** **ALPS#4**	
		(1 Mark for each line of output) ***(Deduct ½ Mark if the entire output is correct but the formatting or line break or separating characters is/are incorrect)***	
23.	(A)	Write the Python statement for each of the following tasks using built-in functions/methods only : (i) To remove the item whose key is **"NISHA"** from a dictionary named **Students**. For example, if the dictionary **Students** contains **{"ANITA":90, "NISHA":76, "ASHA":92}**, then after removal the dictionary should contain **{"ANITA":90,"ASHA":92}** (ii) To display the number of occurrences of the substring **"is"** in a string named **message**. For example if the string **message** contains **"This is his book"**, then the output will be **3**.	1+1=2
Ans	(A)	**(i)** **Students.pop("NISHA")** **OR** **del(Students["NISHA"])** **OR** **del Students["NISHA"]** **OR** Any other correct variation of the code **(ii)** **print(message.count("is"))** **OR** **message.count("is")** **OR** Any other correct variation of the code	
		(1 Mark for each correct command)	
		OR	
	(B)	A tuple named **subject** stores the names of different subjects. Write the Python commands to convert the given tuple to a list and thereafter delete the last element of the list.	
Ans	(B)	**subject=list(subject)** **subject.pop()** **OR** **subject=list(subject)** **subject.pop(-1)** **OR** **subject=list(subject)** **del(subject[-1])** **OR** **subject=list(subject)** **del subject[-1]**	

		OR Any other correct variation of the code	
		(1 Mark for correctly converting to list) ***(1 Mark for correctly popping the last element/name)***	
24.	(A)	Ms. Veda created a table named **Sports** in a MySQL database, containing columns **Game_id, P_Age and G_name.** After creating the table, she realized that the attribute, **Category** has to be added. Help her to write a command to add the **Category** column. Thereafter, write the command to insert the following record in the table : **Game_id : G42** **P_Age : Above 18** **G_name : Chess** **Category : Senior**	2
Ans	(A)	**ALTER TABLE SPORTS** **ADD CATEGORY VARCHAR(10);** OR **ALTER TABLE SPORTS** **ADD COLUMN CATEGORY VARCHAR(10);** OR **ALTER TABLE SPORTS** **ADD CATEGORY CHAR(10);** OR **ALTER TABLE SPORTS** **ADD COLUMN CATEGORY CHAR(10);** **INSERT INTO SPORTS** **VALUES("G42","Above 18","Chess","Senior");** OR **INSERT INTO SPORTS(Game_id, P_Age, G_name, Category)** **VALUES("G42","Above 18","Chess","Senior");**	
		(½ Mark for* ALTER TABLE *command) ***(½ Mark for* ADD CATEGORY part*)*** ***(½ Mark for* INSERT INTO *command)*** ***(½ Mark for* VALUES part *)***	
		OR	
	(B)	Write the SQL commands to perform the following tasks : (i) View the list of tables in the database, **Exam**. (ii) View the structure of the table, **Term1**.	
Ans	(B)	**(i)** **SHOW TABLES;** **(ii)** **DESCRIBE Term1** OR **DESC Term1** **Note: Ignore** **USE Exam;** **if not written**	
		(1 Mark for each correct command)	

25		Predict the output of the following code :
		`def callon(b=20,a=10) :`
		`  b=b+a`
		`  a=b-a`
		`  print(b, "#" , a )`
		`  return b`
		`x=100`
		`y=200`
		`x=callon(x,y)`
		`print (x,"@", y)`
		`y=callon(y)`
		`print (x,"@",y)`
			2

Ans		`300#100`
		`300@200`
		`210#200`
		`300@210`	

		(½ Mark for each line of correct output) *(Deduct ½ mark only if all the numeric parts of the output are correct but formatting or/and separators are incorrect)*	

SECTION C

26.		Write the output on execution of the following Python code:
		`S="Racecar Car Radar"`
		`L=S.split()`
		`for W in L :`
		`  x=W.upper()`
		`  if x==x[::-1]:`
		`  for I in x:`
		`  print(I,end="*")`
		`  else:`
		`  for I in W:`
		`  print(I,end="#")`
		`  print()`
			3

Ans		`R*A*C*E*C*A*R*`
		`C#a#r#`
		`R*A*D*A*R*`	

		(1 Mark for each line of correct output) *Note:* ● *Deduct ½ mark only if all the alphabets are correct but some cases - lower/upper are incorrectly written* ● *Deduct ½ mark only if all the alphabets are correct but separators - */# are incorrectly written OR new line not considered*	

27 — Consider the table ORDERS given below and write the output of the SQL queries that follow: — 1x3 =3

ORDNO	ITEM	QTY	RATE	ORDATE
1001	RICE	23	120	2023-09-10
1002	PULSES	13	120	2023-10-18
1003	RICE	25	110	2023-11-17
1004	WHEAT	28	65	2023-12-25
1005	PULSES	16	110	2024-01-15
1006	WHEAT	27	55	2024-04-15
1007	WHEAT	25	60	2024-04-30

```
(i)  SELECT ITEM, SUM(QTY) FROM ORDERS GROUP BY ITEM;
(ii) SELECT ITEM, QTY FROM ORDERS WHERE  ORDATE BETWEEN
     '2023-11-01' AND '2023-12-31';
(iii) SELECT ORDNO, ORDATE FROM ORDERS WHERE ITEM = 'WHEAT'
     AND RATE>=60 ;
```

Ans

(i)

ITEM	SUM(QTY)
RICE	48
PULSES	29
WHEAT	80

(ii)

ITEM	QTY
RICE	25
WHEAT	28

(iii)

ORDNO	ORDATE
1004	2023-12-25
1007	2024-04-30

(1 Mark for writing each correct output)

Note:

- ***Ignore output heading***
- ***Ignore order of rows***

28 (A) Write a user defined function in Python named **showInLines()** which reads contents of a text file named **STORY.TXT** and displays every sentence in a separate line. — 3

Assume that a sentence ends with a full stop (.), a question mark (?), or an exclamation mark (!).

		For example, if the content of file **STORY.TXT** is as follows : **Our parents told us that we must eat vegetables to be healthy.And it turns out, our parents were right! So, what else did our parents tell?** Then the function should display the file's content as follows : **Our parents told us that we must eat vegetables to be healthy. And it turns out, our parents were right! So, what else did our parents tell?**	

Ans		**def showInLines():
		  with open("STORY.TXT",'r') as F:
		  S=F.read()
		  for W in S:
		  if W=="." or W=="?" or W=="!":
		  print(W)
		  elif W=="\n":
		  print(end="")
		  else:
		  print(W,end="")
		  F.close()**
		
		OR
		
		**def showInLines():
		  F = open("STORY.TXT",'r')
		  S=F.read()
		  for W in S:
		  if W.endswith(".") or W.endswith("?") or W.endswith("!"):
		  print(W)
		  elif W=="\n":
		  print(end="")
		  else:
		  print(W,end="")
		  F.close()**
		
		OR
		Any other correct variation of the code	

		(½ Mark for correctly opening the file) ***(½ Mark for reading the content of file using any correct method/mode)*** ***(½ Mark for the correct loop)*** ***(½ Mark for correctly checking end of sentence terminating characters)*** ***(½ Mark for correctly printing normal text without sentence terminator)*** ***(½ Mark for correctly printing text with sentence terminator)***	
		OR	
	(B)	Write a function, **c_words()** in Python that separately counts and displays the number of uppercase and lowercase alphabets in a text file, **Words.txt**.	

Ans

```
def c_words():
  f=open("Words.txt","r")
  Txt=f.read()
  CLower=CUpper=0
  for i in Txt:
    if i.islower():
      CLower+=1
    elif i.isupper():
      CUpper+=1
  print(CLower, CUpper)
  f.close()
OR
def c_words():
  with open("Words.txt","r") as F:
    Txt=f.read()
    CL=CU=0
    for i in Txt:
      if i.islower():  # if i>="a" and i<="z":
        CL+=1
      elif i.isupper():# if i>="A" and i<="Z":
        CU+=1
    print(CL, CU)
```

(½ Mark for correctly opening the file)
(½ Mark for reading the content of file using any correct method/mode)
(½ Mark for the correct loop)
(½ Mark for correctly checking and incrementing for uppercase alphabets)
(½ Mark for correctly checking and incrementing for lowercase alphabets)
(½ Mark for printing/returning required output)

29 — 1x3 =3

Consider the table **Projects** given below:

Table: Projects

P_id	Pname	Language	Startdate	Enddate
P001	School Management System	Python	2023-01-12	2023-04-03
P002	Hotel Management System	C++	2022-12-01	2023-02-02
P003	Blood Bank	Python	2023-02-11	2023-03-02
P004	Payroll Management System	Python	2023-03-12	2023-06-02

Based on the given table, write **SQL** queries for the following:

(i) Add the constraint, **primary key** to column **P_id** in the existing table **Projects**.

(ii) To change the language to **Python** of the project whose id is **P002**.

(iii) To delete the table **Projects** from MySQL database along with its data.

Ans

(i)
```
ALTER TABLE Projects
ADD PRIMARY KEY (P_id);
```

(½ Mark for **ALTER TABLE** *part)*
(½ Mark for **ADD PRIMARY KEY** *part)*

		(ii) `UPDATE Projects`
		`SET LANGUAGE= "Python"`
		`WHERE P_id = "P002";`	

		(½ Mark for **`UPDATE - SET`** *part)* *(½ Mark for* **`WHERE`** *part)*	
		(iii) `DROP TABLE Projects;`	
		(1 Mark for **`correct command`***)* ***OR*** *(½ Mark for partial answer such as* **`DROP Projects`** *or* **`DROP TABLE`***)*	
30		Consider a list named **`Nums`** which contains random integers. Write the following user defined functions in Python and perform the specified operations on a stack named **`BigNums`.** (i) **`PushBig()`:** It checks every number from the list **`Nums`** and pushes all such numbers which have **5 or more digits** into the stack, **`BigNums`.** (ii) **`PopBig()`:** It pops the numbers from the stack, **`BigNums`** and displays them. The function should also display **`"Stack Empty"`** when there are no more numbers left in the stack. For example: If the list **`Nums`** contains the following data: **`Nums = [213, 10025, 167, 254923, 14, 1297653, 31498, 386, 92765]`** Then on execution of **`PushBig()`**, the stack **`BigNums`** should store: **`[10025, 254923, 1297653, 31498, 92765]`** And on execution of **`PopBig()`**, the following output should be displayed: **`92765`** **`31498`** **`1297653`** **`254923`** **`10025`** **`Stack Empty`**	3

Ans		`def PushBig(Nums,BigNums):`
		`  for N in Nums:`
		`  if len(str(N)) >= 5:`
		`  BigNums.append(N)`
		`def PopBig(BigNums):`
		`  while BigNums:`
		`  print(BigNums.pop())`
		`  else:`
		`  print("Stack Empty")`
		
		OR	

```
def PushBig():
  for N in Nums:
      if N >= 10000:
        BigNums.append(N)

def PopBig():
  while BigNums:
    print(BigNums.pop())
  print("Stack Empty")
```

OR
Any other correct variation of the code

(½ Mark for the correct loop in the function PushBig)
(½ Mark for correctly checking the number of digits in the function PushBig)
(½ Mark for pushing the correct number into BigNums in the function PushBig)

(½ Mark for the correct loop in the function PopBig)
(½ Mark for correctly checking the underflow condition and printing `"Stack Empty"` ***in the function PopBig)***
(½ Mark for popping and printing the correct number in the function PopBig)

Note:
Ignore the declarations of `Nums` ***and/or*** `BigNums`

SECTION D

31 — 1x4 =4

Consider the tables **Admin** and **Transport** given below:

Table: Admin

S_id	S_name	Address	S_type
S001	Sandhya	Rohini	Day Boarder
S002	Vedanshi	Rohtak	Day Scholar
S003	Vibhu	Raj Nagar	NULL
S004	Atharva	Rampur	Day Boarder

Table: Transport

S_id	Bus_no	Stop_name
S002	TSS10	Sarai Kale Khan
S004	TSS12	Sainik Vihar
S005	TSS10	Kamla Nagar

Write SQL queries for the following:

(i) Display the student name and their stop name from the tables **Admin** and **Transport**.

Ans

```
SELECT S_name, Stop_name FROM Admin, Transport
WHERE Admin.S_id = Transport.S_id;
```

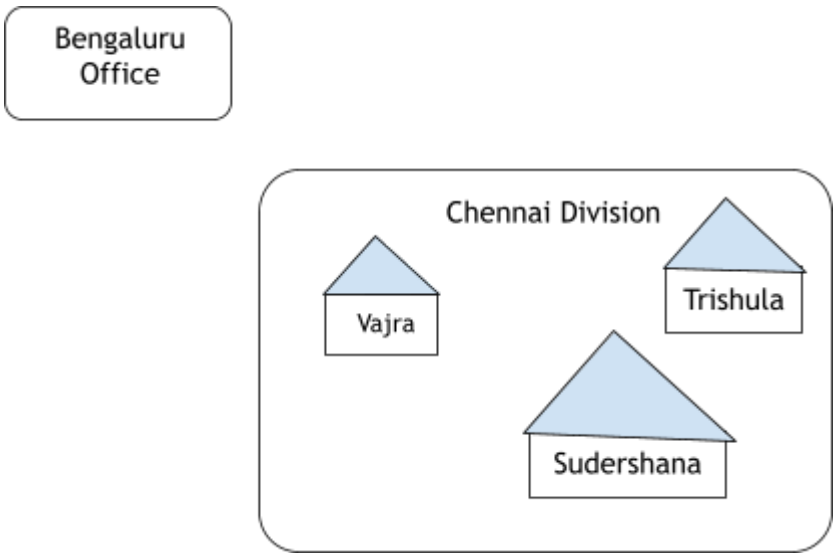
(½ Mark for `SELECT - FROM` ***part)***
(½ Mark for `WHERE` ***part)***

	(ii)	Display the number of students whose **S_type** is not known.	
Ans		**SELECT COUNT(*) FROM Admin** **WHERE S_type IS NULL;**	
		(½ Mark for **SELECT – FROM** *part)* *(½ Mark for* **WHERE** *part)*	
	(iii)	Display all details of the students whose name starts with **'V'** .	
Ans		**SELECT * FROM Admin** **WHERE S_name LIKE 'V%';** OR Any other correct query using/without using join	
		(½ Mark for **SELECT – FROM** *part)* *(½ Mark for* **WHERE** *part)*	
	(iv)	Display student id and address in alphabetical order of student name, from the table **Admin**.	
Ans		**SELECT S_id, Address FROM Admin** **ORDER BY S_name;**	
		(½ Mark for **SELECT – FROM** *part)* *(½ Mark for* **ORDER BY** *part)*	
32		Sangeeta is a Python programmer working in a computer hardware company. She has to maintain the records of the peripheral devices. She created a csv file named **Peripheral.csv**, to store the details. The structure of **Peripheral.csv** is: **[P_id,P_name,Price]** where **P_id** is Peripheral device ID (integer) **P_name** is Peripheral device name (String) **Price** is Peripheral device price (integer) Sangeeta wants to write the following user defined functions : **Add_Device()** : to accept a record from the user and add it to a csv file, **Peripheral.csv** **Count_Device()** : To count and display number of peripheral devices whose price is less than 1000.	4

Ans		**import csv** **def Add_Device():** **F=open("Peripheral.csv","a",newline='')** **W=csv.writer(F)** **P_id=int(input("Enter the Peripheral ID"))** **P_name=input("Enter Peripheral Name")** **Price=int(input("Enter Price"))** **L=[P_id,P_name,Price]** **W.writerow(L)** **F.close()** **def Count_Device():** **F=open("Peripheral.csv","r")** **L=list(csv.reader(F))** **Count=0** **for D in L:** **if int(D[2])<1000:** **Count+=1** **print(Count)** **F.close()** OR Any other correct variation of the code	
		(½ Mark for opening the csv file correctly in the function Add_Device()) ***(½ Mark for reading the data from the user in the function Add_Device())*** ***(½ Mark for writing the data correctly into the csv file in the function Add_Device())*** ***(½ Mark for opening the csv file correctly in the function Count_Device())*** ***(½ Mark for reading the data from the file in the function Count_Device())*** ***(½ Mark for loop in the function Count_Device())*** ***(½ Mark for checking the condition and counting correctly in the function Count_Device())*** ***(½ Mark for printing the output correctly in the function Count_Device())*** ***Note:*** ***Full 4 mark should be awarded if the examinee has mentioned that there is no mention of the task in the question***	

SECTION-E

33		Infotainment Ltd. is an event management company with its prime office located in Bengaluru. The company is planning to open its new division at three different locations in Chennai named as - **Vajra, Trishula and Sudershana.** You, as a networking expert need to suggest solutions to the questions in part (i) to (v), keeping in mind the distances and other given parameters.	1x5 =5



Distances between various locations:

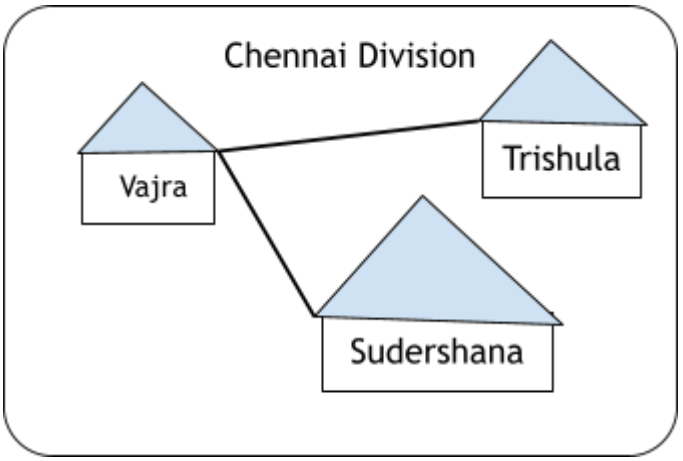
Vajra to Trishula	350 m
Trishula to Sudershana	415 m
Sudershana to Vajra	300 m
Bengaluru Office to Chennai	2000 km

Number of Computers installed at various locations :

Vajra	120
Sudershana	75
Trishula	65
Bengaluru Office	250

(i) Suggest and draw the cable layout to efficiently connect various locations in Chennai division for connecting the digital devices.

Ans



(Full 1 Mark for drawing any valid layout with OR without mentioning topology)
OR
(Only ½ mark for mentioning only topology without cable layout)

	(ii)	Which block in Chennai division should host the server? Justify your answer.	
Ans		**Vajra** can host the server as it has a maximum number of computers. OR Any other answer with valid justification	
		(½ Mark for the correct answer) ***(½ Mark for the correct justification)***	
	(iii)	Which fast and effective wired transmission medium should be used to connect the prime office at Bengaluru with the Chennai division?	
Ans		Optical Fiber	
		(1 Mark for the correct answer)	
	(iv)	Which network device will be used to connect the digital devices within each location of Chennai division so that they may communicate with each other ?	
Ans		Switch/Hub/Router	
		(1 Mark for the correct answer)	
	(v)	A considerable amount of data loss is noticed between the different locations of the Chennai division, which are connected in the network. Suggest a networking device that should be installed to refresh the data and reduce the data loss during transmission to and from different locations of Chennai division.	
Ans		Repeater **OR** **Mentioning any other valid reason or solution for data loss**	
		(1 Mark for the correct answer)	
34	(A)	(i) Differentiate between '**w**' and '**a**' file modes in Python.	2+3 =5
Ans		'**w**': Open the file in write mode. If the file doesn't exist, then a new file will be created. The file pointer is in the beginning of the file. If the file exists, the contents of the file, if any, are lost/truncated and the new data is added as fresh data into the file. '**a**': Open the file in append mode. If the file doesn't exist, then a new file will be created. The file pointer is at the end of the file. If the file exists, the new data is added at the end of the file without deleting the previous contents of the file.	
		(1 Mark each for any one correct characteristics of 'w' mode)	

(1 Mark each for any one correct characteristics of 'a' mode)

(ii)
Consider a binary file, **items.dat,** containing records stored in the given format :
{item_id: [item_name,amount]}

Write a function, **Copy_new()**, that copies all records whose amount is greater than 1000 from **items.dat** to **new_items.dat**.

Ans

```
import pickle
def Copy_new():
  F2=open("new_items.dat","wb")
  try:
    F1=open("items.dat","rb")
    Data1=pickle.load(F1)
    Data2={}
    for K,V in Data1.items():
      if V[1]>1000:
        Data2[K]=V
    pickle.dump(Data2,F2)
    F2.close()
  except:
    print("File not found!")
  F1.close()
```

OR

```
def Copy_new():
  try:
    F1=open("items.dat","rb")
    F2=open("new_items.dat","wb")
    D2={}
    try:
      while True:
        D1=pickle.load(F1)
        for k,v in D1.items():
          if v[1]>1000:
            D2[k]=v
    except:
      pickle.dump(D2,F2)
      F1.close()
      F2.close()
 except:
   print('File Opening Error')
```

OR

```
def Copy_new():
 f=open("items.dat","rb")
 f1=open("new_items.dat","wb")
 while True:
   try:
     r=pickle.load(f)
     for k,v in r.items():
       if v[1]>1000:
         pickle.dump(r, f1)
```

```
   except:
     break
 f.close()
 f1.close()
```

OR
Any other correct variation of the code

(½ Mark for opening the file items.dat in correct mode)
(½ Mark for opening the file new_items.dat in correct mode)
(½ Mark for reading the content of the file items.dat)
(½ Mark for the correct loop)
(½ Mark for checking the condition)
(½ Mark for writing the required contents into the file new_items.dat)

Note: Ignore f.close() and f1.close()

OR

(B) (i) What is the advantage of using **with** clause while opening a data file in Python ? Also give syntax of **with** clause.

The advantage of using **with** clause is that any file that is opened using this clause is closed automatically, once the control comes outside the with clause.
Example:

```
with open("myfile.txt","r+") as file_object:
   content = file_object.read()
```

In Python, we can open a file using **with** clause/statement.

The syntax of **with** clause is:

```
with open (file_name, access_mode) as file_object:
```

(1 Mark for writing any one advantage of with statement)
(1 Mark for writing syntax OR any valid example of with statement)

(ii)
A binary file, **EMP.DAT** has the following structure :
[Emp_Id, Name, Salary]

where
Emp_Id : Employee id
Name : Employee Name
Salary : Employee Salary

Write a user defined function, **disp_Detail()**, that would read the contents of the file **EMP.DAT** and display the details of those employees whose salary is below **25000**.

```
def disp_Detail():
  try:
    with open("EMP.DAT","rb") as F:
```

```
      Data=pickle.load(F)
      for D in Data:
        if D[2]<25000:
           print(D)
  except:
    print("File Not Found!!!")
```

OR

```
def disp_Detail():
  try:
    with open("EMP.DAT","rb") as F:
      try:
        while True:
          Data=pickle.load(F)
          if Data[2]<25000:
             print(Data)
      except:
        print("File ended")
  except:
    print("File Not Found!!!")
```

OR

```
def disp_Detail():
  try:
    with open("EMP.DAT","rb") as F:
      try:
        while True:
          Data=pickle.load(F)
          for D in Data:
            if D[2]<25000:
              print(D)
      except:
        print("File ended")
  except:
    print("File Not Found!!!")
```

OR
Any other correct variation of the code

(½ Mark for opening the file items.dat in correct mode)
(1 Mark for reading the content of the file items.dat)
(½ Mark for the correct loop)
(½ Mark for checking the condition)
(½ Mark for printing the desired output)

35	(A)	(i) Define Cartesian Product with respect to RDBMS.	1+4 =5
	Ans	Cartesian Product operation combines rows/tuples from two tables/relations. It results in all the pairs of rows from both the tables. It is denoted by 'X'.	
		(1 Mark each for the correct definition)	

		(ii) Sunil wants to write a program in Python to update the quantity to **20** of the records whose item code is **111** in the table named **shop** in MySQL database named **Keeper**. The table **shop** in MySQL contains the following attributes : • **Item_code**: Item code (Integer) • **Item_narne**: Name of item (String) • **Qty**: Quantity of item (Integer) • **Price**: Price of item (Integer) Consider the following to establish connectivity between Python and MySQL: Username: **admin** Password : **Shopping** Host: **localhost**	
Ans		**import pymysql as pm** **DB=pm.connect(host="localhost",user="admin",** **passwd="Shopping", database="Keeper")** **MyCursor=DB.cursor()** **SQL=f"UPDATE SHOP SET QTY=%S WHERE ITEM_CODE=%S"%(20,111)** **# OR** **SQL="UPDATE SHOP SET QTY=20 WHERE ITEM_CODE=111"** **MyCursor.execute(SQL)** **DB.commit()**	
		(½ mark for importing with any correct module/method) ***(1 mark for correct connect())*** ***(½ mark for creating the cursor)*** ***(1 mark for the correct SQL command - ½ Mark for UPDATE SET and ½ Mark for WHERE)*** ***(1 mark for correctly executing SQL)***	
		OR	
	(B)	(i) Give any two features of SQL.	
		Any two of the following • Full form is Structured Query Language. • Is used to retrieve and view specific data from a table in a database. • Is case insensitive • Each query in SQL ends with a semicolon (;) • It contains DDL and DML	
	Ans	***(½ Mark each for the any two correct feature as mentioned above or any other correct feature)***	
		(ii) Sumit wants to write a code in Python to display all the details of the passengers from the table **flight** in MySQL database, **Travel**. The table contains the following attributes: **F_ code** : Flight code (String)	

		F_name: Name of flight (String) **Source:** Departure city of flight (String) **Destination:** Destination city of flight (String) Consider the following to establish connectivity between Python and MySQL: ● Username : **root** ● Password : **airplane** ● Host : **localhost**	
Ans		**import pymysql as pm** **DB=pm.connect(host="localhost",user="root",** **password="airplane",database="Travel")** **MyCursor=DB.cursor()** **MyCursor.execute("SELECT * FROM Flight ")** **Rec=MyCursor.fetchall()** **for R in Rec:** **print (R)** OR Any other correct variation of the code	
		(½ mark for importing any correct module/method pymysql or any other) ***(1 mark for correct connect())*** ***(1 mark for correctly executing the query)*** ***(½ mark for correctly fetching the data)*** ***(1 mark for correctly displaying data)*** ***Note:*** ***Full 4 mark should be awarded if the examinee has mentioned that there is no mention of the task in the question***	