ANSWERS

Section – A

1. False

Explanation: The break statement terminates the immediate loop and comes out of it. It does not terminate all loops.

2. Correct option is (b)

Explanation: Length of keshav is 6 characters. We know that varchar is variable length. So even if width of the field is 20, it will use only the length of number of characters in the data. Instead, char is fixed length. It will occupy as many characters as is reserved for the field, irrespective of the number of characters in the actual data.

3. Correct option is (c)

Explanation: 3 – 2**2**3 + 99/11

=3 – 2**8 + 9.0

=3 – 256 + 9.0

=–253 + 9.0

=–244.0

4. Correct option is (b)

Explanation: s="Python is fun"

l=s.split()

s_new="-".join([l[0].upper(), l[1],l[2].capitalize()])

print(s_new)

s.split() returns a list L=["Python","is","fun"]

L[0].upper() returns PYTHON , L[1] returns "is" and L[2].capitalize() returns Fun. "-" character joins all these three words.

5. Correct option is (b)

Explanation: In case of Cartesian product: Degree of resultant is D1+ D2, and cardinality is C1xC2

6. Correct option is (a)

Explanation: A personal area network is a network of personal devices within a small area. Here the user connects her mobile to the laptop, hence it forms a PAN.

7. Correct option is (b)

Explanation: The del function requires the key of the key:value pair to be deleted.

8. Correct option is (b)

Explanation:

#	G	2	0	P	r	e	s	i	d	e	n	c	y
–14	–13	–12	–11	–10	–9	–8	–7	–6	–5	–4	–3	–2	–1
–0	1	2	3	4	5	6	7	8	9	10	11	12	13

print(pride[-2:2:-2]) extracts characters from position -2 up to position 2 jumping by 2 positions

9. Correct option is (d)

Explanation: Tuples are immutable and statement4 tries to change the value of a tuple.

10. Correct option is (b)

Explanation: In the code the loop: "for I in color" , picks values starting from "YELLOW" . Hence first value in output must be "YELLOW". So options b) or d) can be correct. Now since in both options YELLOW is printed once , it indicates that the value of my Number must be 2 , therefore the inner loop will go for only once . Hence all the other values in the color list will be printed once.

11. **Correct option is (b)**

Explanation: The modem at the sender's end modulates signals , that is converts signals from Digital to Analog and at the receiver's end de modulates , that is converts from Analog to Digital.

12. **Correct option is (c)**

Explanation: The global keyword will make the global variable b accessible in the function.

13. **True**

Explanation: An exception is an unusual situation that might happen in a program execution. Since exceptions are of different types and may occur for various reasons such as "File not found", "Invalid Inputs" "Keyboard error" etc. , it might happen even if the program is syntactically correct.

14. **Correct option is (c)**

Explanation: A candidate key that is not a primary key is an Alternate key.

15. **Circuit switching**

Explanation: In case of circuit switching, before a communication starts, a dedicated path is established between the sender and the receiver ends.

16. **Correct option is (c)**

Explanation: The seek() function is used to place the file pointer at a specific byte position.

17. **Correct option is (d)**

Explanation: A List is mutable. Immutable types, if tried to be modified, the old variable is destroyed and a new memory is allocated.

18. **Correct option is (b)**

Explanation: Both the statements are True , but (R) does not explain (A)

Section – B

19. **(i) POP3 – Post Office Protocol 3** 2
URL – Uniform Resource Locator

(ii) HTML(Hyper text MarkUp Language)

- We use pre-defined tags
- Static web development language – only focuses on how data looks
- It use for only displaying data, cannot transport data
- Not case sensitive

XML (Extensible Markup Language)

- We can define our own tags and use them
- Dynamic web development language – as it is used for transporting and storing data
- Case sensitive

20. 2

```
def revNumber(num):
rev = 0
rem = 0
while num > 0:
rem =num %10
rev = rev*10 + rem num = num//10
return rev
print(revNumber(1234))
```

Explanation: The keyword to define a function is "def" not "define"
The keyword is "while" not "While"
The return statement should be after the end of the while loop. So indentation should be done accordingly.

21.

```
PLACES = {1: "Delhi", 2: "London", 3: "Pariaos", 4: "New York", 5: "Dubai"}
def count Now (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
```

2

Explanation: 1st option code:

The for loop traverses through the values of the dictionary and checks them if their length is more than 5. In these cases the values are converted to uppercase and printed.

2nd option code:

The split() function parses the string into words and returns a list L carrying each word as a list item. The for loop picks each word from the list and finds the length of each. It then adds the length found as an Explanation:

1st option code:

The for loop traverses through the values of the dictionary and checks them if their length is more than 5. In these cases the values are converted to uppercase and printed.

2nd option code:

The split() function parses the string into words and returns a list L carrying each word as a list item. The for loop picks each word from the list and finds the length of each. It then adds the length found as an element of the tuple.

Note: Any other correct logic may be marked

22.

```
4*L
33*4
21*S
10*6
```

2

Explanation: S="LOST"

```
L=[10,21,33,4]
D={ }
for I in range(len(S)):
    if I%2==0:
     D[L.pop()]=S[I]
else
    D[L.pop()]=I+3
for K,V in D.items():
    print(K,Vsep="*")
```

In the code, the for loop traverses through the length of the string that is, I picks values from 0 to 3. If I is even the statement D[L.pop()] pops the last element from the list L and adds it as a dictionary key . The value of the key is taken as S[I] that is character at index I of the string S.

In the output for loop , each dictionary key and values is picked from D.items() and printed with a separator ="*"

23. (a) L1.insert(2,200)

(b) message.endswith('.') 2

Explanation: The insert() function inserts an element at the index specified.

The endswith() function checks whether a string ends with a specific string or not and returns True/False.

24.
```
Part 1:
ALTER TABLE Employee ADD EmpId INTEGER PRIMARY KEY;
Part 2:
INSERT INTO Employee VALUES("Shweta","Production",26900,999);
                                    OR
INSERT INTO Employee(EmpId,Ename,Department,Salary)
VALUES(999,"Shweta","Production",26900);
```

Explanation: The Alter table command can be used to add, modify and remove a field . It can also be used to add a constraint.

Syntax: ALTER TABLE <tablename> ADD <Field> <Type> PRIMARY KEY;

The syntax to add a record to a table is: Insert into <table> values (val1,val2,val3,.....)

25. **Output:** 10.0$20 2

10.0$2.0###

```
def Changer (P, Q = 10):
    P = P/Q
    Q = P%Q
    return P
A = 200
B = 20
A = Changer (A, B)
print (A, B, sep = '$')
B = Changer (B)
Print (A, B, sep = '$', end - '###')
```

Explanation: The Changer function receives a value P and a default argument in the form of Q=10.

In the first call the values received in P, Q are 200,20 . The statement P=P/Q makes P = 200/20 = 10.0 and Q=P%Q = 10.0%10 =0. The value of P is returned back to A=10.0.

The print statement prints values of A, B which are 10.0 and 20 with a separator $.

Similarly in the 2nd call the values received by P and Q are 20, 10(The default value). P=P/Q makes P as 2.0. Value of P=2.0 is returned into B . Hence the values printed are 10.0 , 2.0 with separator '$' and ending with "###"

Section – C

26. ND-*34

Explanation: Text1 = "IND-23" 3

```
Text2=""
I=0
while 1<len(Text1):
    if Text1[I]>="0" and Text1[I]<="9":
        Val = int (Text1 [I])
```

```
        Text2=Text2 + str(Val)
elif Text1 [I]>-"A" and Text1 [I]<-"z":
    Text2=Text2 + (Text1 [I + 1])
else:
    Text2=Text2 + "*"
  It=1
print (Text2)
```

In the above code the while loop traverses through the length of Text1 with I taking the values 0 to 5.

For each character in the text:

(a) If it is a digit , it converts it into integer form, increments the digit by 1 and appends it to Text2

(b) If it is an uppercase alphabet the next character in the text is appended to Text2 , otherwise a "*" is appended to Text2.

27. (a)

COUNT(DISTINCT SPORTS)
4

(b)

CNAME	SPORTS
AMINA	CHESS

(c)

CNAME	AGE	PAY
AMRIT	28	1000
VIRAT	35	1050

Explanation: (i) SELECT COUNT (DISTINCT SPORTS) FROM CLUB;

(ii) SELECT CNAME, SPORTS FROM CLUB WHERE

DOAPPE<"2006-04-30" AND CNAME LIKE "%NA";

(iii) SELECT CNAME, AGE, PAY FROM CLUB WHERE

GENDER = "MALE" AND PAY BETWEEN 1000 AND 1200;

(i) The count(distinct sports) function call counts the number of unique sports values in the club table.

(ii) The SQL query displays CNAME and SPORTS from the CLUB table for records whose DOAPP is before the date specified and whose CNAME ends with "NA". The like clause with "%NA" specifies values which end with "NA"

(iii) The SQL query displays CNAME, AGE and PAY of the records who are "MALE" and whose PAY ranges between 1000 and 1200. The Between cause matches values in a range.

28. 3

```
def test ():
    f0bj1 = open ("Alpha.text", "r")
    data = f0bj1.readlines ()
    for line in data:
    L = line. split ()
    if L[0] == "You":
     print (line)
f0bj1. close ()
```

Explanation: The open() function opens the text file in read mode. The readlines() function reads all the data of the text file as separate lines. The for loop picks each line from the data read and the split() functions creates a list L of the words of each line . L[0] gives the first word of each line. The if condition checks if the word is "You", and prints the line in case of a match.

OR

```
def vowel Count ():
    f0bj = open ("Alpha.text",:r")
    data = str (F0bj.read())
    cnt = 0
    for ch in data:
      in ch in "aeiouAEIOU":
          cnt=cnt+1
    print (cnt)
    f0bj.close ()
```

Explanation: The open() function opens the text file in read mode. The read () function reads all the data of the text into data. The for loop picks each character from the data and checks whether it is in the domain of vowels. In case of a match a counter is incremented, which is finally printed.

29. 3

```
(i)  UPDATE Personal
     SET Salary=Salary*0.5
     WHERE Allowance IS NOT NULL;
(ii) SELECT Name, Salary+Allowance AS
     "Total Salary" FROM Personal;
(iii) DELETE FROM Personal
     WHERE Salary>25000
```

Explanation:

(i) The UPDATE command is used to change or update values in a table. It may be used along with a where clause to update certain records.

(ii) Calculated values may be displayed in a select query by specifying the expression for it in the query which is done here by writing "Salary + Allowance" . The heading for a column in the display may be changed by specifying the new heading with "AS" clause.

(iii) The DELETE command is used to remove records from a table. It may be used along with a where clause to delete certain records.

30. 3

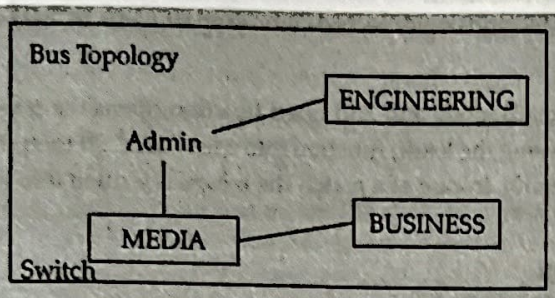
```
travel = []
def Push_element (NList):
    for L in NList:
      if L[1] ! = "India" and L[2] < 3500:
          travel.append ([L[0], L[1]])
def Pop_element ():
    while len (travel):
    print (travel.pop())
else:
    print ("Stack Empty")
```

Explanation: The Push_element() function receives the nested list of elements and picks each list of journey into L using the for loop. The if condition checks the City from L[1] and "distance from delhi" from L[2] . The condition for outside India is checked as "!=India" and the condition for distance<3500 is checked as L[2]<3500. Such records are pushed to the stack using the append() function.

The Pop_element() function loops through the number of elements in the list by using len(travel) and pops the last elements from the list using "travel.pop()"

Section – D

31. (a) 5



(b) Switch

(c) Admin block, as it has maximum number of computers.

(d) Microwave

(e) Firewall

Explanation:

(a) To draw the cable layout , the least three distances have to be considered, and connected. Here the least three distances are 45m, 50m and 55m. This saves cable costs and brings better connectivity.

(b) A switch is a device that connects multiple computers in a local area network, efficiently.

(c) ADMIN block, as it houses the maximum number of computers and hence they will have internal connectivity.

(d) Since the distance between DELHI and CHENNAI blocks is 2175 KMS, a Microwave transmission will be required, as Microwave can travel between cities, states and countries.

(e) A Firewall is a security system that can filter a private network from malware and security threats.

32. (a) 5

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:

- 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

(b)
```
def copyData ():
    f0bj = open ("SPORT.DAT", "rb")
    f0bj1 = open ("BASKET.DAT", "wb")
    cnt = 0
    try:
        while True:
            data = pickle. load (f0bj)
            print (data)
            if data [0] == "Basket Ball":
                pickle.dump (data, f0bj1)
                cnt + = 1
    except:
```

```
    f0bj.close ()
    f0bj1.close ()
  return cnt.
```

Explanation: The code reads records from one binary file and copies records whose sport name is "Basketball" to another file.

The 1st open function opens the source file in "rb" (read) mode and 2nd open function opens the target file in "wb" (write) mode. The while loop reads each record using the load() function into a list "data" . It then checks for the sportname which is the 1st element of the list – data[0]. In case of a match the record is written into the target file using the dump() function.

OR

```
def Search type (mtype):
    f0bj = open ("CINEMA.DAT", "rb")
try:
    while True:
      data = pickle.load (f0bj)
      if data [2] = = mtype:
            print ("Movie number:", data [0])
            print ("Movie Name:", data [1])
            print ("Movie Type:", data [2])
    except EOFError:
            f0bj.close ()
```

33. (a) 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. 5

(b)
```
import mysql.connector as mysql
con1 = mysql.connect (host = "localhost", user="root", password="tiger", database="sample 2023")
mycursor=conl.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")
conl. close ()
```

Explanation:

(1) In order to use functions for a python – mysql connectivity , the mysql.connector package needs to be imported , which is done first in the code.

(2) The connect() function establishes the connection between python application and mysql with the connection string carrying the database name , userid and password.

(3) The data to be inserted is input using the input function.

(4) The SQL query string is created in the variable "query" with the SQL insert command.

(5) Finally the execute function executes the SQL command.

(6) The commit() function makes the changes permanent.

Section – E

34. 4

(a)
```
SELECT PName, BName FROM PRODUCT P,
BRAND B WHERE P.BID=B.BID;
```

(b)
```
DESC PRODUCT;
```

(c)
```
SELECT BName, AVG(Rating) FROM PRODUCT
P, BRAND B
WHERE P.BID=B.BID
GROUP BY BName
HAVING BName='Medimix' OR
BName='Dove';
```

(d)
```
SELECT PName, UPrice, Rating
FROM PRODUCT
ORDER BY Rating DESC;
```

Explanation:

(i) The product name and corresponding brand names are to be displayed. So the query selects the respective fields from both the tables by matching the joining column(BID) - P.BID=B.BID

(ii) The describe command displays the structure of the table

(iii) In this command the GROUP BY clause is used to group the records on brandname . The having clause filters only those groups whose Bname is "Medimix" or "Dove". Since the command brings records from two tables the joining condition "P.BID=B.BID" is also used.

(iv) Since the records have to be displayed in descending order of Rating, The "Order by Rating Desc " clause is used.

35. 4

```
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)
```

```
    cswriter.writerow (headings)
    csvwriter.writerow (data)
    f.close ()
def wonCount ():
    f=open ('Result.csv', 'r')
    csvreader = csv.reader (f, delimiter = ',')
    head = list (csvreader)
    pring (head [0])
    for x in head:
        if x [3] = = "WON":
            print (x)
    f.close ()
```

Explanation: The code inputs the details of a student. It opens the csv file "Result.csv" in append mode to add records. A list of the data input is created in "data" and list of the column headings is created in "headings". The csv writer object "csvwriter" is created using the writer() function. The heading and the data input is written to the csv file using writerow() function. The file object is closed at the end.

❑❑❑