

## Solution Sample paper- 08

Q.1	(c) 4ever	
Q.2	(b) False	
Q.3	True	
Q.4	(c) tp1 = tp+tp*2	
Q.5	(b) Python \$pro\$gramming is fun!	
Q.6	(a) myfile = open('c:\\test.txt','rb+')	
Q.7	(b) LIKE	
Q.8	(b) SELECT DISTINCT	
Q.9	(a) Statement 1 and 2	
Q.10	(c) Candidate keys	
Q.11	(c) dump()	
Q.12	(a) ALTER TABLE	
Q.13	(a) LAN	
Q.14	(b) 13.0 bool(True) is equal to 1 and bool(False) is equal to 0	
Q.15	(d) Both (a) and (c)	
Q.16	(b) fetchone()	
Q.17	A is True but R is False	
Q.18	<p>(a) Both A and R are true and R is the correct explanation for A</p> <p>Explanation: LEGB rule mean Python resolve the variables as per sequences.</p> <ol style="list-style-type: none"> <li>1. Local variable</li> <li>2. Enclosing(nested function) Variable</li> <li>3. Global Variable,</li> <li>4. Built-in variable (keywords, exceptions etc)</li> </ol>	
Q.19	<pre>def prime (num):          # def should be in small letters     factorial = 1     if num &lt; 0:         print("Sorry, factorial does not exist for negative numbers")     elif( num == 0):      # elif required for condition and : required         print("The factorial of 0 is 1")     else:         for i in range(1,num + 1):             factorial = factorial*i      # indentation is not correct         print("The factorial of",num,"is",factorial)</pre>	
Q.20	<b>Hub</b>	<b>Switch</b>
	<ul style="list-style-type: none"> <li>*. Non intelligent device</li> <li>*. Collisions occur commonly in setups using hubs.</li> <li>*. Lower-performance</li> <li>*. Data pass through all network traffic to each of the computers over network</li> <li>*. Not Better perform on busy network</li> </ul>	<ul style="list-style-type: none"> <li>*. Intelligent device</li> <li>*. No collisions occur</li> <li>*. Effectively a higher-performance</li> <li>*. Capable of determining the destination and selectively forwarding data to the one computer that actually needs</li> <li>*. Better perform on busy network</li> </ul>
	<b>Bus Topology</b>	<b>Star Topology</b>
	<ul style="list-style-type: none"> <li>*. Each device is connected to a single cable which is known as the backbone.</li> </ul>	<ul style="list-style-type: none"> <li>*. All devices are connected to a central hub</li> </ul>

	<ul style="list-style-type: none"> <li>*. The failure of the network cable will cause the whole network to fail.</li> <li>*. There is a linear arrangement of nodes in a network.</li> <li>*. Bus topology is less expensive than a star topology</li> <li>*. Data is transmitted slower as compared to a star topology.</li> <li>*. Expansion of network i.e. addition of new node is difficult.</li> <li>*. Fault identification and isolation are relatively not easier.</li> <li>*. Data collisions occur frequently</li> <li>*. It is used in Ethernet networks</li> </ul>	<ul style="list-style-type: none"> <li>*. If the central hub fails then the whole network fails</li> <li>*. There is a non-linear arrangement of nodes in a network.</li> <li>*. High implementation cost because of the central hub and extra wires required for connection.</li> <li>*. Data transmission is faster in a star topology</li> <li>*. Expansion is easier.</li> <li>*. Fault identification and isolation are relatively easier.</li> <li>*. Chances of data collision are less</li> <li>• *. Star topology is frequently used in high-speed LANs.</li> </ul>
--	---	--

Q.21	<p>(a)</p> <pre>name = ' Karim Benzema' size=(len(name)) index=0 while(index&lt;size):     print(name[index].upper(),end="")</pre> <p>(B)</p> <p>Suchitra scored 66 marks in Pre Board  Prakul scored 62 marks in Pre Board  Arun scored 55 marks in Pre Board  Prashant scored 49 marks in Pre Board and needs improvement.  Sidhi scored 65 marks in Pre Board</p>
------	--

Q.22	<p>Explain Referential Integrity in a Relational Database Management System.</p> <ul style="list-style-type: none"> <li>• Referential integrity can be enforced by working with primary key and foreign key In Relational Database Management Systems (RDBMS).</li> <li>• Each foreign key match with primary key to reference from one table to another.</li> <li>• With referential integrity the database contents will be joined together to be jointly updated.</li> </ul> <p><u>Why DBMS is better than File System</u></p> <p><b>File System:</b>  A File Management system allows access to single file at a time.  Data is directly stored in a set of files  It contains flat files that have no relation to other files</p> <p><b>Database Management System (DBMS)</b>  It is a system that allows to efficiently define, create, maintain and share databases</p> <p><b>Advantages of DBMS over File system</b></p> <ul style="list-style-type: none"> <li>• Reduce Data redundancy Data inconsistency</li> <li>• Support Data sharing and Data searching</li> <li>• Data integrity</li> <li>• Data security</li> <li>• Backup</li> <li>• Easy Maintenance</li> </ul>
------	--

Q.23 (a) (i) SMTP : Simple Mail Transfer Protocol  
(ii) IMAP : Internet Message Access Protocol

(b) What is MAC address: Media Access Control.  
It is the unique type of 12 character address that identify the device at network.  
It is in hexa-decimal notation and 6 Byte long address  
Each byte is separated with colon (:)  
Example: 01:A5:F3:4D:2C:8B

Q.24 Output is 190  
OR -----  
9 60 P\$R\$\$

Q.25

CHAR	VARCHAR
CHAR datatype is used to store character strings of fixed length	VARCHAR datatype is used to store character strings of variable length
If the length of the string is less than set or fixed-length then it is padded with extra memory space.	If the length of the string is less than the set or fixed-length then it will store as it is without padded with extra memory spaces.
CHAR stands for "Character"	VARCHAR stands for "Variable Character"
Storage size of CHAR datatypes is equal to n bytes i.e. set length	The storage size of the VARCHAR datatype is equal to the actual length of the entered string in bytes.
CHAR waste space when there is variation in the length of values	VARCHAR saves space when there is variation in the length of values

OR

**Candidate key:** -

- All attribute combinations inside a relation that can serve as primary key are candidate key
- All attribute with no redundant values
- One of key from Candidate keys will be selected as Primary key.

**Alternate Key:** –

- It is a column or group of columns in a table that uniquely identify every row in that table.
- The candidate keys, other than Primary key, are known as Alternate keys.

Q.26

(a)

NAME	TDATE	AMOUNT
AAKASH	2022-08-31	1500
INDIRA	2022-09-15	2000

(b)

(i)

ANO	ANAME
103	Ali Reza
105	Simran Kaur

(ii)

DISTINCT ANO

101  
102  
103

(iii)

ANO	COUNT(*)	MIN(AMOUNT)
101	2	2500
103	2	1000

(iv)

COUNT(*)	SUM(AMOUNT)
2	5000

Q.27	<pre>def ChangeGender():     f=open("BIOPIC.TXT",'rt')     rec=f.read()     print("file content is:\n")     rec=rec.replace(' he ',' she ')     print(rec)     f.close()  OR ----- def BIGWORDS():     f=open("CODE.TXT ','rt')     data=f.read()     words=data.split()     count=0     for w in words:         if(len(w)&gt;=5):             count+=1     print("\Count of Big words are: ",count)</pre> <p><b>BIGWORDS()</b></p>
Q.28	<p>(A)</p> <p>(i) <u>DISTINCT TRAVELDATE</u>  2018-12-25  2018-11-10  2018-10-12</p> <p>(ii) <u>MIN (TRAVELDATE)</u>                      <u>MAX (TRAVELDATE)</u>  2018-10-12                                      2018-12-25</p> <p>(iii) <u>START</u>                      <u>COUNT(*)</u>  Pune Junction                      2  New Delhi                              2</p> <p>(iv) <u>TNAME</u>                      <u>PNAME</u>                      .  Amritsar Mail                      R N AGRAWAL</p> <p>(B) DESC table_name ;</p>
Q.29	<pre>def ZeroEnding(SCORES):     sum=0     for n in SCORES:         if(n%10==0):             sum+=n     print("Sum of numbers: ",sum)</pre> <p>ZeroEnding([200, 456, 300, 100, 234, 678])</p>
Q.30	<pre>def Push_element():     global status     status=[]     print("List of details are:")     for rec in list:         print(rec)</pre>

```
if(rec[1]>75 and rec[2]=='CS'):
    r=[rec[0],rec[1]]
    status.append(r)
```

```
def Pop_element():
    print("\nStack contains:")
    while(len(status)>0):
        print(status.pop())
    else:
        print("Stack Empty")
```

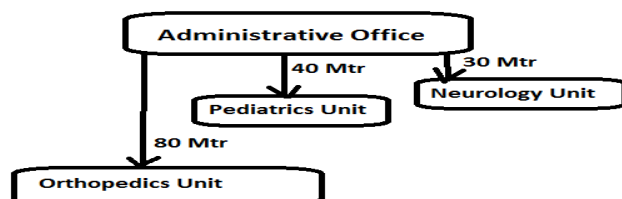
```
list=[["Danish",80,"Maths"],
["Hazik",79,"CS"],
["Parnik",95,"Bio"],
["Danish",70,"CS"],
["Sidhi",99,"CS"]]
Push_element()
Pop_element()
```

**OR** -----

```
def Push(emp):
    stack=[]
    count=0
    for key in emp:
        if(emp[key]<15000):
            stack.append(key)
            count+=1
    for i in range(-1,-len(stack)-1,-1):
        print(stack[i])
    print("\ntotal records in Stack: ",count)
```

```
emp={"Sohan":20000,"Mohan":9000,"Rohan":25000,"Aman":5000}
Push(emp)
```

Q.31 (i) Administrative Office  
(ii)



(iii) Switch  
(iv) Firewall  
(v) Ethernet Cable

Q.32	<p>(a) dcba</p> <p>(b)</p> <p>Statement 1: cur=db1.cursor()  Statement 2: sql = 'UPDATE EMP SET salary=salary+1000 WHERE salary&lt;8000'  Statement 1: cur.execute(sql)</p> <p><b>OR</b> -----</p> <p>(a) Minimum Value of R: 0 , if K=2  Maximum Value of R: 1 , if K=2  Output of code: (b) Wait # Stop #</p> <p>(b)</p> <p>Statement 1: mycursor=con1.cursor()  Statement 2: mycursor.execute(query)  Statement 1: con1.commit()</p>
Q.33	<p>full form of CSV: comma Separated Values</p> <pre>import csv def WRITEREC():     f=open("PLANTS.csv",'w')     ID=input("Enter ID: ")     name=input("Enter Name: ")     price=float(input("Enter Price: "))     rec=[ID,name,price]     write_obj=csv.writer(f)     write_obj.writerow(rec)     f.close()  def SHOWHIGH():     f=open("PLANTS.csv",'r')     read_obj=csv.reader(f)     for rec in read_obj:         if(rec[2]&gt;500):             print(rec)     f.close()</pre> <p><b>OR</b> -----</p> <p><b>Text file</b> stores data in the form of alphabets, digits and other special symbols by storing their ASCII values and are in a human readable format.  A small error in a <b>textual file</b> can be recognized and eliminated  For example, any file with a .txt, .c, etc extension</p> <p><b>Binary file</b> contains a sequence or a collection of bytes which are not in a human readable format.  small error in a <b>binary file</b> corrupts the file and is not easy to detect  For example, files with .exe, .mp3, etc extension</p>

	<pre>import csv def countrec():     f=open("PATIENTS.csv",'r')     read_obj=csv.reader(f)     count=0     for rec in read_obj:         if(rec[2]=='COVID-19'):             count+=1             print(rec)     f.close()</pre>
Q.34	<p>(a) Degree: 5 Cardinality: 6</p> <p>(b). MOVIEID field can be used as Primary Key, because it contains unique and not null values.</p> <p>(C)</p> <p>(i) DELETE FROM MOVIEDETAILS WHERE rating &lt;4 ;</p> <p>(ii) UPDATE MOVIEDETAILS SET rating=5 WHERE MOVIEID = 'M020' ;</p> <p><b>OR</b> -----</p> <p>(i) ALTER TABLE MOVIEDETAILS ADD Recommended CHAR(20) DEFAULT 'Recommended' ;</p> <p>(ii) UPDATE MOVIEDETAILS SET rating=rating+1 WHERE language = 'HINDI' ;</p>
Q.35	<p>(a) Line-1: import pickle</p> <p>(b) Line-2A: fout=open("Mydata.dat",'wb')</p> <p>Line-2 B: fin=open("Mydata.dat",'rb')</p> <p>(c) Line-3: pickle.dump(Sqlist,fout)</p> <p>(d) Line-4: mylist=pickle.load(fin)</p>