



MARKING SCHEME

Section - A

Q01	False	(1)
Q02	(A) eval	(1)
Q03	(D) dict_student.update(dict_marks)	(1)
Q04	(B) False	(1)
Q05	(C) mail2@kvsangathan.	(1)
Q06	(D) close()	(1)
Q07	(C) alter	(1)
Q08	(B) DROP DATABASE	(1)
Q09	(C) S4	(1)
Q10	(C) Foreign Key	(1)
Q11	(D) file_object.seek(offset [, reference_point])	(1)
Q12	(A) DISTINCT	(1)
Q13	(B) VoIP	(1)
Q14	(C) 30.6	(1)
Q15	(D) count(*)	(1)
Q16	(B) connect	(1)
Q17	(A) Both A and R are true and R is the correct explanation for A	(1)
Q18	(C) A is True but R is False	(1)

Section - B

Q19 .	<pre> Def checkNumber(N): be defstatus = N%2 return return status #main-code num=int(input(" Enter a number to check :)) # Message not enclosed within quotationmark k=checkNumber(num) if k = 0: print("This is EVEN number")else: print("This is ODD number") # must be k = = 0 </pre> <p>(½ mark for each correct correction made and underlined.)</p>	(2)
Q20 .	1 mark for each correct point of difference	(2)
Q21 .	(A) 322ADORSF	(2)
	(B) dict_keys(['name', 'age', 'dept', 'rno']) 1 Marks for each correct answer.	
Q22.	<p>A foreign key is used to set or represent a relationship between two relations (or tables) in a database. Its value is derived from the primary key attribute of another relation.</p> <p>(1 mark for explanation and 1 mark for example) (Any relevant correct example may be marked)</p>	(2)
Q23.	<p>(A) (i) HTTP: Hyper Text Transfer Protocol (ii) FTP: File Transfer Protocol (½ mark for every correct full form)</p> <p>(B) TELNET is used to access a remote computer / network. (1 mark for correct answer)</p>	(2)
Q24.	<p>1 20 L@ 4 60 L@M@ 9 120 L@M@N@</p> <p>(½ M + ½ M + 1 M) means ½ - ½ marks for first two lines and 1 mark for last line.</p> <p>OR</p> <p>[(2, 4), (4, 16)]</p> <p>(½ mark for each correct pair of tuple , ½ mark for enclosing in parenthesis) means concept of tuple and list</p>	(2)
Q25.	<p>1 mark for the difference and 1 mark for appropriate example OR</p> <p>DDL- ALTER, DROP DML – INSERT, UPDATE</p> <p>(½ mark for each correct categorization)</p>	(2)
Section – C		

Q26.	<p>½ Marks for each correct answer</p> <p>(i) BRAND_NAME FLAVOUR LAYS TOMATO UNCLE CHIPS SPICY HALDIRAM TOMATO</p> <p>(ii) BRAND_NAME FLAVOUR PRICE QUANTITY HALDIRAM TOMATO 25 30</p> <p>(iii) BRAND_NAME LAYS</p> <p>(iv) count(distinct (BRAND_NAME)) 3</p> <p>(v) PRICE PRICE*1.5 10 15</p>	(3)
	<p>(vi) distinct (BRAND_NAME) UNCLE CHIPS LAYS HALDIRAM</p>	

27.	<pre> def countINDIA(): f=open('d:\\myfile.txt') data=f.read() data=data.split() ctr=0 for w in data: if w.upper()=='INDIA': ctr=ctr+1 print('Frequency of India is ',ctr) #main countINDIA() </pre> <p style="text-align: center;">OR</p> <pre> def countVowel(): ctr=0 f=open('d:\\myfile.txt') data=f.read() for ch in data: if ch.lower() in 'aeiou': ctr=ctr+1 print('Total number of vowels are : ', ctr) </pre>	(3)
Q28.	<p>(A)select bname, auname, price from books where bid like "comp%"; (i) update books set price = price + 50 where bid like "hist%"; (ii) select * from books order by price; (iii) select bid, bname, qty_issued from books, issued where books.bid = issued.bid; (1/2 mark for each correct SQL)</p> <p>(B) SHOW TABLES; (1 mark for correct answer)</p>	(2+1)
Q29.	<pre> def lenFOURword(L): _____ indexList=[] for i in range(len(L)): if len(L[i])==4: indexList.append(i) return indexList </pre>	(3)

	<p>½ mark for function header</p> <p>½ mark for declaration of indexList</p> <p>½ mark for loop</p> <p>½ mark for checking condition</p> <p>½ mark for appending</p> <p>½ mark for returning</p>	
Q30	<pre> xiiia=[] student=[['Rajveer', '99999999999','XI', 'B'],['Swatantra', '88888888888' ,'XII', 'A'], ['Sajal','77777777777','VIII','A'],['Yash', '1010101010','XII','A']] def pushElement(student): for d in student: if d[2]=='XII' and d[3]=='A': xiiia.append([d[0],d[1]]) def popElement(): while len(xiiia)!=0: print(xiiia.pop()) else: print('Stack Empty') pushElement(student) print(xiiia) popElement() (1.5 marks for correct pushElement() and 1.5 marks for correct popElement()) OR stackItem=[] def Push(SItem): count=0 for k in SItem: if (SItem[k]>=25): </pre>	(3)

```

stackItem.appe
nd(k)
count=count+1
print("The count of elements in the stack is : ", count)

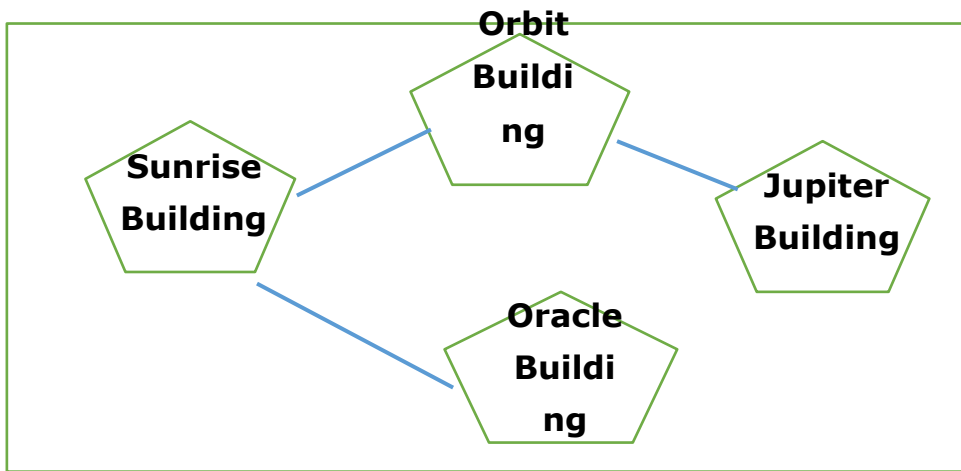
```

(1 mark for correct function
header1 mark for correct
loop
½ mark for correct If statement
½ mark for correct display of count)

Section – D

Q31. i) Suggest a cable layout of connections between the buildings.

(5)



ii) Orbit Building

iii) iii)

- a. Internet Connecting Device/Modem- Orbit Building
- b. Switch- Each Building

iv) MAN, it is formed to connect various locations of the city via various communication media.

v) PAN is "Personal Area Network", basically configured at home area.

Q32.	<p>(A)</p> <p>100</p> <p>5</p> <p>12</p> <p>(B)</p> <p>Statement 1: con1.cursor()</p> <p>Statement 2: mycursor.execute(query)</p> <p>Statement 3: con1.commit()</p> <p>(1 mark for each correct answer)</p> <p style="text-align: center;">OR</p> <p>(A)</p> <p>c&&vVpP</p> <p>(B) Statement 1:con1.cursor()</p> <p>Statement 2: mycursor.execute("select * from student where Marks>75")</p> <p>Statement 3: mycursor.fetchall()</p> <p>(1 mark for each correct statement)</p>	(5)
------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----

33. Advantage of a csv file: (5)

- * It is human readable – can be opened in Excel and Notepad applications
- * It is just like

text file Program:

```
import
```

```
csv def
```

```
ADD():
```

```
    fout=open("teacher.csv","a",newlin
```

```
    e="\n") wr=csv.writer(fout)
```

```
    tid=int(input("Enter teacher id ::
```

```
    ")) name=input("Enter name :: ")
```

```
    mobile=int(input("Enter mobile
```

```
    number :: "))
```

```
    lst=[tid, name, mobile] ----- 1/2 mark
```

```
    wr.writerow(lst) ----- 1/2 mark
```

```
    fout.close()
```

```
def COUNTR():
```

```
    fin=open("teacher.csv","r",newline
```

```
    ="\n") data=csv.reader(fin)
```

```
    d=list(data)
```

```
    print("No of records
```

```
    :",len(d)) fin.close()
```

```
ADD()
```

```
COUNTR()
```

(1 mark for advantage

1/2 mark for importing csv module

1 1/2 marks each for correct definition of ADD() and COUNTR()

1/2 mark for function call statements

) OR

Difference between binary file and csv file: (Any one

difference may be given) Binary file:


```

* Extension is .dat
* Not human readable
* Stores data in the form of
0s and 1s CSV file
* Extension is .csv
* Human readable
* Stores data like a
text file
Program:
import
csv
def
add():
    fout=open("employee.csv","a",newli
ne='\n')
    wr=csv.writer(fout)
    eid=int(input("Enter Employee
Id :: "))
    name=input("Enter
employee name :: ")
    salary
=int(input("Enter salary :: "))
    FD=[eid, name, salary]
    wr.writerow(FD)
    fout.close()

def search():
    fin=open("employee.csv","r",newlin
e='\n')
    data=csv.reader(fin)
    found=False
    print("The Details
are")
    for i in data:
        if int(i[2])>40000:
            found=True
            print(i[0], i[1],
i[2])
    if found==False:
        print("Record not
found")
    fin.close
()
add()
print("Now

```

displaying")

search()

(1 mark for difference

½ mark for importing csv module

1 ½ marks each for correct definition of add() and search()

Section – E

Q34.	<p>(i) stockid</p> <p>(ii) degree = 8, cardinality = 3</p> <p>(iii) (a) insert into stock values(201,'2022-10-18','neckphone','boat',500);</p> <p>(b) update stock set price=price*0.95 where year(dopurchase)=2020; OR</p> <p>(a) delete from stock where year(dopurchase) < 2015;</p> <p>(b) alter table stock add column STATUS char(1);</p>	<p>(1+</p> <p>½+</p> <p>½</p> <p>2)</p>
Q35.	<p>(i) pickle (1 mark for correct module)</p> <p>(ii) fout=open('temp.dat', 'wb') (1 mark for correct statement)</p> <p>(iii) Statement 3: pickle.load(fin)</p> <p>Statement 4: pickle.dump(rec,fout) (1 mark for each correct statement)</p>	(4)

0-O-o- End of Paper -o-O-0