SAMPLE QUESTION PAPER (SOLVED)

CLASS XII
COMPUTER SCIENCE (083)

TIME: 3 hrs **M.M: 70**

1. (a) Differentiate between mutable and immutable objects in Python language with example. (2)

Ans. Every variable in Python holds an instance of an object. There are two types of objects in Python, *i.e.*, **Mutable and Immutable objects.** Whenever an object is instantiated, it is assigned a unique object id. The type of the object is defined at the runtime and it can't be changed afterwards. However, its state can be changed if it is a mutable object.

For example, **int**, **float**, **bool**, **string**, **unicode**, **tuple** are immutable objects in Python. In simple words, an immutable object can't be changed after it is created. Lists, dictionaries and sets are mutable types.

(b) Identify and write the name of the module to which the following functions belong: (1)

(i) ceil() (ii) findall()

Ans. (i) ceil() – math module (ii) findall() – re module

(c) Observe the following Python code very carefully and rewrite it after removing all syntactical errors with each correction underlined. (2)

```
DEF execmain():
    x = input("Enter a number:")
    if (abs(x)= x):
        print"You entered a positive number"
    else:
        x=*-1
        print"Number made positive:"x
execmain()
```

Ans. Corrected code:

```
def execmain():
    x= input("Enter a number:")
    if(abs(x)== x):
        print("You entered a positive number")
    else:
        x *= -1
        print("Number made positive:",x)
execmain()
```

(d) Find the output of the following: (2)

```
L1 = [100,900,300,400,500]
START = 1
SUM = 0
for C in range(START,4):
    SUM = SUM + L1[C]
    print(C, ":", SUM)
    SUM = SUM + L1[0]*10
    print(SUM)
```

Ans. Output is:

```
1 : 900
1900
2 : 2200
3200
3 : 3600
4600
```

(e) Write the output of the following Python program code: (3)

```
def ChangeList():
    L=[]
    L1=[]
    L2=[]
    for i in range(1,10):
        L.append(i)
    for i in range(10,1,-2):
        L1.append(i)
    for i in range(len(L1)):
        L2.append(L1[i]+L[i])
    L2.append(len(L)-len(L1))
    print(L2)
ChangeList()
```

Ans. Output is:

```
[11,10,9,8,7,4]
```

(f) Study the following program and select the possible output(s) from the options (i) to (iv) following it. Also, write the maximum and the minimum values that can be assigned to the variable Y. (2)

```
import random
X= random.random()
Y= random.randint(0,4)
print(int(X),":",Y+int(X))
```

(i) 0 : 0 (ii) 1 : 6

(iii) 2 : 4 (iv) 0 : 3

Ans. (i) and (iv) are the possible outputs. Minimum value that can be assigned is – Y = 0. Maximum value that can be assigned is – Y = 3

2. (a) Explain operator overloading with the help of an example. (2)

Ans. Operator overloading is a feature in Python that allows the same operator to have a different meaning according to the context. It signifies giving extended meaning beyond the predefined operational meaning. *For example*, operator + is used to add two integers as well as join two strings and merge two lists.

For example,

```
>>print(44 + 2)
46
# concatenate two strings
print("Python"+"Programming")
PythonProgramming
# Product of two numbers
print(5 * 5)
25
# Repeat the String
print("Hello"*3)
HelloHelloHello
```

(b) Find the output of following: (2)

```
colors = ["violet", "indigo", "blue", "green", "yellow", "orange", "red"]
del color[4]
colors.remove("blue")
colors.pop(3)
print(colors)
```

Ans. Output is:

```
['violet', 'indigo', 'green', 'red']
```

(c) Find the output of the following: (2)

```
str = "Pythonforbeginners is easytolearn"
str2 = "easy"
print("The first occurrence of str2 is at : ", end="")
print(str.find( str2, 4))
print("The last occurrence of str2 is at : ", end="")
print(str.rfind( str2, 4))
```

Ans. Output is:

```
The first occurrence of str2 is at : 22
The last occurrence of str2 is at : 22
```

3. (a) Write the definition of a function Reverse(X) in Python to display the elements in reverse order such that each displayed element is twice of the original element (element *2) of the List X in the following manner: (2)

Example:

If List X contains 7 integers as follows:

X[0]	X[1]	X[2]	X[3]	X[4]	X[5]	X[6]
4	8	7	5	6	2	10

After executing the function, the array content should be displayed as follows:

If List

Ans.

20	4	12	10	14	16	8

```
    def Reverse(X):
        for i in range(len(X)-1,-1,-1):
            print(X[i]*2)
```

(b) Consider the following unsorted list: 95 79 19 43 52 3. Write the passes of bubble sort for sorting the list in ascending order till the 3rd iteration. (3)

Ans. [79, 19, 43, 52, 3, 95]

[19, 43, 52, 3, 79, 95]

[19, 43, 3, 52, 79, 95]

(c) Write a user-defined function to generate odd numbers between a and b (including b). (3)

Note: a and b are received as an argument by the function.

Ans.

```
def generateodd(a,b):
    for i in range(a, b+1):
        if(i%2 != 0):
            print(i)
```

(d) Observe the following code and answer the questions that follow: (1)

```
File = open("Mydata","a") _______________________ #Blank1
File.close()
```

(i) What type (Text/Binary) of file is Mydata?

(ii) Fill in Blank 1 with a statement to write “ABC” in the file “Mydata”.

Ans. (i) Text File

(ii) File.write("ABC")

4. (a) Write any one advantage and one disadvantage of Coaxial cable. (1)

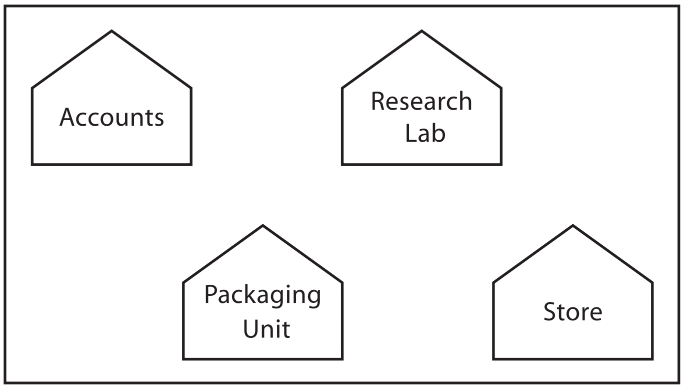
Ans. *Advantages:*

- It is less susceptible to noise or interference (EMI or RFI) as compared to twisted pair cable.
- It supports high bandwidth signal transmission as compared to twisted pair.
- It is easy to wire and easy to expand due to its flexibility.

Disadvantages:

- It is bulky.
- It is expensive to install for longer distances due to its thickness and stiffness.

(b) Riana Medicos Centre has set up its new centre in Dubai. It has four buildings as shown in the diagram given below: (4)



Distances between various buildings are as follows:

Accounts to Research Lab	55 m
Accounts to Store	150 m
Store to Packaging Unit	160 m
Packaging Unit to Research Lab	60 m
Accounts to Packaging Unit	125 m
Store to Research Lab	180 m

Number of computers:

Accounts	25
Research Lab	100
Store	15
Packaging Unit	60

As a network expert, provide the best possible answer for the following queries:

(i) Suggest the type of network established between the buildings.

Ans. LAN (Local Area Network)

(ii) Suggest the most suitable place (*i.e.*, building) to house the server of this organization.

Ans. Research Lab as it has the maximum number of computers.

(iii) Suggest the placement of the following devices with justification: (a) Repeater (b) Hub/Switch

Ans. (a) **Repeater:** It should be placed between Accounts and Packaging Unit, Accounts to Research Lab, Store to Research Lab and Accounts to Packaging Unit.

(b) Switch should be placed in each of the buildings for better traffic management.

(iv) Suggest a system (hardware/software) to prevent unauthorized access to or from the network.

Ans. Firewall.

(c) Expand the following: (2)

(i) VoIP

Ans. Voice over Internet Protocol

(ii) SMTP

Ans. Simple Mail Transfer Protocol

(iii) TDMA

Ans. Time Division Multiple Access

(iv) TCP/IP

Ans. Transmission Control Protocol/Internet Protocol

(d) The following is a 32-bit binary number usually represented as 4 decimal values, each representing 8 bits, in the range 0 to 255 (known as octets) separated by decimal points. (1)

140.179.220.200

What is it? What is its importance?

Ans. It is an IP Address. It is used to identify the computers on a network.

(e) Name the network tools used in the given situations: (4)

(i) To troubleshoot internet connection problems

Ans. ping

(ii) To see the IP address associated with a domain name

Ans. nslookup

(iii) To look up registration record associated with a domain name.

Ans. whois

(iv) To test the speed of internet connection

Ans. speedtest.net

(f) What is a cloud? (1)

Ans. A cloud is a combination of networks, hardware, services, storage and interfaces that helps in delivering computing as a service. It has three users:

(i) End users

(ii) Business management users

(iii) Cloud service provider

(g) What are the effects of Network Congestion? (1)

Ans. **Network congestion** in data networking and queuing theory is the reduced quality of service that occurs when a network node or link is carrying more data than it can handle. Typical effects include queuing delay, packet loss or the blocking of new connections.

5. (a) Write the difference between GET and POST method. (1)

Ans. A web browser may be the client and an application on a computer that hosts a website may be the server.

So, to request a response from the server, there are mainly two methods:

(i) **GET :** to request data from the server

(ii) **POST :** to submit data to be processed to the server

(b) Write a MySQL-Python connectivity to retrieve data, one record at a time, from city table for employees with id less than 10. (2)

Ans.

```
import MySQLdb as my
try:
    db = my.connect(host="localhost",
                    user="root",
                    passwd="",
                    database="India")
    cursor = db.cursor()
    sql = "select * from city where id < 10"
    number_of_rows = cursor.execute(sql)
    print(cursor.fetchone()) # fetch the first row only
    db.close()
except my.DataError as e:
    print("DataError")
    print(e)
```

(c) What are the basic steps to connect Python with MYSQL using table Members present in the database ‘Society’? (3)

Ans.

```
import MySQLdb
conn = MySQLdb.connect(host=”localhost”, user=‘root’, password =‘ ’,
                        database=”Society”)
cursor = conn.cursor()
cursor.execute('SELECT COUNT(MemberID) as count FROM Members WHERE id = 1')
row = cursor.fetchone()
conn.close()
print(row)
```

(d) What is the role of Django in website design? (1)

Ans. Django is a high-level Python web framework, designed to help build complex web applications simply and quickly. Django makes it easier to build better web apps quickly and with less code.

6. (a) Write the steps to connect with database “testdb” with Python programming. (2)

Ans. Steps to be followed are:

- Import mysqldb as db
- Connect
- Cursor
- Execute
- Close

(b) Which method is used to retrieve all rows and single row? (1)

Ans. fetchall(),fetchone()

(c) Consider the table ‘empsalary’. (1)

eid	esalary
101	40000
102	NULL
104	51000
107	NULL

To select tuples with some esalary, Arun has written the following erroneous SQL statement:

SELECT eid, esalary FROM empsalary WHERE esalary = something;

Write the correct SQL statement.

Ans. The correct SQL statement is:

SELECT eid, esalary FROM empsalary WHERE esalary is NOT NULL;.

(d) Table COACHING is shown below. Write commands in SQL for (i) to (iii) and output for (iv) and (v) (4)

ID	NAME	AGE	CITY	FEE	PHONE
P1	SAMEER	34	DELHI	45000	9811076656
P2	ARYAN	35	MUMBAI	54000	9911343989
P4	RAM	34	CHENNAI	45000	9810593578
P6	PREMLATA	36	BHOPAL	60000	9910139987
P7	SHIKHA	36	INDORE	34000	9912139456
P8	RADHA	33	DELHI	23000	8110668888

(i) Write a query to display name in descending order whose age is more than 23.

Ans. select name from coaching where age>23 order by name desc;

(ii) Write a query to find the average fee grouped by age from customer table.

Ans. select avg(fee) from coaching group by age;

(iii) Write query details from coaching table where fee is between 30000 and 40000.

Ans. Select * from coaching table where fee is between 30000 and 40000;

(iv) Select sum(Fee) from coaching where city like “%O% ;

Ans. 94000

(v) Select name, city from coaching group by age having count(age)>2;

Ans. Empty set

7. (a) What are the proper methods and steps for the disposal of used electronic items? (1)

Ans. Explanation about any methods like:

- Landfilling
- Incineration
- Reuse of electronic devices
- Acid Bath
- Recycling of e-waste

(b) When did IT Act come into force? (1)

Ans. 17th October, 2000.

(c) Explain the gender and disability issues while using computers in the classroom. (1)

Ans. It has been seen in many studies that in many countries computer use in schools is dominated by men. Female teachers have less regard for their skills and knowledge than their male colleagues.

Females know less about information technology, enjoy using the computer less than male students, and perceive more problems with software.

(d) How can we recycle e-waste safely? (2)

Ans. (i) Use a certified e-waste recycler.

(ii) Visit civic institutions. Check with your local government, schools and universities for additional responsible recycling options.

(iii) Explore retail options.

(iv) Donate your electronics.

(e) Name some threats to computer security. (1)

Ans.
- Snooping
- Virus and Trojans
- Identity theft
- Child pornography, etc.
- Phishing

(f) What is meant by the term Cyber Forensics? (2)

Ans. Cyber forensics is an electronic discovery technique used to determine and reveal technical criminal evidence. It often involves electronic data storage extraction for legal purposes. Although still in its infancy, cyber forensics is gaining traction as a viable way of interpreting evidence.

Cyber forensics is also known as computer forensics.

(g) Write any two categories of cyber crime. (1)

Ans. Cyber crime encompasses a wide range of activities, but these can generally be broken into two categories:

- Crimes that target computer networks or devices: Such types of crimes include viruses and denial-of-service (DoS) attacks.
- Crimes that use computer networks to advance other criminal activities: These types of crimes include cyberstalking, phishing and fraud or identity theft.

(h) How does phishing happen? (1)

Ans. Phishing is a fraudulent attempt to obtain sensitive information such as usernames, passwords, and credit card details (and money), often for malicious reasons, by disguising as a trustworthy entity in an electronic communication. Phishing is typically carried out by email spoofing or instant messaging, and it often directs users to enter personal information at a fake website, the look and feel of which is identical to the legitimate one and the only difference is the URL of the website in question.

(i) Define and name some open source software. (1)

Ans. GPl, Apache, Creative Commons, Mozilla Firefox, Open Office.org, etc.

(j) What are Intellectual Property Rights (IPR)? (1)

Ans. IPR is a general term covering patents, copyright, trademark, industrial designs, geographical indications, layout design of integrated circuits, undisclosed information (trade secrets) and new plant varieties.

COMPUTER SCIENCE WITH PYTHON (083)
BLUE PRINT

Type of Questions	Marks Per Question	Total Number of Questions	Total Marks
VSA	1	15	15
SA I	2	17	34
SA II	3	03	09
LA	4	03	12
Total		**38**	**70**

- **VSA: Very Short Answer Type**
- **SA : Short Answer Type**
- **LA : Long Answer Type**

BLUE PRINT

Topic / Unit	VSA (1 mark)	SAI (2 marks)	SA II (3 marks)	LA (4 marks)	Total
Programming and Computational Thinking	2(2)	9(18)	2(6)	1(4)	14(30)
Computer Networks	4(4)	2(4)	1(3)	1(4)	8(15)
Data Management	7(7)	2(4)	-	1(4)	10(15)
Society, Law and Ethics	2(2)	4(8)	-	-	6(10)
Total					38(70)

Difficulty Level :

Easy : 15%

Average : 70%

Difficult : 15%

HOTS-based Questions : 20%

SAMPLE PAPER - II
COMPUTER SCIENCE with PYTHON (083)
CLASS –XII

MM.-70 **TIME: 3 HRS**

1 (a) Which of the following can be used as valid variable identifier(s) in Python? 2

(i) 4thSum

(ii) Total

(iii) Number#

(iv) Data

Ans.
ii) Total iv) Data

(b) Name the Python Library modules which need to be imported to invoke the following functions 1

(i) floor()

(ii) randn()

Ans.
(i) math (ii) random

OR

Write the modules that will be required to be imported to execute the following code in Python.

```
def main( ):
        for i in range (len(string)) ):
                if string [i] = = ‘’
                        print
                elif
                        c=string[i].upper()
        print “string is:”c
        print    “String length=”,len(sring.floor())
```

Ans. Math module and String module

(c) Rewrite the following code in python after removing all syntax error(s).Underline each correction done in the code. 3

```
STRING=""HAPPY NEW YEAR"
for S in range[0,8]:
print STRING(S)
print S+STRING
```

Ans.

```
STRING = "HAPPY NEW YEAR"
for S in range(0,14):
      print(STRING[S])
      print(S,STRING)
```

OR

Write a function mul ( ) which accepts list L, odd_L, even_L as its parameters. Transfer all even values of list L to even_L and all odd values of L to odd_L.

3

eg. If L = [10, 22, 24, 36, 37, 43, 51]

odd_L = [37, 43,51]

and even_L = [10, 22, 24, 26]

Ans. defmul(l, odd_l, even_l):

```
            Odd_l=[]
            Even_l=[]
            fori in range(len(l)):
                  if (l[i]%2==0):
                        Even_l+=[l[i]]
                  else:
                        Odd_l+=[l[i]]
```

(d) Find and write the output of the following python code: 2

```
TXT = ["20","50","30","40"]
CNT = 3
TOTAL = 0
for C in [7,5,4,6]:
      T = TXT[CNT]
      TOTAL = float(T) + C
      print(TOTAL)
      CNT -= 1
```

Ans.

```
47.0
35.0
54.0
```

```
26.0
```

OR

Find the output of the following program: 2

```
          def main ( ) :
                    Moves=[11, 22, 33, 44]
                    Queen=Moves
                    Moves[2]+=22
                    L=Len(Moves)
                    For i in range (L)
                              print "Now@", Queen[L-i-1], "#", Moves [i]
```

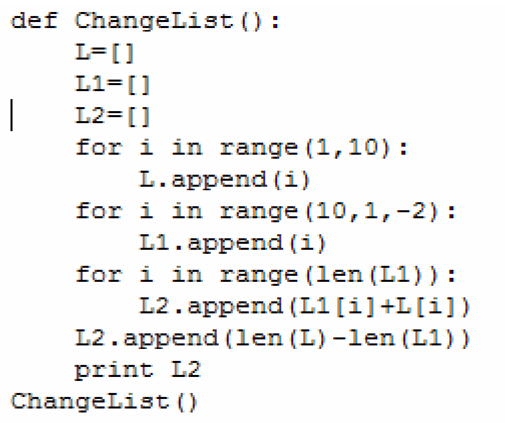
Ans. Now @ 44 # 11

Now @ 55 # 22

Now @ 22 # 55

Now @ 11 # 44

(e) What output will be generated when the following Python code is executed? 2



```
def ChangeList():
    L=[]
    L1=[]
    L2=[]
    for i in range(1,10):
        L.append(i)
    for i in range(10,1,-2):
        L1.append(i)
    for i in range(len(L1)):
        L2.append(L1[i]+L[i])
    L2.append(len(L)-len(L1))
    print L2
ChangeList()
```

Ans.

```
[11, 10, 9, 8, 7, 4]
```

(f) What are the possible outcome(s) executed from the following code? Also specify the maximum and minimum values that can be assigned to variable N. 2

```
import random
NAV = ["LEFT","FRONT","RIGHT","BACK"]
NUM = random.randint(1,3)
NAVG = ""
for C in range(NUM,1,-1):
     NAVG = NAVG + NAV[C]
     print(NAVG)
```

Ans.
No Output

or

RIGHT

or

BACK

BACKRIGHT

Max value 3 and minimum value 1 for variable NUM

2 (a) List one similarity and one difference between List and Dictionary datatype. 2

Ans.
Similarity: Both List and Dictionary is mutable datatypes.
Dissimilarity: List is a sequential data type i.e. they are indexed. Dictionary is a mapping datatype. It consists of key: value pair.
For example, L =[1,2,3,4,5] is a list
D = {1:"Ajay",2:"Prashant,4:"Himani"} is a dictionary where 1,2,4 are keys and "Ajay",Prashant,"Himani" are their corresponding values.

(b) Rewrite the following Python program after removing all the syntactical errors (if any), underlining each correction.: 2

```
def checkval:
     x = input("Enter a number")
     if x % 2 = 0:
           print x,"is even"
     else if x<0:
           print x,"should be positive"
     else;
           print x,"is odd"
```

Ans. Corrected code:

```
def checkval():
      x = int(input("Enter a number:"))
      if x % 2 == 0:
           print(x," is even")
      elif x<0:
           print(x, "should be positive")
      else:
           print(x, "is odd")
```

(c) Find the output of the following Python program: 3

```
def makenew(mystr):
   newstr = ""
   count = 0
   for i in mystr:
         if count%2 != 0:
               newstr = newstr + str(count)
```

```
      else:
          if i.islower():
              newstr = newstr + i.upper()
          else:
              newstr = newstr + i
      count += 1
  newstr = newstr + mystr[:1]
  print("The new string is:", newstr)
makenew("sTUdeNT")
```

Ans.

```
The new string is: S1U3E5Ts
```

(d) Discuss the strategies employed by python for memory allocation? 3

Ans.
Python uses two strategies for memory allocation- Reference counting and Automatic garbage collection:
Reference Counting: works by counting the number of times an object is referenced by other objects in the system. When an object's reference count reaches zero, Python collects it automatically.
Automatic Garbage Collection: Python schedules garbage collection based upon a threshold of object allocations and object de-allocations. When the numbers of allocations minus the number of deallocations are greater than the threshold number, the garbage collector is run and the unused block of memory is reclaimed.

3 (a) Write a user defined function findname(name) where name is an argument in Python to delete phone number from a dictionary phonebook on the basis of the name, where name is the key. 3

Ans.

```
def findname(name):
    if phonebook.has_key():
        del phonebook[name]
    else:
        print("Name not found")
    print("Phonebook Information")
    print("Name",'\t', "Phone number")
    for i in phonebook.keys():
        print(i,'\t', phonebook[i])
```

(b) Write the specific purpose of functions used in plotting: 2

i) plot()
ii) Legend()

Ans.
i) plot(): A line chart or line graph can be created using the plot() function available in pyplot library. For example, the basic syntax for creating line plots is plt.plot(x,y), where x and y are the points or specify the (x, y) pairs that form the line.
ii) Legend(): legend is the text or string that “has to be read” to understand the graph. Legends are used in line graphs to explain the function or the values underlying the different lines of the graph.

(c) Write a python program to plot the algebraic equation: 10x + 14. 2

Ans.

```
# Program to evaluate an algebraic expression

# 10x + 14 using Line Chart

import numpy as np

from matplotlib import pyplot as plt

x = np.arange(12, 20)

y = 10 * x + 14

plt.title("Graph for an Algebraic Expression")

plt.xlabel("x axis")

plt.ylabel("y label")

plt.plot(x,y)

plt.show()
```

(d) Write definition of a Method MSEARCH(STATES) to display all the state names from a list of STATES, which are starting with alphabet M. 3

For example:

If the list STATES contains

["MP","UP","WB","TN","MH","MZ","DL","BH","RJ","HR"]

The following should get displayed

MP
MH
MZ

Ans.

```
def MSEARCH(STATES):
      for i in STATES:
            if i[0]=='M':
                  print(i)
```

(e) Differentiate between file modes r+ and w+ with respect to Python. 2

Ans.

r+ Opens a file for both reading and writing. The file pointer placed at the beginning of the file.
w+ Opens a file for writing. The file pointer placed at the beginning of the file.

4 (a) Write about types of network depending upon geographical location. 1

Ans.

Types of network depending upon geographical location: PAN, LAN, MAN, WAN.

(b) Write down difference between private cloud and public cloud. 2

Ans.

Public Cloud	**Private Cloud**
Public cloud refers to a common cloud service made available to multiple subscribers.	Consist of computing resources used exclusively owned by one business or organization.

Cloud resources are owned and operated by third party cloud service provider and delivered over the internet.	Services and infrastructure are always maintained on a private network and the hardware and software are dedicated solely to one organization
Microsoft Azure, Google drive, Amazon Cloud Drive, iCloud etc.	Used by Government agencies, financial institutions, mid and large size organization

(c) Write one restriction of wired and one wireless networks. 2

Ans.
Limitation of wireless network is cost, security
Limitation of wired network is Speed ,extra cost for expansion etc

(d) How IP address differ from MAC address? 2

Ans.
IP address is a number assigned to the connection in a network. The basic difference between MAC address and IP address is that a MAC address uniquely identifies a device that wants to take part in a network. On the other hand, an IP address uniquely defines a connection of a network with an interface of a device.

(e) Expand the following terms: 3

1. MAC
2. DNS
3. URL

Ans.

1. MAC-Media Access Control,
2. DNS is Domain Name Server,
3. URL is Uniform Resource Locator.

5 (a) Write difference between IP v-4 and IPv-6. 2

Ans.
The difference between IPv4 and IPv6 Addresses. An IP address is binary numbers but can be stored as text for human readers. For example, a 32-bit numeric address (IPv4) is written in decimal as four numbers separated by periods. IPv6 addresses are 128-bit IP address written in hexadecimal and separated by colons.

(c) Write the purpose of following commands 3

1. whois
2. ipconfig
3. nslookup

Ans.

1. **whois**: Lookup tool finds contact information for the owner of a specified IP address. The ipwhois Lookup tool displays as much information as possible for a given IP address.
2. **ipconfig**: In Windows, ipconfig is a console application designed to run from the Windows command prompt. This utility allows you to get the IP address information of a Windows computer. It also allows some control over active TCP/IP connections.

3. **nslookup**: is a network administration command-line tool available for many computer operating systems. It is used for querying the Domain Name System (DNS) to obtain domain name or IP address mapping information.

(d) Write SQL queries for (i) to (iv) and find outputs for SQL queries (v) to (vii), which are based on the table. 5

Table: Transact

TRNO	ANO	AMOUNT	TYPE	DOT
T001	101	2500	Withdraw	2017-12-21
T002	103	3000	Deposit	2017-06-01
T003	102	2000	Withdraw	2017-05-12
T004	103	1000	Deposit	2017-10-22
T005	101	12000	Deposit	2017-11-06

(i) To display minimum amount transaction from the table.

Ans.

select min(amount) from Transact;

(ii) To display total amount withdrawn from table.

Ans.

select sum(amount) from Transact where type = “Withdraw”;

(iii) To display number of deposited transaction.

Ans.

select sum(amount) from Transact where type=”Deposit”;

(iv) To display ANO,DOT, AMOUNT for maximum amount transaction.

Ans.

select ANO, DOT, AMOUNT from Transact where AMOUNT = max(AMOUNT);

(v) SELECT ANO, COUNT(*), MIN(AMOUNT) FROM TRANSACT
GROUP BY ANO HAVING COUNT(*)> 1

Ans.

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

(vi) SELECT COUNT(*), SUM(AMOUNT) FROM TRANSACT
WHERE DOT <= '2017-06-01';

Ans.

```
COUNT(*) SUM(AMOUNT)
```

2 5000

6 (a) Is the Django installation as same as other packages? 2

Ans.
No

(b) Start project command creates four basic Django project in Directory. Write any two file Names. 2

Ans.
The startproject command creates a basic Django project directory structure with the following files:

- manage.py
- settings.py

(c) What is Django? 2

Ans.
Django is a Web framework written in Python. But that's an abstract definition. In practice, Django is a Python package that lives inside the site-packages directory of your current Python installation.

(d) Write difference between GET and POST method. 2

Ans.

GET method	POST method
all form data is encoded into the URL appended to the action URL as query string parameters	form data appears within the message body of the HTTP request
Parameters remain in browser history hence cannot be used to send password like sensitive information.	Parameters are not saved in browser history hence can be used to send sensitive information
can be bookmarked	cannot be bookmarked
Easier to hack for script kiddies	More Difficult to hack
Can be cached	Cannot be cached

7 (a) What are the environmental issues of e-waste? 2

Ans.
E-waste, or electronic waste, is waste from all sorts of electronics ranging from computers and mobile phones, to household electronics such as food processors, pressure, cookers etc.
The effects of improper disposal of this E-waste on the environment are little known; these impacts nonetheless pose very real threats and dangers to the global environment at large.

(b) What do you understand by the term Plagiarism? Write 2 software's used as Plagiarism checker. 3

Ans.
Plagiarism is "copying and publication" of another author's "language, thoughts, ideas, or

		expressions" and the representation of them as one's own original work. Plagiarism is considered academic dishonesty and a breach of journalistic ethics Here is a list of the top 10 free plagiarism checker tools available today. DupliChecker. Grammarly. Paperrater. Plagiarisma.	
	(c)	What is Identity Theft? How to prevent this? Ans. Identity theft is known as identity fraud, is a crime in which a cracker obtains key pieces of personally identifiable information, such as Social Security or driver's license numbers, in order to impersonate someone else. Prevention Techniques from Identity theft: 1. Don't share personal information (birth date, Social Security number, or bank account number) just because someone asks for it. 2. Collect mail every day. 3. Pay attention to your billing cycles.	3
	(d)	List down some points about Societal changes introduced by technology. Technology and Social Change. Ans. Technology is the application of scientific knowledge to the making of tools to solve specific problems. Technological advances such as automobiles, airplanes, radio, television, cellular phones, computers, modems, and fax machines have brought major advances and changes to the world.	2

MODEL TEST PAPER–I

CLASS XI

COMPUTER SCIENCE (083)

TIME: 3 hrs **M.M: 70**

1. (a) Differentiate between Syntax Error and Run-Time Error. Also, write a suitable example in Python to illustrate both. (2)

(b) Name the Python Library modules which need to be imported to invoke the following functions: (1)

(i) log() (ii) match()

(c) Rewrite the following code in Python after removing all syntax error(s). Underline each correction done in the code. (2)

```
Num = int(rawinput("Number:"))
Sum = 0
for i in range(10,Num,3) Sum+=i
    if i%2=0:
        print ( i*2)
    Else:
        print ( i*3 print Sum)
```

(d) Find and write the output of the following Python code: (3)

```
L =["X",20,"Y",10,"Z",30]
CNT = 0
ST = ""
INC = 0
for C in range(1,6,2):
    CNT= CNT + C
    ST= ST + L[C-1]+"@"
    INC = INC + L[C]
    print (CNT,INC,ST)
```

(e) Carefully observe the following Python code and answer the questions that follow. (2)

```
global x
x=5
def fun2():
    x=3
    print(x)
x=x+1
print(x)
```

On execution, the above code produces the following output:

6

3

Explain the output with respect to the scope of the variables.

(f) Observe the following Python code and find out which of the given options (i) to (iv) are the expected correct output(s). Also, assign maximum and minimum values that can be assigned to the variable ‘Go’. (2)

```
import random
X=[100,75,10,125]
Go = random.randint(0,3)
for i in range(Go):
    print(X[i],"$$",)
```

What will be the minimum and maximum value for ‘i’? Write all the possible output(s).

2. (a) Kritika was asked to accept a list of even numbers but she did not put the relevant condition while accepting the list. Write a user-defined function oddtoeven(L) that accepts the List L as an argument and converts all the odd numbers into even by multiplying them by 2. (2)

(b) Write a Python function generatefibo(n) where n is the limit, using a generator function Fibonacci (max) (where max is the limit n) that produces the Fibonacci series. (2)

(c) Consider the following unsorted list: 95, 79, 19, 43, 52, 3. Write the passes of bubble sort for sorting the list in ascending order till the 4th iteration. (2)

(d) Write a program to perform insert and delete operations on a Queue containing Members details as given in the following definition of item node: (4)

Member No	integer
Member Name	String
Age	integer

```
def isEmpty(Qu ):
        if Qu == []:
            return True
    else:
        return False
def Enqueue(Qu, item):
# Write the code to insert member details using Queue.
def Dequeue(Qu):
# Write the code to delete a member using Queue.
```

(e) Evaluate the following Postfix expression: (2)

20, 4, +, 3, -, 7, 1

3. (a) Observe the following code and answer the questions that follow: (1)

```
File = open("Mydata","a")
________________________ #Blank1
File.close()
```

(i) What type (text/binary) of file is Mydata?

(ii) Fill in Blank 1 with a statement to write “ABC” in the file “Mydata”.

(b) Write a user-defined function in Python that displays the number of lines starting with ‘H’ in the file Para.txt. *For example*, if the file contains: (2)

Whose woods these are I think I know.
His house is in the village though;
He will not see me stopping here
To watch his woods fill up with snow.

Then the line count should be 2.

OR

Write a Python program to copy file1.txt into file2.txt.

(c) Consider a binary file Employee.dat containing details such as empno:ename:salary (separator ‘ :’). Write a Python function to display details of those employees who are earning between ₹ 20,000 and ₹ 40,000 (both values inclusive). (3)

OR

Write the Python code to implement stack of integers.

4. (a) Expand the following: (2)

(i) VoIP (ii) SMTP

(iii) TDMA (iv) TCP/IP

(b) Write one advantage and one disadvantage of Bus topology. (1)

OR

Give one example each of wireless and wired communication media.

(c) The following is a 32-bit binary number, usually represented as 4 decimal values, each representing 8 bits, in the range 0 to 255 (known as octets) separated by decimal points. (1)

140.179.220.200

What is it? What is its importance?

OR

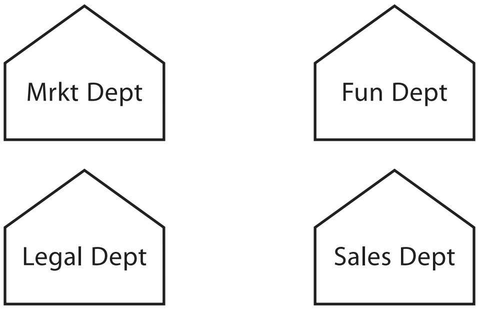
What kind of data gets stored in cookies and how is it useful?

(d) Daniel has to share data among various computers of his two office branches situated in the same city. Name the network (out of LAN, WAN, PAN and MAN) which should be formed in this process. (1)

OR

Which protocol helps us to transfer files to and from a remote computer?

(e) SunRise Pvt. Ltd. is setting up the network in Ahmedabad. There are four departments—MrktDept, FunDept, LegalDept, SalesDept. (4)



Distance between various buildings is as under:

MrktDept to FunDept	80 m
MrktDept to LegalDept	180 m
MrktDept to SalesDept	100 m
LegalDept to SalesDept	150 m
LegalDept to FunDept	100 m
FunDept to SalesDept	50 m

Number of computers in the buildings:

MrktDept	20
LegalDept	10
FunDept	08
SalesDept	42

(i) Suggest the network type between the Departments and specify topology.

(ii) Suggest the most suitable building to place the server with a suitable reason.

(iii) Suggest the placement of i) modem, ii) Hub/Switch in the network.

(iv) The organization is planning to link its sale counters situated in various parts of the same city. Which type of network out of LAN, WAN, MAN will be formed? Justify.

(f) Give two differences between public cloud and private cloud. (2)

(g) Name the network tools used in the given situation— (3)

(i) To troubleshoot internet connection problems

(ii) To see the IP address associated with a domain name

(iii) To look up the registration record associated with a domain name

(h) What makes a protocol have a higher bandwidth? (1)

5. Study the following tables STAFF and SALARY and write SQL command for the questions (a) to (d) and give output for (e) and (f). (6)

Table: STAFF

ID	NAME	DEPT	SEX	EXPERIENCE
101	Siddharth	Sales	M	12
104	Raghav	Finance	M	6
107	Naman	Research	M	10
114	Nupur	Sales	F	3
109	Jhanvi	Finanace	F	9
105	Rama	Research	M	10
117	James	Sales	F	3
111	Binoy	Finance	F	12
130	Samuel	Sales	M	15

Table: SALARY

ID	Basic	Allowance	Commission
101	15000	1500	800
104	17000	1200	500
107	16000	1000	200
114	20000	2200	550
109	18000	1000	250
105	15000	1200	150
117	18000	1700	100
111	20000	1500	300
130	18000	1200	500

(a) Display NAME of all the staff members who are in SALES having more than 10 years' experience from the table staff.

(b) Display the average Basic Salary of all staff members working in "Finance" department using the tables staff and salary.

(c) Display the minimum ALLOWANCE of female staff.

(d) Display the highest commission among all male staff.

(e) Select count(*) from STAFF group by sex;

(f) SELECT NAME,DEPT,BASIC FROM STAFF,SALARY WHERE DEPT='SALES' AND STAFF.ID=SALARY.ID;

6. (a) Write the necessary command to incorporate SQL interface within Python. (1)

(b) Give the Python Statement to: (4)

(i) establish a connection with the database
(ii) create a cursor for the connection
(iii) execute create table SQL query
(iv) insert data into the table created in the above question

(c) What is Django? (2)

(d) Give any two differences between GET and POST submission methods of HTML form. (2)

7. (a) What is meant by intellectual property? What are the provisions for protecting intellectual property in India? (2)

(b) How does technology affect society? Give two points in favour of technology and two points against it. (2)

(c) How can we recycle e-waste safely? (2)

(d) Name any one open source operating system and open source browser. (1)

(e) How does phishing happen? (1)

(f) What is cyber forensics? What is its use? (2)

MODEL TEST PAPER–II

CLASS XII

COMPUTER SCIENCE (083)

TIME: 3 hrs **M.M: 70**

Instructions:

(i) Attempt all three sections and all questions are compulsory.

(ii) Programming Language: Python

SECTION–A

1. Answer the following questions:

(a) Differentiate between Syntax Error and Run-Time Error. Also, write a suitable example in Python to illustrate both. (2)

(b) Name the Python Library modules which need to be imported to invoke the following functions: (1)

(i) fabs()

(ii) bar()

(c) Rewrite the following code in Python after removing all syntax error(s). Underline each correction done in the code. (2)

```
a = int(input("Value:"))
b = 0
for c in range(1,a,2)
    b + = c if c%2 = 0:
        Print (c*3) Else:
        print (c*) print (b)
```

(d) Which string method is used to implement the following? (2)

(i) To count the number of characters in the string.

(ii) To change the first character of the string in capital letter.

(iii) To check whether the given character is a letter or a number.

(iv) To change lower case to upper case letter.

(v) To change one character into another character.

(e) Differentiate between append() and extend() functions with examples. (2)

(f) Differentiate between read() and write() functions with examples. (2)

(g) What possible output(s) is/are expected to be displayed on screen at the time of execution of the program from the following code? Also specify the maximum values that can be assigned to each of the variables **start** and **end**. (2)

```
import random
POINTS=[20,40,10,30,15]; POINTS=[30,50,20,40,45]
start=random.randint(1,3)
end=random.randint(2,4)
for c in range(start,end+1):
    print(POINTS[c],"#",)
```

(i) 50# 20#	(ii) 40# 30#
(iii) 50# 40# 20#	(iv) 10# 15#

2. Answer the following questions:

(a) How many values may a function return in Python? Illustrate the same with an example in Python. (2)

(b) Write a program in Python to print the Fibonacci series till n terms, where n is entered by users. (2)

(c) Write a program in Python to input a list of numbers from the user and display only those numbers which are divisible by 3 but not by 5. (2)

3. Answer the following questions:

(a) Write a program in Python to search a number from the entered sorted list using binary search. (2)

OR

Write a Python function **generatefibo(n)** where n is the limit, using a generator function **Fibonacci(max)**, where max is the limit n, that produces Fibonacci series.

(b) Write the definition of a method COUNTNOW(PLACES) to find and display those place names, in which there are more than 7 characters. *For example*: If the list PLACES contains (3)

["SYDNEY","TOKYO","PINKCITY","BEIJING","SUNCITY"]

The following should get displayed : **PINKCITY**

OR

Consider the following unsorted list, 95 79 19 43 52 3. Write the passes of bubble sort along with the respective algorithm for sorting the list in ascending order till the 4th iteration.

4. Answer the following questions:

(a) Write a statement in Python to open a text file STORY.TXT so that new contents can be added at the end of it. (1)

OR

Neena intends to position the file pointer at the beginning of a text file. Write Python statement for the same assuming F is the File object.

(b) Explain all file modes in Data File Handling in Python for Data Handling. (2)

OR

Write a function **countthe()** in Python to read the text file “DATA.TXT” and count the number of times “the” occurs in the file.

For example, if the file “DATA.TXT” contains:

“This is my website. I have displayed my preferences in the CHOICE section. The website looks good.”

The **countthe()** function should display the output as:

“the occurs 2 times”.

(c) Write a program which inputs two lists ‘FirstName’ and ‘LastName’ and returns answer in list ‘name’ with ‘FirstName’ and ‘LastName’ concatenated. (3)

OR

Write a user-defined function named Count() that will read the contents of text file named “Report.txt” and count the number of lines which start with either “J” or “M” and display the count, *e.g.*,

In the following paragraph, there are 3 lines starting with “J” or “M”.

“India is the fastest growing economy. India is looking for more investments from around the globe.

The whole world is looking at India as a great market.

Most of the Indians can foresee the dizzy heights that India is capable of reaching.”

SECTION–B

5. Answer the following questions:

(a) Differentiate between having and Group By clauses of a table with an example. (2)

(b) What do you mean by group/aggregate functions? Give any four examples of these functions. (2)

(c) Explain with the help of examples: (2)

(i) Primary Key

(ii) Foreign key

(d) Differentiate between GET and POST method with examples. (2)

(e) What do you mean by MVC? (1)

(f) Consider the following tables ACTIVITY and COACH. Write SQL commands for the statements (i) to (iv) and give outputs for SQL queries (v) to (viii). (6)

Table: ACTIVITY

ACode	ActivityName	ParticipantsNum	PrizeMoney	ScheduledDate
1001	Relay 100x4	16	10000	23-Jan-2004
1002	High jump	10	12000	12-Dec-2003
1003	Shot Put	12	8000	14-Feb-2004
1005	Long Jump	12	9000	01-Jan-2004
1008	Discuss Throw	10	15000	19-Mar-2004

Table: COACH

PCode	Name	ACode
1	Ahmad Hussain	1001
2	Ravinder	1008
3	Janila	1001
4	Naaz	1003

(i) To display the name of all activities with their Acodes in descending order.

(ii) To display sum of PrizeMoney for each of the Number of participants groupings (as shown in column ParticipantsNum 10,12,16)

(iii) To display the coach's name and ACodes in ascending order of ACode from the table COACH

(iv) To display the content of the GAMES table whose ScheduledDate is earlier than 01/01/2004 in ascending order of ParticipantNum

(v) SELECT COUNT(DISTINCT ParticipantsNum) FROM ACTIVITY;

(vi) SELECT MAX(ScheduledDate),MIN(ScheduledDate) FROM ACTIVITY;

(vii) SELECT SUM(PrizeMoney) FROM ACTIVITY;

(viii) SELECT DISTINCT ParticipantNum FROM COACH;

SECTION–C

6. Answer the following questions:

(a) What do you understand by the following? (2)

(i) cyber bullying

(ii) cookies

(b) Differentiate between digital signature and digital certificate. Explain with examples. (2)

(c) What do you mean by web browser and web server? Explain with examples. (2)

(d) What do you mean by Wi-Fi and IR? Explain with examples. (2)

(e) What do you mean by firewall? Explain its use in modern scenario. (2)

7. Answer the following questions:

(a) Differentiate between Bus Topology and Star Topology of Networks. What are the advantages and disadvantages of Star Topology over Bus Topology? (2)

(b) Classify each of the following Web Scripting as Client Side Scripting and Server Side Scripting: (2)

(i) Java Scripting (ii) ASP

(iii) VB Scripting (iv) JSP

(c) Write the expanded forms of the following abbreviated terms used in networking and communications: (2)

(i) SMTP (ii) VoIP

(iii) GSM (iv) WLL

(d) Differentiate between guided and unguided media with examples of each. (2)

(e) Differentiate between bridge and router with examples of each. (2)

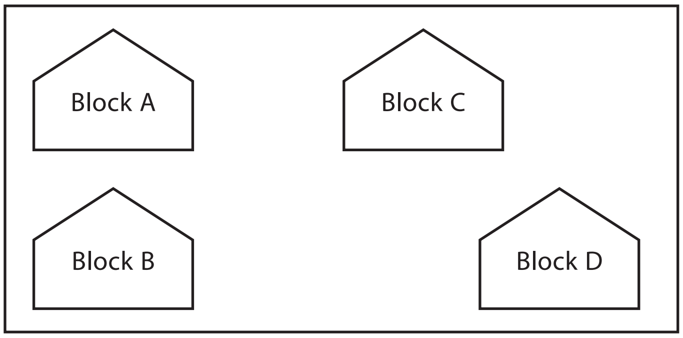
(f) Knowledge Supplement Organization has set up its new centre at Mangalore for its office and web-based activities. It has 4 blocks of buildings as shown in the diagram below:

Centre to centre distances between various blocks:

Block A to Block B	50 m
Block B to Block C	150 m
Block C to Block D	25 m
Block A to Block D	170 m
Block B to Block D	125 m
Block A to Block C	90 m

Number of computers:

Block A	25
Block B	50
Block C	125
Block D	10



(i) Suggest the network implemented. (1)

(ii) Suggest the most suitable place (*i.e.*, block) to house the server of this organization with a suitable reason. (1)

(iii) Suggest the placement of the following devices with justification: (1)

(a) Repeater

(b) Hub/Switch

(iv) The organization is planning to link its front office situated in the city in a hilly region where cable connection is not feasible. Suggest an economic way to connect it with reasonably high speed. (1)

(v) Which topology do we use for establishing this architecture? (1)

CLASS – XII

MODEL TEST PAPER - III

SUBJECT: COMPUTER SCIENCE (083)

M.TIME :03HRS **M.MARKS:70**

1. (a) Which of the following can be used as valid variable identifier(s) in Python? 2

```
My.file, _Count, For, 2digits, 4thSu, Total, Number#,
Name1
```

(b) Name the Python Library modules which need to be imported to invoke the following functions: 1

(i) `floor()` (ii) `find()`

c) Rewrite the following python code after removing all syntax error(s). Underline the corrections done. 2

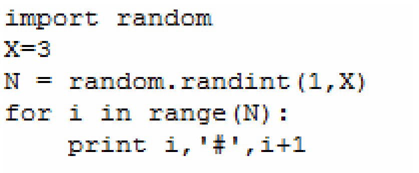
```
def main():
     r = input('enter any radius: ')
     a = pi * math.pow(r,2)
     print("Area = " + a)
```

(d) Find and write the output of the following Python code: 2

```
     L1 = [500,800,600,200,900]
     START = 1
     SUM = 0
     for C in range(START,4):
          SUM = SUM + L1[C]
          print(C,":",SUM)
          SUM = SUM + L1[0]*10
          print(SUM)
```

OR

(d) Observe the following program and answer the questions that follow:



```
import random
X=3
N = random.randint(1,X)
for i in range(N):
    print i,'#',i+1
```

What is the minimum and maximum number of times the loopwill execute?
b. Find out, which line of output(s) out of (i) to (iv) will not be expected from the program?

i. 0#1
ii. 1#2
iii. 2#3
iv. 3#4

(e) Find and write the output of the following Python code: 2

```
     def rec(x):
          if(x>1):
               return x*rec(x-2)
          print(x)
          return x
      print(rec(5))
```

(f) which module needs to be imported for showing data in chart form? 2

OR

(f) Which string method is used to implement the following:

a) To count the number of characters in the string.

b) To change the first character of the string in capital letter.

c) To check whether given character is letter or a number.

d) To change lower case to upper case letter.

e) Change one character into another character

2. (a) Write the definition of a function Reverse(X) in Python, to display the elements in reverse order such that each displayed element is the Four time of the original element (element * 4) of the List X in the following manner: 2

Example:

If List X contains 7 integers is as follows:

X[0]	X[1]	X[2]	X[3]	X[4]	X[5]	X[6]
4	6	9	12	5	8	7

After executing the function, the array content should be displayed as follows:

28 32 20 48 36 24 16

(b) Consider the following sorted list: 3

[22, 54, 66, 90, 155, 178]

Write the passes of binary search for searching the list for value 25 till the 3rd iteration.

(c) Write AddCustomer(Customer) and DeleteCustomer(Customer) methods in Python to add a new Customer and delete a Customer from a List of CustomerNames, considering them to act as push and pop operations of the stack data structure. 4

(d) Write definition of a Method COUNTNOW(STATES) to find and display names of those STATES, in which there are less than or equal to 8 characters. 2

For example:

If the list STATES contains

["GOA","NEW DELHI","RAJASTHAN","TAMILNADU","GUJARAT"]

The following should get displayed

GOA

GUJARAT

(e) Evaluate the following postfix expression using a stack. Show the contents of stack after execution of each operation:

20,80,50,-,*,30,8,*,+ 2

3. (a) Neena intends to position the file pointer to the beginning of a text file. Write Python statement for the same assuming F is the File object. 1

(b) Write a function **countthe()** in Python to read the text file “DATA.TXT” and count the number of times “the” occurs in the file. 2
For example if the file “DATA.TXT” contains:
“This is my website. I have displayed my preferences in the CHOICE section. The website looks good.”
The **countthe**( ) function should display the output as:
“the occurs 2 times”.
OR
(b) .A text file “Quotes.Txt” has the following data written in it:

Living a life you can be proud of

Doing your best

Spending your time with people and activities that are important to you

Standing up for things that are right even when it’s hard

Becoming the best version of you

Write a user defined function to display the total number of words present in the file.

(c) Write a user-defined function named Count() that will read the contents of text file named “Report.txt” and count the number of lines which starts with either “I“ or “M“ and displays the count. 2
E.g. In the following paragraph, there are 3 lines starting with “I“ or “M“:

“India is the fastest growing economy. India is looking for more investments around the globe.
The whole world is looking at India as a great market.
Most of the Indians can foresee the heights that India is capable of reaching.”

OR

(c) Observe the following code and answer the questions that follow:

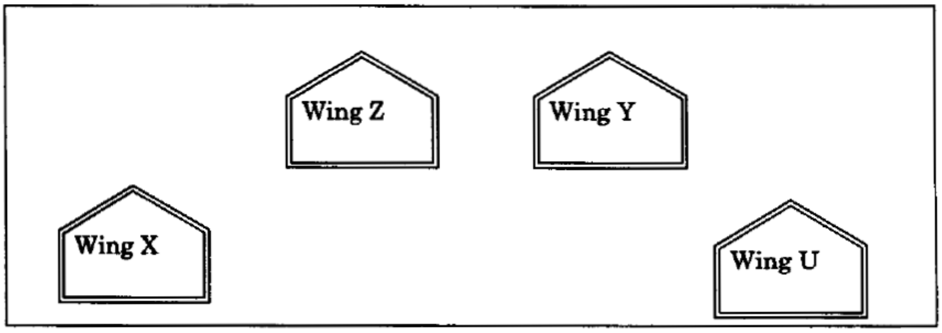
```
File = open("Mydata","a")
_____________________                    #Blank1
File.close()
```

i. What type (Text/Binary) of file is Mydata?
ii. Fill the Blank 1 with statement to write “ABC” in the file “Mydata”

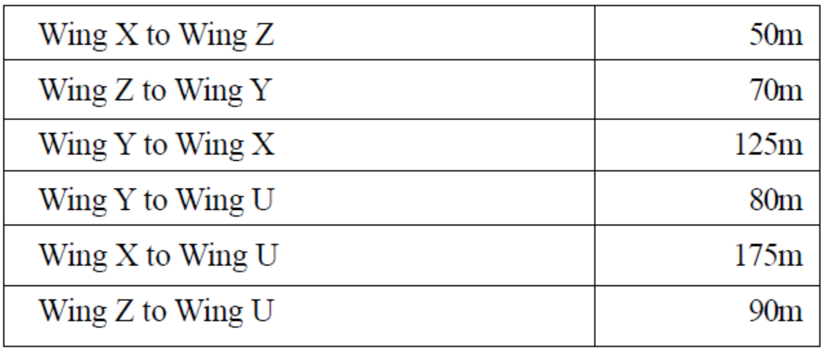
4. (a) A teacher provides **“http://www.publicschools.nic.in/staff.aspx”** to his/her students. Identify the URL & domain name. 1

(b) Expand the following terminologies: 2
(i) FTP (ii) SCP (iii) NFC (iv) IMAP

(c) The ABC Pvt Ltd has set up its new Branch at Jammu for its office and web based activities. It has 4 Wings of buildings as shown in the diagram: 4

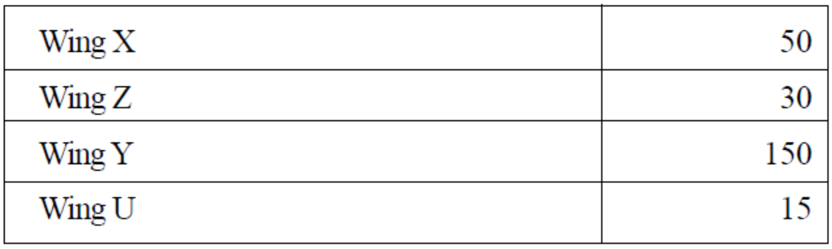


Center to center distances between various blocks



Wing X to Wing Z	50m
Wing Z to Wing Y	70m
Wing Y to Wing X	125m
Wing Y to Wing U	80m
Wing X to Wing U	175m
Wing Z to Wing U	90m

Number of Computers



Wing X	50
Wing Z	30
Wing Y	150
Wing U	15

(i) Suggest a most suitable connection between the Wings, and topology.

(ii) Suggest the most suitable place (i.e. Wing) to house the server of this organization with a suitable reason, with justification.

(iii) Suggest the placement of the following devices with justification:

(1) Repeater (2) Hub/Switch

(iv) The organization is planning to link its head office situated in Delhi with the offices at Jammu. Suggest an economic way to connect it; the company is ready to compromise on the speed of connectivity. Justify your answer.

(d) How MAC address is different from IP address? Why IP address is needed? 2

(e) What is IoT? 1

(f) What is cloud? How a private cloud is different from public cloud? 2

(g) What is the use of traceroute and ping network tools? 2

(h) What is SIM? 1

5. (a) What is MVT in respect of Django Framework? 1

(b) Which module needs to be imported from making MySQL connectivity with python? 1

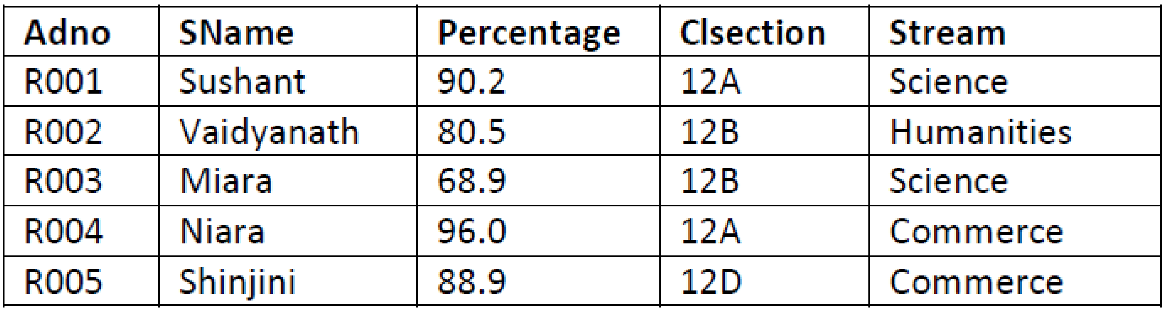
(c) What is the difference between flat file and csv file? 1

(d) What are aggregate functions in SQL? Give examples? 2

(e) How is having clause different from where clause? Explain with help of an example. 2

(f) Consider the following table names EXAM with details of marks. Rite command of MySQl for (i) to (iv) and output for (v) to (viii). 8

Table EXAM



Adno	SName	Percentage	Clsection	Stream
R001	Sushant	90.2	12A	Science
R002	Vaidyanath	80.5	12B	Humanities
R003	Miara	68.9	12B	Science
R004	Niara	96.0	12A	Commerce
R005	Shinjini	88.9	12D	Commerce

(i) To display all information of the students of humanities in descending order of percentage.
(ii) To display Adno, Name, Percentage and Stream of those students whose name starts with S and ends with t.
(iii) To display SName, Percentage, Clsection of students who have highest percentage in each stream.
(iv) To display details of students who have Percentage in range of 80 and 90(both inclusive) in decreasing order of Percentage.
(v) SELECT COUNT(*) FROM EXAM;
(vi) SELECT Sname, Percentage FROM EXAM WHERE Name LIKE “N%”;
(vii) SELECT ROUND(Percentage,0) FROM EXAM WHERE Adno=”R005”;
(viii) SELECT DISTINCT Clsection FROM EXAM.

6. (a) What is plagiarism? 1

(b) What is Creative Commons and GPL? 2

(c) What is open source and open data? 2

(d) What is E-waste management? 1

(e) What is cyber forensics? 1

(f) What is Identity theft? 1

(g) What is Gender and disability issues while teaching and using computers? 2

```
#Brain Teasers- Project for conducting quiz

'''
Before creating and executing the code, create the required database and the associated table for
MySQL as shown below:
..................................................................
create database brain_teaser;
use brain_teaser;
create table question(qid int(4) primary key, question varchar(500) not null,op1 varchar(100) not
null,op2 varchar(100) not null,op3 varchar(100),op4 varchar(100),ans varchar(100) not null);
'''

import sys
import mysql.connector
import random
mydb=mysql.connector.connect(host="localhost", user="root", password="",
                                         database="brain_teaser")
mycursor=mydb.cursor()

# Function definition for Home screen
def Home():
    f=1
    while f!=3:
       print("Welcome to Quiz")
       print("********************")
       print("1. Enter Questions")
       print("2. Take Quiz")
       print("3. Exit")
       f=int(input("Enter your choice: "))
       if f == 1:
          Question()
       elif f == 2:
          Quiz()
       elif f == 3:
          print("Exiting the Quiz")
          mycursor.close()
          mydb.close()
          sys.exit()
       else:
          Home()
```

```
def Question():
    ch='Y'
    while ch=='Y' or ch=='y':
      print("Welcome to Question Portal")
      print("***********************")
      q=input("Enter the question :")
      op1=input("Enter the option 1 :")
      op2=input("Enter the option 2 :")
      op3=input("Enter the option 3 :")
      op4=input("Enter the option 4 :")
      ans=0
      while ans == 0:
             op=int(input("Which option is correct answer (1,2,3,4) :"))
            if op==1:
              ans=op1
            elif op==2:
              ans=op2
             elif op==3:
              ans=op3
             elif op==4:
              ans=op4
            else:
              print("Please choose the correct option as answer")
       mycursor.execute("Select * from question")
       data=mycursor.fetchall()
       qid=(mycursor.rowcount)+1
       mycursor.execute("Insert into question values
                         (%s,%s,%s,%s,%s,%s,%s)",(qid,q,op1,op2,op3,op4,ans))
     mydb.commit()
     ch=input("Question added successfully. Do you want to add more (Y/N)")
     Home()


# Function definition for Quiz
def Quiz():
     print("Welcome to Quiz portal")
     print("***********************")
     mycursor.execute("Select * from question")
     data=mycursor.fetchall()
```

```
name=input("Enter your name :")
rc=mycursor.rowcount
noq=int(input("Enter the number of questions to attempt (max %s):"%rc))
l=[]
while len(l) != noq:
      x=random.randint(1,rc)
      if l.count(x)>0:
           l.remove(x)
     else:
          l.append(x)
          print("Quiz has started")
          c=1
          score = 0
          for i in range(0,len(l)):
                   mycursor.execute("Select * from question where qid=%s",(l[i],))
                   ques=mycursor.fetchone()
                   print("------------------------------------------------------------------------------------------------------------------------")
                   print("Q.",c,":
                          ",ques[1],"\nA.",ques[2],"\t\tB.",ques[3],"\nC.",ques[4],"\t\tD.",ques[5])
                  print("------------------------------------------------------------------------------------------------------------------------")
                  c += 1
          ans=None
          while ans==None:
                    choice=input("Answer (A,B,C,D) :")
                     if choice=='A' or choice=='a':
                              ans=ques[2]
                    elif choice=='B' or choice=='b':
                              ans=ques[3]
                    elif choice=='C' or choice=='c':
```

```
                ans=ques[4]
        elif choice=='D' or choice=='d':
                ans=ques[5]
        else:
            print("Kindly select A,B,C,D as option only")
            if ans==ques[6]:
                print("Correct")
                score=score+1
            else:
                print("Incorrect.. Correct answer is :",ques[6])
    print("Quiz has ended !! Your final score is :",score)
    input("Press any key to continue")

Home()
```

INVENTORY MANAGEMENT

About Project

Inventory management is the practice of ordering, storing, tracking, and controlling inventory. Inventory management applies to every item a business uses to produce its products or services – from raw materials to finished goods. In other words, inventory management covers every aspect of a business's inventory.

The project contains following modules:-

1. Product Management: This module is used to add, update and delete the products.

2. Purchase Management: This module is used to manage the purchase system.

3. Sales Management: This module is used to manage the sale of the products.

4. User Management: This module is used to add/delete the user/staff.

5. Database setup: This module is used to setup the database in the system for the first time.

SOFTWARE SPECIFICATION:-

Operating System	:	Windows 8
Platform	:	Python IDLE 3.7
Database	:	MySQL
Languages	:	Python

Note: For Python-MySQL connectivity, following data have been used:-

Host- localhost, user- root, password- ' ', database- stock

CODING

```
# INVENTORY MANAGEMENT
import os
import mysql.connector
import datetime
now = datetime.datetime.now()


def product_mgmt():
    while True:
        print("\t\t\t 1. Add New Product")
        print("\t\t\t 2. List Product")
        print("\t\t\t 3. Update Product")
        print("\t\t\t 4. Delete Product")
        print("\t\t\t 5. Back (Main Menu)")
        p = int(input("\t\t Enter Your Choice :"))
        if p == 1:
            add_product()
        if p == 2:
            search_product()
        if p == 3:
            update_product()
        if p == 4:
            delete_product()
        if p == 5:
            break


def purchase_mgmt():
    while True:
        print("\t\t\t 1. Add Order")
        print("\t\t\t 2. List Order")
        print("\t\t\t 3. Back (Main Menu)")
        o = int(input("\t\t Enter Your Choice :"))
        if o == 1:
            add_order()
        if o == 2:
            list_order()
        if o == 3:
            break
```

```
def sales_mgmt( ):
    while True:
        print("\t\t\t 1. Sale Items")
        print("\t\t\t 2. List Sales")
        print("\t\t\t 3. Back (Main Menu)")
        s = int (input("\t\t Enter Your Choice :"))
        if s == 1:
            sale_product()
        if s == 2:
            list_sale()
        if s == 3:
            break


def user_mgmt():
    while True:
        print("\t\t\t 1. Add user")
        print("\t\t\t 2. List user")
        print("\t\t\t 3. Back (Main Menu)")
        u = int(input("\t\t Enter Your Choice :"))
        if u == 1:
            add_user()
        if u == 2:
            list_user()
        if u == 3:
            break


def create_database():
     mydb = mysql.connector.connect(host="localhost", user="root",
                                    password="", database="stock")
    mycursor = mydb.cursor()
    print(" Creating PRODUCT table")
    sql = "CREATE TABLE if not exists product(pcode int(4) PRIMARY KEY,
                                              pname char(30) NOT NULL,
                                              price float(8,2),
                                              pqty int(4),
                                              pcat char(30));"
    mycursor.execute(sql)
    print("Creating ORDER table")
    sql = "CREATE TABLE if not exists orders(orderid int(4)PRIMARY KEY,
                                             orderdate DATE,
                                             pcode char(30) NOT NULL ,
                                             pprice float(8,2),
                                             pqty int(4),
                                             supplier char(50),
                                             pcat char(30));"
    mycursor.execute(sql)
    print("ORDER table created")
    print("Creating SALES table")
    sql = "CREATE TABLE if not exists sales(salesid int(4) PRIMARY KEY,
                                           salesdate DATE,
                                           pcode char(30) references product(pcode),
                                           pprice float(8,2),
                                           pqty int(4),
                                           Total double(8,2));"
    mycursor.execute(sql)
    print("SALES table created")
    sql = "CREATE TABLE if not exists user (uid char(6) PRIMARY KEY,
                                           uname char(30) NOT NULL,
                                           upwd char(30));"
    mycursor.execute(sql)
    print("USER table created")


def list_database():
    mydb = mysql.connector.connect(host="localhost", user="root",
                                   password="",database="stock")
    mycursor = mydb.cursor()
    sql = "show tables;"
    mycursor.execute(sql)
    for i in mycursor:
        print(i)


def add_order():
    mydb = mysql.connector.connect(host="localhost", user="root",
                                   password="", database="stock")
    mycursor = mydb.cursor()
    now = datetime.datetime.now()
    sql = "INSERT INTO orders (orderid, orderdate, pcode,
                              pprice, pqty, supplier, pcat) values
                              (%s,%s,%s,%s,%s,%s,%s)"
    code = int(input("Enter product code :"))
    oid = now.year+now.month+now.day+now.hour+now.minute+now.second
    qty = int(input("Enter product quantity : "))
    price = float(input("Enter Product unit price: "))
    cat = input("Enter product category: ")
    supplier = input("Enter Supplier details: ")
    val = (oid, now, code, price, qty, supplier, cat)
    mycursor.execute(sql, val)
    mydb.commit()


def list_order():
    mydb = mysql.connector.connect(host="localhost", user="root",
                                   password="", database="stock")
    mycursor = mydb.cursor()
    sql = "SELECT * from orders"
    mycursor.execute(sql)
    print("\t\t\t\t\t\t\t ORDER DETAILS")
    print("-"*85)
    print("orderid   date   product code   price    quantity   supplier   category")
    print("-" * 85)
    for i in mycursor:
        print(i[0], "\t", i[1], "\t", i[2], "\t", i[3], "\t", i[4], "\t", i[5], "\t", i[6])
    print("-" * 85)


def db_mgmt( ):
    while True:
        print("\t\t\t 1. Database creation")
        print("\t\t\t 2. List Database")
        print("\t\t\t 3. Back (Main Menu)")
        p = int(input("\t\t Enter Your Choice :"))
        if p == 1:
            create_database()
        if p == 2:
            list_database()
        if p == 3:
```

```
                    break
def add_product():
        mydb = mysql.connector.connect(host="localhost", user="root",
                                        password="",database="stock")
        mycursor = mydb.cursor()
        sql = "INSERT INTO product(pcode,pname,pprice,pqty,pcat) values
             (%s,%s,%s,%s,%s)"
        code = int(input("\t\t Enter product code :"))
        search = "SELECT count(*) FROM product WHERE pcode=%s;"
        val = (code,)
        mycursor.execute(search,val)
        for x in mycursor:
              cnt = x[0]
        if cnt == 0:
              name = input("\t\t Enter product name :")
              qty = int(input("\t\t Enter product quantity :"))
              price = float(input("\t\t Enter product unit price :"))
              cat = input("\t\t Enter Product category :")
              val = (code,name,price,qty,cat)
              mycursor.execute(sql,val)
              mydb.commit()
        else:
              print("\t\t Product already exist")


def update_product():
        mydb = mysql.connector.connect(host="localhost", user="root",
                                        password="", database="stock")
        mycursor = mydb.cursor()
        code = int(input("Enter the product code :"))
        qty = int(input("Enter the quantity :"))
        sql = "UPDATE product SET pqty=pqty+%s WHERE pcode=%s;"
        val = (qty,code)
        mycursor.execute(sql,val)
        mydb.commit()
        print("\t\t Product details updated")
```

```
def delete_product():
        mydb = mysql.connector.connect(host="localhost", user="root",
                                                   password="", database="stock")
        mycursor=mydb.cursor()
        code = int(input("Enter the product code :"))
        sql = "DELETE FROM product WHERE pcode = %s;"
        val = (code,)
        mycursor.execute(sql, val)
        mydb.commit()
        print(mycursor.rowcount,"record(s) deleted");


def search_product():
        while True:
                print("\t\t\t 1. List all product")
                print("\t\t\t 2. List product code wise")
                print("\t\t\t 3. List product category wise")
                print("\t\t\t 4. Back (Main Menu)")
                s = int(input("\t\t Enter Your Choice :"))
                if s == 1:
                        list_product()
                if s == 2:
                         code=int(input(" Enter product code :"))
                         list_prcode(code)
                if s == 3:
                         cat=input("Enter category :")
                         list_prcat(cat)
                if s == 4:
                        break


def list_product():
        mydb = mysql.connector.connect(host="localhost", user="root",
                                                   password="",database="stock")
        mycursor = mydb.cursor()
        sql = "SELECT * from product"
        mycursor.execute(sql)
        print("\t\t\t\t PRODUCT DETAILS")
        print("\t\t", "-" * 47)
        print("\t\t code    name    price   quantity      category")
        print("\t\t", "-" * 47)
        for i in mycursor:
                print("\t\t", i[0], "\t", i[1], "\t", i[2], "\t", i[3], "\t\t", i[4])
        print("\t\t", "-" * 47)


def list_prcode(code):
        mydb = mysql.connector.connect(host="localhost", user="root",
                                                        password="", database="stock")
        mycursor = mydb.cursor()
        sql = "SELECT * from product WHERE pcode=%s"
        val = (code,)
        mycursor.execute(sql, val)
        print("\t\t\t\t PRODUCT DETAILS")
        print("\t\t", "-" * 47)
        print("\t\t code    name    price   quantity     category")
        print("\t\t", "-" * 47)
        for i in mycursor:
                print("\t\t", i[0], "\t", i[1], "\t", i[2], "\t", i[3], "\t\t", i[4])
        print("\t\t", "-" * 47)


def sale_product():
        mydb = mysql.connector.connect(host="localhost", user="root",
                                                        password="", database="stock")
        mycursor = mydb.cursor()
        pcode = input("Enter product code: ")
        sql = "SELECT count(*) from product WHERE pcode=%s;"
        val = (pcode,)
        mycursor.execute(sql,val)
        for x in mycursor:
                cnt = x[0]
        if cnt != 0 :
                sql = "SELECT * from product WHERE pcode=%s;"
                val = (pcode,)
                mycursor.execute(sql, val)
                for x in mycursor:
                        print(x)
                        price = int(x[2])
                        pqty = int(x[3])
                    qty = int(input("Enter no of quantity :"))
                    if qty <= pqty:
                        total = qty * price
                        print("Collect Rs. ", total)
                        sql = "INSERT into sales values(%s,%s,%s,%s,%s,%s)"
                        val = (int(cnt)+1,datetime.datetime.now(),pcode,price,qty,total)
                        mycursor.execute(sql,val)
                        sql = "UPDATE product SET pqty=pqty-%s WHERE pcode=%s"
                        val = (qty, pcode)
                        mycursor.execute(sql, val)
                        mydb.commit()
                else:
                        print("Quantity not available")
        else:
                print("Product is not available")

def list_sale():
        mydb = mysql.connector.connect(host="localhost", user="root",
                                       password="", database="stock")
        mycursor = mydb.cursor()
        sql = "SELECT * FROM sales"
        mycursor.execute(sql)
        print("\t\t\t\t SALES DETAILS")
        print("-" * 80)
        print("Sales ID    Date    Product Code    Price       Quantity       Total")
        print("-" * 80)
        for x in mycursor:
                print(x[0], "\t", x[1], "\t", x[2], "\t", x[3], "\t\t", x[4], "\t\t", x[5])
        print("-" * 80)


def list_prcat(cat):
        mydb = mysql.connector.connect(host="localhost", user="root",
                                       password="", database="stock")
        mycursor = mydb.cursor()
        print(cat)
        sql="SELECT * from product WHERE pcat =%s"
        val = (cat,)
        mycursor.execute(sql, val)
        clrscr()
        print("\t\t\t\t PRODUCT DETAILS")
        print("\t\t", "-" * 47)
        print("\t\t code    name    price   quantity    category")
        print("\t\t", "-" * 47)
        for i in mycursor:
                print("\t\t", i[0], "\t", i[1], "\t", i[2], "\t", i[3], "\t\t", i[4])
        print("\t\t", "-" * 47)


def add_user():
        mydb = mysql.connector.connect(host="localhost", user="root",
                                       password="", database="stock")
        mycursor = mydb.cursor()
        uid = input("Enter emaid id :")
        name = input("Enter Name :")
        password = input("Enter Password :")
        sql = "INSERT INTO user values (%s,%s,%s);"
        val = (uid, name, password)
        mycursor.execute(sql, val)
        mydb.commit()
        print(mycursor.rowcount, "user created")


def list_user():
        mydb = mysql.connector.connect(host="localhost", user="root",
                                       password="", database="stock")
        mycursor = mydb.cursor()
        sql = "SELECT uid, uname from user"
        mycursor.execute(sql)
        clrscr()
        print("\t\t\t\t USER DETAILS")
        print("\t\t", "-" * 27)
        print("\t\t UID       name    ")
        print("\t\t", "-" * 27)
        for i in mycursor:
                print("\t\t", i[0], "\t", i[1])
        print("\t\t", "-" * 27)
```

```
def clrscr():
    print("\n"*5)

while True:
    clrscr()
    print("\t\t\t STOCK MANAGEMENT")
    print("\t\t\t ****************\n")
    print("\t\t 1. PRODUCT MANAGEMENT")
    print("\t\t 2. PURCHASE MANAGEMENT")
    print("\t\t 3. SALES MANAGEMENT")
    print("\t\t 4. USER MANAGEMENT")
    print("\t\t 5. DATABASE SETUP")
    print("\t\t 6. EXIT\n")
    n = int(input("Enter your choice :"))
    if n == 1:
        product_mgmt()
    if n == 2:
        os.system('cls')
        purchase_mgmt()
    if n == 3:
        sales_mgmt()
    if n == 4:
        user_mgmt()
    if n == 5:
        db_mgmt()
    if n == 6:
        break
```

PROJECT WORK

Project on Library Management System

ABC School library offers reading services and membership to the students on roll. The students are issued a library ticket for getting the books issued and library fee is included in the fees. There is also a provision for teachers to get the books issued as per their requirement. Library Members can avail of the following facilities:

1. Providing books/journals for reading/reference.
2. Issue of books to students so that they can take them home and return them on or before the due date.

When a new book is brought to the library, its record is created, which can be modified or deleted if the book is no longer available in the library. If a student requests for a book, it is first searched for its availability through a unique key field "Bno", *i.e.*, Book code. If it is available (*i.e.*, not issued to any other member) then it is issued and details are entered in the Issue module.

The books are issued for 10 days after which they are required to be returned on or before the due date. The details of the same are entered in the Issue module. A new member can be added, his record can be modified, and an existing member can be removed from the library membership, for which necessary changes are made in the Member module.

The project (Front-end) has been developed using Python version 3.7.1. The RDBMS (Relational Database Management System) used is MySQL. Thus, the entire project is based on Python-MySQL connectivity.

The important modules used in our project are listed below:

☞ MENULIB

☞ LIBRARY MANAGEMENT

☞ BOOK

☞ ISSUE

☞ MEMBER

The three MySQL tables used for storing the book details, member details and issue details in this project are:

➢ bookRecord

➢ issue

➢ member

Name of the Database: Library

About Python-MySQL Connectivity

While designing real-life applications, certain situations arise pertaining to storing some important and necessary information by the user. Usually, the data inputted by the user along with the generated output are displayed but are not stored, since all the program execution takes place inside the RAM, which is a temporary memory, and as soon as we close the form, its contents (form input and generated output) get erased. They can't be retrieved since they are not getting saved on a hard disk (or any secondary storage device). Thus, when the application is executed for the second time, it requires a new set of inputs from the user. This limitation can be overcome by sending the output generated and saving the input fetched from the user in a database created at the back-end of the application. The input is fetched from the user using Python Interface. This is termed as the **Front End Interface** of the application.

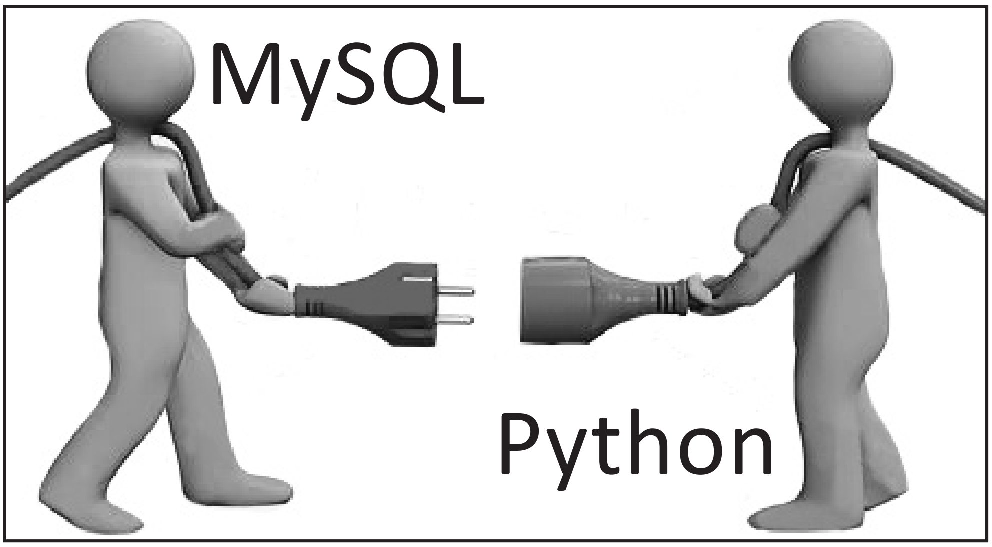


Fig.: Front-end (Python) and Back-end (MySQL)

Database (The Back-end)

While working with an application, it is required to save data permanently on some secondary storage device, which is usually the hard disk, so that the data can be retrieved for future reference, modification, deletion, etc. An application usually stores a lot of data in the form of a database which is not directly accessible to the user. This database is used by the application to give suitable responses to the user. This database is called **Back-end Database**.

In Class XI, we have learnt how to create databases, tables and to perform query processing on these tables using SQL commands like CREATE, UPDATE, ALTER, INSERT, DELETE, SELECT and so on in various forms according to their specific syntax structures. We shall be implementing all these DDL and DML commands of SQL through Python Interface.

Why Python?

Python is a flexible, portable, easy to learn and modifiable language. So, we are integrating MySQL with Python Interface for executing any database applications. The various reasons to use Python for programming database applications are:

- Programming in Python is arguably more efficient and faster compared to other languages.
- Python is famous for its portability.
- It is platform-independent.
- Python supports SQL cursors.

- In many programming languages, the application developer needs to take care of the open and closed connections of the database, to avoid further exceptions and errors. In Python, these connections are taken care of.
- Python supports relational database systems.
- Python database APIs are compatible with various databases, so it is very easy to migrate and port database application interfaces.

Coding

```
#Project on Library Management System
#-----------------------------------------------------------------------
#MODULE : LIBRARY MANAGEMENT
import MenuLib
import Book
import issue

while True:
    Book.clrscreen()
    print("\t\t\t Library Management\n")
    print("============================================================")
    print("1. Book Management")
    print("2. Members Management")
    print("3. Issue/Return Book")
    print("4. Exit")
    print("============================================================")
    choice=int(input("Enter Choice between 1 to 4-------> : "))
    if choice==1:
        MenuLib.MenuBook()
    elif choice==2:
        MenuLib.MenuMember()
    elif choice==3:
        MenuLib.MenuIssueReturn()
    elif choice==4:
        break
    else:
        print("Wrong Choice......Enter Your Choice again")
        x=input("Enter any key to continue")
```

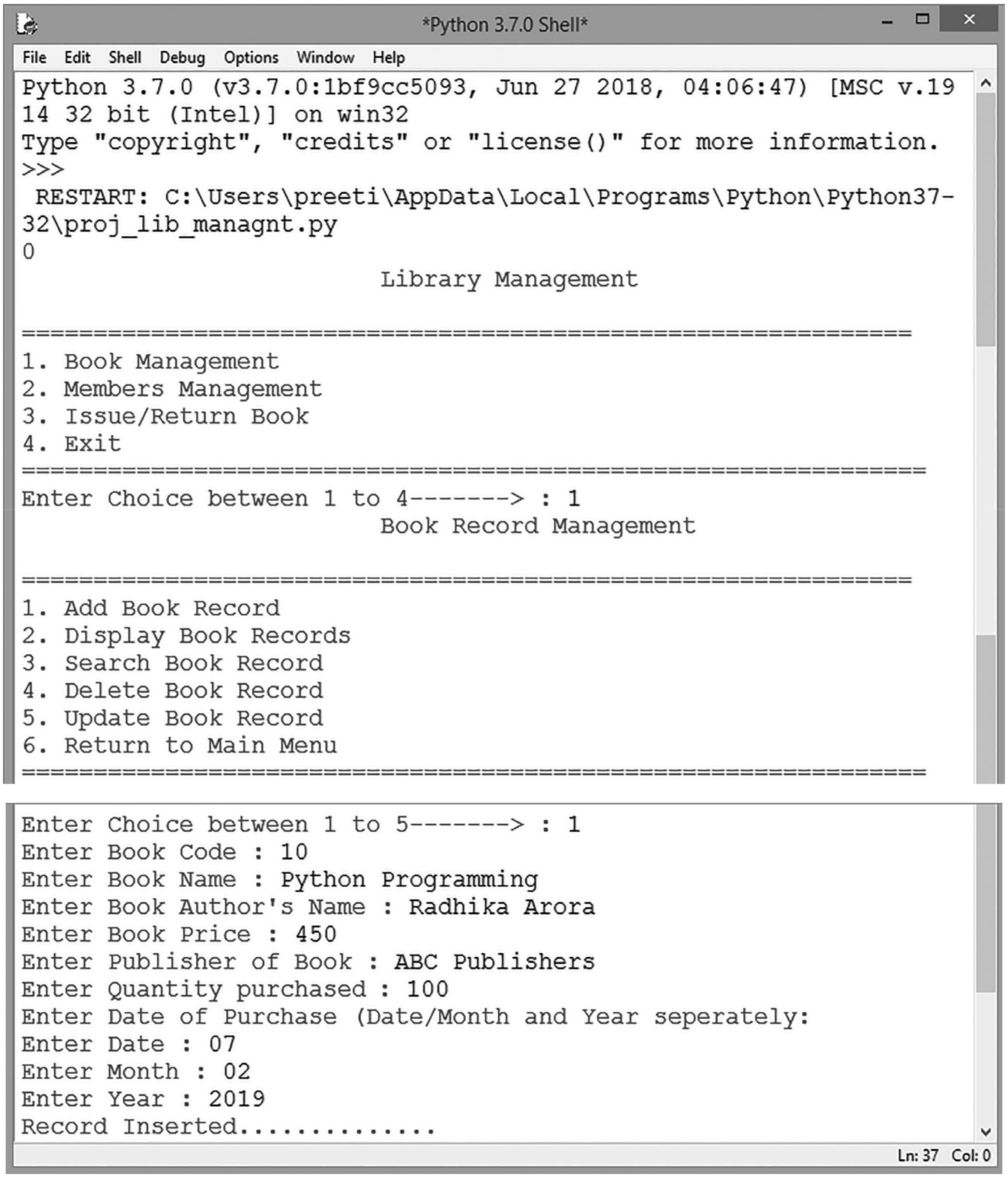
```python
#PYTHON MODULE: MENULIB
import Book
import Member
import issue

def MenuBook():
    while True:
        Book.clrscreen()
        print("\t\t\t Book Record Management\n")
        print("======================================================")
        print("1. Add Book Record")
        print("2. Display Book Records")
        print("3. Search Book Record")
        print("4. Delete Book Record")
        print("5. Update Book Record")
        print("6. Return to Main Menu")
        print("======================================================")
        choice=int(input("Enter Choice between 1 to 5-------> : "))
        if choice==1:
            Book.insertData()
        elif choice==2:
            Book.display()
        elif choice==3:
            Book.SearchBookRec()
        elif choice==4:
            Book.deleteBook()
        elif choice==5:
            print("No such Function")
        elif choice==6:
            return
        else:
            print("Wrong Choice......Enter Your Choice again")
            x=input("Enter any key to continue")

def MenuMember():
    while True:
        Book.clrscreen()
        print("\t\t\t Member Record Management\n")
        print("=========================================================")
        print("1. Add Member Record")
        print("2. Display Member Records")
        print("3. Search Member Record")
        print("4. Delete Member Record")
        print("5. Update Book Record")
        print("6. Return to Main Menu")
        print("=========================================================")
        choice=int(input("Enter Choice between 1 to 5-------> : "))
        if choice == 1:
            Member.insertData()
        elif choice == 2:
            Member.display()
        elif choice == 3:
            Member.SearchMember()
        elif choice == 4:
            Member.deleteMember()
        elif choice == 5:
            print("No such Function")
        elif choice==6:
            return
        else:
            print("Wrong Choice......Enter Your Choice again")
            x=input("Enter any key to continue")
#--------------------------------------------------------------------

def MenuIssueReturn():
    while True:
        Book.clrscreen()
        print("\t\t\t Member Record Management\n")
        print("======================================================")
        print("1. Issue Book")
        print("2. Display Issued Book Records")
        print("3. Return Issued Book")
        print("4. Return to Main Menu")
        print("======================================================")
        choice=int(input("Enter Choice between 1 to 4------> : "))
        if choice==1:
            issue.issueBookData()
        elif choice==2:
            issue.ShowIssuedBooks()
        elif choice==3:
            issue.returnBook()
        elif choice==4:
            return
        else:
            print("Wrong Choice......Enter Your Choice again")
            x=input("Enter any key to continue")

# PYTHON MODULE : BOOK
from mysql.connector import errorcode
from datetime import date, datetime, timedelta
from mysql.connector import(connection)
import os
import platform
def clrscreen():
    if platform.system()=="Windows":
        print(os.system("cls"))


def display():
    try:
        os.system('cls')
        cnx = connection.MySQLConnection(user='root', password='',
                                         host='localhost',
                                         database='Library')
        Cursor = cnx.cursor()
        query = ("SELECT * FROM BookRecord")
        Cursor.execute(query)
        for (Bno,Bname,Author,price,publ,qty,d_o_purchase) in Cursor:
        print("===================================================")
        print("Book Code : ",Bno)
        print("Book Name : ",Bname)
        print("Author of Book : ",Author)
        print("Price of Book : ",price)
        print("Publisher : ",publ)
        print("Total Quantity in Hand : ",qty)
        print("Purchased On : ",d_o_purchase)
        print("===================================================")
        Cursor.close()
        cnx.close()
        print("You have done it!!!!!!")
    except mysql.connector.Error as err:
        if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
            print("Something is wrong with your user name or password")
        elif err.errno == errorcode.ER_BAD_DB_ERROR:
            print("Database does not exist")
        else:
            print(err)
    else:
        cnx.close()

import mysql.connector


def insertData():
    try:
        cnx = connection.MySQLConnection(user='root',password='',
                                         host='localhost',
                                         database='Library')
        Cursor = cnx.cursor()
        bno=input("Enter Book Code : ")
        bname=input("Enter Book Name : ")
        Auth=input("Enter Book Author's Name : ")
        price=int(input("Enter Book Price : "))
        publ=input("Enter Publisher of Book : ")
        qty=int(input("Enter Quantity purchased : "))
        print("Enter Date of Purchase (Date/Month and Year separately): ")
        DD=int(input("Enter Date : "))
        MM=int(input("Enter Month : "))
        YY=int(input("Enter Year : "))
        Qry = ("INSERT INTO BookRecord VALUES (%s, %s, %s, %s, %s, %s, %s)")
        data = (bno,bname,Auth,price,publ,qty,date(YY,MM,DD))
        Cursor.execute(Qry,data)
    # Make sure data is committed to the database cnx.commit()
        Cursor.close()
        cnx.close()
        print("Record Inserted..............")
    except mysql.connector.Error as err:
        if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
            print("Something is wrong with your user name or password")
        elif err.errno == errorcode.ER_BAD_DB_ERROR:
            print("Database does not exist")
        else:
            print(err)
    cnx.close()


def deleteBook():
    try:
        cnx = connection.MySQLConnection(user='root',password='',
                                         host='localhost',
                                         database='Library')
        Cursor = cnx.cursor()
        bno=input("Enter Book Code of Book to be deleted from the Library : ")
        Qry =("""DELETE FROM BookRecord WHERE BNO = %s""")
        del_rec=(bno,)
        Cursor.execute(Qry,del_rec)

# Make sure data is committed to the database cnx.commit()
        Cursor.close()
        cnx.close()
        print(Cursor.rowcount,"Record(s) Deleted Successfully..........")
    except mysql.connector.Error as err:
        if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
            print("Something is wrong with your user name or password")
        elif err.errno == errorcode.ER_BAD_DB_ERROR:
            print("Database does not exist")
        else:
            print(err)
    cnx.close()


def SearchBookRec():
    try:
        cnx = connection.MySQLConnection(user='root',password='',
                                         host='localhost',
                                         database='Library')
        Cursor = cnx.cursor()
        bno=input("Enter Book No to be Searched from the Library : ")
        query = ("SELECT * FROM BookRecord where BNo = %s ")
        rec_srch=(bno,)
        Cursor.execute(query,rec_srch)
        Rec_count=0
        for (Bno,Bname,Author,price,publ,qty,d_o_purchase) in Cursor:
            Rec_count+=1
            print("==================================================")
            print("Book Code : ",Bno)
            print("Book Name : ",Bname)
            print("Author of Book : ",Author)
            print("Price of Book : ",price)
            print("Publisher : ",publ)
            print("Total Quantity in Hand : ",qty)
            print("Purchased On : ",d_o_purchase)
            print("==================================================")
            if Rec_count%2==0:
                input("Press any key to continue")
                clrscreen()
                print(Rec_count, "Record(s) found")
# Make sure data is committed to the database cnx.commit()
        Cursor.close()
        cnx.close()
    except mysql.connector.Error as err:
        if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
            print("Something is wrong with your user name or password")
        elif err.errno == errorcode.ER_BAD_DB_ERROR:
            print("Database does not exist")
        else:
            print(err)
    cnx.close()


def UpdateBook():
    try:
        cnx = connection.MySQLConnection(user='root',password='',
                                         host='localhost',
                                         database='Library')
        Cursor = cnx.cursor()
        bno=input("Enter Book Code of Book to be Updated from the Library : ")
        query = ("SELECT * FROM BookRecord where BNo = %s ")
        rec_srch=(bno,)
        print("Enter new data ")
        bname=input("Enter Book Name : ")
        Auth=input("Enter Book Author's Name : ")
        price=int(input("Enter Book Price : "))
        publ=input("Enter Publisher of Book : ")
        qty=int(input("Enter Quantity purchased : "))
        print("Enter Date of Purchase (Date/MOnth and Year seperately: ")
        DD=int(input("Enter Date : "))
        MM=int(input("Enter Month : "))
        YY=int(input("Enter Year : "))
        Qry = ("UPDATE BookRecord SET bname=%s,Author=%s,price=%s,
                publisher=%s,qty=%s,d_o_purchase=%s,WHERE BNO=%s")
        data = (bname,Auth,price,publ,qty,date(YY,MM,DD),bno)
        Cursor.execute(Qry,data)
    # Make sure data is committed to the database cnx.commit()
        Cursor.close()
        cnx.close()
        print(Cursor.rowcount,"Record(s) Updated Successfully........")
    except mysql.connector.Error as err:
        if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
            print("Something is wrong with your user name or password")
        elif err.errno == errorcode.ER_BAD_DB_ERROR:
            print("Database does not exist")
        else:
            print(err)
        cnx.close()
        UpdateBook()

#PYTHON MODULE : ISSUE
import mysql.connector
from mysql.connector import errorcode
from datetime import date
from mysql.connector import (connection)
import os

def clrscreen():
    print('\n' *5)



def ShowIssuedBooks():
    try:
        os.system('cls')
        cnx = connection.MySQLConnection(user='root', password='',
                                         host='localhost',
                                         database='Library')
        Cursor = cnx.cursor()
        query = ("SELECT B.bno,bname,M.mno,mname,d_o_issue,d_o_ret FROM
                bookRecord B,issue I,member M where B.bno=I.bno and
                I.mno=M.mno")
        Cursor.execute(query)
        for (Bno,Bname,Mno,Mname,doi,dor) in Cursor:
        print("=====================================================")
        print("Book Code : ",Bno)
        print("Book Name : ",Bname)
        print("Member Code : ",Mno)
        print("Member Name : ",Mname)
        print("Date of issue : ",doi)
        print("Date of return : ",dor)
        print("=====================================================")
        Cursor.close()
        cnx.close()
        print("You have done it!!!!!!")
    except mysql.connector.Error as err:
        if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
            print("Something is wrong with your user name or password")
        elif err.errno == errorcode.ER_BAD_DB_ERROR:
            print("Database does not exist")
        else:
            print(err)
    else:
        cnx.close()


def issueBook():
    try:
        cnx = connection.MySQLConnection(user='root',password='',
                                         host='localhost',
                                         database='Library')
        Cursor = cnx.cursor()
        bno=input("Enter Book Code to issue : ")
        mno=input("Enter Member Code : ")
        print("Enter Date of Issue (Date/Month and Year separately): ")
        DD=int(input("Enter Date : "))
        MM=int(input("Enter Month : "))
        YY=int(input("Enter Year : "))
        Qry = ("INSERT INTO issue (bno,mno,d_o_issue)VALUES(%s, %s, %s)")
        data = (bno,mno,date(YY,MM,DD))
        Cursor.execute(Qry,data)
        cnx.commit()
        Cursor.close()
        cnx.close()
        print("Record Inserted.............")
    except mysql.connector.Error as err:
        if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
            print("Something is wrong with your user name or password")
        elif err.errno == errorcode.ER_BAD_DB_ERROR:
            print("Database does not exist")
        else:
            print(err)
    cnx.close()


def returnBook():
    try:
        cnx = connection.MySQLConnection(user='root',password='',
                                         host='localhost',
                                         database='Library')
        Cursor = cnx.cursor()
        bno=input("Enter Book Code of Book to be returned to the Library : ")
        Mno=input("Enter Member Code of Member who is returning Book : ")
        retDate=date.today()
        Qry =("""Update Issue set d_o_ret= %s WHERE BNO = %s and Mno= %s """)
        rec=(retDate,bno,Mno)
        Cursor.execute(Qry,rec)
    # Make sure data is committed to the database cnx.commit()
        Cursor.close()
        cnx.close()
        print(Cursor.rowcount,"Record(s) Deleted Successfully.........")
    except mysql.connector.Error as err:
        if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
            print("Something is wrong with your user name or password")
        elif err.errno == errorcode.ER_BAD_DB_ERROR:
            print("Database does not exist")
        else:
            print(err)
        cnx.close()
```

```
#PYTHON MODULE MEMBER
from mysql.connector import errorcode
from datetime import date, datetime, timedelta
from mysql.connector import(connection)
import os

def clrscreen():
   print('\n' *5)


def display():
   try:
      os.system('cls')
      cnx = connection.MySQLConnection(user='root', password='',
                                       host='localhost',
                                       database='Library')
      Cursor = cnx.cursor()
      query = ("SELECT * FROM Member")
      Cursor.execute(query)
      for (Mno,Mname,MOB,DOP,ADR) in Cursor:
         print("==================================================")
         print("Member Code : ",Mno)
         print("Member Name : ",Mname)
         print("Mobile No.of Member : ",MOB)
         print("Date of Membership : ",DOP)
         print("Address : ",ADR)
         print("==================================================")
      Cursor.close()
      cnx.close()
      print("You have done it!!!!!!")
   except mysql.connector.Error as err:
      if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
         print("Something is wrong with your user name or password")
      elif err.errno == errorcode.ER_BAD_DB_ERROR:
         print("Database does not exist")
      else:
         print(err)
   else:
      cnx.close()
```

```
def insertMember():
    try:
        cnx = connection.MySQLConnection(user='root',password='',
                                         host='localhost',
                                         database='Library')
        Cursor = cnx.cursor()
        mno=input("Enter Member Code : ")
        mname=input("Enter Member Name : ")
        mob=input("Enter Member Mobile No. : ")
        print("Enter Date of Membership (Date/Month and Year seperately): ")
        DD=int(input("Enter Date : "))
        MM=int(input("Enter Month : "))
        YY=int(input("Enter Year : "))
        addr=input("Enter Member Adress : ")
        Qry = ("INSERT INTO Member VALUES(%s, %s, %s, %s, %s)")
        data = (mno,mname,mob,date(YY,MM,DD),addr)
        Cursor.execute(Qry,data)
    # Make sure data is committed to the database cnx.commit()
        Cursor.close()
        cnx.close()
        print("Record Inserted.............")
    except mysql.connector.Error as err:
        if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
            print("Something is wrong with your user name or password")
        elif err.errno == errorcode.ER_BAD_DB_ERROR:
            print("Database does not exist")
        else:
            print(err)
        cnx.close()


def deleteMember():
    try:
        cnx = connection.MySQLConnection(user='root',password='',
                                         host='localhost',
                                         database='Library')
        Cursor = cnx.cursor()
        mno=input("Enter Member Code to be deleted from the Library : ")
        Qry =("""DELETE FROM Member WHERE MNO = %s""")
        del_rec=(mno,)
        Cursor.execute(Qry,del_rec)
    # Make sure data is committed to the database cnx.commit()
```

```
        Cursor.close()
        cnx.close()
        print(Cursor.rowcount,"Record(s) Deleted Successfully..........")
    except mysql.connector.Error as err:
        if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
            print("Something is wrong with your user name or password")
        elif err.errno == errorcode.ER_BAD_DB_ERROR:
            print("Database does not exist")
        else:
            print(err)
        cnx.close()


def SearchMember():
    try:
        cnx = connection.MySQLConnection(user='root',password='',
                                         host='localhost',
                                         database='Library')
        Cursor = cnx.cursor()
        mnm=input("Enter Book Name to be Searched from the Library : ")
        query = ("SELECT * FROM Member where MName = %s ")
        rec_srch=(mnm,)
        Cursor.execute(query,rec_srch)
        Rec_count=0
        for (Mno,Mname,MOB,DOP,ADR) in Cursor:
            print("================================================")
            print("Member Code : ",Mno)
            print("Member Name : ",Mname)
            print("Mobile No.of Member : ",MOB)
            print("Date of Membership : ",DOP)
            print("Address : ",ADR)
            print("================================================")
            if Rec_count%2==0:
                input("Press any key to continue")
                clrscreen()
        print(Rec_count, "Record(s) found")
    # Make sure data is committed to the database cnx.commit()
        Cursor.close()
        cnx.close()
    except mysql.connector.Error as err:
        if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
            print("Something is wrong with your user name or password")
        elif err.errno == errorcode.ER_BAD_DB_ERROR:
            print("Database does not exist")
        else:
            print(err)
        cnx.close()
```

Once the above code is written and all modules are created, upon execution it will show the following output window:



```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.19
14 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
 RESTART: C:\Users\preeti\AppData\Local\Programs\Python\Python37-
32\proj_lib_managnt.py
0
                         Library Management

==============================================================
1. Book Management
2. Members Management
3. Issue/Return Book
4. Exit
===============================================================
Enter Choice between 1 to 4-------> : 1
                         Book Record Management

==============================================================
1. Add Book Record
2. Display Book Records
3. Search Book Record
4. Delete Book Record
5. Update Book Record
6. Return to Main Menu
===============================================================
```

```
Enter Choice between 1 to 5-------> : 1
Enter Book Code : 10
Enter Book Name : Python Programming
Enter Book Author's Name : Radhika Arora
Enter Book Price : 450
Enter Publisher of Book : ABC Publishers
Enter Quantity purchased : 100
Enter Date of Purchase (Date/Month and Year seperately:
Enter Date : 07
Enter Month : 02
Enter Year : 2019
Record Inserted.............
```

In the above project, we have used formatting with print() statement, which we will discuss now.

FORMATTING DATA

Formatting Data with print()Statement

Usually we align strings to the left and numbers to the right. Python supports formatting values into strings. This operator has two forms:

- A formatting string followed by one argument, or
- A tuple of arguments.

Python overloads the modulus operator (%) for use with strings to produce formatted output. While using %, a number of values may follow the % operator. These values are used up by % placeholders in the format string. When print statement uses the % sign with numbers, the per cent sign acts as a remainder operation. With strings, the per cent sign is a string formatting operator.

In general the string formatting operator is used like this:

<string>%<values>

Table shows Python's data format characters.

Symbol	**Description**
%c	character
%s	string
%d	signed decimal integer
%i	integer
%u	unsigned decimal integer
%o	octal integer
%f	floating number
%e	exponential notation with lowercase e
%E	exponential notation with uppercase E
%x	hexadecimal integer in lowercase
%X	hexadecimal integer in uppercase

Note: The % operator is sometimes called string interpolation (we prefer to call it string formatting) since it interpolates literal text and converted values.

For example,

```
print("string %d, %f, %.2f" <value1, value2, value3>)
```

Here, the print statement uses three numeric data including one integer and two floating-point numbers. The function sues same number values for each format specifier. We can format the data using width specifier.

The general format is:

```
%<width>.<precision><type-char>
```

The specifier starts with a % and ends with a character that indicates the data type of the value being inserted. We will see three different format types: string, integer and float.

For example:

```
>>># A string format
>>> print (“These are %s %s”%(“CBSE”, “Schools”))
These are CBSE Schools
>>># An integer format with width specifier 7
>>> print (“Number is: %7” % 4590)  #Notice the output with space.
Number is: 4590
>>># A Floating point number with width specifier 10.
>>> import math                # Module imported to use math constant pi
>>> print (“The value Pl is: %10.5f” % math.pi)
The value Pl is: 3.14159
```

- %s. It is used when a string is expected. This is the most basic usage to insert values into a string with the %s placeholder. *For example*, the string %s is replaced by an argument that is type casted to a string (*i.e.*, converted from any type to the string type) and then printed.

  ```
  >>>Age = 21
  >>>print (“Your age is: %s” %Age)
  Your age is: 21
  ```

  Here, the %s is replaced by the value Age.

  Let us take another example.

  ```
  >>>FName = “Naman”
  >>>LName = “Singhal”
  >>>print(“Name is: %s %s” (FName, LName)
  Name is : Naman Singhal
  ```

  Here the (FName, LName) is a tuple and the whole expression evaluates to a string. The first %s is replaced by the value of FName. All other characters in the string stay as they are %i & %d. The code i and d both are used for integers. String formatting also works with integers by specifying %i or %d instead of %s.

Practical File

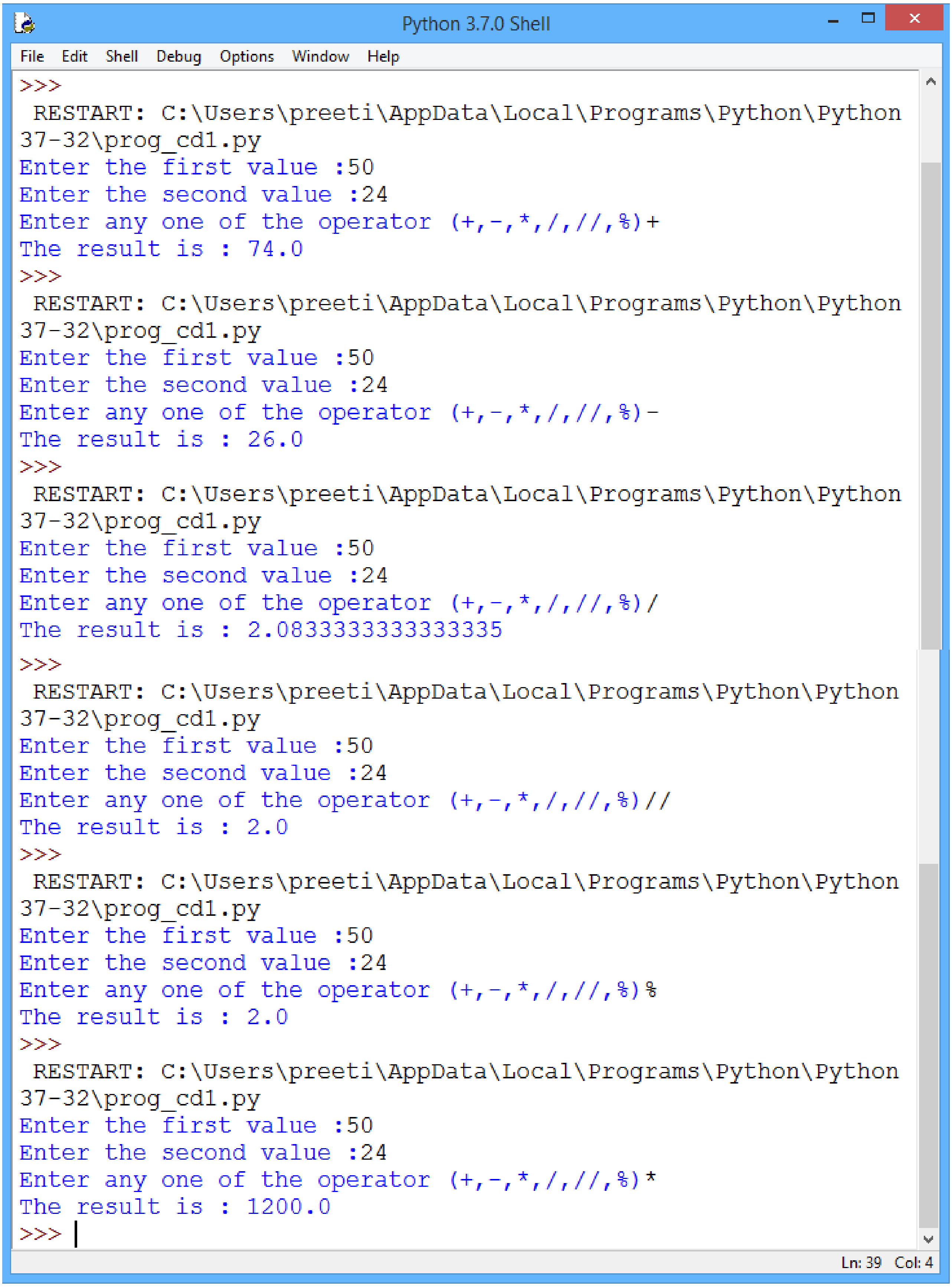
Class XII - Computer Science with Python(083)

Program 1: Program to enter two numbers and print the arithmetic operations like +,-,*, /, // and %.

Solution:

```
#Program for Arithmetic Calculator

result = 0

val1 = float(input("Enter the first value :"))

val2 = float(input("Enter the second value :"))

op = input("Enter any one of the operator (+,-,*,/,//,%)")

if op == "+":

    result = val1 + val2

elif op == "-":

    result = val1 - val2

elif op == "*":

    result = val1 * val2

elif op == "/":

    if val2 == 0:

        print("Please enter a value other than 0")

    else:

        result = val1 / val2

elif op == "//":

    result = val1 // val2

else:

    result = val1 % val2
```

```
print("The result is :",result)
```

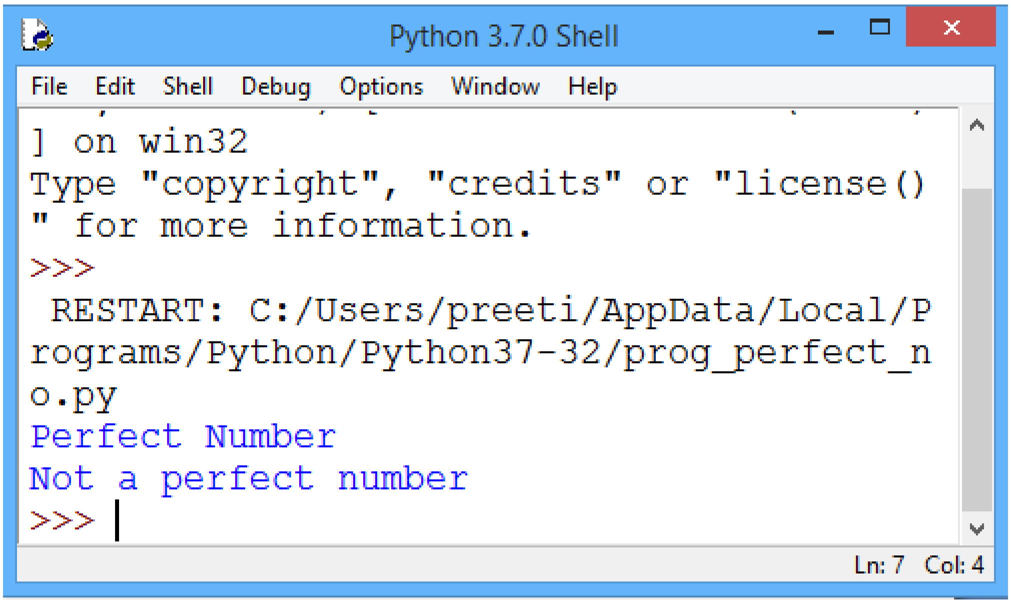


```
>>>
 RESTART: C:\Users\preeti\AppData\Local\Programs\Python\Python
37-32\prog_cd1.py
Enter the first value :50
Enter the second value :24
Enter any one of the operator (+,-,*,/,//,%)+
The result is : 74.0
>>>
 RESTART: C:\Users\preeti\AppData\Local\Programs\Python\Python
37-32\prog_cd1.py
Enter the first value :50
Enter the second value :24
Enter any one of the operator (+,-,*,/,//,%)-
The result is : 26.0
>>>
 RESTART: C:\Users\preeti\AppData\Local\Programs\Python\Python
37-32\prog_cd1.py
Enter the first value :50
Enter the second value :24
Enter any one of the operator (+,-,*,/,//,%)/
The result is : 2.0833333333333335
>>>
 RESTART: C:\Users\preeti\AppData\Local\Programs\Python\Python
37-32\prog_cd1.py
Enter the first value :50
Enter the second value :24
Enter any one of the operator (+,-,*,/,//,%)//
The result is : 2.0
>>>
 RESTART: C:\Users\preeti\AppData\Local\Programs\Python\Python
37-32\prog_cd1.py
Enter the first value :50
Enter the second value :24
Enter any one of the operator (+,-,*,/,//,%)%
The result is : 2.0
>>>
 RESTART: C:\Users\preeti\AppData\Local\Programs\Python\Python
37-32\prog_cd1.py
Enter the first value :50
Enter the second value :24
Enter any one of the operator (+,-,*,/,//,%)*
The result is : 1200.0
>>>
```

Program 2: Write a program to find whether an inputted number is perfect or not.

Solution:

```
# To find whether a number is perfect or not

def pernum(num):

    divsum=0

    for i in range(1,num):

        if num%i == 0:

            divsum+=i

    if divsum==num:

        print('Perfect Number')

    else:

        print('Not a perfect number')


pernum(6)

pernum(15)
```

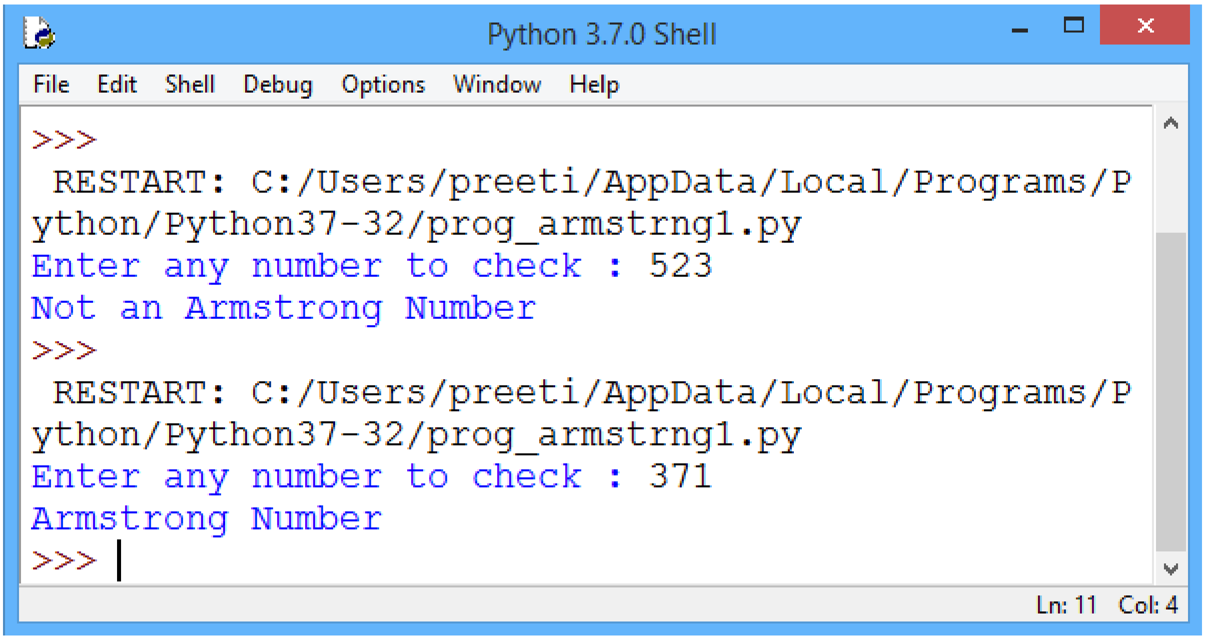


```
] on win32
Type "copyright", "credits" or "license()
" for more information.
>>>
 RESTART: C:/Users/preeti/AppData/Local/P
rograms/Python/Python37-32/prog_perfect_n
o.py
Perfect Number
Not a perfect number
>>> 
```

Program 3: Write a Program to check if the entered number is Armstrong or not.

Solution:

```
# Program to check if the entered number is Armstrong or not.

#An Armstrong number has sum of the cubes of its digits is equal to the number itself

no=int(input("Enter any number to check : "))

no1 = no

sum = 0

while(no>0):

    ans = no % 10;

    sum = sum + (ans * ans * ans)

    no = int (no / 10)

if sum == no1:

    print("Armstrong Number")

else:

    print("Not an Armstrong Number")
```

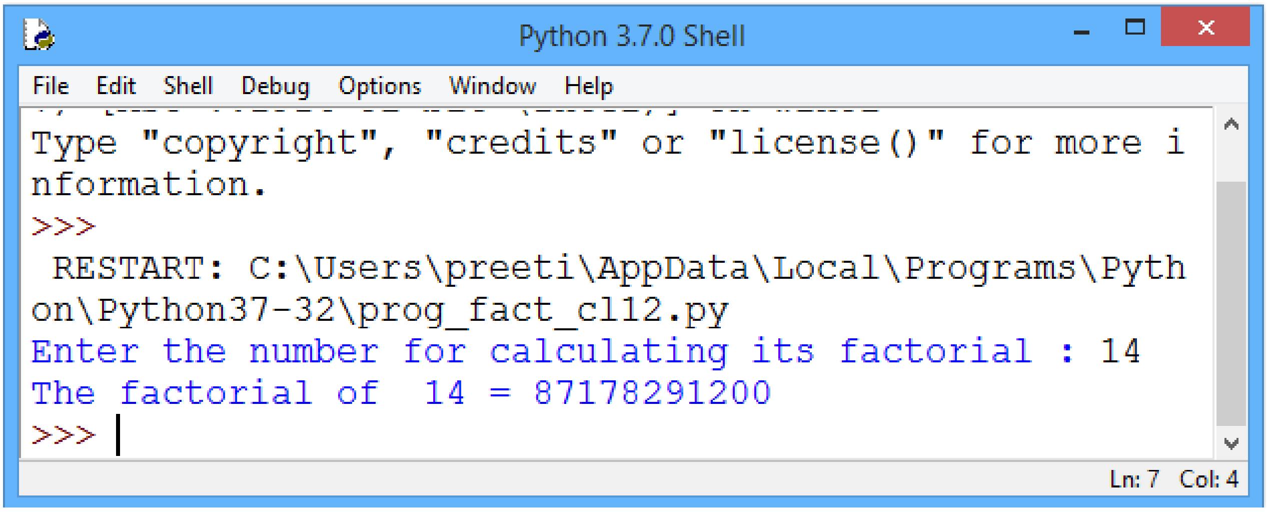


```
>>>
 RESTART: C:/Users/preeti/AppData/Local/Programs/P
ython/Python37-32/prog_armstrng1.py
Enter any number to check : 523
Not an Armstrong Number
>>>
 RESTART: C:/Users/preeti/AppData/Local/Programs/P
ython/Python37-32/prog_armstrng1.py
Enter any number to check : 371
Armstrong Number
>>> 
```

Program 4: Write a Program to find factorial of the entered number.

Solution:

```
#Program to calculate the factorial of an inputted number (using while loop)

num = int(input("Enter the number for calculating its factorial : "))

fact = 1

i = 1

while i<=num:

    fact = fact*i

    i = i + 1

print("The factorial of ",num,"=",fact)
```



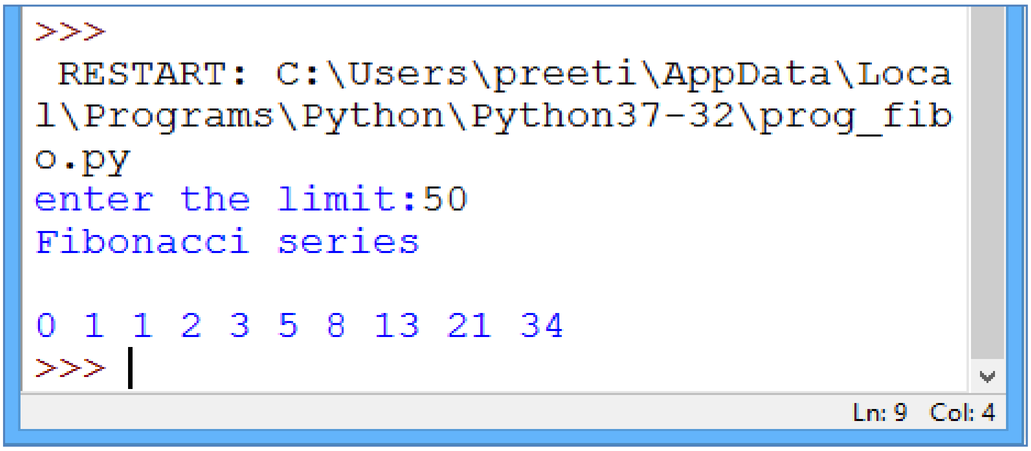
```
Type "copyright", "credits" or "license()" for more i
nformation.
>>>
 RESTART: C:\Users\preeti\AppData\Local\Programs\Pyth
on\Python37-32\prog_fact_cl12.py
Enter the number for calculating its factorial : 14
The factorial of  14 = 87178291200
>>> 
```

Program 5: Write a Program to enter the number of terms and to print the Fibonacci Series.

Solution:

```
#fibonacci

i =int(input("enter the limit:"))

x = 0
```

```
y = 1

z = 1

print("Fibonacci series \n")

print(x, y,end= " ")

while(z<= i):

   print(z, end=" ")

   x = y

   y = z

   z = x + y
```



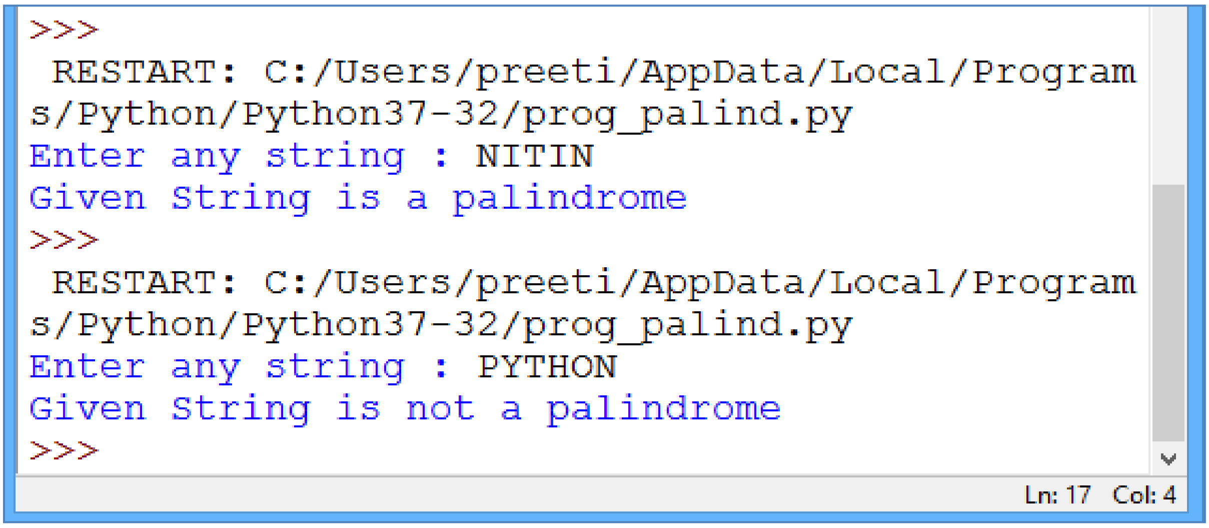
```
>>> 
 RESTART: C:\Users\preeti\AppData\Loca
l\Programs\Python\Python37-32\prog_fib
o.py
enter the limit:50
Fibonacci series 

0 1 1 2 3 5 8 13 21 34
>>> 
```

Program 6: Write a Program to enter the string and to check if it’s palindrome or not using loop.

Solution:

```
# Program to enter the string and check if it’s palindrome or not using ‘for’ loop.

msg=input("Enter any string : ")

newlist=[]

newlist[:0]=msg

l=len(newlist)

ed=l-1

for i in range(0,l):
```

```
if newlist[i]!=newlist[ed]:
   print ("Given String is not a palindrome")
   break
if i>=ed:
   print ("Given String is a palindrome")
   break
l=l-1
ed = ed - 1
```

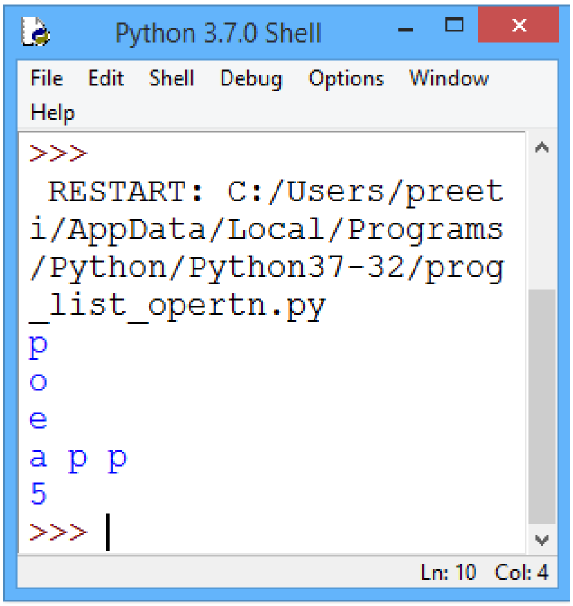


```
>>> 
 RESTART: C:/Users/preeti/AppData/Local/Program
s/Python/Python37-32/prog_palind.py
Enter any string : NITIN
Given String is a palindrome
>>> 
 RESTART: C:/Users/preeti/AppData/Local/Program
s/Python/Python37-32/prog_palind.py
Enter any string : PYTHON
Given String is not a palindrome
>>> 
```

Program 7: Write a Program to show the outputs based on entered list.

Solution:

```
my_list = ['p','r','o','b','e']
# Output: p
print(my_list[0])
# Output: o
print(my_list[2])
# Output: e
print(my_list[4])
# Error! Only integer can be used for indexing
# my_list[4.0]

# Nested List

n_list = ["Happy", [2,0,1,5]]

# Nested indexing

# Output: a

print(n_list[0][1],n_list[0][2],n_list[0][3])

# Output: 5

print(n_list[1][3])
```

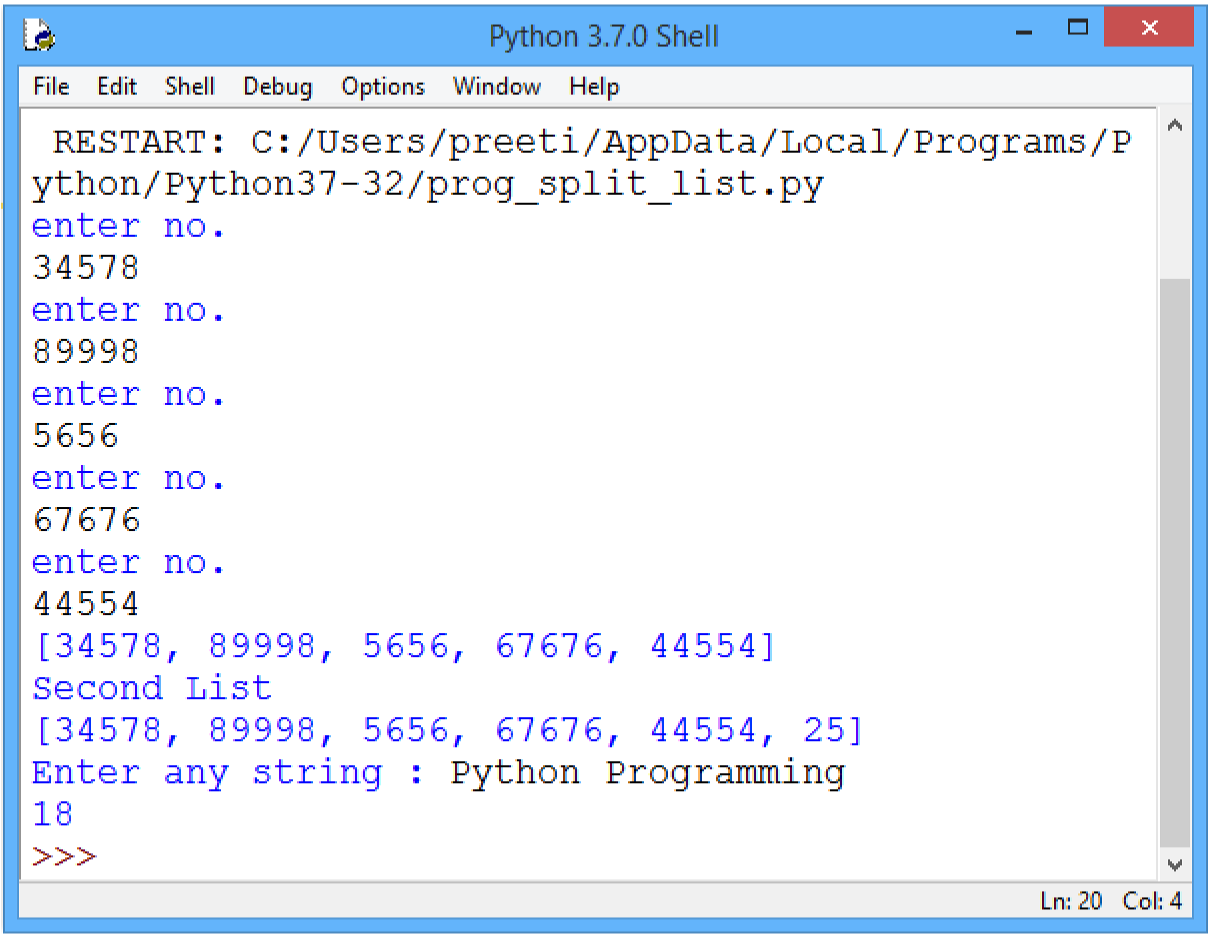


```
>>>
 RESTART: C:/Users/preet
i/AppData/Local/Programs
/Python/Python37-32/prog
_list_opertn.py
p
o
e
a p p
5
>>> 
```

Program 8: Write a Program to enter the numbers in a list using split () and to use all the functions related to list.

Solution:

```
#Program to enter the numbers in a list using split () and to use all the functions related to list.

# numbers = [int(n, 10) for n in input().split(",")]

# print (len(numbers))

 memo=[]

for i in range (5):
    x=int(input("enter no. \n"))
    memo.insert(i,x)
    i+=1
print(memo)
memo.append(25)
print("Second List")
print(memo)
msg=input("Enter any string : ")
newlist=[]
newlist[:0]=msg
l=len(newlist)
print(l)
```

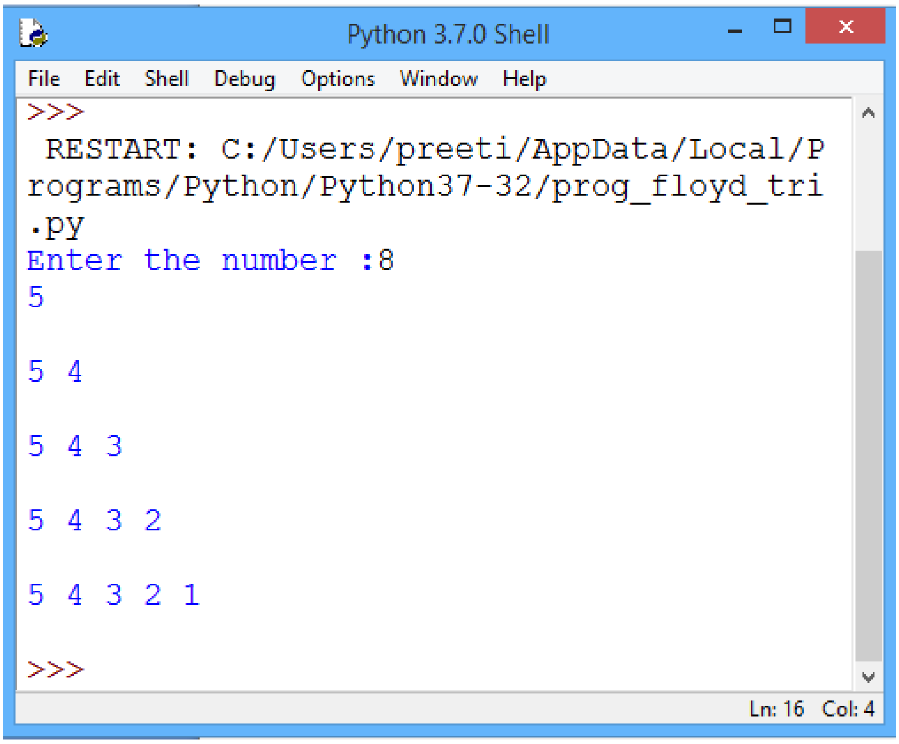


```
 RESTART: C:/Users/preeti/AppData/Local/Programs/P
ython/Python37-32/prog_split_list.py
enter no.
34578
enter no.
89998
enter no.
5656
enter no.
67676
enter no.
44554
[34578, 89998, 5656, 67676, 44554]
Second List
[34578, 89998, 5656, 67676, 44554, 25]
Enter any string : Python Programming
18
>>>
```

Program 9: Write a Program to enter the number and print the Floyd's Triangle in decreasing order.

Solution:

```
#Floyd's triangle

n=int(input("Enter the number :"))

for i in range(5,0,-1):

   for j in range(5,i-1,-1):

      print (j,end=' ')

   print('\n')
```



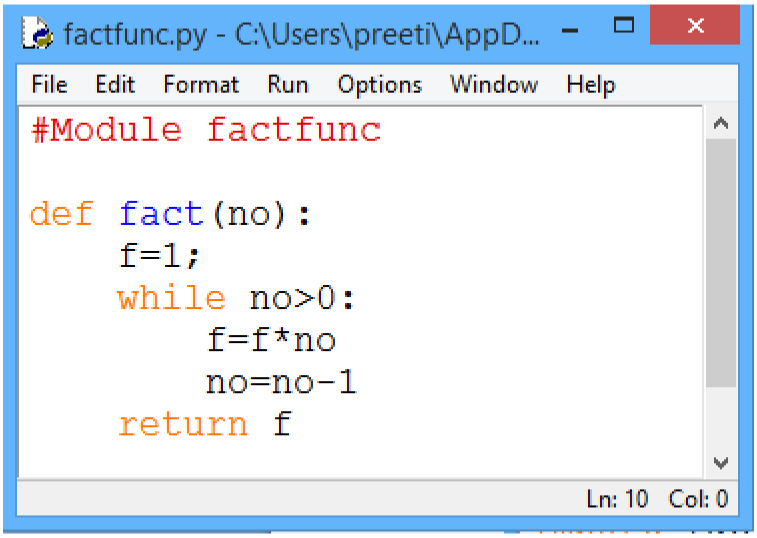
```
>>>
 RESTART: C:/Users/preeti/AppData/Local/P
rograms/Python/Python37-32/prog_floyd_tri
.py
Enter the number :8
5

5 4

5 4 3

5 4 3 2

5 4 3 2 1

>>>
```

Program 10: Write a Program to find factorial of entered number using user-defined module fact().

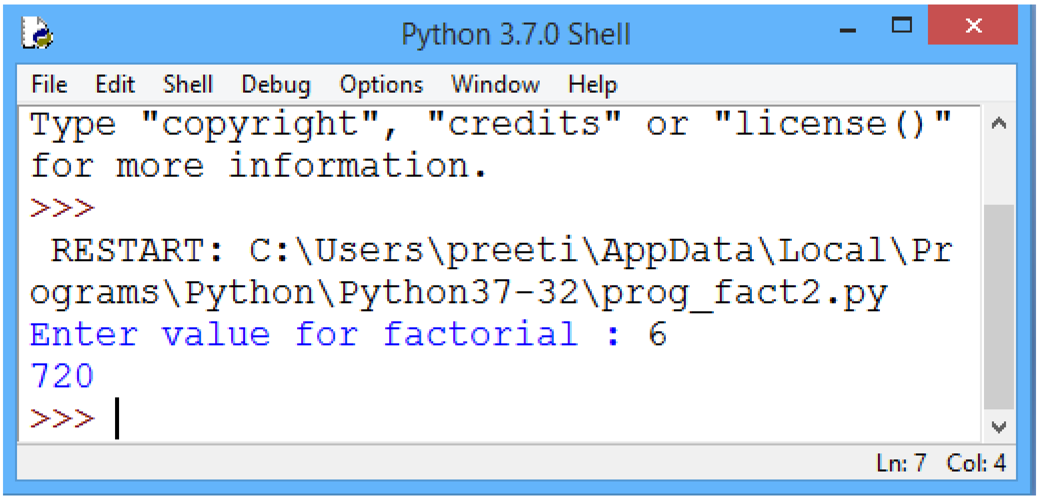
Solution:

```
#Using function

import factfunc

x=int(input("Enter value for factorial : "))

ans=factfunc.fact(x)
```

print (ans)


factfunc.py - C:\Users\preeti\AppD...
File Edit Format Run Options Window Help
Ln: 10 Col: 0

```
#Module factfunc

def fact(no):
    f=1;
    while no>0:
        f=f*no
        no=no-1
    return f
```



```
Type "copyright", "credits" or "license()"
for more information.
>>>
 RESTART: C:\Users\preeti\AppData\Local\Pr
ograms\Python\Python37-32\prog_fact2.py
Enter value for factorial : 6
720
>>> 
```

Program 11: Write a Program to enter the numbers and find Linear Search, Binary Search, Lowest Number and Selection Sort using list/array code.

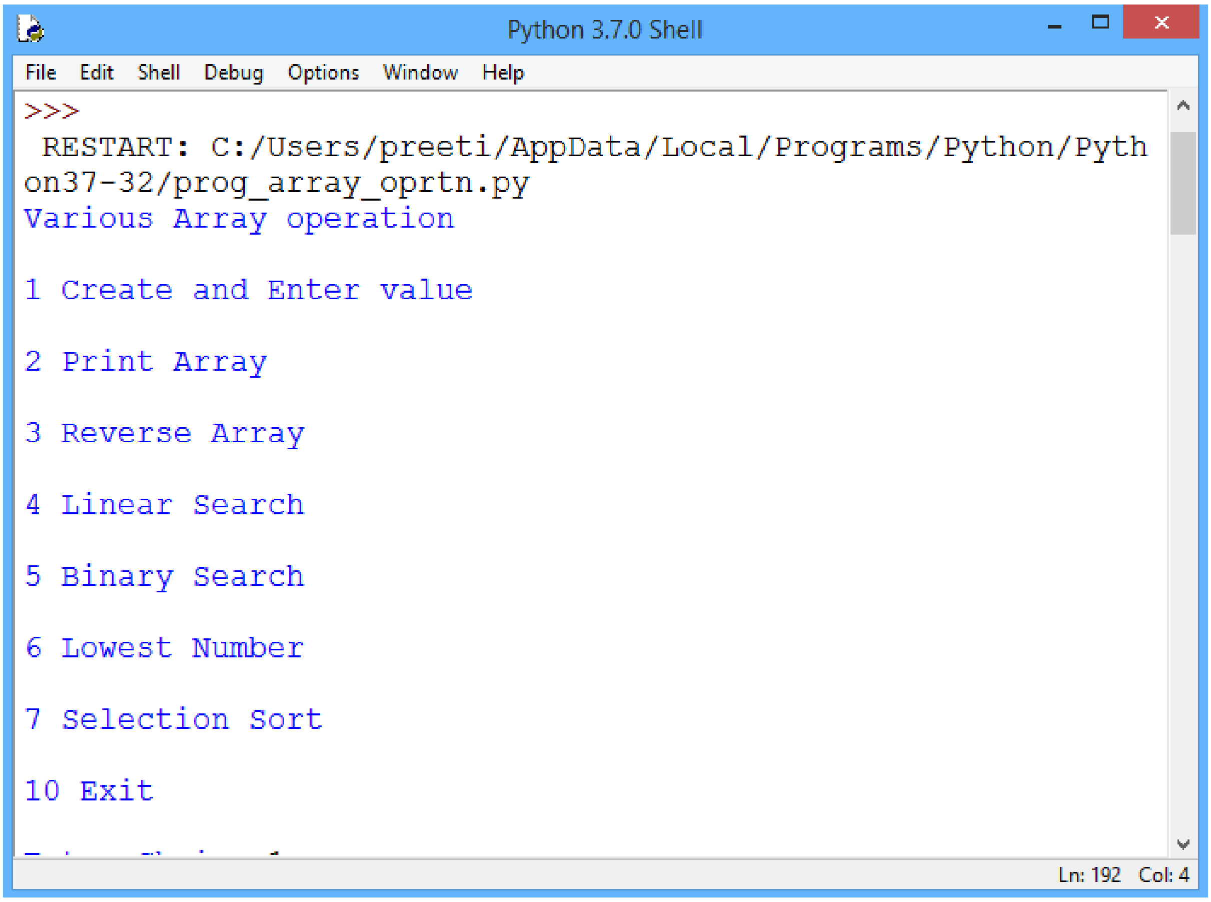
Solution:

```
arr=[]

def array_operation():

  ch=1

  while ch!=10:

    print('Various Array operation\n')

    print('1 Create and Enter value\n')

    print('2 Print Array\n')

    print('3 Reverse Array\n')
```

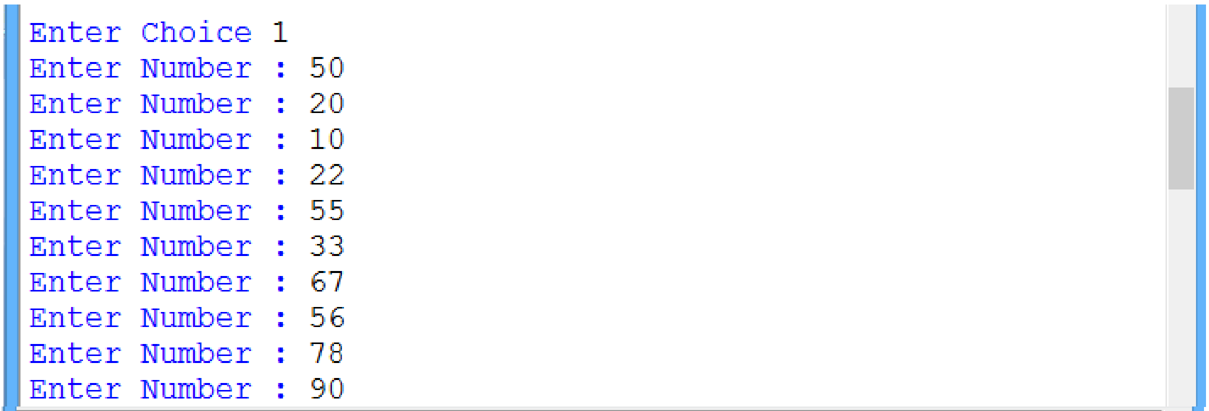
```
        print('4 Linear Search\n')

        print('5 Binary Search\n')

        print('6 Lowest Number \n')

        print('7 Selection Sort\n')

        print('10 Exit\n')

        ch=int(input('Enter Choice '))

        if ch==1 :

           appendarray()

        elif ch==2 :

           print_array()

        elif ch==3 :

           reverse_array()

        elif ch==4 :

           linear_search()

        elif ch==5 :

           binary_search()

        elif ch==6 :

           min_number()

        elif ch==7 :

           selection_sort()


  def appendarray():

     for i in range(0,10):

        x=int(input('Enter Number : '))

        arr.insert(i,x)
```

```
#------------------------------------------------------------------------------------------------------------------------------------------
def print_array():
    for i in range(0,10):
        print(arr[i]),
#------------------------------------------------------------------------------------------------------------------------------------------
def reverse_array():
    for i in range(1,11):
        print(arr[-i]),
#------------------------------------------------------------------------------------------------------------------------------------------
def lsearch():
    try:
        x=int(input('Enter the Number You want to search : '))
        n=arr.index(x)
        print ('Number Found at %d location'% (i+1))
    except:
        print('Number Not Exist in list')
#------------------------------------------------------------------------------------------------------------------------------------------
def linear_search():
    x=int(input('Enter the Number you want to search : '))
    fl=0
    for i in range(0,10):
        if arr[i]==x :
            fl=1
```

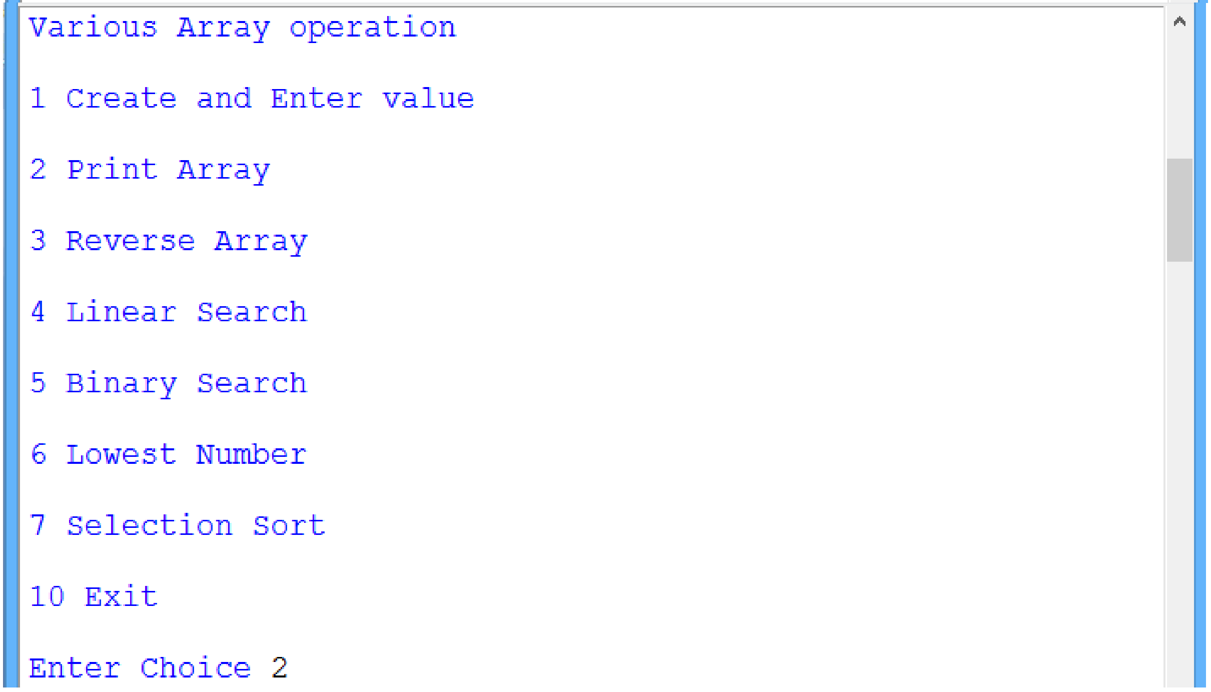
```
         print ('Number Found at %d location'% (i+1))
         break
   if fl==0 :
       print ('Number Not Found')
#-------------------------------------------------------------------------------------------------------------------------
----------------
def binary_search():
   x=int(input('Enter the Number you want to search : '))
   fl=0
   low=0
   heigh=len(arr)
   while low<=heigh :
      mid=int((low+heigh)/2)
      if arr[mid]==x :
         fl=1
         print ('Number Found at %d location'% (mid+1))
         break
      elif arr[mid]>x :
         low=mid+1
      else :
         heigh=mid-1

   if fl==0 :
      print ('Number Not Found')
#-------------------------------------------------------------------------------------------------------------------------
----------------
def min_number():
    n=arr[0]

    k=0

    for i in range(0,10):

        if arr[i]<n :

            n=arr[i]

            k=i

    print('The Lowest number is %d '%(n))


#--------------------------------------------------------------------------------------------------------------------
----------------

def selection_sort():

    for i in range(0,10):

        n=arr[i]

        k=i

        for j in range(i+1,10):

            if arr[j]<n :

                n=arr[j]

                k=j

        arr[k]=arr[i]

        arr[i]=n

array_operation()
```



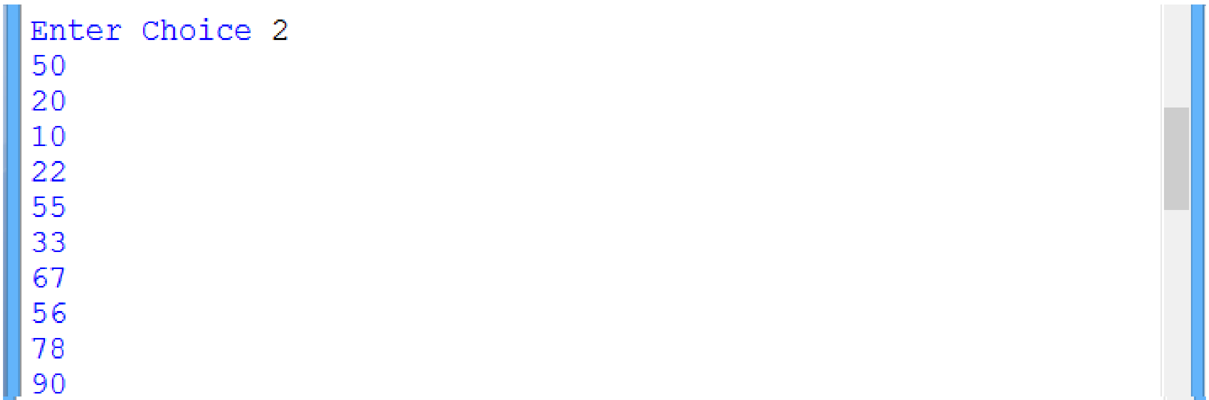
```
>>>
 RESTART: C:/Users/preeti/AppData/Local/Programs/Python/Python37-32/prog_array_oprtn.py
Various Array operation

1 Create and Enter value

2 Print Array

3 Reverse Array

4 Linear Search

5 Binary Search

6 Lowest Number

7 Selection Sort

10 Exit
```



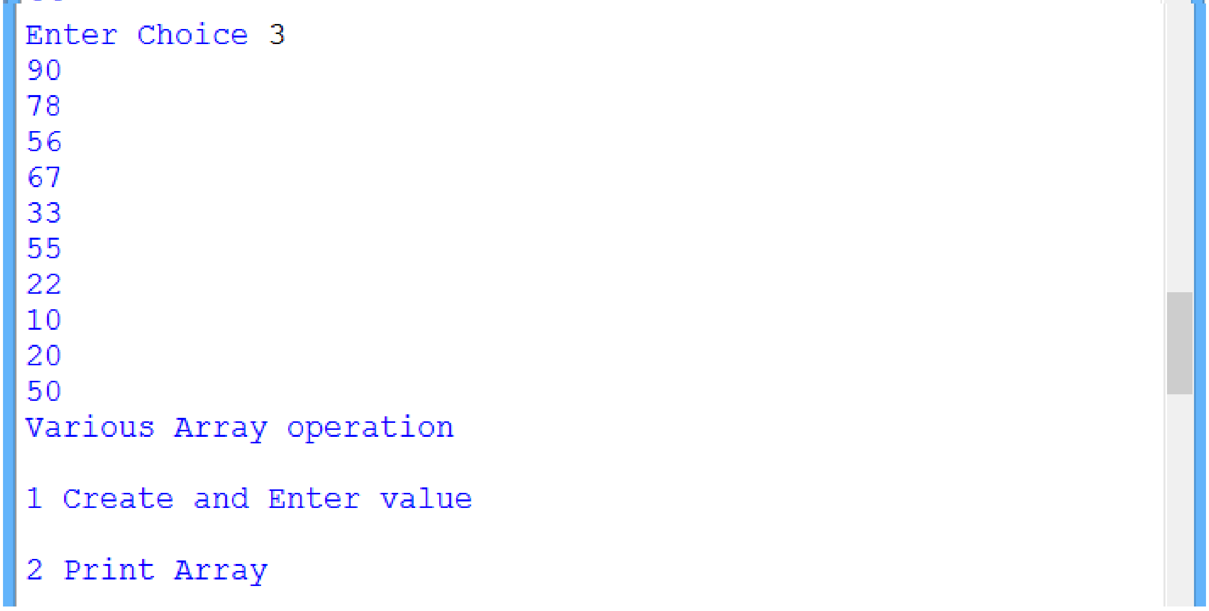
```
Enter Choice 1
Enter Number : 50
Enter Number : 20
Enter Number : 10
Enter Number : 22
Enter Number : 55
Enter Number : 33
Enter Number : 67
Enter Number : 56
Enter Number : 78
Enter Number : 90
```



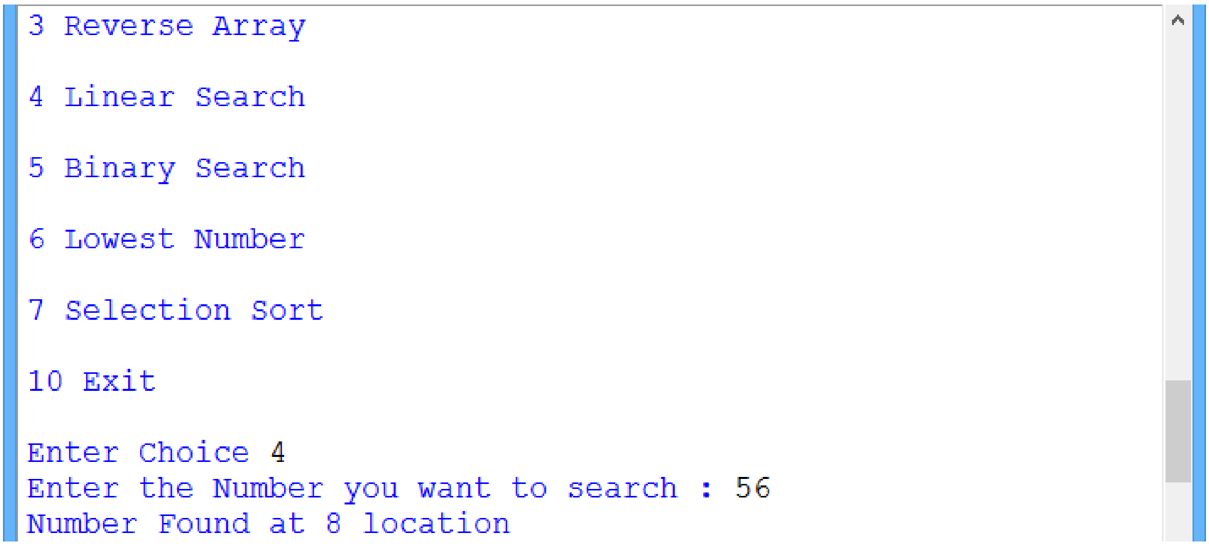
```
Various Array operation

1 Create and Enter value

2 Print Array

3 Reverse Array

4 Linear Search

5 Binary Search

6 Lowest Number

7 Selection Sort

10 Exit

Enter Choice 2
```



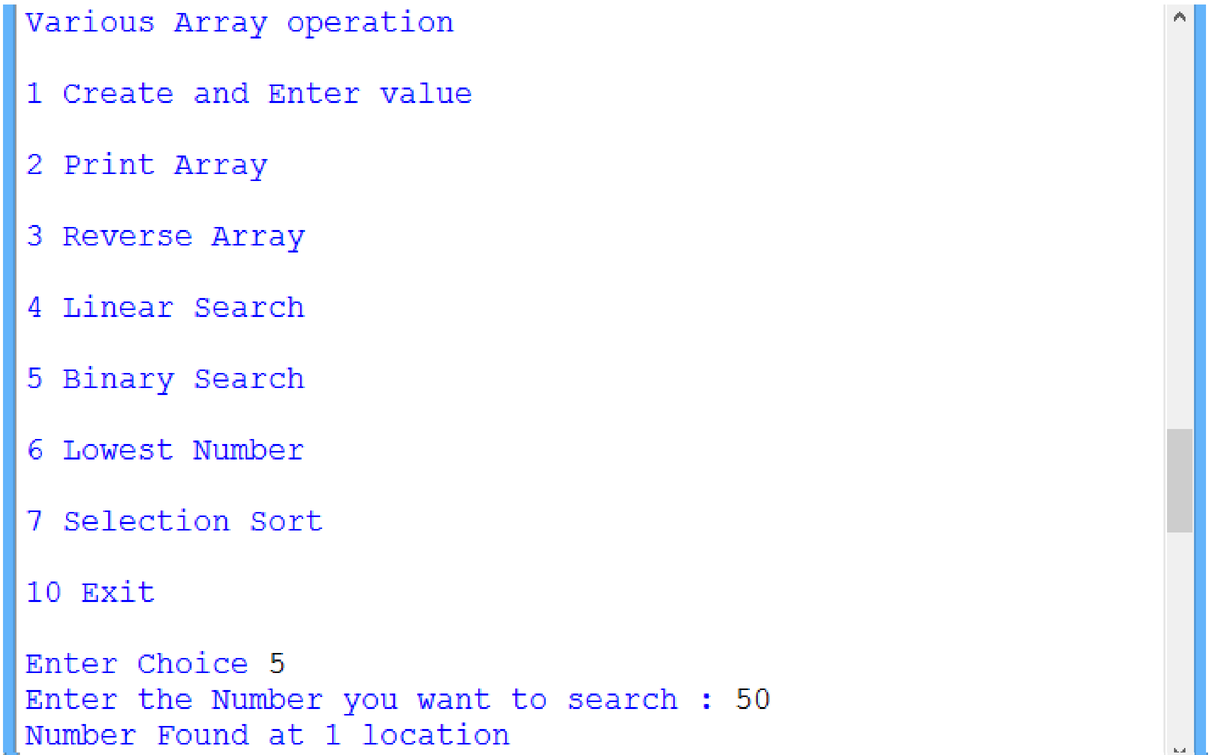
```
Enter Choice 2
50
20
10
22
55
33
67
56
78
90
```



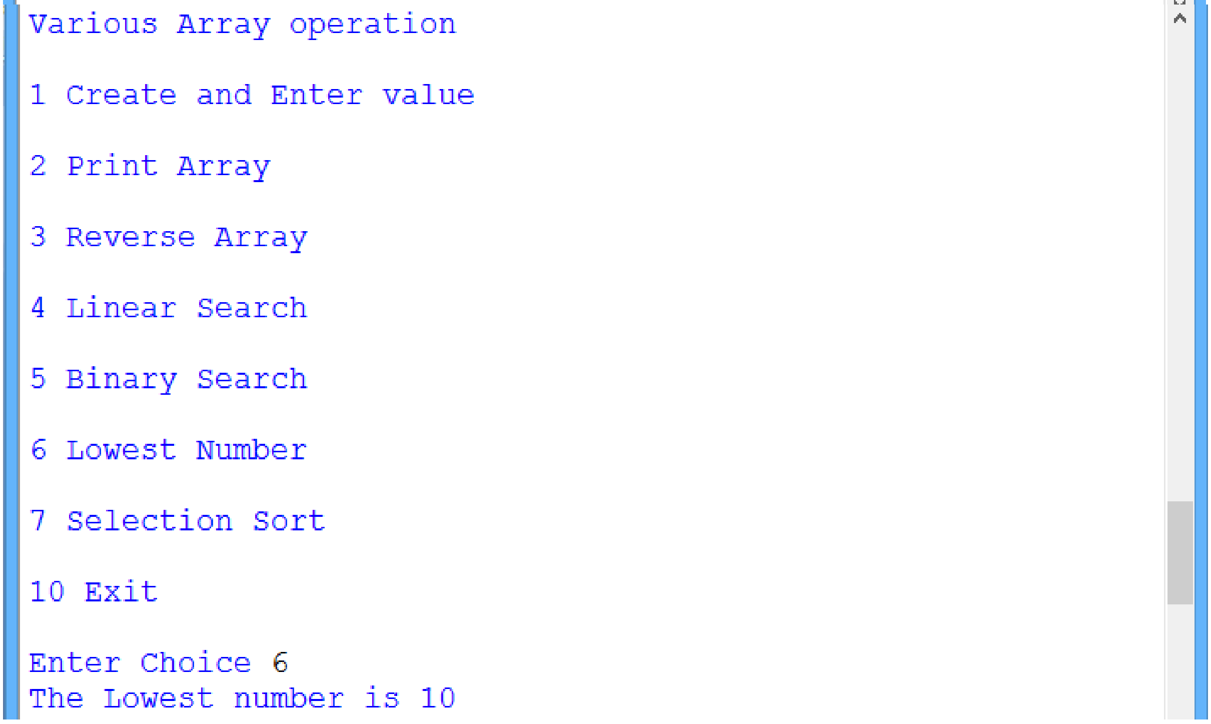
```
Enter Choice 3
90
78
56
67
33
55
22
10
20
50
Various Array operation

1 Create and Enter value

2 Print Array
```



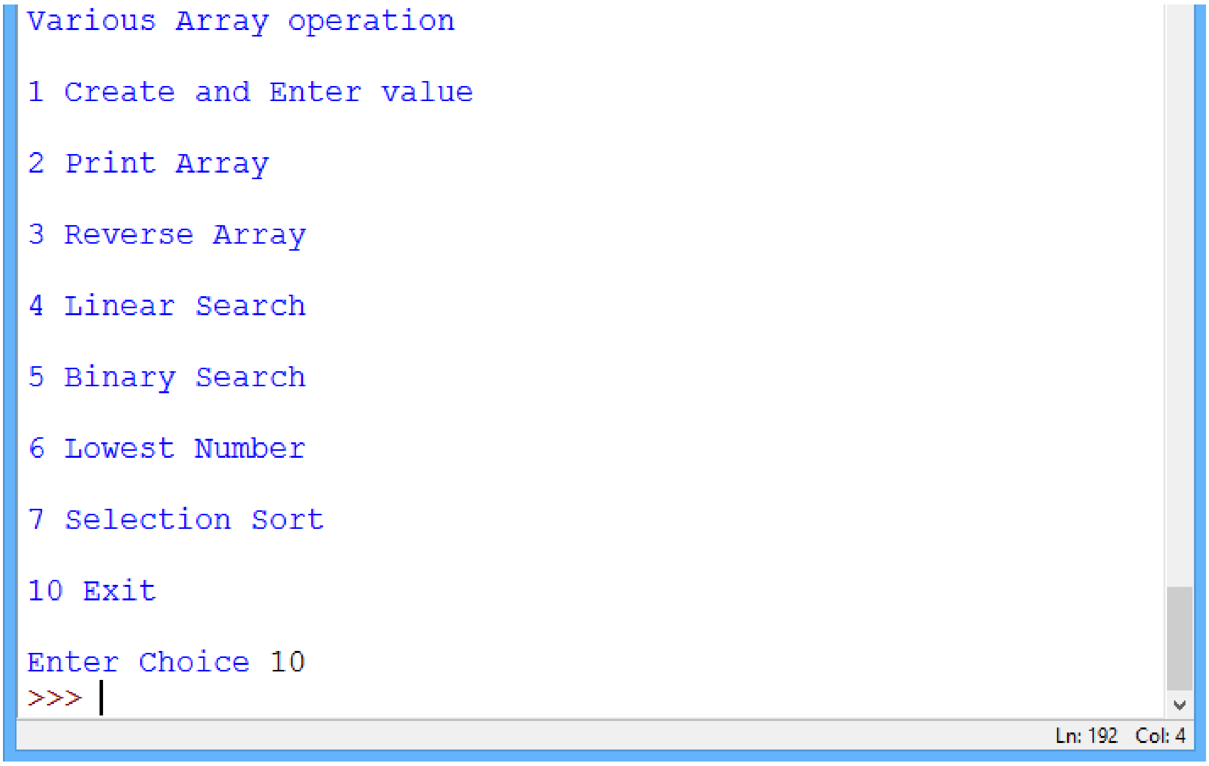
```
3 Reverse Array

4 Linear Search

5 Binary Search

6 Lowest Number

7 Selection Sort

10 Exit

Enter Choice 4
Enter the Number you want to search : 56
Number Found at 8 location
```



```
Various Array operation

1 Create and Enter value

2 Print Array

3 Reverse Array

4 Linear Search

5 Binary Search

6 Lowest Number

7 Selection Sort

10 Exit

Enter Choice 5
Enter the Number you want to search : 50
Number Found at 1 location
```



```
Various Array operation

1 Create and Enter value

2 Print Array

3 Reverse Array

4 Linear Search

5 Binary Search

6 Lowest Number

7 Selection Sort

10 Exit

Enter Choice 6
The Lowest number is 10
```



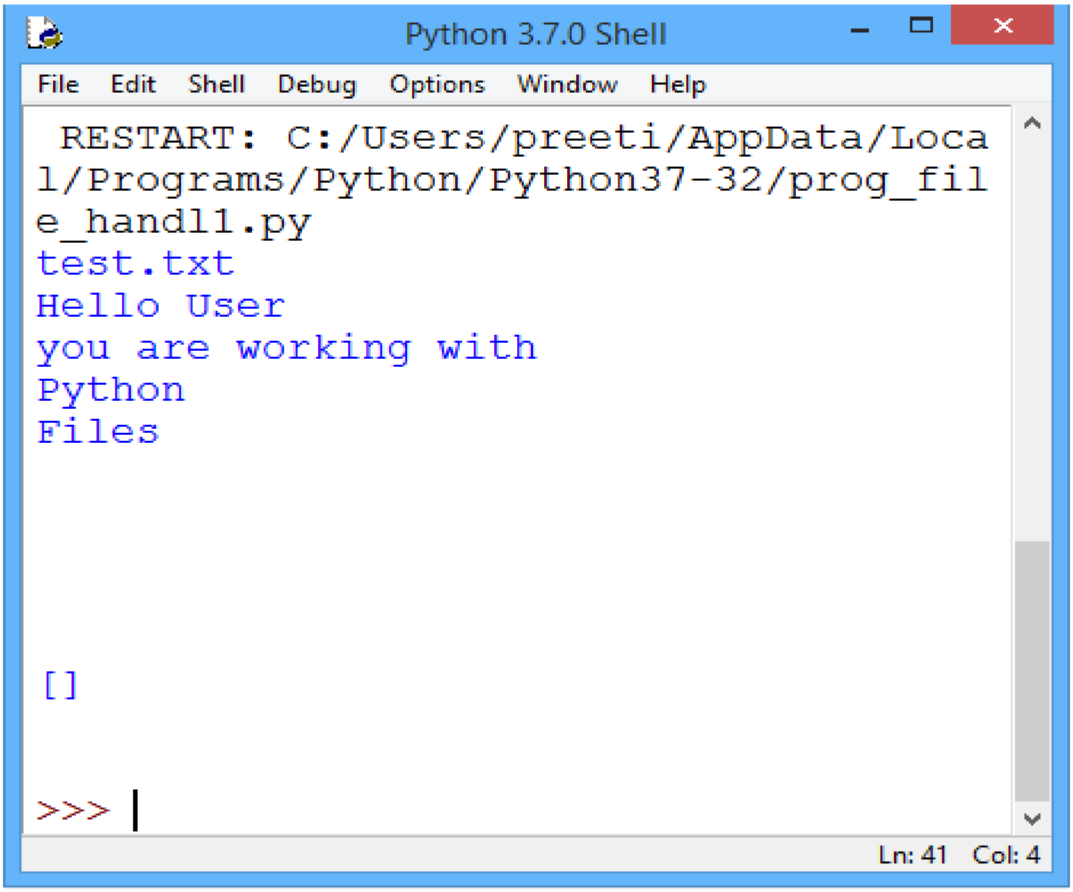
```
Various Array operation

1 Create and Enter value

2 Print Array

3 Reverse Array

4 Linear Search

5 Binary Search

6 Lowest Number

7 Selection Sort

10 Exit

Enter Choice 10
>>> 
```

Program 12: Write a Program to read data from data file and show Data File Handling related functions utility in python.

Solution:

```
f=open("test.txt",'r')

print(f.name)

f_contents=f.read()

print(f_contents)

f_contents=f.readlines()

print(f_contents)

f_contents=f.readline()

print(f_contents)

for line in f:

   print(line, end='')

f_contents=f.read(50)

print(f_contents)
```

```
size_to_read=10

f_contents=f.read(size_to_read)

while len(f_contents)>0:

   print(f_contents)

   print(f.tell())

   f_contents=f.read(size_to_read)
```

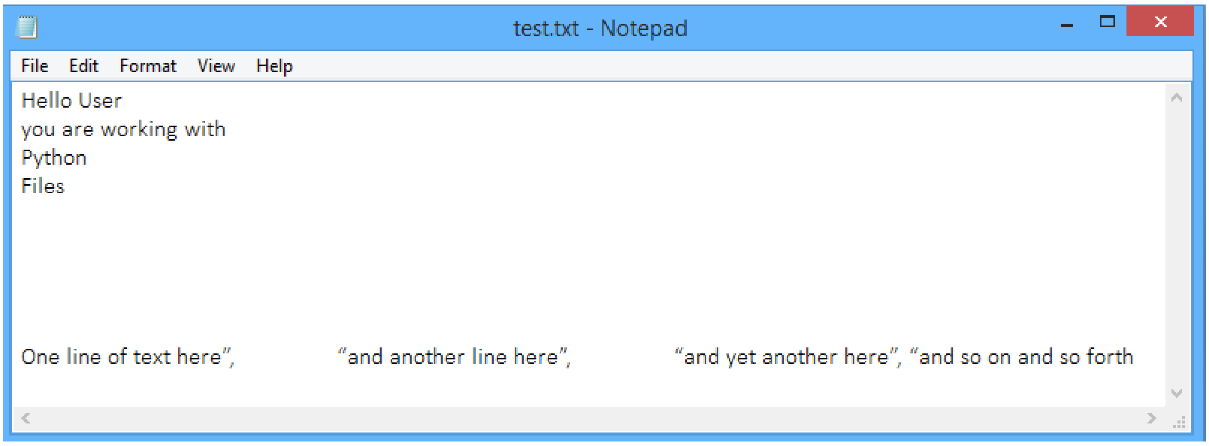


```
 RESTART: C:/Users/preeti/AppData/Loca
l/Programs/Python/Python37-32/prog_fil
e_handl1.py
test.txt
Hello User
you are working with
Python
Files






 []


>>> 
```

Program 13: Write a Program to read data from data file in append mode and use writeLines function utility in python.

Solution:

```
#Program to read data from data file in append mode

af=open("test.txt",'a')

lines_of_text = ("One line of text here”,\

            “and another line here”,\

            “and yet another here”, “and so on and so forth")

af.writelines('\n' + lines_of_text)

af.close()
```



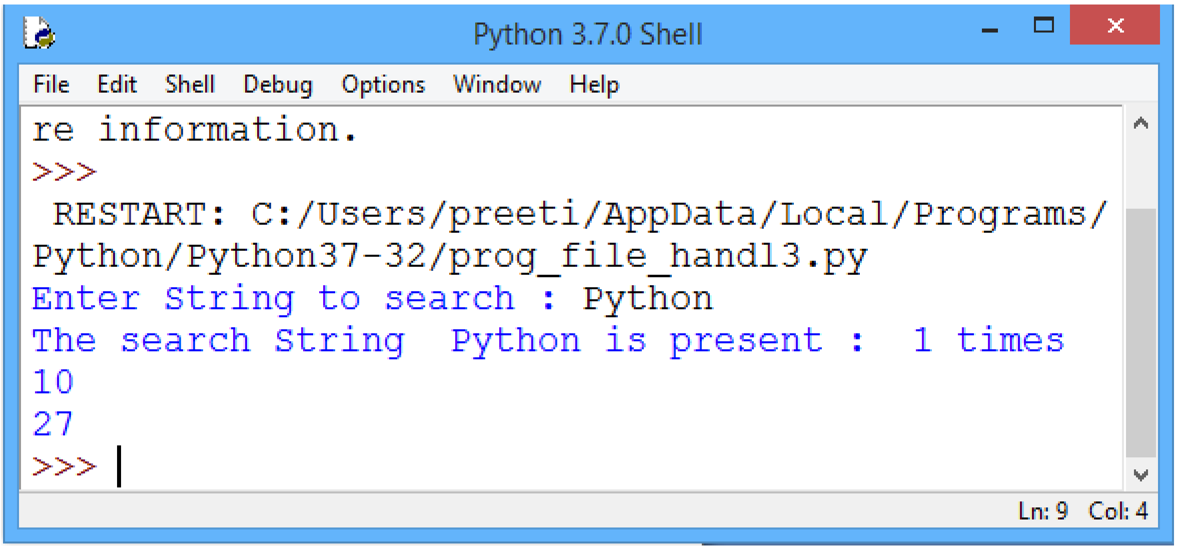
```
Hello User
you are working with
Python
Files



One line of text here”,            “and another line here”,            “and yet another here”, “and so on and so forth
```

Program 14: Write a Program to read data from data file in read mode and count the particular word occurrences in given string, number of times in python.

Solution:

```
#Program to read data from data file in read mode and

#count the particular word occurrences in given string,

#number of times in python.

f=open("test.txt",'r')

read=f.readlines()

f.close()

times=0 #the variable has been created to show the number of times the loop runs

times2=0 #the variable has been created to show the number of times the loop runs

chk=input("Enter String to search : ")
```

```
count=0
for sentence in read:
    line=sentence.split()
    times+=1
    for each in line:
        line2=each
        times2+=1
        if chk==line2:
            count+=1
print("The search String ", chk, "is present : ", count, "times")
print(times)
print(times2)
```

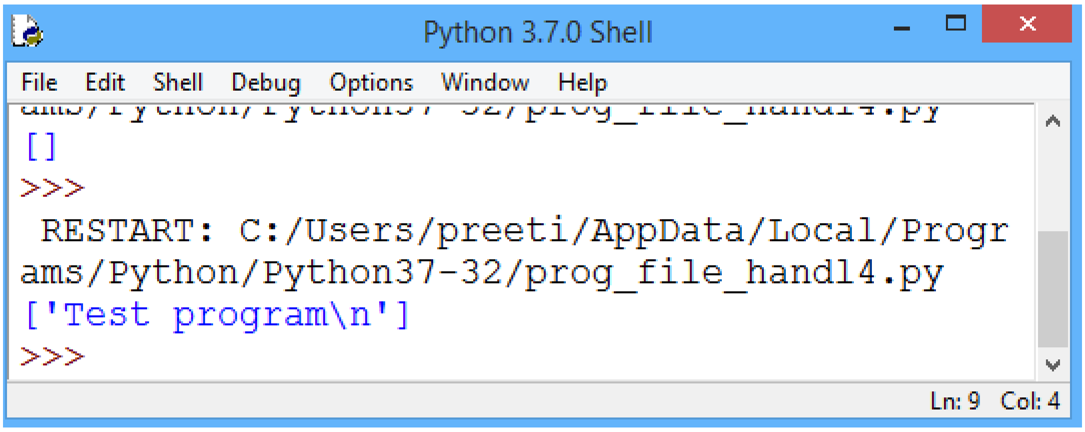


```
re information.
>>>
 RESTART: C:/Users/preeti/AppData/Local/Programs/
Python/Python37-32/prog_file_handl3.py
Enter String to search : Python
The search String  Python is present :  1 times
10
27
>>>
```

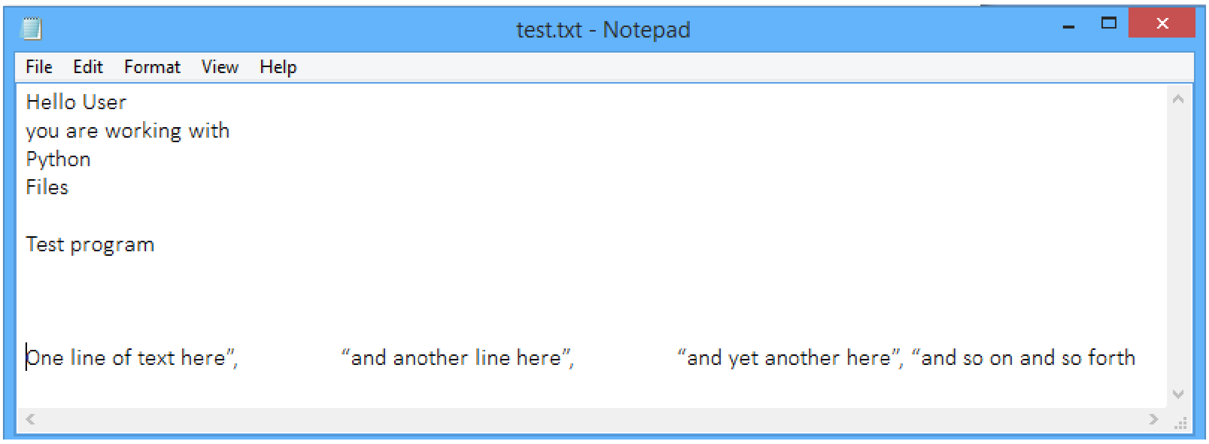
Program 15: Write a Program to read data from data file in read mode and append the words starting with letter ‘T’ in a given file in python.

Solution:

```
#Program to read data from data file in read mode and

#append the words starting with letter ‘T’

#in a given file in python

f=open("test.txt",'r')

read=f.readlines()

f.close()

id=[]

for ln in read:

    if ln.startswith("T"):

        id.append(ln)

print(id)
```



```
ams/Python/Python37-32/prog_file_handl4.py
[]
>>>
 RESTART: C:/Users/preeti/AppData/Local/Progr
ams/Python/Python37-32/prog_file_handl4.py
['Test program\n']
>>>
```

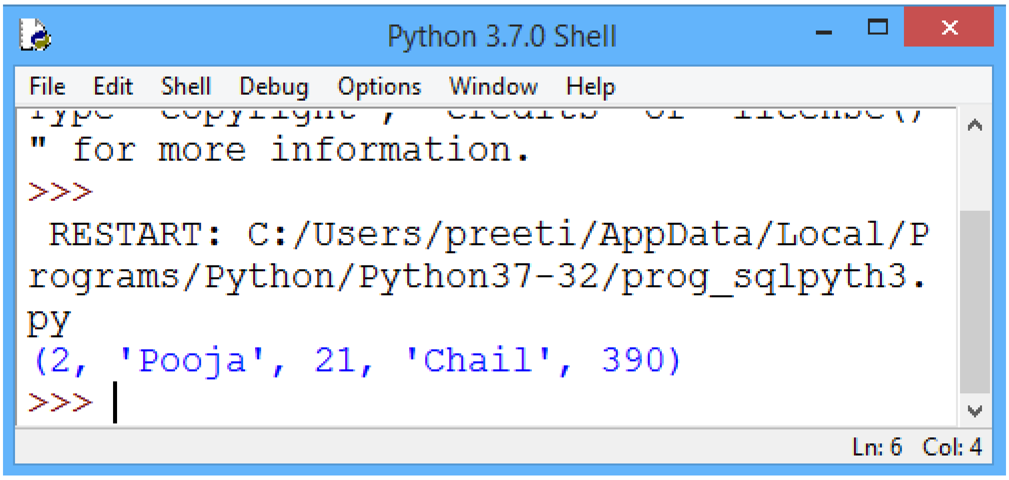


```
Hello User
you are working with
Python
Files

Test program



One line of text here”,           “and another line here”,           “and yet another here”, “and so on and so forth
```

Program 16: Write a Program to show MySQL database connectivity in python.

Solution:

```
import mysql.connector

con=mysql.connector.connect(host='localhost',user='root',password='',db='school')

stmt=con.cursor()

query='select * from student;'

stmt.execute(query)

data=stmt.fetchone()

print(data)
```

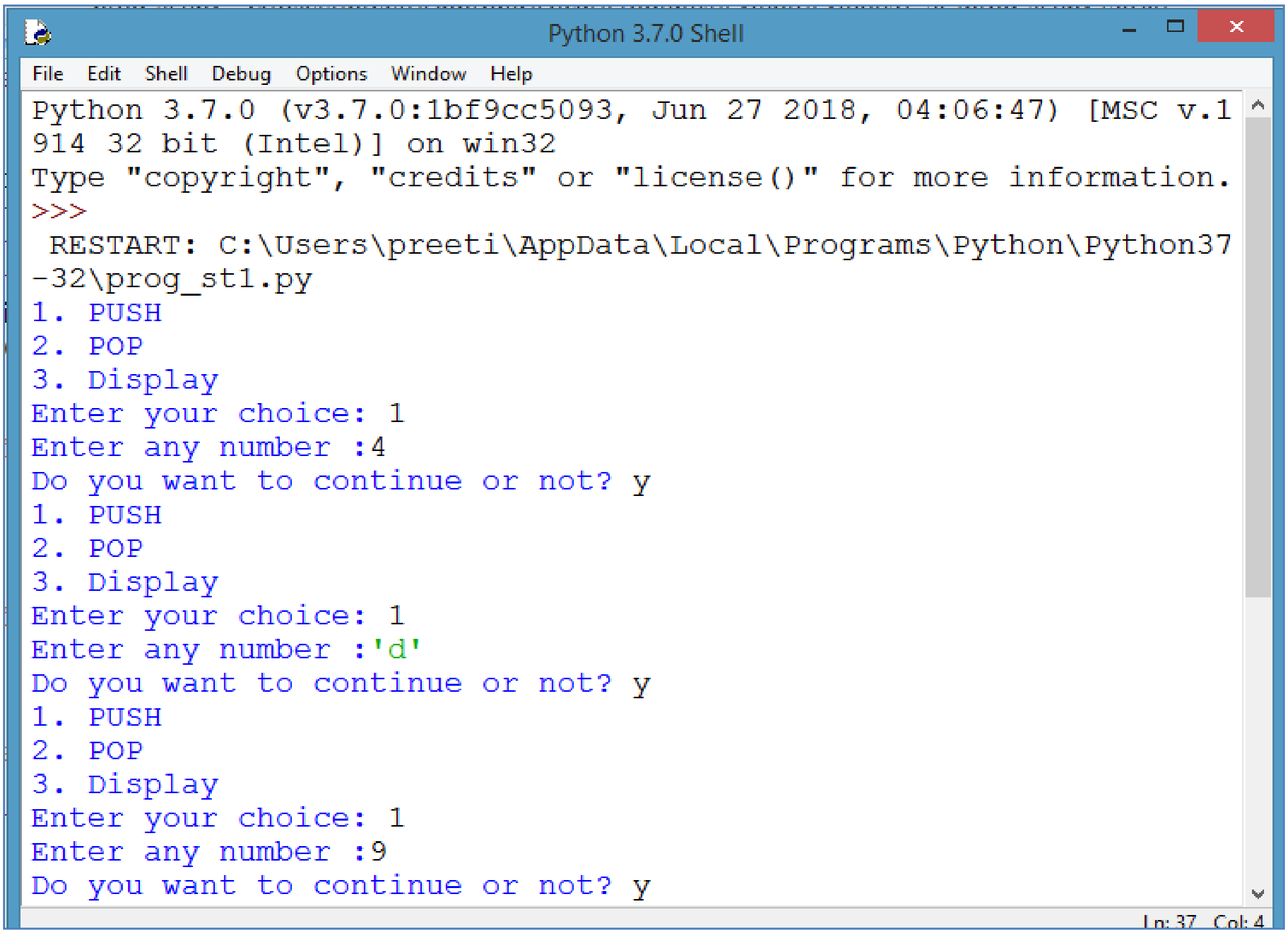


```
Type "copyright", "credits" or "license()
" for more information.
>>>
 RESTART: C:/Users/preeti/AppData/Local/P
rograms/Python/Python37-32/prog_sqlpyth3.
py
(2, 'Pooja', 21, 'Chail', 390)
>>> 
```

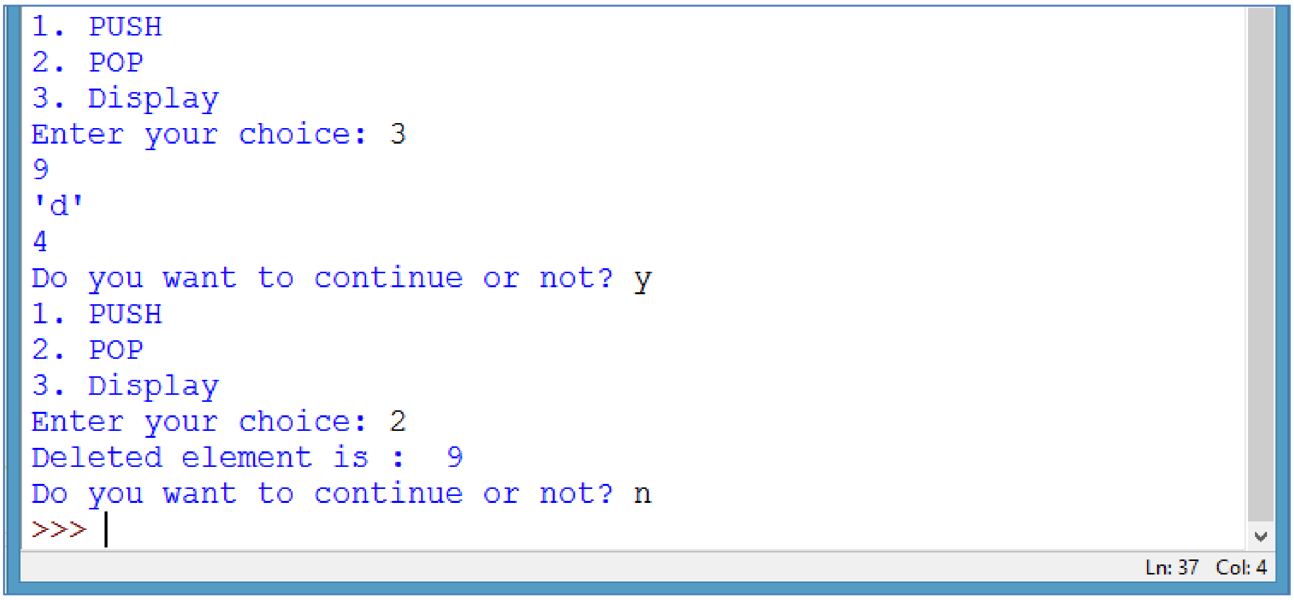
Program 17: Write a Python program to implement all basic operations of a stack, such as adding element (PUSH operation), removing element (POP operation) and displaying the stack elements (Traversal operation) using lists.

Solution:

```
#Implementation of List as stack

s=[]

c="y"

while (c=="y"):

   print ("1. PUSH")
   print ("2. POP ")

   print ("3. Display")

   choice=int(input("Enter your choice: "))

   if (choice==1):

       a=input("Enter any number :")

       s.append(a)

   elif (choice==2):

       if (s==[]):

           print ("Stack Empty")

       else:

           print ("Deleted element is : ",s.pop())

   elif (choice==3):

       l=len(s)

       for i in range(l-1,-1,-1):  #To display elements from last element to first

           print (s[i])

   else:

       print("Wrong Input")

   c=input("Do you want to continue or not? ")
```



```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1
914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
 RESTART: C:\Users\preeti\AppData\Local\Programs\Python\Python37
-32\prog_st1.py
1. PUSH
2. POP
3. Display
Enter your choice: 1
Enter any number :4
Do you want to continue or not? y
1. PUSH
2. POP
3. Display
Enter your choice: 1
Enter any number :'d'
Do you want to continue or not? y
1. PUSH
2. POP
3. Display
Enter your choice: 1
Enter any number :9
Do you want to continue or not? y
```

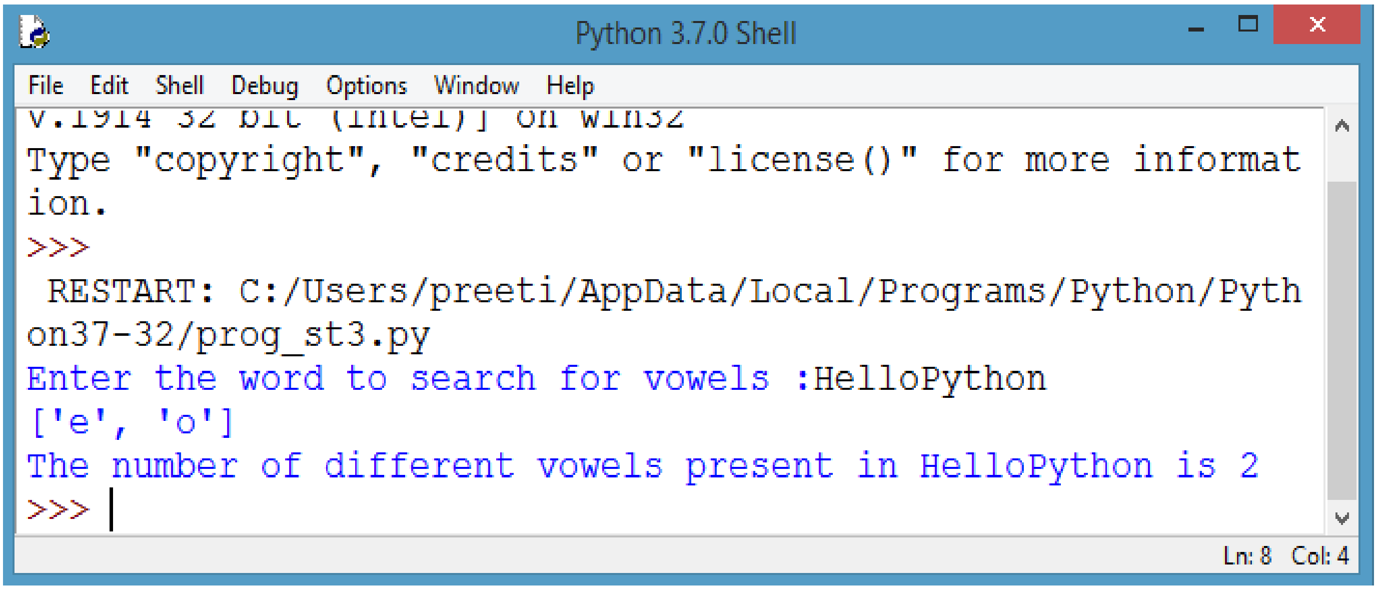


```
1. PUSH
2. POP
3. Display
Enter your choice: 3
9
'd'
4
Do you want to continue or not? y
1. PUSH
2. POP
3. Display
Enter your choice: 2
Deleted element is :  9
Do you want to continue or not? n
>>> 
```

Program 18: Write a program to display unique vowels present in the given word using Stack.

Solution:

```
#Program to display unique vowels present in the given word

#using Stack


vowels =['a','e','i','o','u']

word = input("Enter the word to search for vowels :")

Stack = []

for letter in word:

    if letter in vowels:

        if letter not in Stack:

            Stack.append(letter)

print(Stack)

print("The number of different vowels present in",word,"is",len(Stack))
```



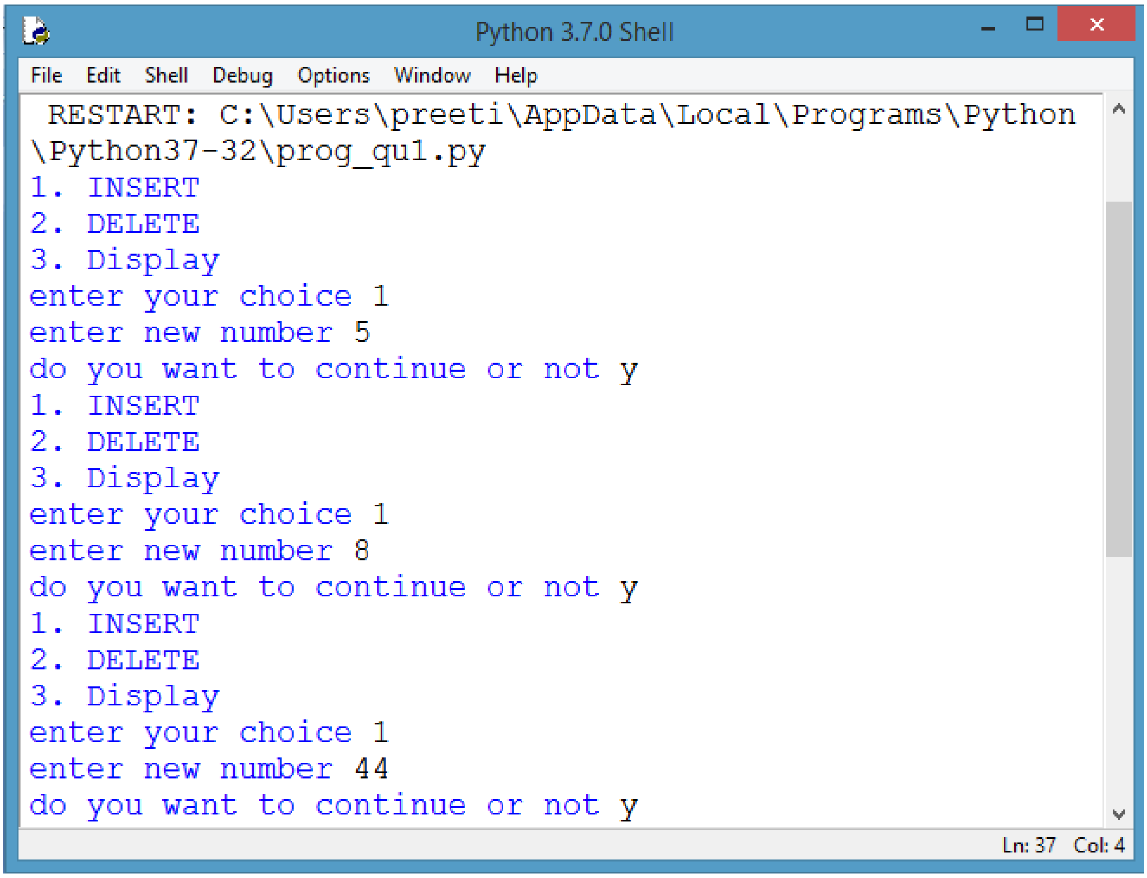
```
v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more informat
ion.
>>>
 RESTART: C:/Users/preeti/AppData/Local/Programs/Python/Pyth
on37-32/prog_st3.py
Enter the word to search for vowels :HelloPython
['e', 'o']
The number of different vowels present in HelloPython is 2
>>> |
```

Program 19: Write a program in Python to add, delete and display elements from a queue using list.

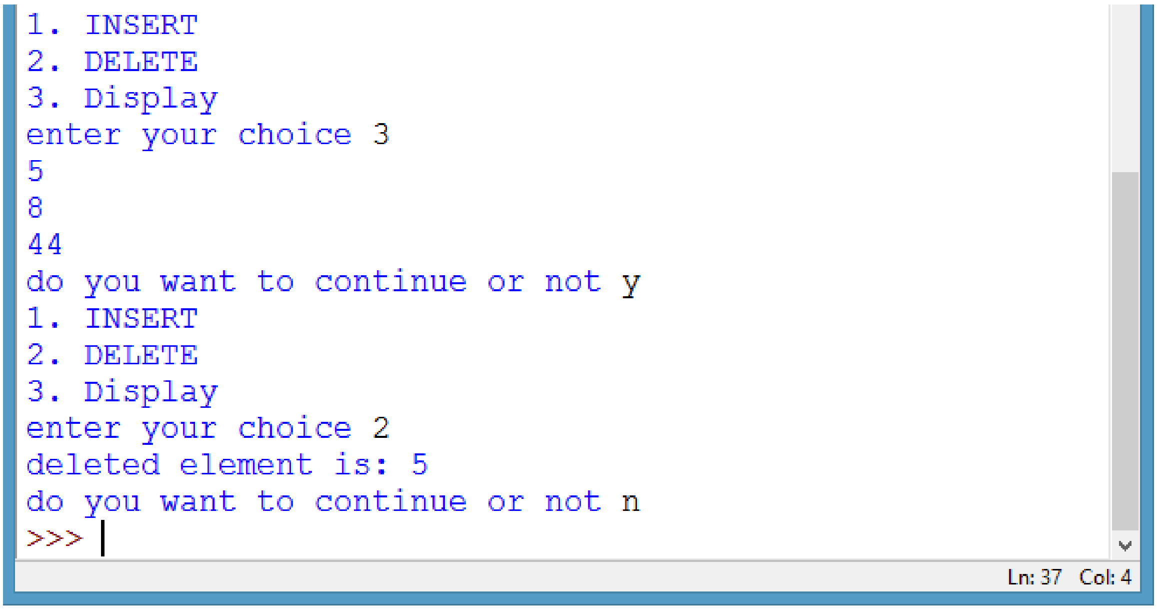
Solution:

```
#Implementing List as a Queue - using function append() and pop()

a=[]

c='y'

while (c=='y'):

   print ("1. INSERT")

   print ("2. DELETE ")

   print ("3. Display")

   choice=int(input("Enter your choice: "))

   if (choice==1):

      b=int(input("Enter new number: "))

      a.append(b)

   elif (choice==2):

      if (a==[]):

         print("Queue Empty")

      else:

         print ("Deleted element is:",a[0])

         a.pop(0)

   elif (choice==3):

      l=len(a)

      for i in range(0,l):

         print (a[i])

   else:

      print("wrong input")
```

c=input("Do you want to continue or not: ")



```
 RESTART: C:\Users\preeti\AppData\Local\Programs\Python
\Python37-32\prog_qu1.py
1. INSERT
2. DELETE
3. Display
enter your choice 1
enter new number 5
do you want to continue or not y
1. INSERT
2. DELETE
3. Display
enter your choice 1
enter new number 8
do you want to continue or not y
1. INSERT
2. DELETE
3. Display
enter your choice 1
enter new number 44
do you want to continue or not y
```

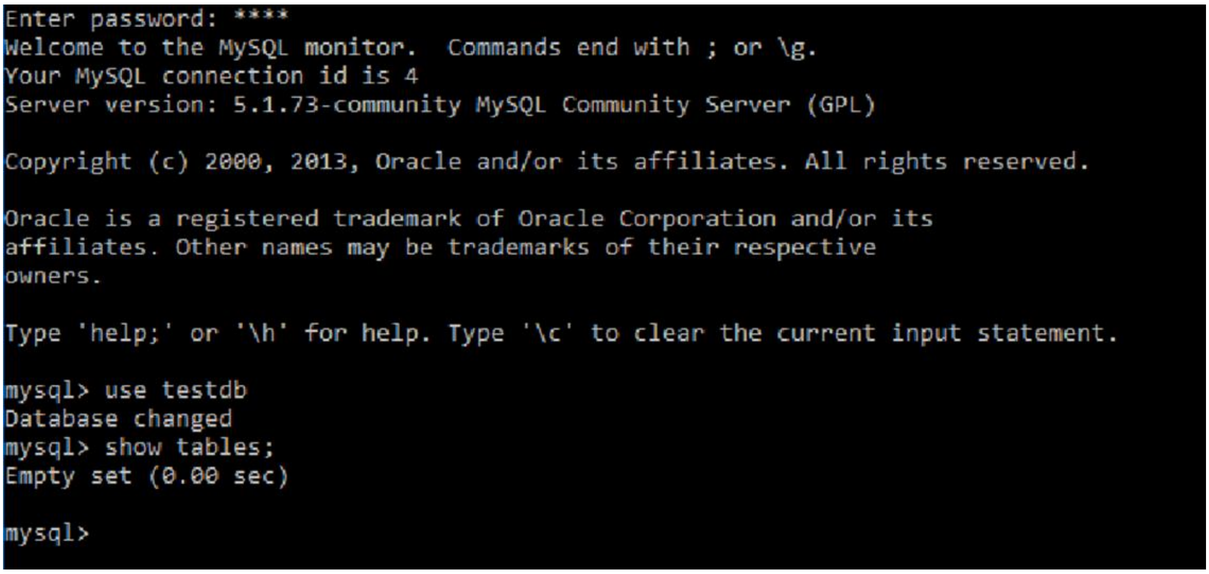


```
1. INSERT
2. DELETE
3. Display
enter your choice 3
5
8
44
do you want to continue or not y
1. INSERT
2. DELETE
3. Display
enter your choice 2
deleted element is: 5
do you want to continue or not n
>>> 
```

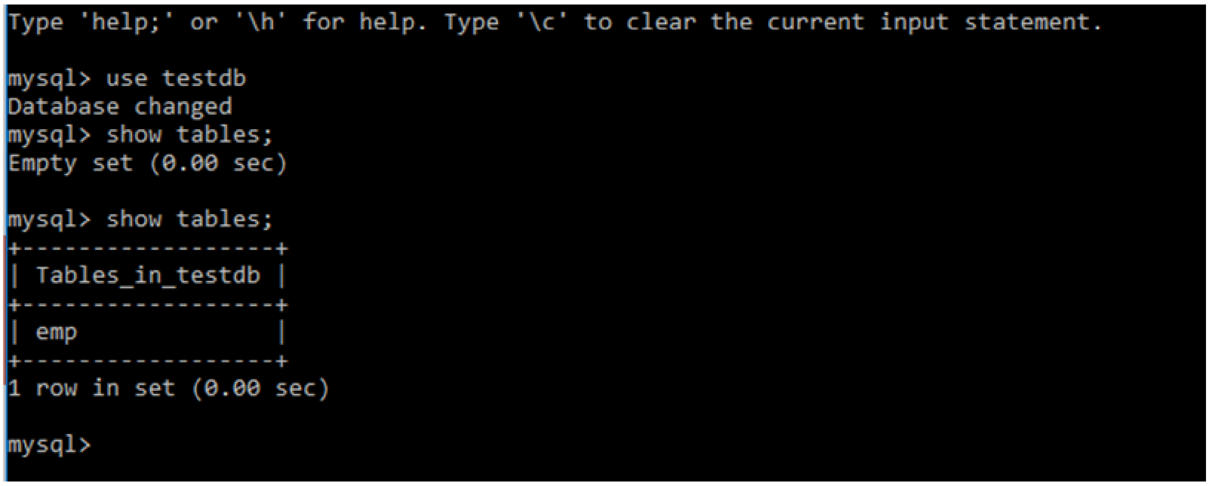
Program 20: Perform all the operations with reference to table ‘Employee’ through MySQL-Python connectivity.

Solution:

```
import MySQLdb

# Using connect method to connect database

db1 = MySQLdb.connect("localhost","root","","TESTDB" )

# using cursor() method for preparing cursor

cursor = db1.cursor()

# Preparing SQL statement to create EMP table

sql = "CREATE TABLE EMP(empno integer primary key,ename  varchar(25) not null,salary float);"

cursor.execute(sql)

# disconnect from server

db1.close()
```



```
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.1.73-community MySQL Community Server (GPL)

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use testdb
Database changed
mysql> show tables;
Empty set (0.00 sec)

mysql>
```

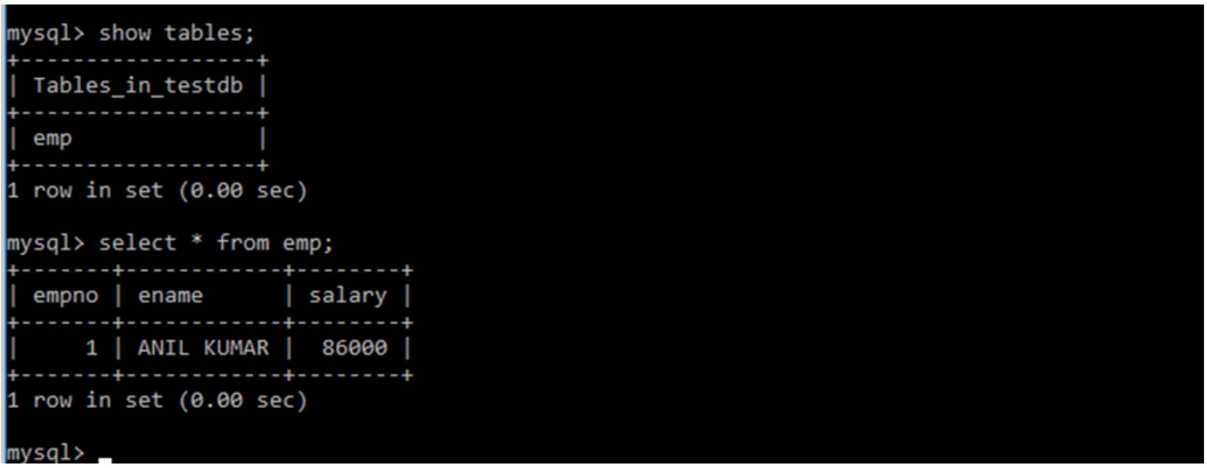


```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use testdb
Database changed
mysql> show tables;
Empty set (0.00 sec)

mysql> show tables;
+-----------------+
| Tables_in_testdb |
+-----------------+
| emp              |
+-----------------+
1 row in set (0.00 sec)

mysql>
```

Inserting a record in ‘emp’

```
import MySQLdb

db1 = MySQLdb.connect("localhost","root","","TESTDB" )

cursor = db1.cursor()

# Prepareing SQL statement to insert one record with the given values

sql = "INSERT INTO EMP VALUES (1,'ANIL KUMAR',86000);"

try:

   cursor.execute(sql)

   db1.commit()
```

```
except:

  db1.rollback()

db1.close()
```



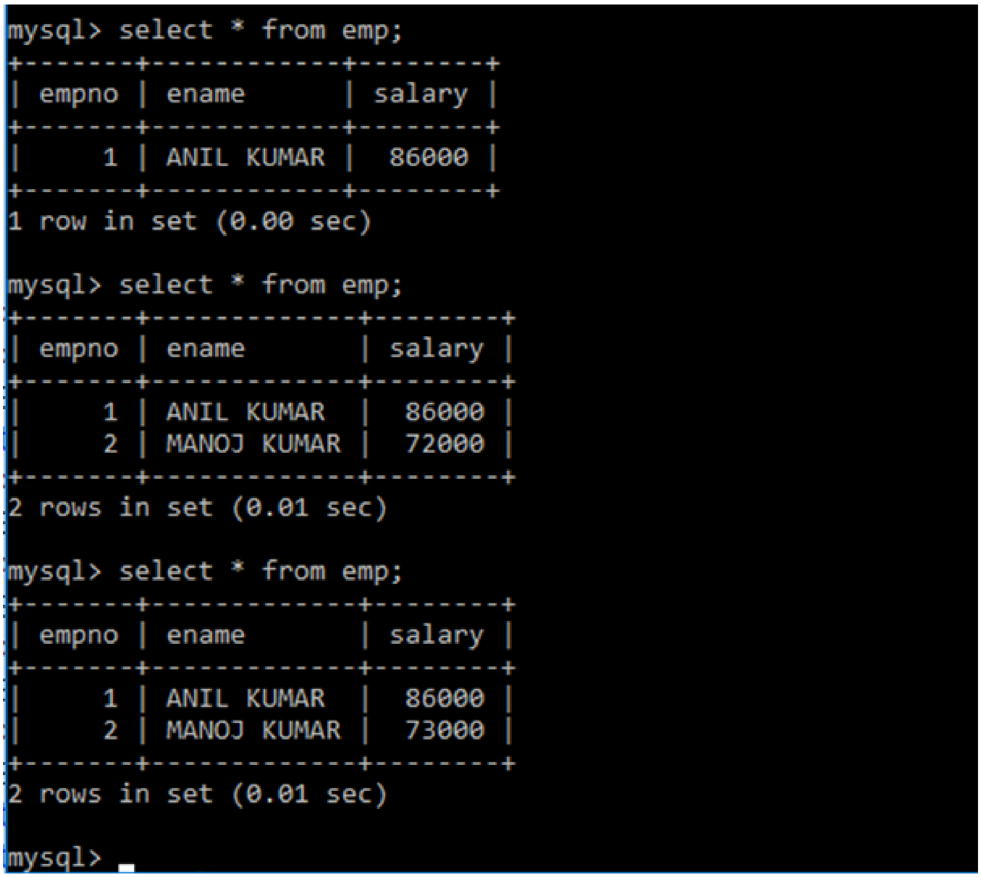
```
mysql> show tables;
+------------------+
| Tables_in_testdb |
+------------------+
| emp              |
+------------------+
1 row in set (0.00 sec)

mysql> select * from emp;
+-------+------------+--------+
| empno | ename      | salary |
+-------+------------+--------+
|     1 | ANIL KUMAR |  86000 |
+-------+------------+--------+
1 row in set (0.00 sec)

mysql> _
```

Fetching all the records from EMP table having salary more than 70000.

```
import MySQLdb

db1 = MySQLdb.connect("localhost","root","","TESTDB" )

cursor = db1.cursor()

sql = "SELECT * FROM EMP WHERE SALARY > 70000;"

try:

    cursor.execute(sql)

    #using fetchall() function to fetch all records from the table EMP and store in resultset

    resultset = cursor.fetchall()

for row in resultset:



     print (row)

except:

  print ("Error: unable to fetch data")

db1.close()
```

Updating record(s) of the table using UPDATE

```
import MySQLdb

db1 = MySQLdb.connect("localhost","root","","TESTDB" )

cursor = db1.cursor()

#Preparing SQL statement to increase salary of all employees whose salary is less than 80000

sql = "UPDATE EMP SET salary = salary +1000 WHERE salary<80000;"

try:

    cursor.execute(sql)


    db1.commit()

except:


   db1.rollback()

db1.close()
```



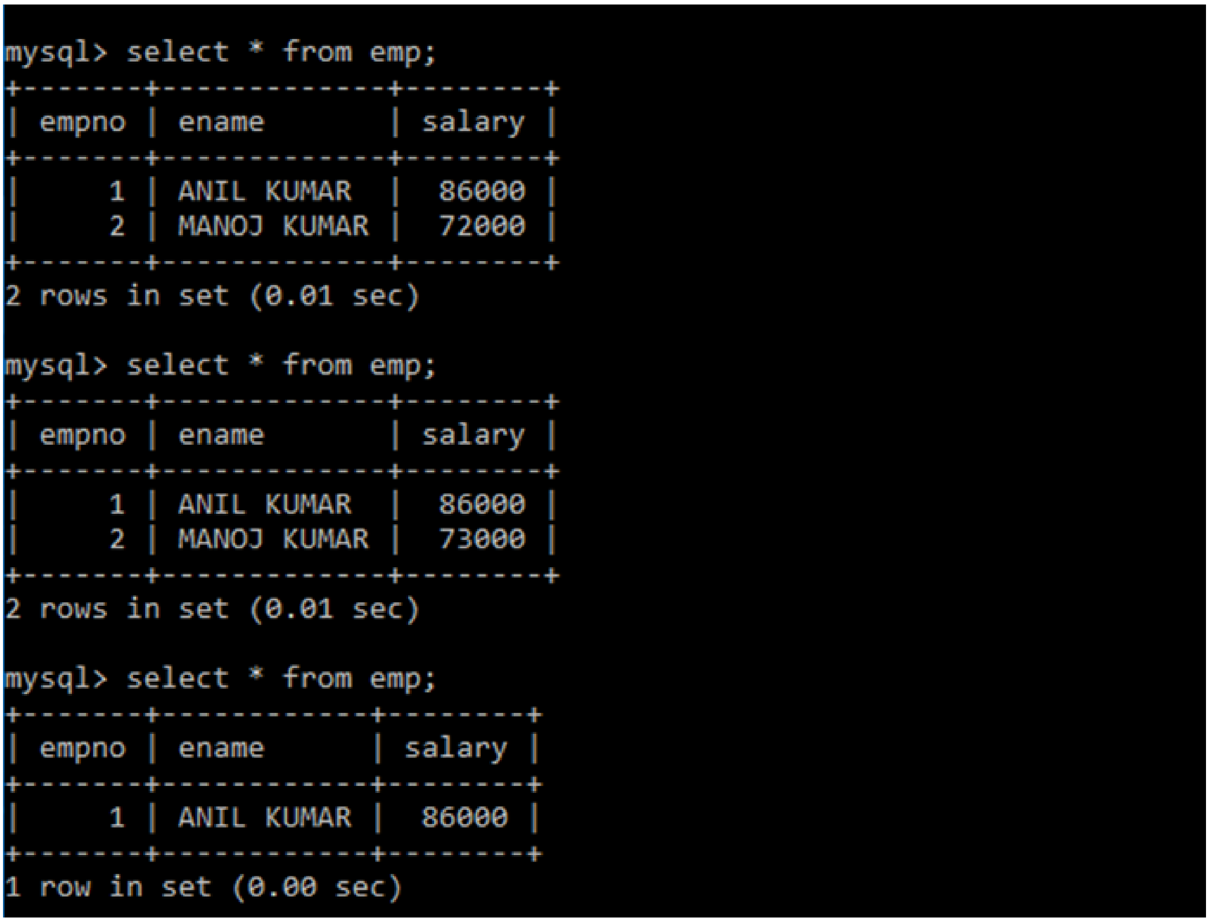
```
mysql> select * from emp;
+-------+------------+--------+
| empno | ename      | salary |
+-------+------------+--------+
|     1 | ANIL KUMAR |  86000 |
+-------+------------+--------+
1 row in set (0.00 sec)

mysql> select * from emp;
+-------+-------------+--------+
| empno | ename       | salary |
+-------+-------------+--------+
|     1 | ANIL KUMAR  |  86000 |
|     2 | MANOJ KUMAR |  72000 |
+-------+-------------+--------+
2 rows in set (0.01 sec)

mysql> select * from emp;
+-------+-------------+--------+
| empno | ename       | salary |
+-------+-------------+--------+
|     1 | ANIL KUMAR  |  86000 |
|     2 | MANOJ KUMAR |  73000 |
+-------+-------------+--------+
2 rows in set (0.01 sec)

mysql> _
```

Deleting record(s) from table using DELETE

```
import MySQLdb

db1 = MySQLdb.connect("localhost","root","","TESTDB" )

cursor = db1.cursor()

sal=int(input("Enter salary whose record to be deleted : "))

#Preparing SQL statement to delete records as per given condition

sql = "DELETE FROM EMP WHERE salary =sal”

try:

    cursor.execute(sql)

    print(cursor.rowcount, end=" record(s) deleted ")

    db1.commit()

except:

     db1.rollback()

db1.close()
```

Output:

```
>>> Enter salary whose record to be deleted: 80000

1 record(s) deleted

>>>
```



```
mysql> select * from emp;
+-------+------------+--------+
| empno | ename      | salary |
+-------+------------+--------+
|     1 | ANIL KUMAR |  86000 |
|     2 | MANOJ KUMAR |  72000 |
+-------+------------+--------+
2 rows in set (0.01 sec)

mysql> select * from emp;
+-------+------------+--------+
| empno | ename      | salary |
+-------+------------+--------+
|     1 | ANIL KUMAR |  86000 |
|     2 | MANOJ KUMAR |  73000 |
+-------+------------+--------+
2 rows in set (0.01 sec)

mysql> select * from emp;
+-------+-----------+--------+
| empno | ename     | salary |
+-------+-----------+--------+
|     1 | ANIL KUMAR |  86000 |
+-------+-----------+--------+
1 row in set (0.00 sec)
```

VIVA VOCE

1. What is Python? What are the benefits of using Python?

Ans. Python is a programming language with objects, modules, threads, exceptions and automatic memory management. The benefits of Python are that it is simple and easy, portable, extensible, built-in data structure and is open source.

2. What is pickling and unpickling?

Ans. Pickle module accepts any Python object and converts it into a string representation and dumps it into a file by using dump function. This process is called pickling. The process of retrieving original Python objects from the stored string representation is called unpickling.

3. How is Python interpreted?

Ans. Python language is an interpreted language. Python program runs directly from the source code. It converts the source code that is written by the programmer into an intermediate language, which is again translated into machine language that has to be executed.

4. How is memory managed in Python?

Ans. Python memory is managed by Python private heap space. All Python objects and data structures are located in a private heap. The programmer does not have access to this private heap and the interpreter takes care of this Python private heap.

- The allocation of Python heap space for Python objects is done by Python memory manager. The core API gives access to some tools for the programmer to code.
- Python also has an inbuilt garbage collector, which recycles all the unused memory, frees up memory, and makes it available to the heap space.

5. What is the difference between list and tuple?

Ans. The difference between list and tuple is that list is mutable while tuple is not. Tuple can be further implemented as a key to dictionaries.

6. What are the built-in types that Python provides?

Ans. There are mutable and immutable types of Python built-in types. Mutable built-in types offered by Python are:

- List
- Sets
- Dictionaries

Immutable built-in types are:

- Strings
- Tuples
- Numbers

7. What is namespace in Python?

Ans. In Python, every name introduced has a place where it lives and can be looked for. This is known as namespace. It is like a box where a variable name is mapped to the object placed. Whenever the variable is searched, this box will be searched to get the corresponding object.

8. What is lambda in Python?

Ans. It is a single expression, anonymous function often used as inline function.

9. What is pass in Python?

Ans. Pass means no-operation Python statement or, in other words, it is a placeholder in compound statement, where there should be a blank left and nothing should be written there.

10. What is slicing in Python?

Ans. A mechanism to select a range of items from sequence types like list, tuple, strings, etc., is known as slicing.

11. What is docstring in Python?

Ans. A Python documentation string is known as docstring. It is a way of documenting Python functions, modules and classes.

12. What is negative index in Python?

Ans. Python sequences can be indexed using both the positive and negative numbers. For positive index, 0 is the first index, 1 is the second index, so on and so forth. For negative index, (–1) is the last index and (–2) is the second last index and so on and so forth.

13. How can you convert a number into a string?

Ans. In order to convert a number into a string, use the inbuilt function str(). If you want an octal or hexadecimal representation, use the inbuilt function oct() or hex().

14. What do you understand by module and package in Python?

Ans. In Python, module is the way to structure a program. Each Python program file is a module which imports other modules like objects and attributes.

The folder of a Python program is a package of modules. A package can have modules or sub-folders.

15. What are the rules for local and global variables in Python?

Ans. **Local variables:** If a variable is assigned a new value anywhere within a function's body, it is assumed to be local.

Global variables: Those variables that are only referenced inside a function are implicitly global.

16. Explain how to delete a file in Python.

Ans. A file can be deleted by using a command os.remove (filename) or os.unlink(filename)

17. Explain how you can generate random numbers in Python.

Ans. To generate random numbers in Python, you need to import command as:

```
import random
random.random()
```

This returns a random floating point number in the range [0,1)

18. What is the use of // operator in Python?

Ans. It is a Floor Division operator which is used for dividing two operands with the result as quotient showing only digits before the decimal point. For instance, 10//5 = 2 and 10.0//5.0 = 2.0.

19. Mention five benefits of using Python.

Ans. (a) Python comprises a huge standard library for most internet platforms like email, HTML, etc.

(b) Python does not require explicit memory management as the interpreter itself allocates memory to new variables and frees them automatically.

(c) Provides easy readability due to the use of square brackets.

(d) Easy-to-learn for beginners.

(e) Having the built-in data types saves programming time and effort from declaring variables.

20. Mention the use of the split function in Python.

Ans. The use of split function in Python is that it breaks a string into shorter strings using the defined separator. It gives a list of all words present in the string.

21. What are literals in Python?

Ans. Python literals can be defined as data which can be assigned to a variable or constant. There are 5 types of literals available in Python:

- String literals
- Numeric literals
- Boolean literals
- Special literals
- Literal Collections

22. Explain Python functions.

Ans. A function is a set of instructions or a block of code that is written once and can be called and executed whenever required in the program. There are two categories of functions in Python:

- Built-in functions
- User-defined functions

23. Name the different file processing modes supported by Python.

Ans. Python provides three modes to work with files:

- Read-only mode
- Write-only mode
- Read-Write mode

24. What is an operator in Python?

Ans. An operator is a particular symbol which is used on some values and produces an output as result.

For example, 10 + 30 = 40

Here, "+" and "=" are operators.

25. What are the different types of operators in Python?

Ans. Following is a list of operators in Python:

- Arithmetic Operators
- Relational Operators
- Assignment Operators
- Logical Operators
- Membership Operators
- Identity Operators
- Bitwise Operators

26. What is a Dictionary in Python?

Ans. Dictionary is an important built-in data type in Python. It defines one-to-one relationship between keys and values. Dictionaries contain a pair of keys and their corresponding values.

Dictionaries are indexed by keys.

27. What is the use of HELP() and DIR() function in Python?

Ans. Both Help() and dir() functions are accessible from the Python interpreter and used for viewing a consolidated collection of built-in functions.

Help(): The help() function is used to display the documentation string and also facilitates you to see the help related to modules, keywords, attributes, etc.

Dir(): The dir() function is used to display the defined symbols.

28. How does Python do compile-time and run-time code checking?

Ans. In Python, some amount of coding is done at compile-time, but most of the checking such as type, name, etc., is held up until the code execution. Consequently, if the Python code references a user-defined function that does not exist, the code will compile successfully. The Python code will fail only with an exception when the code execution path does not exist.

29. Explain the use of TRY: EXCEPT: RAISE: and FINALLY:.

Ans. Try, except and finally blocks are used in Python error-handling mechanism. Code is executed in the try block until an error occurs. Except block is used to receive and handle all errors. Control is transferred to the appropriate except block. In all cases, the finally block is executed. Raise may be used to raise your own exceptions.

30. What is the purpose of PYTHONPATH environment variable?

Ans. PYTHONPATH has a role similar to PATH. This variable tells the Python interpreter where to locate the module files imported into a program. It should include the Python source library directory and the directories containing Python source code.

31. What are the supported data types in Python?

Ans. Python has five standard data types:

- Numbers
- String
- List
- Tuple
- Dictionary

32. What is the difference between lists and tuples?

Ans.

Lists	**Tuples**
Lists are mutable, *i.e.*, they can be edited.	Tuples are immutable (tuples are lists which can't be edited).
Lists are slower than tuples.	Tuples are faster than lists.
Syntax: `list1 = [10, 'Python', 44.5]`	**Syntax:** `tup1 = (10, 'Python' , 44.5)`

33. How will you reverse a list?

Ans. list.reverse() − Reverses items of list in place.

34. What is a string in Python?

Ans. A string in Python is a sequence of alphanumeric characters. They are immutable objects. It means that they don't allow modification once they are assigned a value. Python provides several methods such as join(), replace(), or split() to alter strings.

35. Why is the Return keyword used in Python?

Ans. The purpose of a function is to receive the inputs and return some output.

The return is a Python statement which we can use in a function for sending a value back to its calling function or the operating system.

36. When should you use the "Break" in Python?

Ans. Python provides a break statement to exit from a loop. Whenever the break hits in the code, the control of the program immediately exits from the body of the loop. The break statement in a nested loop causes the control to exit from the inner iterative block.

37. What is a tuple in Python?

Ans. A tuple is a collection of type data structure in Python which is immutable. Tuples are similar to sequences, just like the lists. However, there are some differences between a tuple and a list—the former doesn't allow modifications, the latter does.

Also, the tuples use parentheses for enclosing but the lists have square brackets in their syntax.

38. How do you debug a program in Python? Is it possible to step through the Python code?

Ans. Yes, we can use the Python debugger (**pdb**) to debug any Python program. If we start a program using **pdb**, then it lets us even step through the code.

39. List down some of the PDB commands for debugging Python programs.

Ans. Here are a few PDB commands to start debugging Python code:

- Add breakpoint (b)
- Resume execution (c)
- Step-by-step debugging (s)
- Move to the next line (n)
- List source code (l)
- Print an expression (p)

40. Explain the use of "with" statement.

Ans. In Python, generally "with" statement is used to open a file, process the data present in the file, and also to close the file without calling a close() method. "with" statement makes the exception handling simpler by providing cleanup activities.

General form of with:

with open("filename", "mode") as file-var:

processing statements

41. How can we display the contents of text file in reverse order?

Ans. (a) Convert the given file into a list.

(b) Reverse the list by using reversed()

(c) *Eg*: for line in reversed(list(open("file-name","r")))

(d) print(line)

42. Differentiate between append() and extend() methods.

Ans. Both append() and extend() methods are methods of list. These methods are used to add elements at the end of the list.

- append(element) – adds the given element at the end of the list which has called this method.
- extend(another-list) – adds the elements of another list at the end of the list which is called the extend method.

43. What are the advantages of Python recursion?

Ans. Implementing something using Python recursion requires less effort. The code we write using recursion will be comparatively smaller than the code that is implemented by loops. Again, codes that are written using recursion are easier to understand also.

44. What do you understand by Python modules?

Ans. A file containing Python definitions and statements is called a Python module. So naturally, the filename is the module name which is appended with the suffix **.py**.

45. What do you understand by Python package?

Ans. Python package is a collection of modules in directories that gives a package hierarchy. More elaborately, Python packages are a way of structuring Python's module by using "dotted module names". So A.B actually indicates that B is a sub-module which is under a package named A.

46. How can we get current directory using Python?

Ans. To get current directory in Python, we need to use os module. Then, we can get the location of the current directory by using getcwd() function.

47. What is the difference between del keyword and clear() function?

Ans. The difference between del keyword and clear() function is that while del keyword removes one element at a time, clear function removes all the elements.

48. What is primary key?

Ans. Primary key is a combination of columns that uniquely identifies a row in a relational table.

49. What is candidate key?

Ans. All possible combinations of columns that can possibly serve as the primary key are called candidate keys.

50. What is foreign key?

Ans. A combination of columns where values are derived from primary key of some other table is called the foreign key of the table in which it is contained.

51. What is alternate key?

Ans. A candidate key that is not serving as a primary key is called an alternate key.

52. What is MYSQL?

Ans. MYSQL is an open source RDBMS that relies on SQL for processing the data in the database. The database is available for free under the terms of General Public License (GPL).

53. What is RDBMS?

Ans. Relational Database Management System (RDBMS) facilitates access, security and integrity of data and eliminates data redundancy. *For example*, MYSQL, Oracle, Microsoft Sql Server, etc.

54. What is the use of drop command?

Ans. Drop command is used to delete tables. *For example*, Drop Table Orders. Delete commands are used to delete rows of a table.

55. What do you understand by NOT NULL constraint?

Ans. This constraint ensures that the null values are not permitted on a specified column. This constraint can be defined at the column level and not at the table level.

56. What is the significance of COUNT?

Ans. It is used to count the number of values in a given column or number of rows in a table. *For example*, Select count (Roll No.) from students.

57. How can we delete a table in MYSQL?

Ans. We can delete a table in MYSQL using the drop command.

58. How can we delete a record in MYSQL?

Ans. We can delete a record in MYSQL using the delete command.

59. How can we add a record and add a column in a table?

Ans. We can add a record by using insert command and we can add a column through the alter table command.

60. Give any two differences between GET and POST submission methods of HTML form.

Ans.

GET Method	**POST Method**
All form data is encoded into the URL appended to the action URL as query string parameters.	Form data appears within the message body of the HTTP request.
Parameters remain in browser history, hence cannot be used to send password-like sensitive information.	Parameters are not saved in browser history, hence can be used to send sensitive information.
Can be bookmarked.	Cannot be bookmarked.
Easier to hack for script kiddies.	More difficult to hack.
Can be cached.	Cannot be cached.

61. Give the necessary command to incorporate SQL interface within Python.

Ans. import MySQLdb

62. What is Django?

Ans. Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Developed by a fast-moving online news operation, Django was designed to handle two challenges: the intensive deadlines of a newsroom and the stringent requirements of the experienced web developers who wrote it. It lets you build high-performing, elegant web applications quickly.