# USPC JAIN PUBLIC SCHOOL




## ACADEMIC YEAR: 2026 - 27

# PRACTICAL FILE OF COMPUTER SCIENCE

## SUBJECT CODE: 083

**SUBMITTED BY:**
**NAME :**
**ROLL NO. :**

**SUBMITTED TO:**
**Mr. SUMIT SADANA**

**CLSS & SEC. :**

# ***ACKNOWLEDGEMENT***

I would like to express my special thanks of gratitude to my teacher **Mr. Sumit Sadana** as well as our principal ma'am **Ms. Vinny Malhotra** who gave me this golden opportunity to make this informative practical file of **Computer Science (083)**, which also helped me in doing a lot of research and I came to know about so many new things I am really thankful to them. Secondly, I would like to thank my parents and friends who helped me finalising this practical file within the stipulated time.

**STUDENT NAME**

# ***CERTIFICATE***

**This is to certify that the Practical file has been completed by _____ of class XII C of subject Computer science (083) laid down in the regulations of CBSE for Board Practical Examination of class XII to be held in USPC JAIN PUBLIC SCHOOL ON _____. The practical file is the result of his/her efforts & endeavours.**

**…….** **Date:**

**…….**

**(Signature of the teacher)**

**Examiner:**

**Name: _____**

**Signature_____**

**Date:_____**

# INDEX


| SR.NO. | TOPIC | PAGE NO. |
|---|---|---|
| 1. | OVERVIEW OF PYTHON | 5 |
| 2. | OVERVIEW OF SQL | 6 |
| 3. | SYSTEM REQUIREMENTS AND SPECIFICATIONS | 7 |
| 4. | Program to enter two numbers and print the arithmetic operations like +,-,*, /, // and %. | 8 |
| 5. | Write a program to find whether an inputted number is perfect or not. | 10 |
| 6. | Write a Program to check if the entered number is Armstrong or not. | 11 |
| 7. | Write a Program to find factorial of the entered number. | 12 |
| 8. | Write a Program to enter the number of terms and to print the Fibonacci Series. | 12 |
| 9. | Write a Program to enter the string and to check if it's palindrome or not using loop. | 13 |
| 10. | Write a Program to show the outputs based on entered list. | 14 |
| 11. | Write a Program to enter the numbers in a list using split () and to use all the functions related to list. | 15 |
| 12. | Write a Program to enter the number and print the Floyd's Triangle in decreasing order. | 16 |
| 13. | Write a Program to find factorial of entered number using user-defined module fact(). | 17 |
| 14. | Write a Program to enter the numbers and find Linear Search, Binary Search, Lowest Number and Selection Sort using list/array code. | 18 |
| 15. | Write a Program to read data from data file and show Data File Handling related functions utility in python. | 25 |
| 16. | Write a Program to read data from data file in append mode and use writeLines function utility in python. | 26 |
| 17. | 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. | 27 |
| 18. | 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. | 28 |
| 19. | Write a Program to show MySQL database connectivity in python. | 29 |
| 20. | 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. | 30 |

| | | |
|---|---|---|
| 21. | Write a program to display unique vowels present in the given word using Stack. | 32 |
| 22. | Write a program in Python to add, delete and display elements from a queue using list. | 33 |
| 23. | Perform all the operations with reference to table 'Employee' through MySQL-Python connectivity. | 35 |


# OVERVIEW OF PYTHON

**Python is a general purpose, dynamic, high-level, and interpreted programming language. It supports Object Oriented programming approach to develop applications. It is simple and easy to learn and provides lots of high-level data structures Guido Van Rossum is known as the founder of Python programming.**

## Features of Python:

- **Python is a high-level language. It is a free and open source language.**
- **It is an interpreted language, as Python programs are executed by an interpreter**
- **Python programs are easy to understand as they have a clearly defined syntax and relatively simple structure.**
- **Python is case-sensitive. For example, NUMBER and number are not same in Python.**
- **Python is portable and platform independent, means it can run on various operating systems and hardware platforms.**
- **Python has a rich library of predefined functions.**
- **Python is also helpful in web development. Many popular web services and applications are built using Python.**
- **Python uses indentation for blocks and nested blocks.**

# OVERVIEW OF MYSQL

**MySQL is a relational database management system (RDBMS). It is pronounced as "My Sequel". MySQL was originally founded and developed in Sweden by David Axmark, Allan Larsson and Michael Widenius, who had worked together since the 1980s.**

## Characteristics of MySQL:

- **MySQL is released under an open-source license so it is customizable.**
- **It requires no cost or payment for its usage.**
- **MySQL has superior speed, is easy to use and is reliable.**
- **MySQL uses a standard form of the well-known ANSI-SQL standards.**
- **MySQL is a platform independent application which works on many operating systems like**
- **Windows, UNIX, LINUX etc. and have compatibility with many languages including JAVA, C+ty PHP, PERL, etc.**
- **MySQL is an easy to install RDBMS and is capable of handling large data sets.**

  **Since MySQL is released under an open-source license, it does not require any cost or payment for its usage. Anyone can download and use this software from specific location on Internet.**

# SYSTEM REQUIREMENTS AND SPECIFICATIONS

## Software Requirements:

| | |
|---|---|
| **OPERATING SYSTEM** | **- Window-7 and later version** |
| **LANGUAGE** | **-Python** |
| **PLATEFORM** | **-Python IDLE 3.7 (min)** |
| **DATABASE** | **-MySQL** |
| **DATABASE DRIVER** | **-MySQL Connector** |

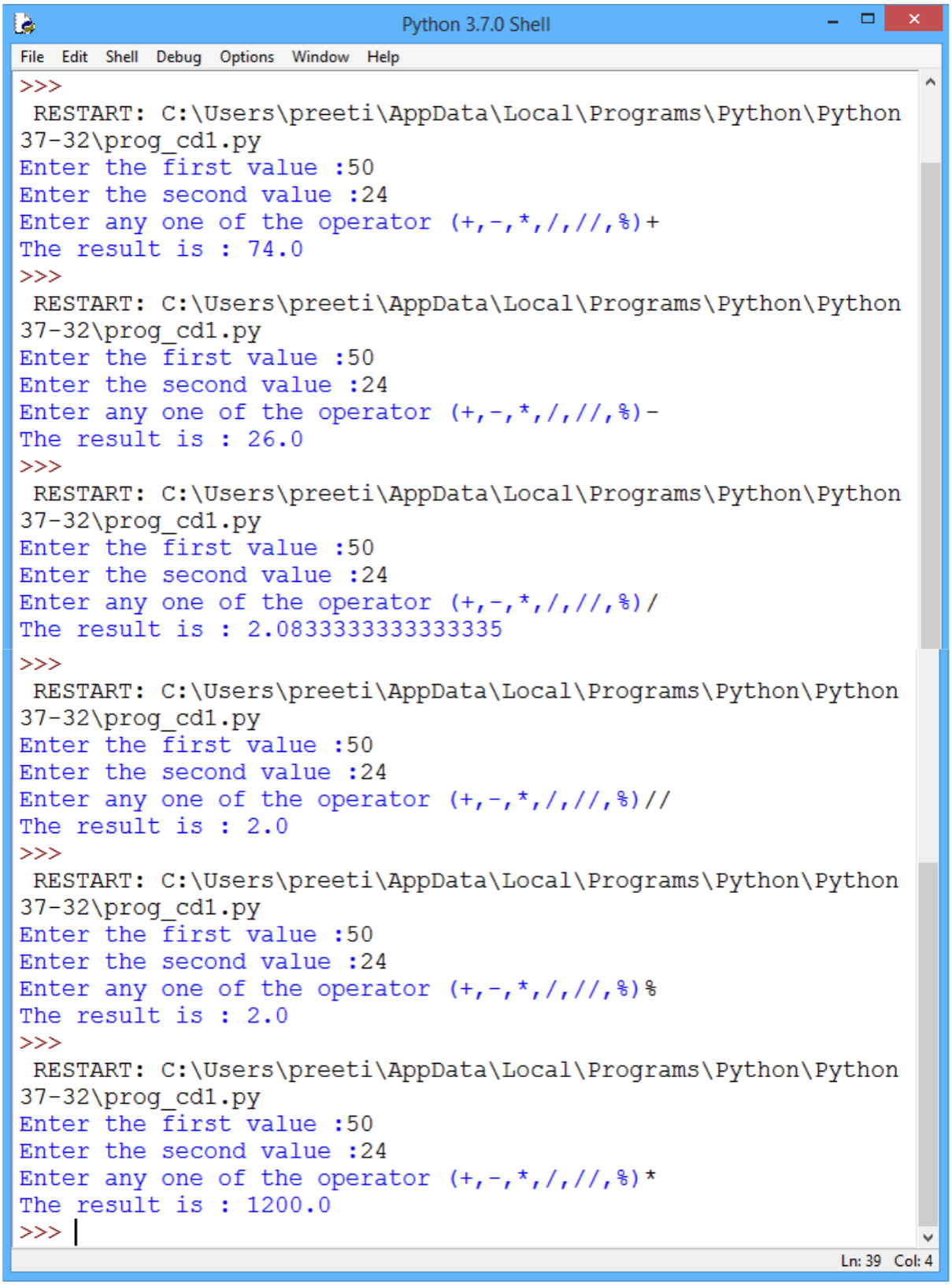
## Hardware Requirements:

| | |
|---|---|
| **PROCESSOR** | **- Pentium Dual Core(min) 32 bit** |
| **HARD DISK** | **-160GB (min)** |
| **RAM** | **-1GB (min)** |

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

Solution:

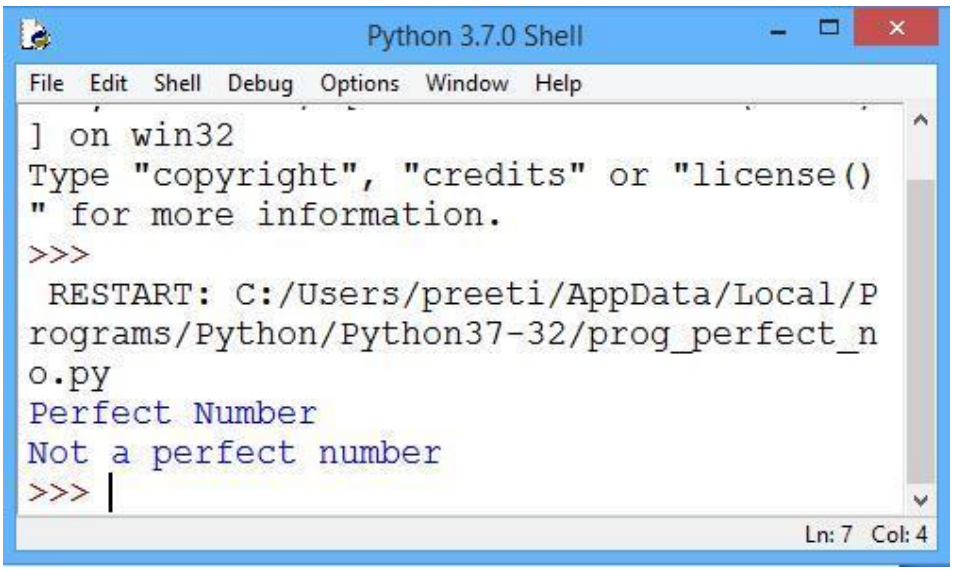
```
#Program for Arithmetic Calculator
result = 0
val1 = float(input("Enter the first value :"))
val2 = float(input("Enter the second value :"))
op = input("Enter any one of the operator (+,-,*,/,//,%)")
if op == "+":
result = val1 + val2
elif op == "-":
result = val1 - val2
elif op == "*":
result = 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)
```


Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
>>>
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
>>>
Ln: 39 Col: 4

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

Solution:

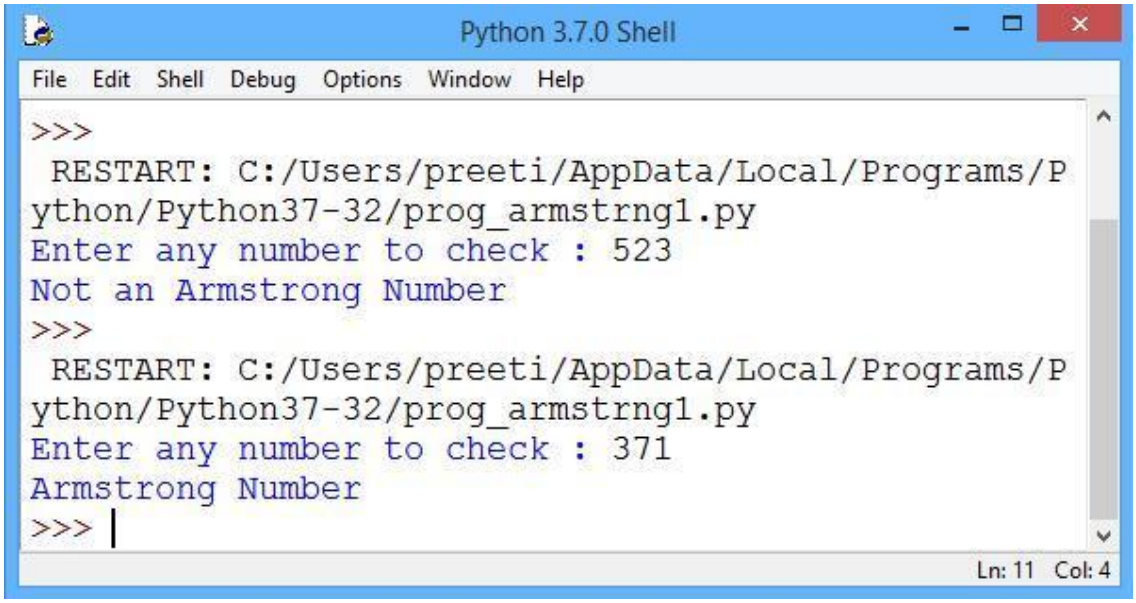
```
#       To find whether a number is perfect or not def pernum(num):
divsum=0
for i in range(1,num): if num%i == 0:
      divsum+=i if divsum==num:
   print('Perfect Number') else:
print('Not a perfect number')
pernum(6)
pernum(15)
```



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

Solution:

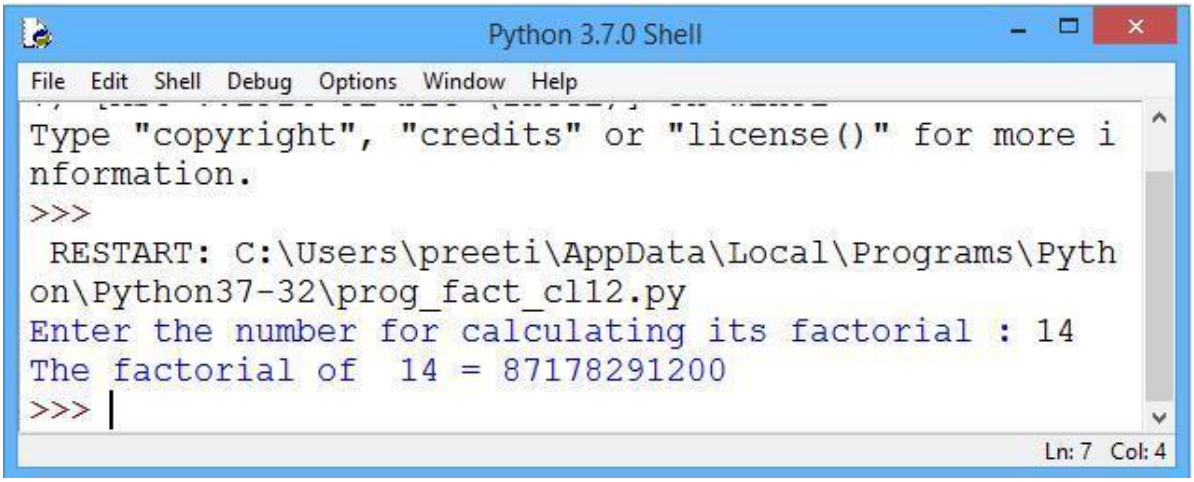
```
# Program to check if the entered number is Armstrong or not.
#An Armstrong number has sum of the cubes of its digits is equal to the number itself
no=int(input("Enter any number to check : ")) no1 = no
sum = 0
while(no>0):
ans = no % 10;
sum = sum + (ans * ans * ans)
no = int (no / 10)
if sum == no1:
print("Armstrong Number")
else:
print("Not an Armstrong Number")
```



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

Solution:

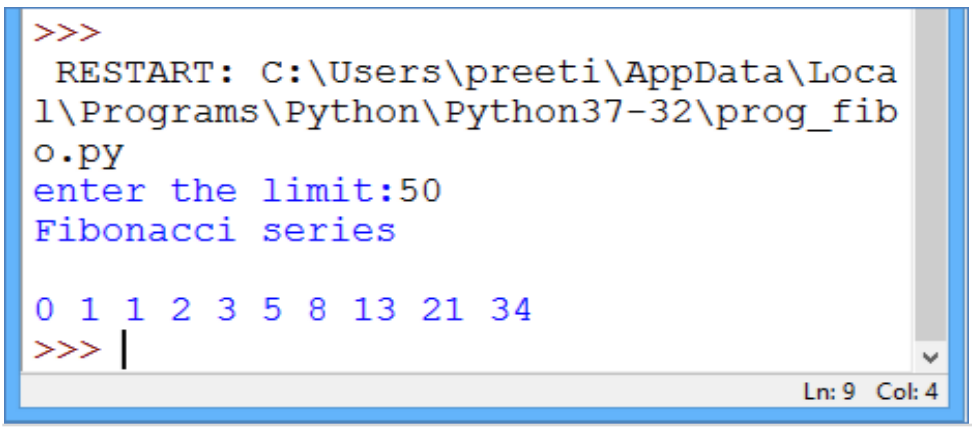
```
#Program to calculate the factorial of an inputted number (using while loop)
num = int(input("Enter the number for calculating its factorial : "))
fact = 1
i = 1
while i<=num:
fact = fact*i
i = i + 1
print("The factorial of ",num,"=",fact)
```




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

Solution:

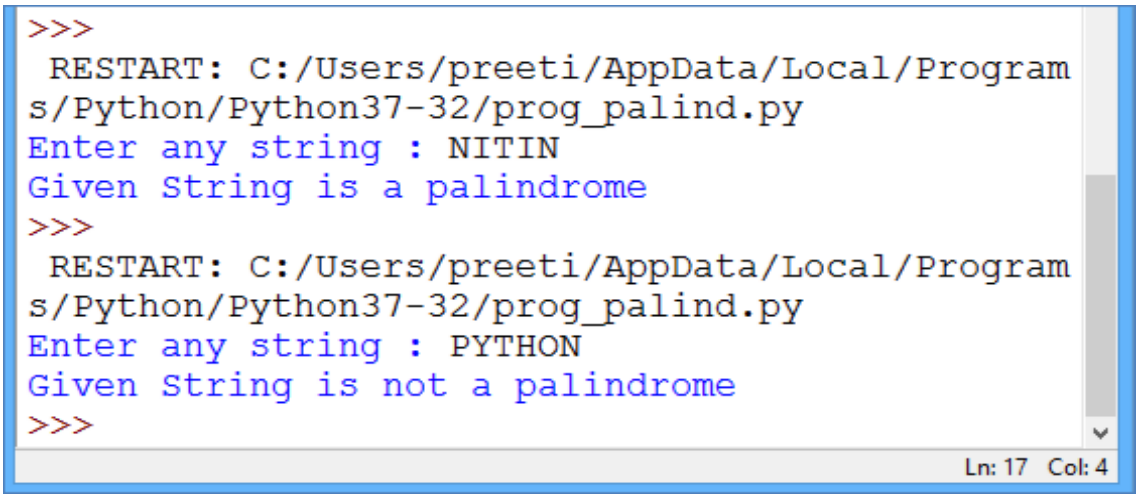
```
#fibonacci
i =int(input("enter the limit:"))
x = 0
y = 1
z = 1
print("Fibonacci series \n")
print(x, y,end= " ")
while(z<= i):
print(z, end=" ")
x = y
y = z
z = x + y
```



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

Solution:

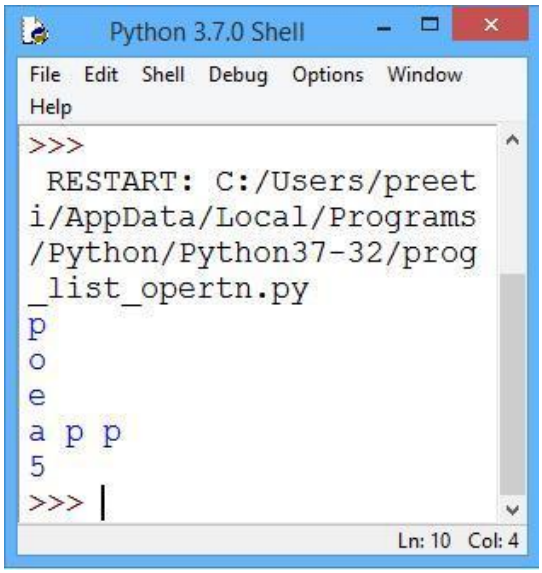
```
# Program to enter the string and check if it's palindrome or not using 'for' loop.
msg=input("Enter any string : ")
newlist=[]
newlist[:0]=msg
l=len(newlist)
ed=l-1
for i in range(0,l):
if newlist[i]!=newlist[ed]:
print ("Given String is not a palindrome")
break
if i>=ed:
print ("Given String is a palindrome")
break
l=l-1
ed = ed - 1
```



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

Solution:

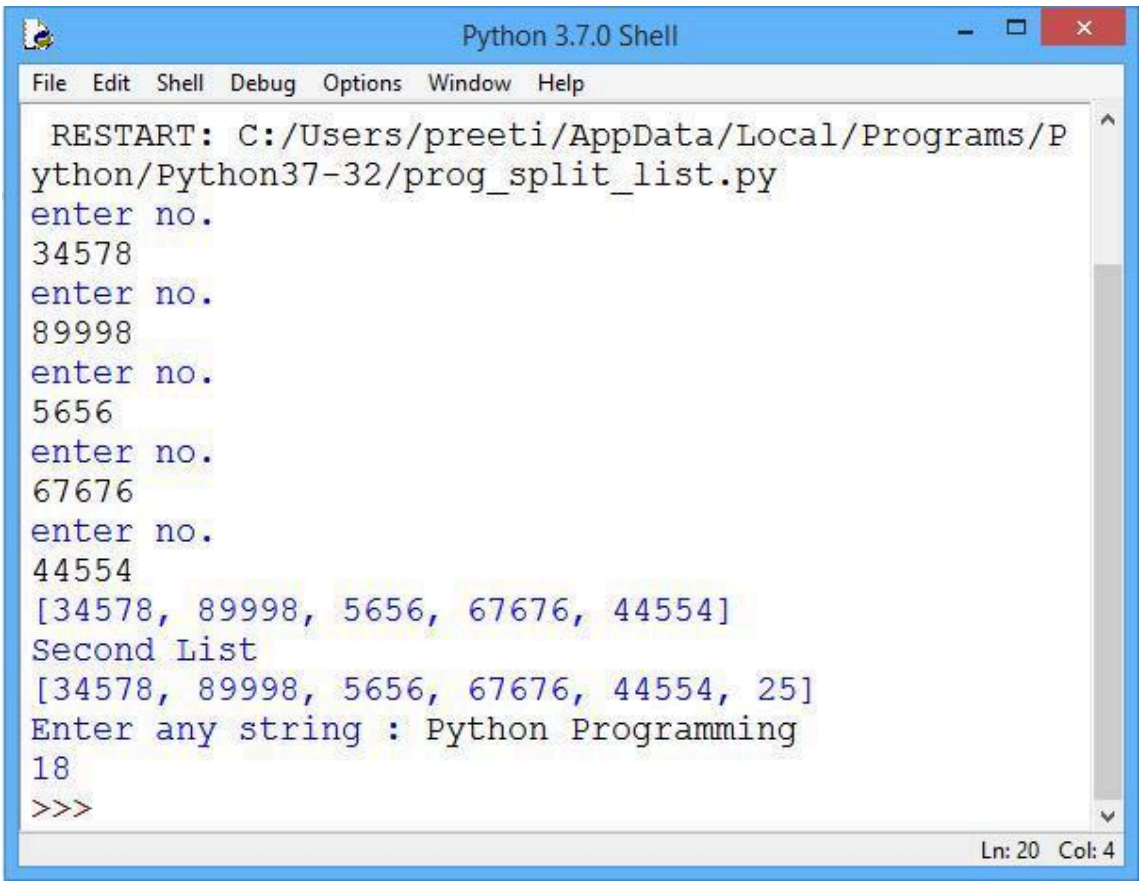
```
my_list = ['p','r','o','b','e']
#         Output: p print(my_list[0])
#         Output: o print(my_list[2])
#         Output: e print(my_list[4])
# Error! Only integer can be used for indexing
# my_list[4.0]
# Nested List
  n_list = ["Happy", [2,0,1,5]]
# Nested indexing
# Output: a
  print(n_list[0][1],n_list[0][2],n_list[0][3])
  #         Output: 5 print(n_list[1][3])
```



**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)
```



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

Solution:

```
#Floyd's triangle

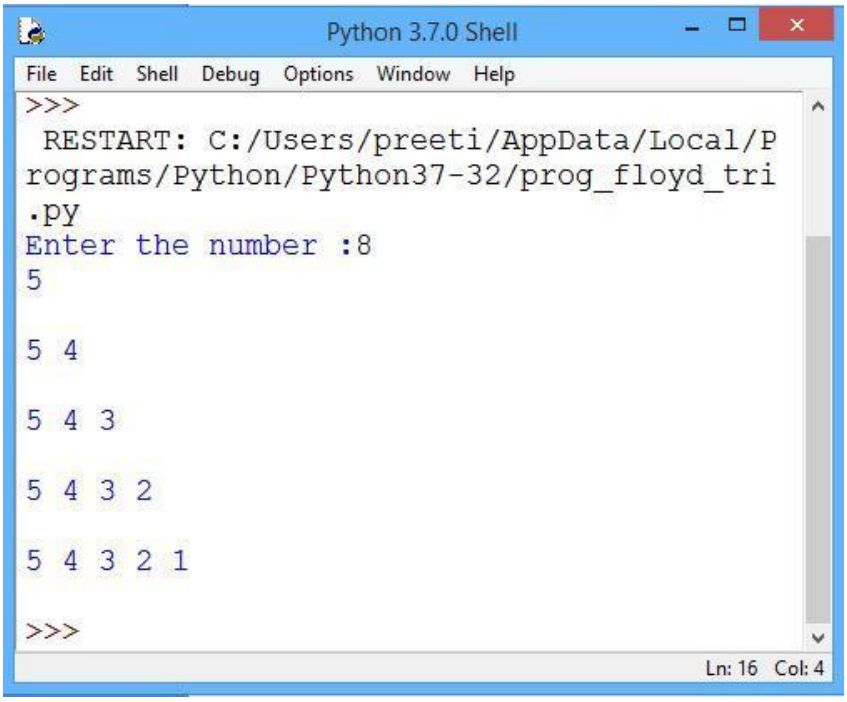
n=int(input("Enter the number :"))

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

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

print (j,end=' ')

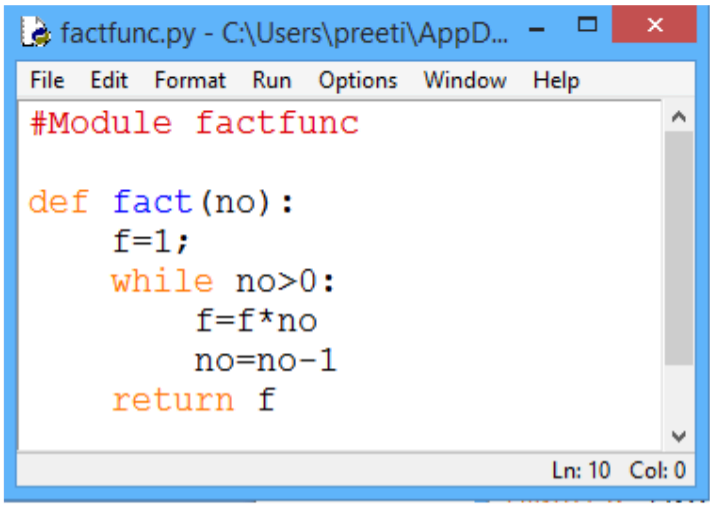
print('\n')
```

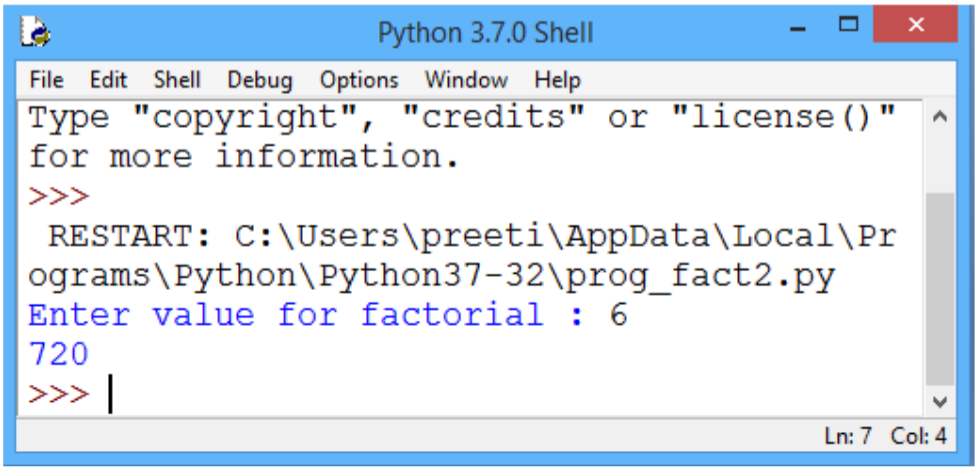


**Program 10: Write a Program to find factorial of entered number using user-defined module fact().**

Solution:

```
#Using function
import factfunc
x=int(input("Enter value for factorial : "))
ans=factfunc.fact(x)
print (ans)
```



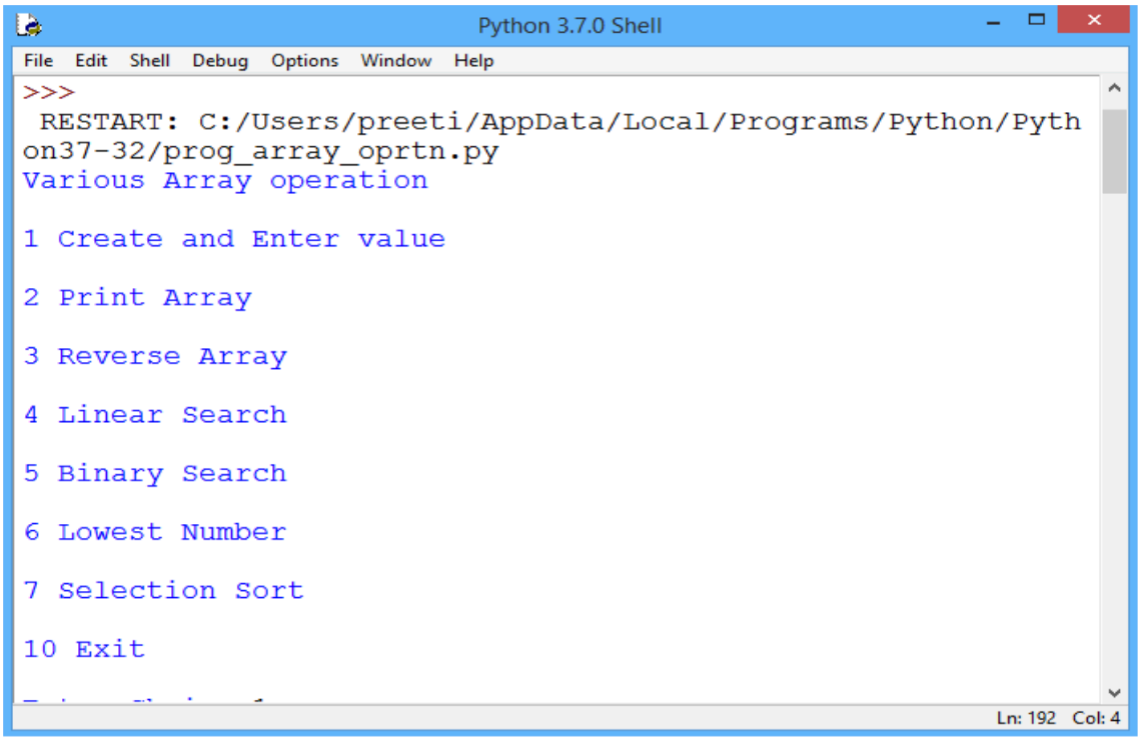
**Program 11: Write a Program to enter the numbers and find Linear Search, Binary Search, Lowest Number and Selection Sort using list/array code.**

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()
```




```
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

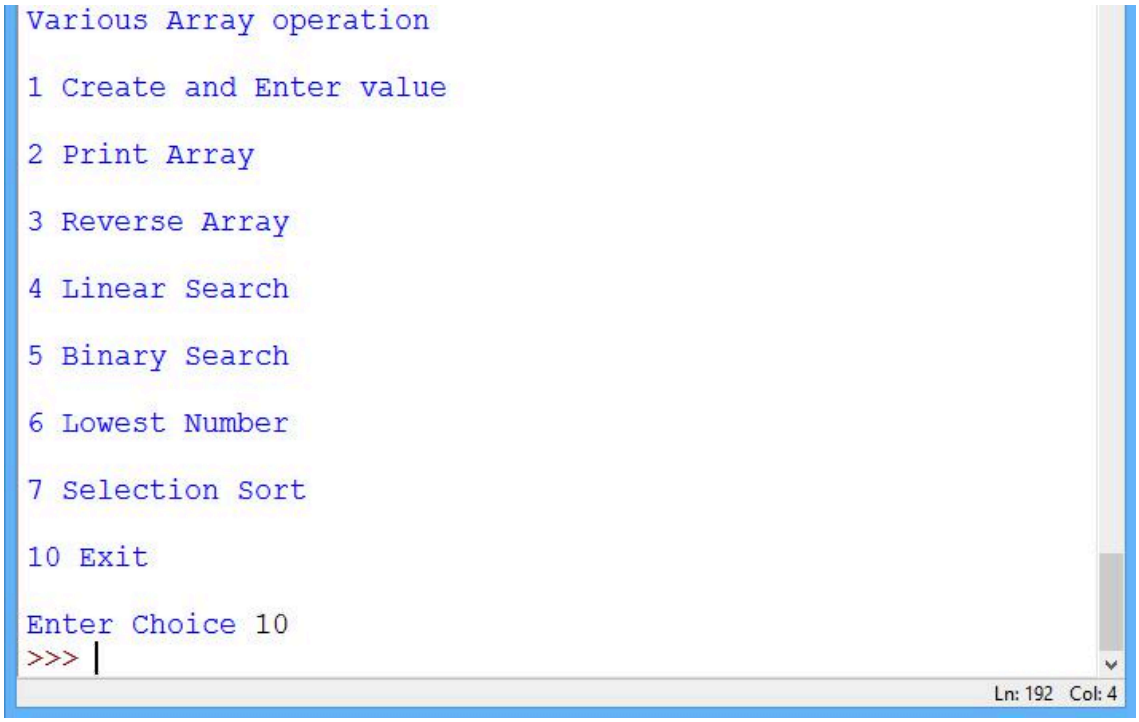
5 Binary Search

6 Lowest Number

7 Selection Sort

10 Exit

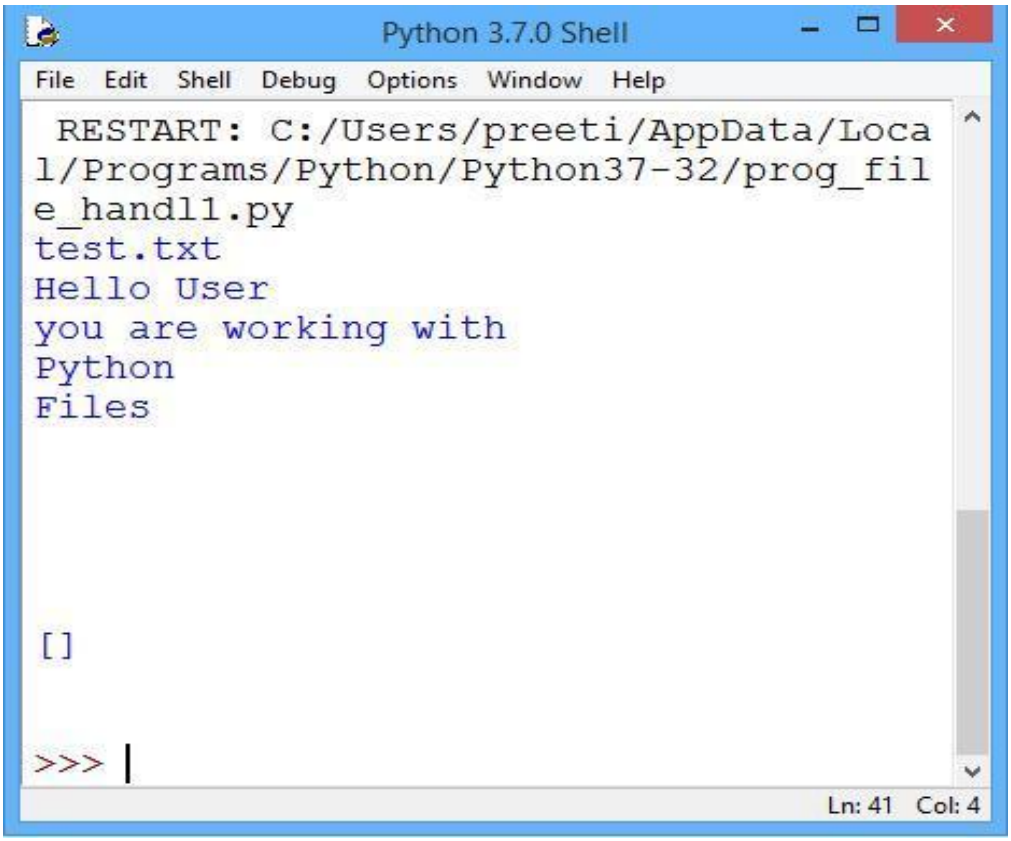
Enter Choice 6
The Lowest number is 10
```


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
>>>
Ln: 192 Col: 4

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

Solution:

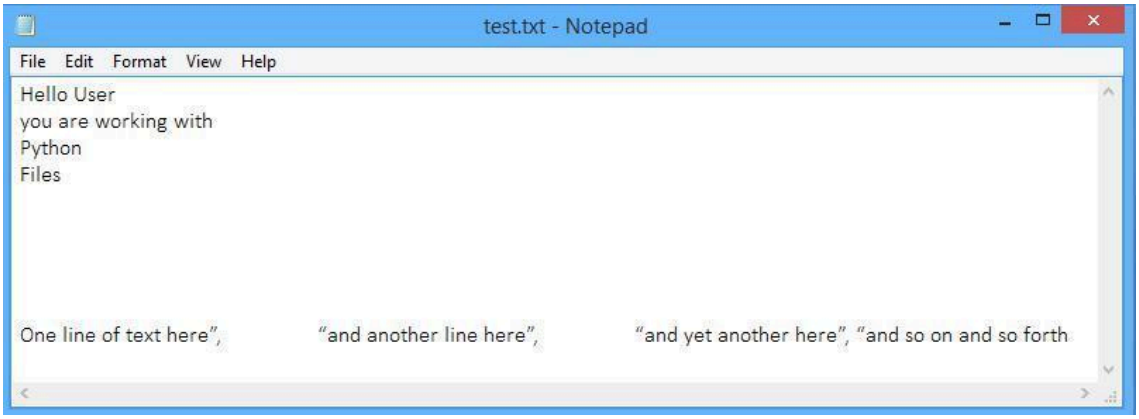
```
f=open("test.txt",'r')
print(f.name)
f_contents=f.read()
print(f_contents)
f_contents=f.readlines()
print(f_contents)
f_contents=f.readline()
print(f_contents)
for line in f:
print(line, end='')
f_contents=f.read(50)
print(f_contents)
size_to_read=10
f_contents=f.read(size_to_read)
while len(f_contents)>0:
print(f_contents)
print(f.tell())
f_contents=f.read(size_to_read)
```



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

Solution:

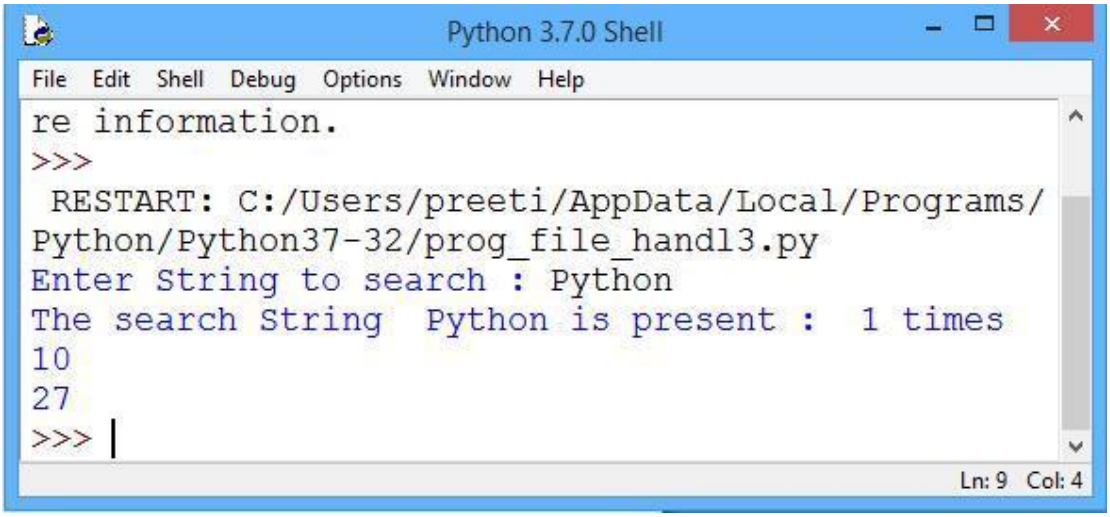
```
#Program to read data from data file in append mode af=open("test.txt",'a')
lines_of_text = ("One line of text here”,\
“and another line here”,\
“and yet another here”, “and so on and so forth")
af.writelines('\n' + lines_of_text)
af.close()
```



**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:

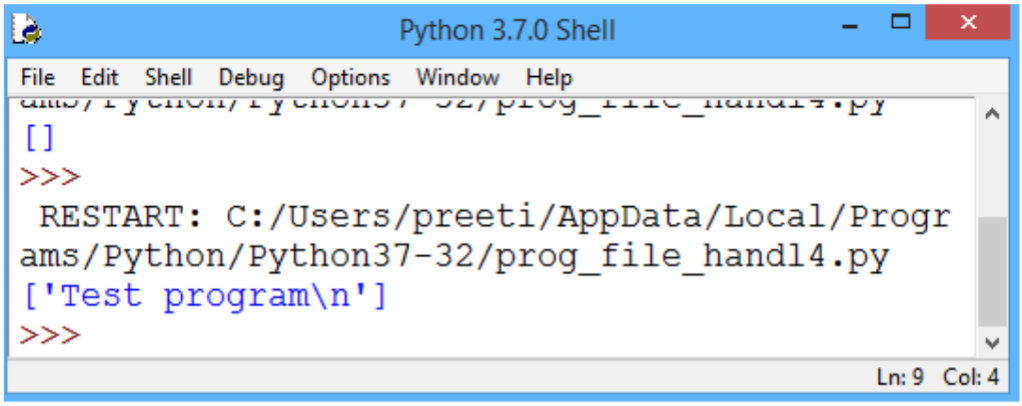
```
#Program to read data from data file in read mode and
#count the particular word occurrences in given string,
#number of times in python.
f=open("test.txt",'r')
read=f.readlines()
f.close()
times=0 #the variable has been created to show the number of times the loop runs times2=0 #the variable has 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)
```

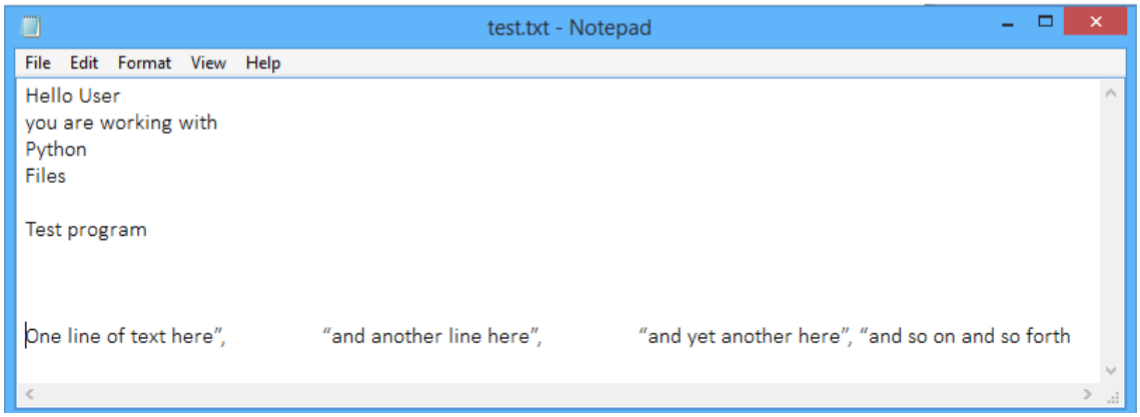


**Program 15: Write a Program to read data from data file in read mode and append the words starting with letter 'T' in a given file in python.**

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)
```

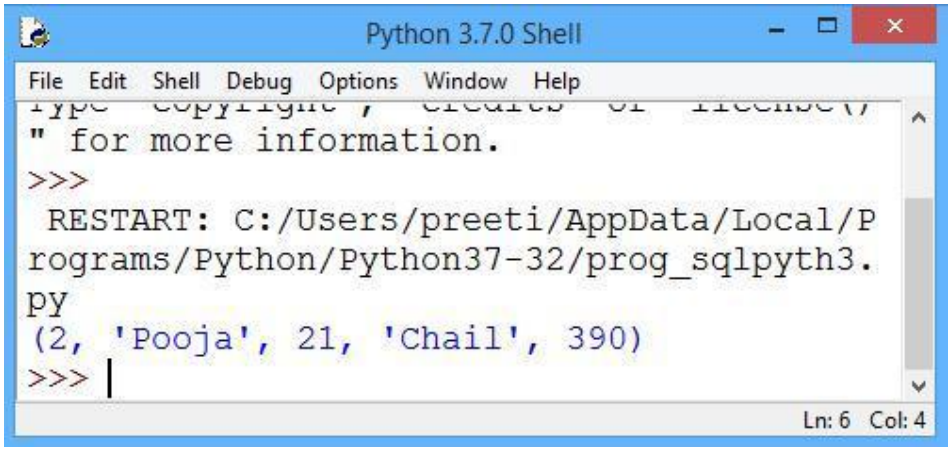


**Program 16: Write a Program to show MySQL database connectivity in python.**

Solution:

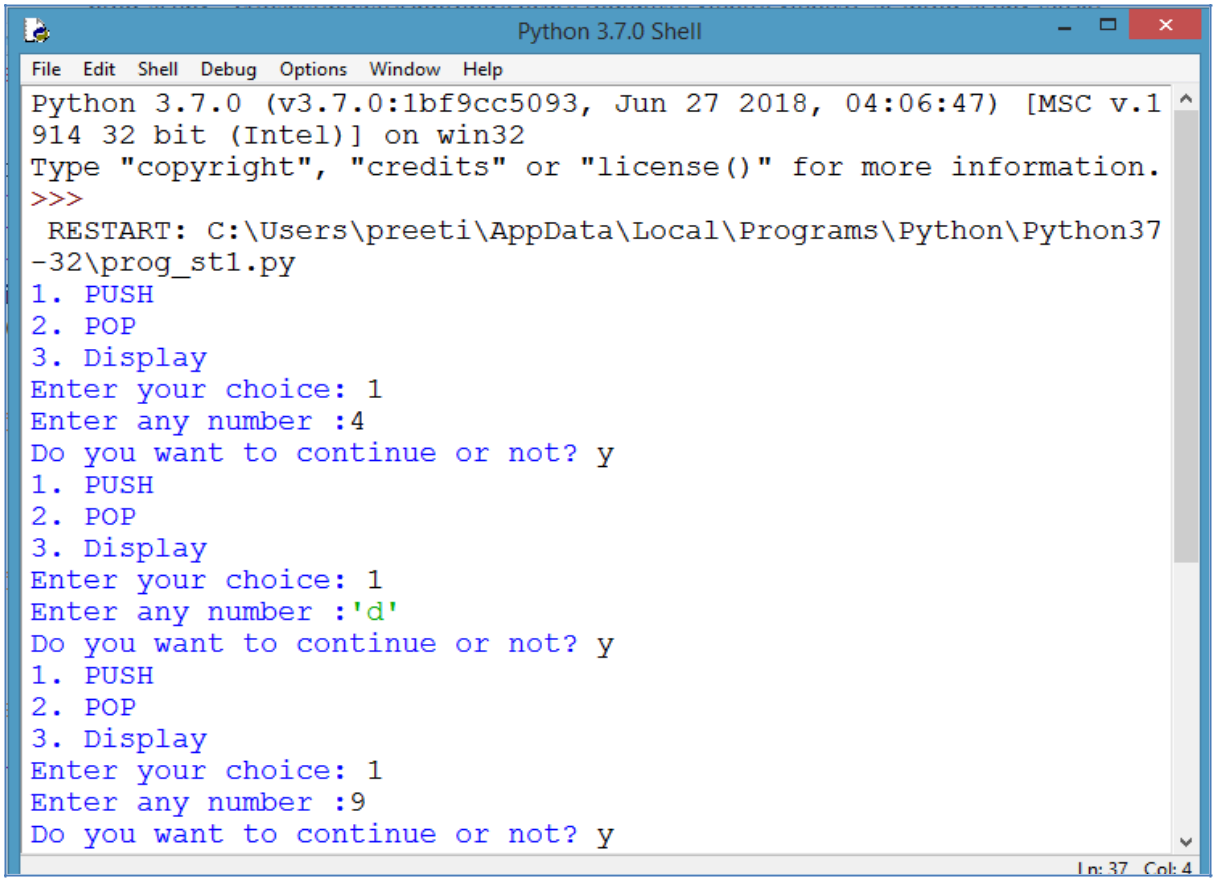
```
import mysql.connector
con=mysql.connector.connect(host='localhost',user='root',password='',db='school')
stmt=con.cursor()
query='select * from student;'
stmt.execute(query)
data=stmt.fetchone()
print(data)
```

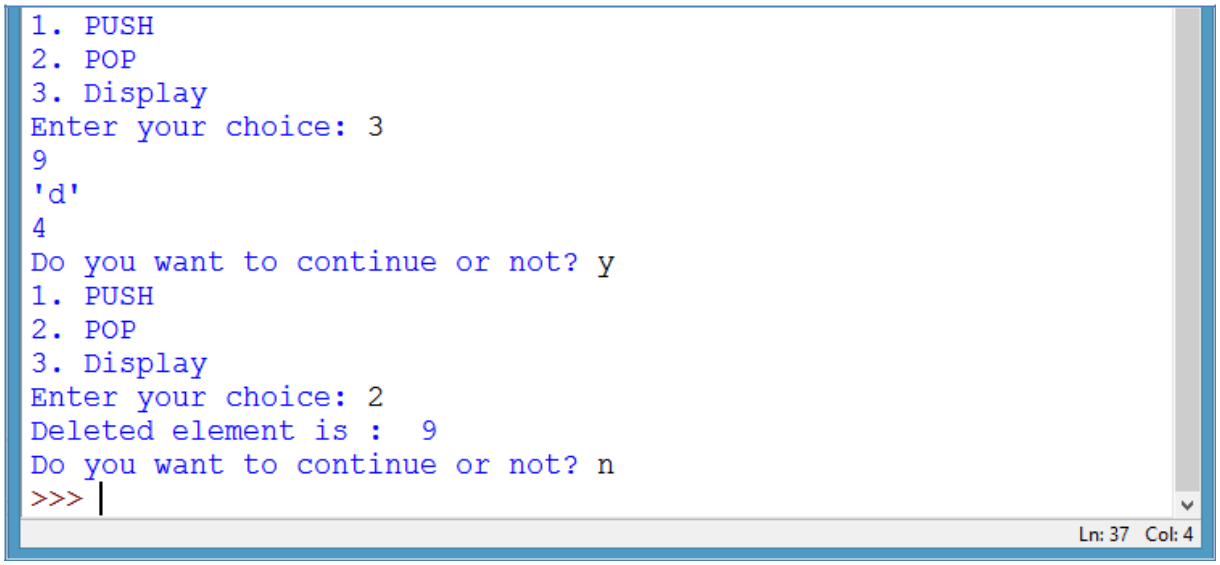


**Program 17: Write a Python program to implement all basic operations of a stack, such as adding element (PUSH operation), removing element (POP operation) and displaying the stack elements (Traversal operation) using lists.**

Solution:

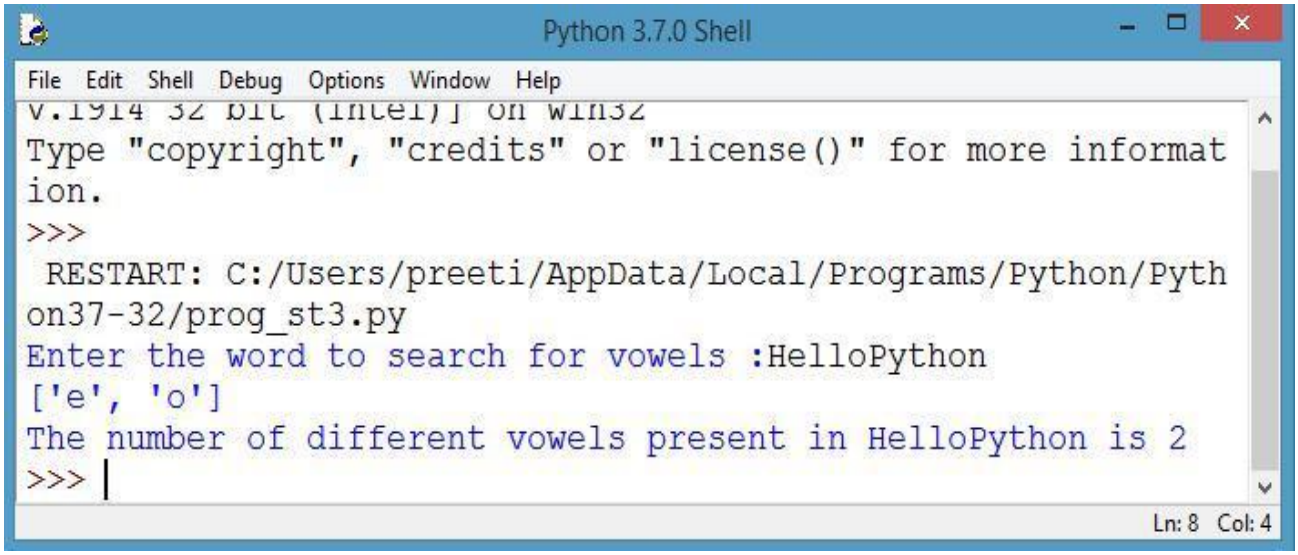
```
#Implementation of List as stack
s=[]
c="y"
while (c=="y"):
print ("1. PUSH")
print ("2. POP ")
print ("3. Display")
choice=int(input("Enter your choice: "))
if (choice==1):
a=input("Enter any number :")
s.append(a)
elif (choice==2):
if (s==[]):
print ("Stack Empty")
else:
print ("Deleted element 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 Shell
File Edit Shell Debug Options Window Help
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
Ln: 37 Col: 4


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
>>>
Ln: 37 Col: 4

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

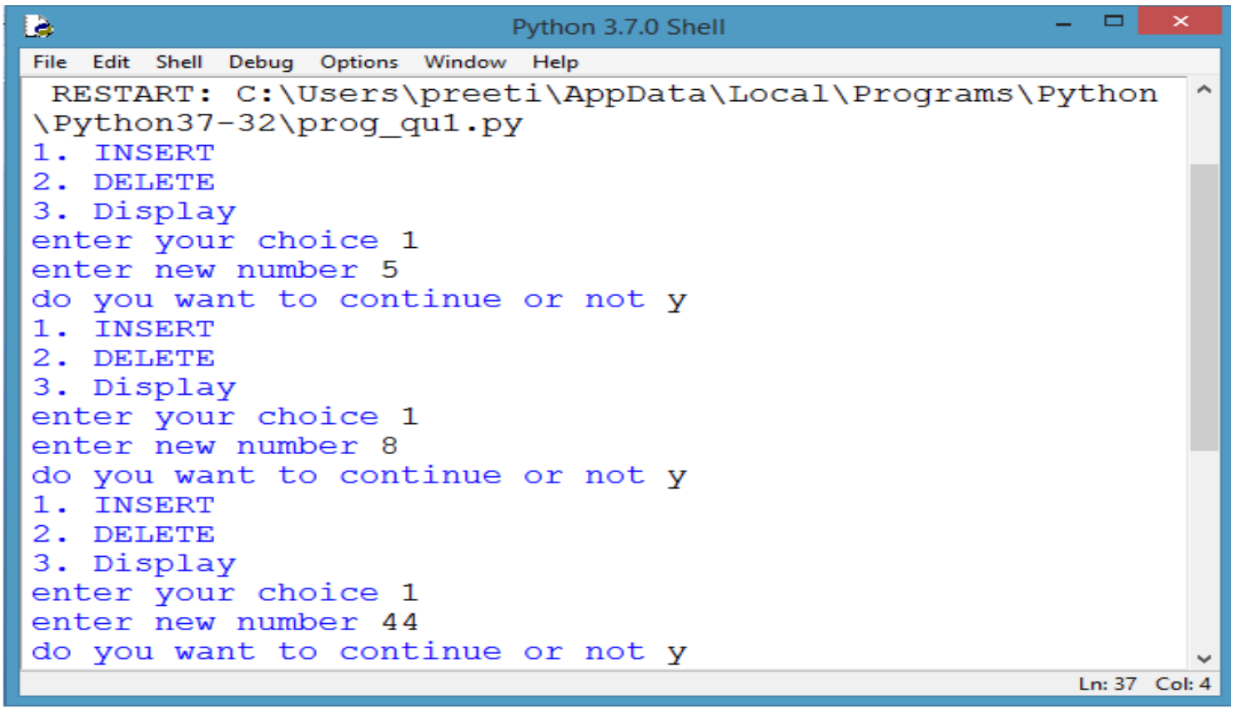
```
#Program to display unique vowels present in the given word #using Stack
vowels =['a','e','i','o','u']
word = input("Enter the word to search for vowels :")
Stack = []
for letter in word:
if letter in vowels:
if letter not in Stack:
Stack.append(letter)
print(Stack)
print("The number of different vowels present in",word,"is",len(Stack))
```

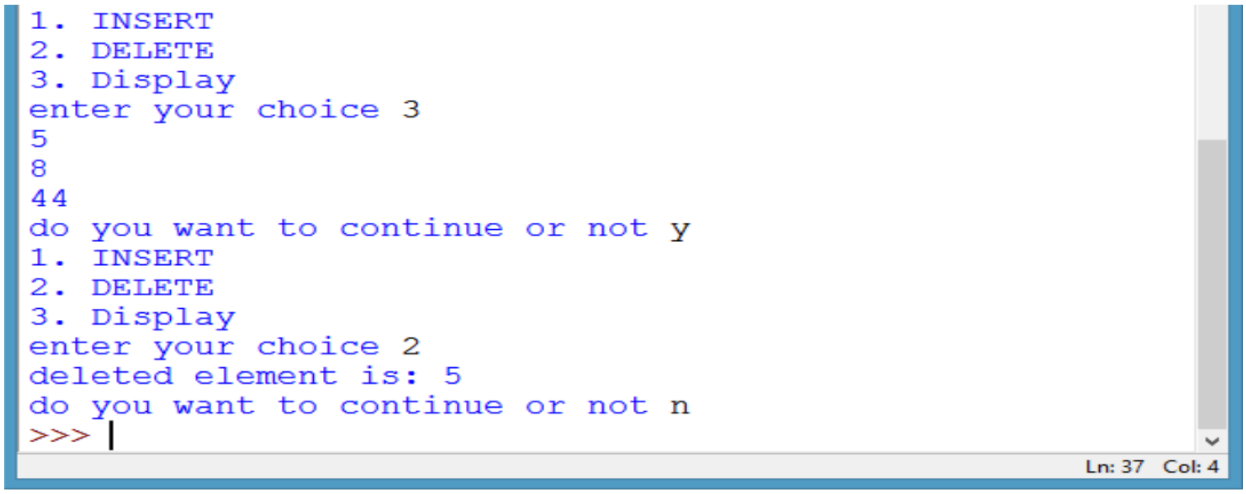


**Program 19: Write a program in Python to add, delete and display elements from a queue using list.**

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: ")
```


Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
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
Ln: 37 Col: 4


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
>>>
Ln: 37 Col: 4

**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)

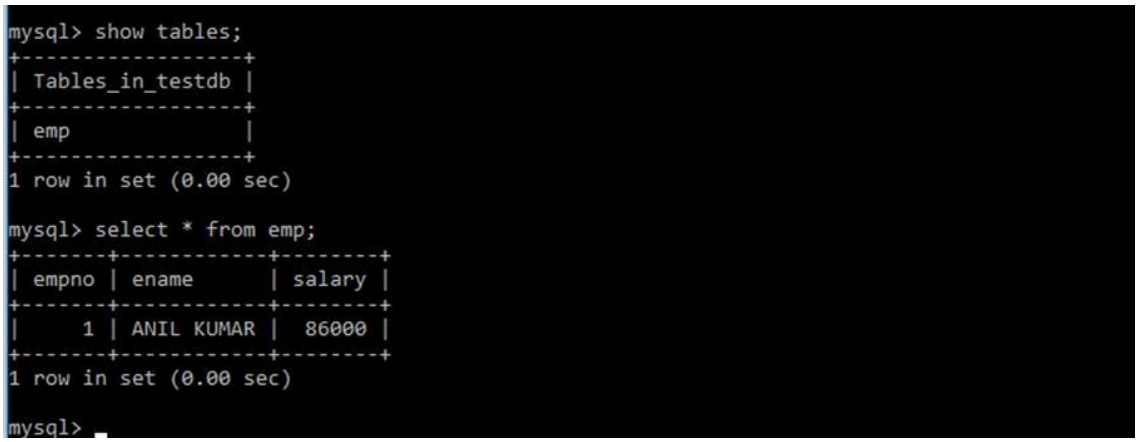
mysql> show tables;
+------------------+
| Tables_in_testdb |
+------------------+
| emp              |
+------------------+
1 row in set (0.00 sec)

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()
```




**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()
```