Name : Nishika prajapati
Enrollment : 190133107022
Class : CE 5th B-3

Practical List

1. Develop a program to understand the control structures of python.
2. Develop a program to learn different types of structures (list, dictionary, tuples) in python.
3. Develop a program to learn concepts of functions scoping, recursion and list mutability.
4. Develop a program to understand working of exception handling and assertions.
5. Develop a program for data structure algorithms using python – searching, sorting, and hash tables.
6. Develop a program to learn regular expressions using python.
7. Develop chat room applications using multithreading.
8. Learn to plot different types of graphs using PyPlot.
9. Implement classical ciphers using python.
10. Draw graphics using Turtle.
11. Develop a program to learn GUI programming using Tkinter.

1. Develop a program to understand the control structures of python.

```
from datetime import datetime
print("Enrollment: 190133107022")


now = datetime.now()

        current_time = now.strftime("%H:%M:%S")
        print("Current Time =", current_time)

#for and if
for i in range(5):
        if i%2 == 0:
                print(str(i) + ' Even') elif
        i%3==0:
                print(str(i) + ' divisible by 3')
        else:
                print(str(i) + ' odd')
i=0
while i < 3:
        print("I'm loopy!")
        i += 1
#boolean syntax x = 0

if x == 1: print('Positive value')
else:
        print("Negative value")
```

Output :

```
Enrollment: 190133107022
Current Time : 18:43:48
0 Even
1 odd
I'm loopy!
I'm loopy!
I'm loopy!
2 Even
3 divisible by 3
4 Even
Negative value
```

2. Develop a program to learn different types of structures (list, dictionary, tuples) in python.

```
from datetime import datetime
print("Enrollment: 190133107022")

now = datetime.now()

current_time = now.strftime("%H:%M:%S")
print("Current Time =", current_time)
lst = list()
for i in range(1, 6):
   lst.append(i)
   print('Our list:', lst)

lst[2] = 'PYTHON'
print('after mutation:', lst)

dct = {}
fruit = ['apple', 'banana', 'grapes']
for i in range(1, 4):
   dct[i] = fruit[i-1]
   print('dict:', dct)

#tpl =  for ft in fruit
#for ft in fruit
 #   print(ft)

thistuple = ("apple", "banana", "cherry")
print("Tuple : ",thistuple)

A = set([1, 2, 3, 'a', 1, 'b'])
print("Set : ",A)
```

Output :

```
Enrollment: 190133107022
Current Time : 18:46:22
Our list: [1]
Our list: [1, 2]
Our list: [1, 2, 3]
Our list: [1, 2, 3, 4]
Our list: [1, 2, 3, 4, 5]
after mutation: [1, 2, 'PYTHON', 4, 5]
dict: {1: 'apple'}
dict: {1: 'apple', 2: 'banana'}
dict: {1: 'apple', 2: 'banana', 3: 'grapes'}
Tuple :  ('apple', 'banana', 'cherry')
Set :  {1, 2, 3, 'b', 'a'}
```

3. Develop a program to learn concepts of functions scoping, recursion and list mutability.

```
 from datetime import datetime
print("Enrollment: 190133107022")

now = datetime.now()

current_time = now.strftime("%H:%M:%S")
print("Current Time =", current_time)
    #function scoping
    import datetime
    x=200
    def myfunc():
      x = 100
      print(x)                    # output x=100
    myfunc()  #inside function

    def print_mydetails(name):                              #global scope
      dt = datetime.datetime.now().date()
      #def printfnc(name):                                  #local scope
      print("Today's date: " ,str(dt))
      print("Enrolment No.:" , 190133107022)
      print("Performing" , str(name))
      print(".....")

        #printfnc(name)
    name="Nishika Prajapati"
    print_mydetails(name)

    def get_date():              #global scope
      return datetime.datetime.now().date()

    def printfnc():              #local scope
      print("Today's date: " , get_date())

    printfnc()

    get_date()        #global scope

    name = 'Nishika'
    def my_name():
      print(name)
      name = 'Nishika Prajapati'
      my_name() #recursion
    def get_itr(num):
      if num <= 0:
        return 1
```

```
    elif num == 1:
        return 1
    else:
        return num*get_itr(num-1)

ans = get_itr(5)
print(ans)



#list
lst = [  i*i for i in range(1, 6) ]              #[1, 4, 9, 16, 25, 36]
print('Before mutation: ' + str(lst[2]))
lst[2] = '1024'
print('After mutation: ' + str(lst[2]))
```

Output :

Enrolment No:**190133107022**
Practical No:**3**
Practical Title: **Develop a program to learn concepts of functions scoping, recursion and list mutability**
System Generated Date and Time:,Date: **Time:18:48:57 , Date:2020-07-28**

Input: (if it there)

Output:

```
Enrollment: 190133107022
Current Time : 18:48:57
100
Today's date:  2020-07-28
Enrolment No.: 190133107022
Performing Nishika Prajapati
.....................
Today's date:  2020-07-28
120
Before mutation: 9
After mutation: 1024
```

4. Develop a program to understand working of exception handling and assertions.

P4_1.py:

```
        from datetime import datetime
    print("Enrollment: 190133107022")

    now = datetime.now()

    current_time = now.strftime("%H:%M:%S")
    print("Current Time =", current_time)

##assertion
        def convert1(Temp):
          assert Temp >= 0, "Colder than absolute zero!"
          return ((Temp-273)*1.8)+32

        #print (convert1(273))
        print (int(convert1(505.78)))
        #print (convert1(-5))
```

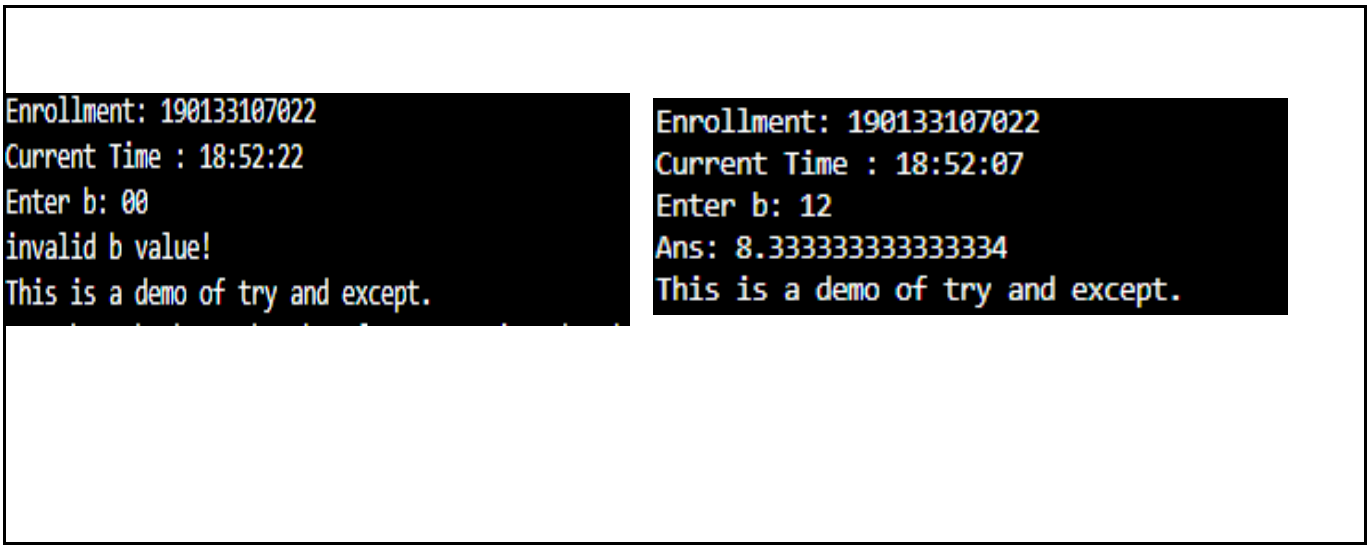
Output :

```
Enrollment: 190133107022
Current Time : 18:50:05
451
```

P4_2.py:

```
from datetime import datetime
        print("Enrollment: 190133107022")

        now = datetime.now()

        current_time = now.strftime("%H:%M:%S")
        print("Current Time =", current_time)
        a = 100
        b = int(input('Enter b: '))
        try:
                ans = a/b
        except:                                 #catch(Exception e) { }
          print('invalid b value!')
        else:
          print('Ans: ' + str(ans))
        finally:
                print('This is a demo of try and except.')
```

Output :

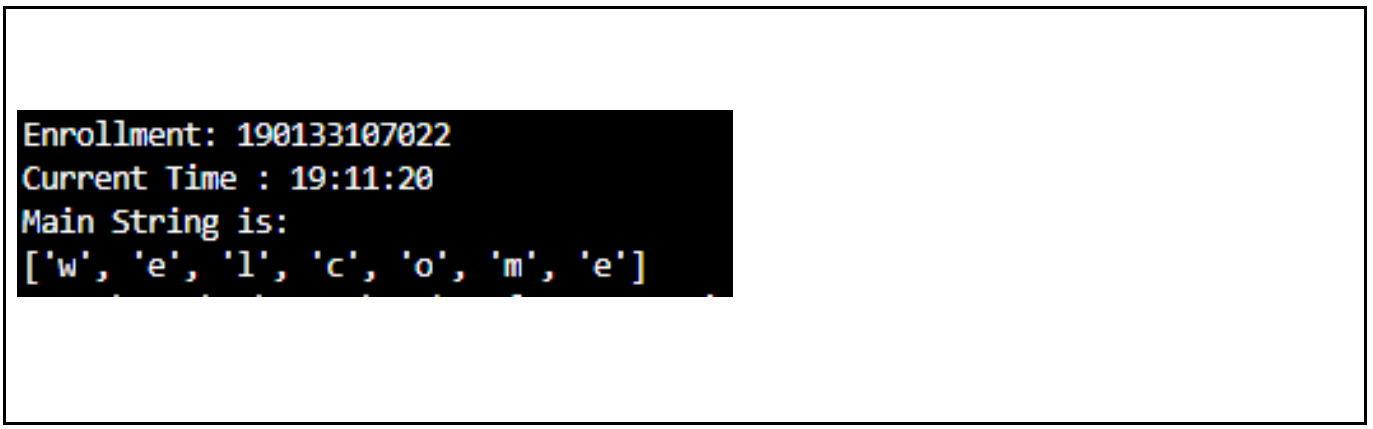


5. Develop program for data structure algorithms using python –searching, sorting, and hash tables.

P5_1.py:

```
from datetime import datetime
        print("Enrollment: 190133107022")

        now = datetime.now()

        current_time = now.strftime("%H:%M:%S")
        print("Current Time =", current_time)

def insertionSort(arr):
        for i in range(1,len(arr)):
                key = arr[i]
                j = i-1
                while j >= 0 and key < arr[j] :                 #[w      e   l c o m e ]
                    arr[j+1] = arr[j]                           #  j=0   i=1
                    j-= 1
                arr[j+1] = key

arr = ['w', 'e', 'l', 'c', 'o', 'm','e']                # 1
print("Main String is: ") print(arr)                            # 2
insertionSort(arr)                                              # 3
        print("Sorted string is:") print(arr)                       # 4
```

Output :



P5_2.py:

```
from datetime import datetime
print("Enrollment: 190133107022")

now = datetime.now()

current_time = now.strftime("%H:%M:%S")
print("Current Time =", current_time)

def merge(arr, l, m, r):
    n1 = m - l + 1                  #left side size of the array
    n2 = r- m
                                    #right side size of the array
    L = [0] * (n1)                  #new array of length n1 and its name is L initialized with 0
    R = [0] * (n2)                  #new array of length n2 and its name is R initialized with 0
    for i in range(0 , n1):
        L[i] = arr[l + i]

    for j in range(0 , n2):
        R[j] = arr[m + 1 + j]
    i = 0
    j = 0
    k = l
    while i < n1 and j < n2 :
        if L[i] <= R[j]:
            arr[k] = L[i]           #updation
            i += 1
        else:
            arr[k] = R[j]           #updation
            j += 1
        k += 1
    while i < n1:
        arr[k] = L[i]                       #appending
        i += 1
        k += 1
    while j < n2:
        arr[k] = R[j]                       #appending
        j += 1
        k += 1

def mergeSort(arr, l, r):
    if l < r:                       #precondition
        m = (l+(r-1))//2    #mid point to divide the list into two parts left and right
        mergeSort(arr, l, m)        #applying recursive merge sort over left side of array
        mergeSort(arr, m+1, r)  #applying recursive merge sort over right side of array
```

```
        merge(arr, l, m, r)       #applied on sorted left and sorted right to combine these two parts.


arr = [14,46,43,27,57,41,45,21,70]
n = len(arr)

print('Unsorted list:')
for i in range(n):
   print ("%d" %arr[i])

mergeSort(arr, 0, n-1)

print('\nSorted List')
for i in range(n):
   print ("%d" %arr[i])
```

Output :

```
Enrollment: 190133107022
Current Time : 19:41:57
Unsorted list:
14
46
43
27
57
41
45
21
70

Sorted List
14
21
27
41
43
45
46
57
70
```

6. Develop a program to learn regular expressions using python.

```
from datetime import datetime
        print("Enrollment: 190133107022")

        now = datetime.now()

        current_time = now.strftime("%H:%M:%S")
        print("Current Time =", current_time)
import re
pattern = input('Enter the pattern to find: ') text =
input('Enter string/paragraph: ')
if re.search(pattern, text):
        print('substring found') else:
        print('substring not found')
```

Output :

```
Enrollment: 190133107022
Current Time : 19:45:04
Enter the pattern to find: how
Enter string/paragraph: Hello.How's you
substring not found
```

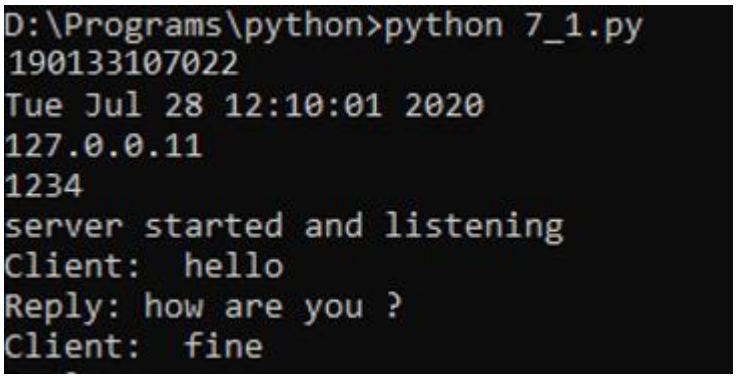
7. Develop chat room applications using multithreading.

P7_server.py:

```
import time
    enroll =
    190133107022

    print(enroll)
    print(time.ctime())
    #P7_server
    import socket
from threading import *
    serversocket = socket.socket(socket.AF_INET,
    socket.SOCK_STREAM) host = "127.0.0.11"
    port =
    1234
    print(host)
    print(port)
    serversocket.bind((host,
    port)) class client(Thread):
      def __init__(self, socket,
        address): Thread.init(self):
        self.sock = socket
        self.addr =
        address
        self.start()
      def
        run(self):
        while 1:
          print('Client: ',
```

```
        self.sock.recv(1024).decode()) m =
        input("Reply: ") self.sock.send(m.encode()
        serversocket.listen(5)
print ('server started and
listening') while 1:
   clientsocket, address =
   serversocket.accept() client(clientsocket,
   address)
```

OUTPUT:



P7_client.py:

```
Enroll=190133107022
import time
print(enroll)
print(time.ctime())
#P7_client
import socket
s = socket.socket(socket.AF_INET,
socket.SOCK_STREAM) host = "127.0.0.11"
port = 1234
s.connect((host,port)) def ts(str):
s.send(str.encode())
data =
s.recv(1024).decode() print
("Server: ",data)
print('Start The Conversation by sending a
message') while 2:
r = str(input('Reply: '))
ts(r)
s.close ()
```

Output:

```
D:\Programs\python>python 7_2.py
190133107022
Tue Jul 28 12:10:05 2020
Start The Conversation by sending a message
Reply: hello
Server:  how are you ?
Reply: fine
```

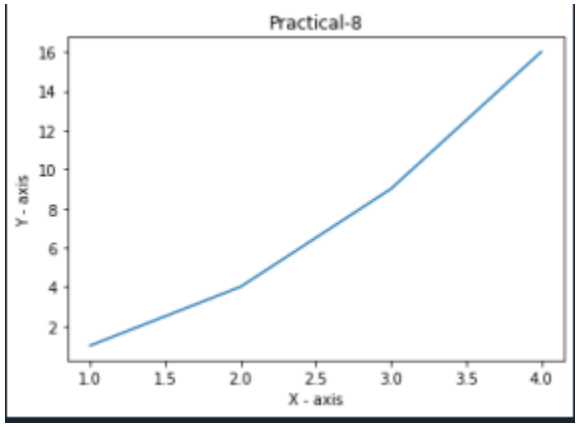
8. Learn to plot different types of graphs using PyPlot.

P8_1.py:

```
Enroll=190133107022

 import time
print(enroll)
print(time.ctime())
import
matplotlib.pyplot as
plt

 x = [1,2,3,4]

 y = [1,4,9,16]
plt.plot(x, y)
plt.xlabel('X - axis')
plt.ylabel('Y - axis')
plt.title('Practical-8')
plt.show()
```

Output :

```
Enrollment: 190133107022
Current Time : 19:59:43
```

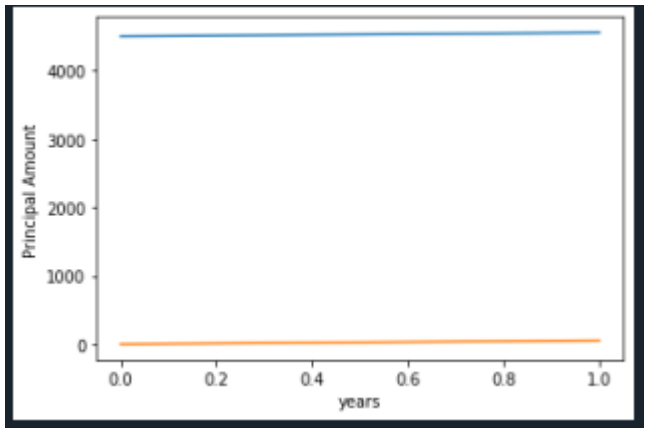


P8_2.py:

```
Enroll=190133107022

 import time
print(enroll)
print(time.ctime())
import matplotlib.pyplot as plt
def compound_interest(principle, rate, time):
    result = principle * (pow((1 + rate / 100), time))
    return result
p = float(input("Enter the principal amount: "))
r = float(input("Enter the interest rate: "))
endyear = float(input("Enter the Year : "))
yearlist = []
pamountlist =
[] interestlist =
[]
for i in
    range(int(endyear)):
    yearlist.append(i)
    amount = compound_interest(p, r, i)
    intamount = int(amount)
    pamountlist.append(intamount)
    interest = amount - p
```

```
        intins = int(interest)
        interestlist.append(intins
        )
    print(yearlist)
    print(pamountlist
    )
    print(interestlist)
    plt.plot(yearlist,pamountlist,interestlist)
    plt.xlabel('years')
    plt.ylabel('Principal
    Amount') plt.show()
```

```
Enrollment: 190133107022
Current Time : 20:01:42

Enter the principal amount: 4500

Enter the interest rate: 1.2

Enter the Year : 2
[0, 1]
[4500, 4554]
[0, 54]
```



9. Implement classical ciphers using python.

```
from datetime import datetime
    print("Enrollment: 190133107022")

    now = datetime.now()

    current_time = now.strftime("%H:%M:%S")
    print("Current Time =", current_time)

def encrypt(text,s):
    result = ""
    for i in range(len(text)):
        char =text[i]
```

```
        if (char.isupper()):
                result += chr((ord(char) + s-65) % 26 + 65)
        else:
                result += chr((ord(char) + s - 97) % 26 + 97)
return result
text = input("hey how are you ") s =
int(input('enter key (0-26): '))
print ("Text : " + text)
print ("Shift : " + str(s))
print ("Cipher: " + encrypt(text,s))
```

Output :

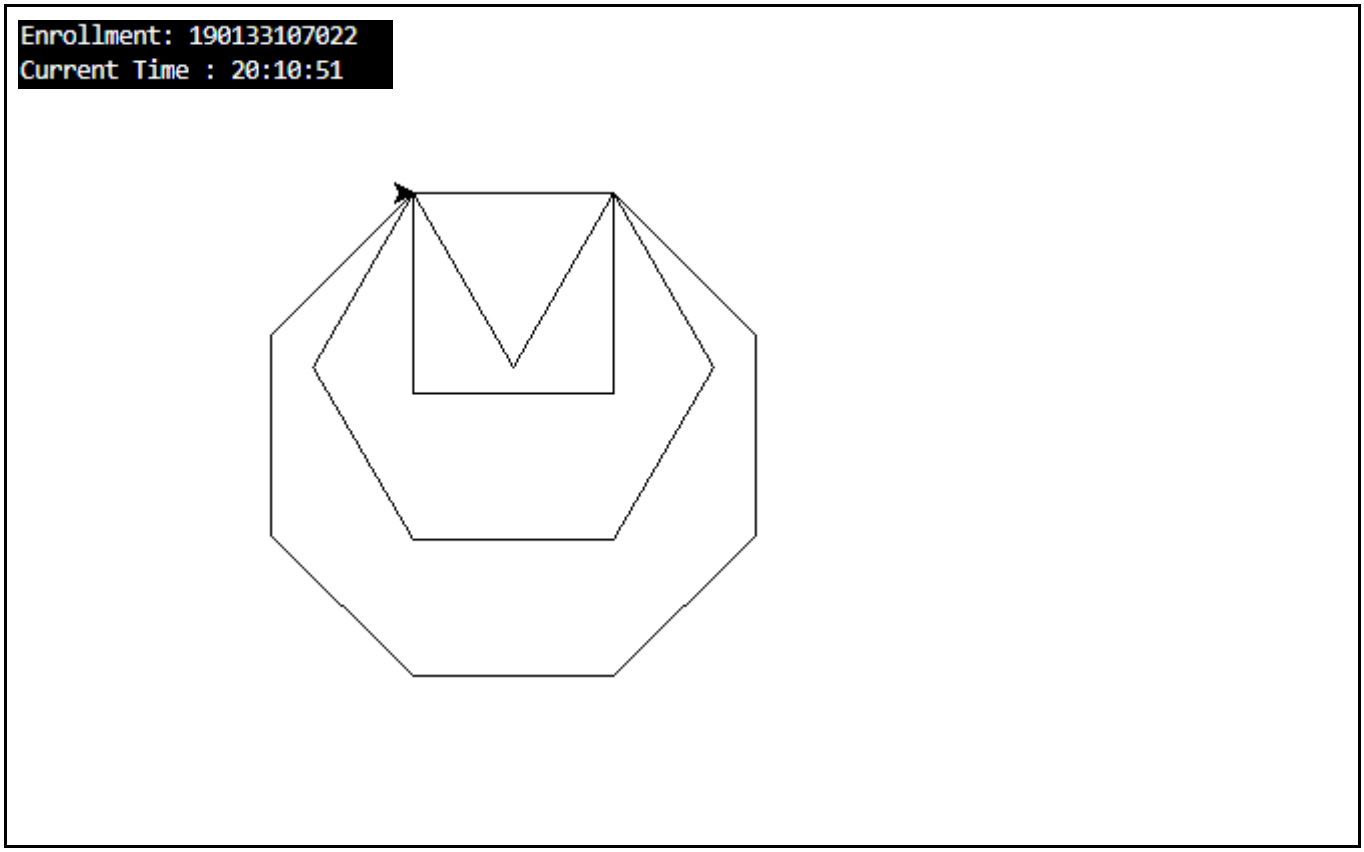
```
Enrollment: 190133107022
Current Time : 20:07:50
Text you want to encrypt: hey how are you
enter key (0-26): 9
Text : hey how are you
Shift : 9
Cipher: q
```

10. Draw graphics using Turtle.

```
from datetime import datetime
        print("Enrollment: 190133107022")

        now = datetime.now()

        current_time = now.strftime("%H:%M:%S")
        print("Current Time =", current_time)

import turtle
triangle = turtle.Turtle()
for i in range(3):
        triangle.forward(100)                   #Assuming the side of a octagon is 100 units
        triangle.right(120) square = turtle.Turtle()
for i in range(4):
        square.forward(100)             #Assuming the side of a octagon is 100 units
        square.right(90) hexagon = turtle.Turtle()
```

```
for i in range(6):
        hexagon.forward(100)            #Assuming the side of a hexagon is 100 units
        hexagon.right(60) octagon = turtle.Turtle()
for i in range(8):
        octagon.forward(100)  #Assuming the side of a octagon is 100 units
        octagon.right(45) turtle.mainloop()
```

Output :



11.Develop a program to learn GUI programming using Tkinter.

```
enroll = 190133107022

 import time

 print(enroll)

 print(time.ctime())

from tkinter import *

master = Tk()

 def red():

   frame.configure(background = "red")

   print('Button1 Clicked!')

def green():

 frame.configure(background = "green")

 print('Buttno2 Clicked!')

def pink():

 frame.configure(background = "pink")

 print('Button3 Clicked!')

 frame = Frame(master, background="blue")

btn1 = Button(frame, command=red, padx = 20)
 btn2 = Button(frame, command=green, padx = 20)

btn3 = Button(frame, command=pink, padx = 20)

 btn1['text'] = 'Button1'

 btn2['text'] = 'Button2'

 btn3['text'] = 'Button3'
```

```
btn1.pack(pady = 20, padx = 20)

btn2.pack(pady = 20, padx = 20)

btn3.pack(pady = 20, padx = 20)

frame.pack()

master.mainloop()
```

Output :