EXPERIMENT:2

PYTHON PROGRAMMING USING SIMPLE STATEMENTS & EXPRESSION

1.EXCHANGING THE VALUES OF TWO VARIABLES

CODE:

p=int(input("enter the first value:"))

q=int(input("enter the second value"))

print("the value before swapping are",p,q)

temp=p

p=q

q=temp

print("the values after swapping are",p,q)

OUTPUT:

enter the first value:48

enter the second value52

the value before swapping are 48 52

the values after swapping are 52 48

2.METHOD-2

CODE:

s=59

t=16

print ("the values before swapping:",s,t)

s,t=s,t

print("the values after swapping:",s,t)

OUTPUT:

the values before swapping: 59 16

the values after swapping: 59 16

3.METHOD-3

CODE:

x=45

y=25

print("the value before swapping are",x,y)

x=x+y

y=x-y

x=x-y

print("the value after swapping are",x,y)

OUTPUT :

the value before swapping are 45 25

the value after swapping are 25 45

4.METHOD-4

CODE:

j=58

k=46

print("the values before swapping are",j,k)

j=j^k

k=j^k

j=j^k

print("the values after swapping are",j,k)

OUTPUT:

the values before swapping are 58 46

the values after swapping are 46 58

2.CIRCULATE THE n VARIABLES

CODE:

s=int(input("enter a the values in the list:"))

list=[]

for i in range(0,s):

element=int(input("enter the value:"))

list.append(element)

print("circulating the list")

for i in range(0,s):

element\_deleted=list.pop(0)

list.append(element\_deleted)

print("the circulated list after",i+1,"rotation",list)

OUTPUT:

enter a the values in the list:8

enter the value:5

enter the value:9

enter the value:2

enter the value:1

enter the value:7

enter the value:0

enter the value:3

enter the value:2

circulating the list

the circulated list after 1 rotation [9, 2, 1, 7, 0, 3, 2, 5]

the circulated list after 2 rotation [2, 1, 7, 0, 3, 2, 5, 9]

the circulated list after 3 rotation [1, 7, 0, 3, 2, 5, 9, 2]

the circulated list after 4 rotation [7, 0, 3, 2, 5, 9, 2, 1]

the circulated list after 5 rotation [0, 3, 2, 5, 9, 2, 1, 7]

the circulated list after 6 rotation [3, 2, 5, 9, 2, 1, 7, 0]

the circulated list after 7 rotation [2, 5, 9, 2, 1, 7, 0, 3]

the circulated list after 8 rotation [5, 9, 2, 1, 7, 0, 3, 2]

METHOD-2

CODE:

def circulate(c,n):

for i in range (1,n+1):

d=c[i:]+c[:i]

print("circulate","=",d)

return

c=[178,289,448,570,698,188,842,956,106]

n=int(input("enter n:"))

circulate(c,n)

OUTPUT:

enter n:6

circulate = [289, 448, 570, 698, 188, 842, 956, 106, 178]

circulate = [448, 570, 698, 188, 842, 956, 106, 178, 289]

circulate = [570, 698, 188, 842, 956, 106, 178, 289, 448]

circulate = [698, 188, 842, 956, 106, 178, 289, 448, 570]

circulate = [188, 842, 956, 106, 178, 289, 448, 570, 698]

circulate = [842, 956, 106, 178, 289, 448, 570, 698, 188]

3.DISTANCE BETWEEN TWO POINTS:

CODE:

x1=int(input("enter the value of x1:"))

x2=int(input("enter the value of x2:"))

y1=int(input("enter the value of y1:"))

y2=int(input("enter the value of y2:"))

d1=(x2-x1)\*\*2

d2=(y2-y1)\*\*2

result=(d1+d2)\*\*0.5

print("distance between",(x1,x2),"and",(y1,y2),"is :",result)

OUTPUT:

enter the value of x1:2

enter the value of x2:6

enter the value of y1:4

enter the value of y2:7

distance between (2, 6) and (4, 7) is : 5.0

4.PRINT THE REAL PART AND IMAGINARY PART OF THE COMPLEX NUMBBERR

CODE:

num=3+4j

print("real part:",num.real)

print("imaginary part:",num.imag)

OUTPUT:

real part: 3.0

imaginary part: 4.0