Experiment – 7

Aim: Implementing numerical operations using Random Number Functions

Theory:

It is required to import the *random* module as : import random

Built-in Random Functions		
S. No.	**Syntax**	**Description**
1.	random.random()	gives a float number that ranges from 0.0 to 1.0
2.	random.randint()	generates a random integer from the given range of numbers
3.	random.randrange(x,y)	selects an item randomly from the given range defined by the start, the stop, and the step parameters.
4.	random.shuffle()	shuffles the entries in a list at random.
5.	random.choice()	generates Random Number from the given list.

```
#Generate Random Float values
import random
num=random.random()
print(num)
```

Output:

```
0.3232640977876686
```

```
#Generate Random Integers
import random
num = random.randint(1, 500)
print( num )
```

Output:

```
210
```

```
#Generate Random Numbers within a Defined Range
import random
num = random.randrange(1, 10)
print( num )
```

Output:

```
4
9
20
```

```
#Shuffle Elements Randomly
import random
a = [34, 23, 65, 86, 23, 43]
random.shuffle(a)
print(a)
```

Output:

```
[23, 43, 86, 65, 34, 23]
[65, 23, 86, 23, 34, 43]
```

Choose Elements Randomly

```
import random

a = [ 1,3,5,'f',5]
c = random.choice(a)
print(c)
```

Objective 1: Write a program to generate 6 digit random OTP.

```
import random

otp = random.randint(100000, 999999)

print("OTP:", otp)
```

Objective 2: Program to flip a coin 10 times and count heads and tails.
Program 2:

```
import random
import itertools

results = { 'heads': 0, 'tails': 0}
a = results.keys()
b = list(a)

print(a)
print(b)

for i in range(10):
   c = random.choice(b)
   print(c)
   results[c] += 1

print('Heads:', results['heads'])
print('Tails:', results['tails'])
```

Output:

```
dict_keys(['heads', 'tails'])
['heads', 'tails']
tails
heads
tails
tails
tails
tails
tails
tails
heads
heads
Heads: 3
Tails: 7
```

Conclusion: Hence learned to implement random module.