Name : Hariharan S

Rollno : 225229111

Concurrent programming in python

In [11]:
```
rand_num = 0
import random as rm
class Thread:
    def _init_(self,rand_num):
        self.num = rand_num
    def generate(self,rand_num):
        self.index = rm.randint(1,100)
    def display(self):
        print(self.index)
```

In [19]:
```
a = Thread()
a.generate(rand_num)
a.display()
```

```
95
```

In [20]:
```
rand_num = 0
import random as rm
class SleepingThread:
    def _init_(Self,rand_num):
        self.num = rand_num

    def count(self):
        return self.count

    def display (self,rand_num):

        self.r=rm.randint(0,1000)
        print("Thread",self.count,"sleeps",self.r,"seconds")

s = SleepingThread()
s.display(rand_num)
```

```
Thread <bound method SleepingThread.count of <__main__.SleepingThread object at
0x000001F235A65DD8>> sleeps 140 seconds
```

In [ ]:
```

```