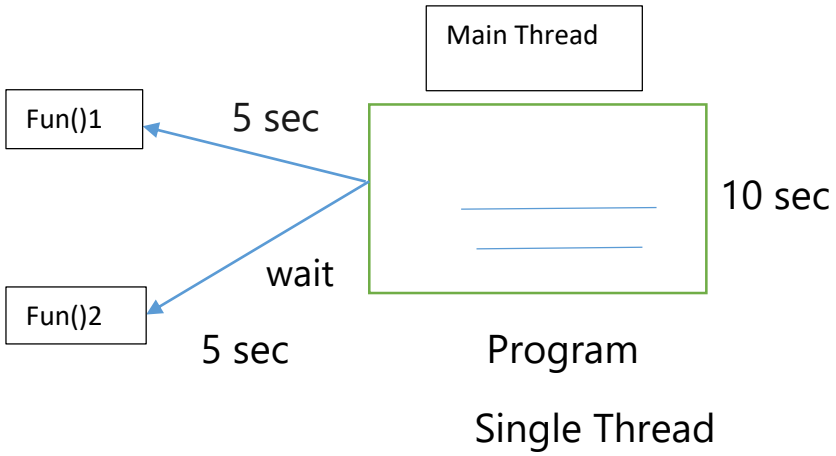
Multithreading in Python

Multithreading is a process of executing multiple threads (small units of a program) concurrently within a single process.

Multithreading is a technique which allows a CPU to execute multiple threads of one process at the same time.

Q.Why Multithreading?

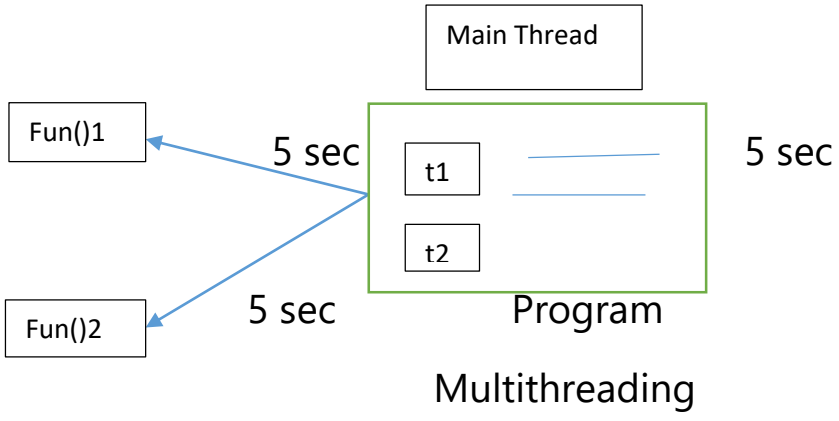
The process of Multithreading is to run multiple task and functions at the same time.



In python every program have 1 thread it is called 'Main Thread'.

it is bydefault thread(i.e. Main thread).

Main thread responsibilities to execute the program.



Q.What is Thread?

Thread is a predefined class which is available in threading in threading module.

Trhread is a basic unit of CPU and it is well know for independent execution.

Thread Class Methods

1.run()-The run() method represents the thread's activity. It may be overridden in a subclass.

2.start()-The start() method automatically invokes the object's run() method in a separate thread.

3.join()-The join() raises a RuntimeError if an attempt is made to join the current thread.

4.isAlive()-This method returns whether the thread is alive or dead.

5.setName()-Set the thread name.

6.getName()-return name (print name).

Program 1-

```
#single thread
Class A
    def run(self):
            for I in range(5):
                print(“Disha”)
Class B
    def run(self):
            for I in range(5):
```

```
                print("Computer")
t1=A()      #object or thread
t2=B()      #object or thread
t1.run()
t2.run()
            # in this program only one thread this is called (main thread).
```

Program 2-

```
#single thread
from time import sleep
Class A
    def run(self):
            for I in range(5):
                print("Disha")
                 sleep(1)
Class B
    def run(self):
            for I in range(5):
                print("Computer")
                sleep(1)
t1=A()      #object or thread
t2=B()      #object or thread
t1.run()
t2.run()
```

Program 3-

```
#Multithreading
from time import sleep
from threading import Thread
Class A(Thread)
    def run(self):
            for I in range(5):
                print("Disha")
                 sleep(1)
Class B(Thread)
    def run(self):
            for I in range(5):
                print("Computer")
                sleep(1)
t1=A()      #object or thread
t2=B()      #object or thread
t1.start()          #call run()function
t2.start()          #call run()function
#print("Institute")     #main thread
```

Program 4-

```
#Multithreading
from time import sleep
from threading import Thread
Class A(Thread)
```

```
    def run(self):
            for I in range(5):
                print("Disha")
                 sleep(1)
Class B(Thread)
    def run(self):
            for I in range(5):
                print("Computer")
                sleep(1)
t1=A()      #object or thread
t2=B()      #object or thread
t1.start()          #call run()function
t2.start()          #call run()function
t1.join()
t2.join()
print("Institute")     #main thread
```

Advantages of Multithreading-

Run multiple task at the same time without depend on each other thread.

Real Time Example-

Game –visual,audio and score at the same time.