Chapter: Threading

What are threads?

- Smallest schedulable unit in a computer
- Threads are contained in processes
- Threads share memory and state of the process

- Two kinds of threads
-   a. Threads created by the kernel / operating system – Kernel Threads
-   b. User threads
- Implemented through Threading Module

Advantages of threads

- ➢Faster Execution
- ➢Let us deliberate on the difference in “return “
- ➢Threads can share memory => threads from a same process have access to the same variables – Both a cause for joy and a cause for worrying

Starting threads

```
import threading
threading.Thread()

(refer program thread1.py)
```

Passing arguments to threads

```
t = threading.Thread(target=somemethod, args=(i,))

(refer program thread1.py)
```

Naming of threads

```
threading.currentThread().getName()


threading.Thread(name = 'myname' + str(i),  target=somemethod,
args=(i,))

(refer program thread3.py)
```

Talk to each other

```
threading.Event()

(refer to program thread4.py)
```

Locking a resource

Execute thread2.py

Note that the output is garbled

Execute thread6.py

Note how lock solved the problem

This has made the code “Thread Safe”