

Multithreading

LifeCycle

- ◆ **new** -> when you create a thread
- ◆ **runnable** -> after invocation of start()
- ◆ **running** -> when thread scheduler actually selects the thread
- ◆ **non-runnable (blocked)** -> sleep(), wait(), suspend()
- ◆ **terminated** -> end of run() or stop()

Creation

- ◆ By extending thread class
- ◆ By implementing runnable interface

Eg1 :-

```
class Example extends Thread {  
    public void run {  
        syso("Thread running..");  
    }  
  
    pvsm(){  
        Example e = new Example();  
        e.start();  
    }  
}
```

Eg2 :-

```
class Example implements Runnable {  
    public void run {  
        syso("Thread running..");  
    }  
  
    pvsm(){  
        Example e = new Example();  
        Thread t = new Thread(e);  
        t.start();  
    }  
}
```