**Lab Sheet 5.1**: Create three threads by setting different priorities to each thread.

## Sol:

class ThreadPrior extends Thread {

public void run()

{

// Print statement

System.out.println("Inside run method");

}

}

public class TestThreadPrior {

public static void main(String[] args)

{

ThreadPrior t1 = new ThreadPrior();

ThreadPrior t2 = new ThreadPrior();

ThreadPrior t3 = new ThreadPrior();

System.out.println("t1 thread priority : "+ t1.getPriority());

System.out.println("t2 thread priority : "+ t2.getPriority());

System.out.println("t3 thread priority : "+ t3.getPriority());

t1.setPriority(2);

t2.setPriority(5);

t3.setPriority(8);

t3.setPriority(21); //error

System.out.println("t1 thread priority : " + t1.getPriority());

System.out.println("t2 thread priority : "+ t2.getPriority());

System.out.println("t3 thread priority : " + t3.getPriority());

// Main thread

System.out.println("Currently Executing Thread : "+Thread.currentThread().getName());

System.out.println(

"Main thread priority : "+ Thread.currentThread().getPriority());

// Main thread priority is set to 10

Thread.currentThread().setPriority(10);

System.out.println(

"Main thread priority : "+ Thread.currentThread().getPriority());

}

}

**Lab Sheet 5.2**: Demonstrate Thread Synchronization for a given resource to avoid race condition.

* Create a Resource class to keep two resources [ and ]. No thread can take ] without [
* Create three threads to access the above resource without synchronization
* Access the above resource using synchronization

## Sol:

class Resource {

void use(String name) {

System.out.print("[" + name);

try {

Thread.sleep(1000);

} catch(InterruptedException e) {

System.out.println("Interrupted");

}

System.out.println("]");

}

}

class MyThread extends Thread {

String name;

Resource r;

MyThread (String name,Resource r){

super(name);

this.name = name;

this.r=r;

}

public void run() {

synchronized(r) {

r.use(name);

}

}

}

public class TestMultiThread {

public static void main(String args[]) {

Resource res=new Resource();

MyThread t1=new MyThread("1st",res);

MyThread t2=new MyThread("2nd",res);

MyThread t3=new MyThread("3rd",res);

t1.start();

t2.start();

t3.start();

try {

t1.join();

t2.join();

t3.join();

} catch (InterruptedException excetion) {

System.out.println("Inturruption occurs in Main Thread");

}

}

}