**Practical-9**

**Multithreading**

|  |
| --- |
| 1. Write a Java program to do the following using threads created by extending Thread class,  * Create 3 threads and display their properties as [thread-name, priority, group-name] – also display similar properties for main thread. Override run() methods in the threads to display numbers from 1 to 5, 1 to 6 and 1 to 7 respectively. * Use following methods in the program as explained, o isAlive() – show whether the three threads are alive before and after using the start() method * sleep() – Use 1000, 2000 and 3000 milliseconds as the argument for the three threads respectively * Display messages to indicate start and end of the 3 threads as well as the main thread. |

**Code:-**

classThreadDemo extends Thread

{

ThreadDemo(ThreadGrouptp,String nm)

{

super(tp,nm);

}

public void run()

{

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Thread = " +Thread.currentThread().getName() + " Start\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println(Thread.currentThread().getName()+" GroupName=" +Thread.currentThread().getThreadGroup());

System.out.println(Thread.currentThread().getName()+" Thread Priority=" +Thread.currentThread().getPriority());

try{Thread.currentThread().sleep(1000); }

catch(InterruptedException e){}

for(inti=1;i<=7;i++)

{

if(i==6 &&Thread.currentThread().getName().equals("Thread Demo-1") )

{

break;

}

if(i==7 &&Thread.currentThread().getName().equals("Thread Demo-2") )

{

break;

}

System.out.println("Thread "+Thread.currentThread().getName()+" =" +i);

}

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" +Thread.currentThread().getName() +" Exit\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" );

}

}

class Lab9\_1

{

public static void main(String args[])throws InterruptedException

{

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Thread= " +Thread.currentThread().getName() + " Start\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

ThreadGrouptp=new ThreadGroup("Main Group");

tp.setMaxPriority(Thread.NORM\_PRIORITY);

System.out.println("Defualt currently Active group=" +tp.activeCount());

ThreadDemo t1=new ThreadDemo(tp,"ThreadDemo");

t1.setName("Thread Demo-1");

t1.setPriority(Thread.MIN\_PRIORITY);

t1.start();

System.out.println("Thread"+t1.getName()+"is alive= " +t1.isAlive());

System.out.println("After thread 1currently Active group=" +tp.activeCount());

ThreadDemo t2=new ThreadDemo(tp,"Thread Demo-2");

t2.setPriority(Thread.NORM\_PRIORITY);

t2.start();

System.out.println("Thread"+t2.getName()+"is alive= " +t2.isAlive());

System.out.println("After thread 2 currently Active group=" +tp.activeCount());

ThreadDemo t3=new ThreadDemo(tp,"ThreadDemo-3");

t3.setPriority(Thread.MAX\_PRIORITY);

t3.start();

System.out.println("After thread 3 currently Active group=" +tp.activeCount());

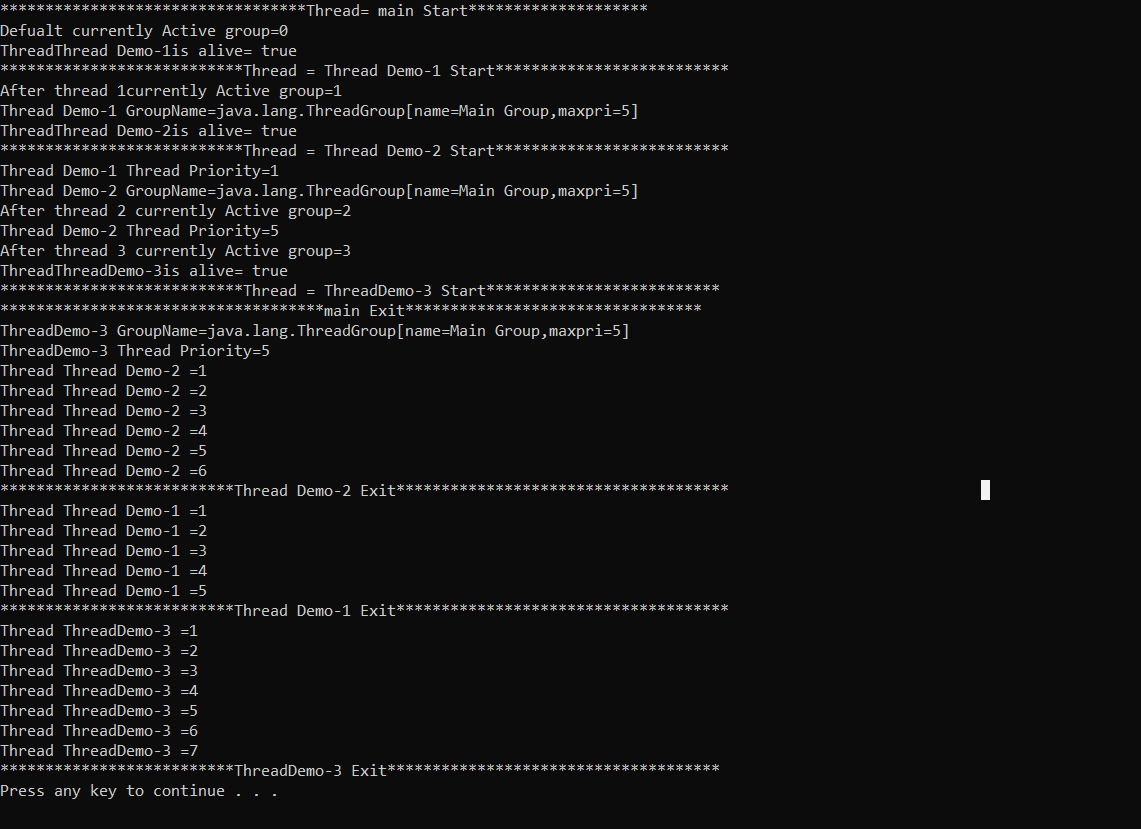
System.out.println("Thread"+t3.getName()+"is alive= " +t3.isAlive());

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" +Thread.currentThread().getName() +" Exit\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" );

}

}

**Output:-**



|  |
| --- |
| 1. Write a Java Program for TrafficLight using Applet and Multithreading. Create three circle to display Red, Green and Yellow light which should simulate traffic light. The switching between Red, Green and Yellow should take place after every 10 seconds |

**Code:-**

importjava.applet.Applet;

importjava.awt.\*;

public class Lab9\_3 extends Applet implements Runnable

{

Thread t;

int a=0;

public void init()

{

t=new Thread(this);

t.start();

}

public void run()

{

while(true)

{

try

{

Thread.sleep(1000);

a++;

repaint();

Thread.sleep(1000);

a++;

repaint();

Thread.sleep(1000);

a++;

repaint();

}

catch(InterruptedException e)

{

e.printStackTrace();

}

}

}

public void paint(Graphics g)

{

g.drawRect(30,30,300,100);

g.setColor(Color.black);

g.drawOval(50,50,50,50);

g.drawOval(150,50,50,50);

g.drawOval(250,50,50,50);

if(a==1)

{

g.setColor(Color.red);

g.fillOval(50,50,50,50);

}

if(a==2)

{

g.setColor(Color.orange);

g.fillOval(150,50,50,50);

}

if(a==3)

{

g.setColor(Color.green);

g.fillOval(250,50,50,50);

}

}

}

**Output:-**

