Program to demonstrate multithreading using Runnable interface. Define three different threads, one to calculate Square of first n integer, another to calculate cube of first n integer and third thread. To find the square root of the first n integers. Apply various thread priority.

import java.io.\*;

class T1 implements Runnable

{

int n;

T1(int p)

{

n=p;

}

public void run()

{

System.out.println("First Thread Started");

for(int i=0;i<n;i++)

{

System.out.println("Square of "+i+" is "+(i\*i));

}

System.out.println("First Thread Ended");

}

}

class T2 implements Runnable

{

int n;

T2(int p)

{

n=p;

}

public void run()

{

System.out.println("Second Thread Started");

for(int i=0;i<n;i++)

{

System.out.println("Cube of "+i+" is "+(i\*i\*i));

}

System.out.println("Second Thread Ended");

}

}

class T3 implements Runnable

{

int n;

T3(int p)

{

n=p;

}

public void run()

{

System.out.println("Third Thread Started");

for(int i=0;i<n;i++)

{

System.out.println("Square root of "+i+" is "+ Math.sqrt(i));

}

System.out.println("Third Thread Ended");

}

}

class ThMethods

{

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

{

int n;

DataInputStream in = new DataInputStream(System.in);

System.out.println("Enter the Value for n");

n = Integer.parseInt(in.readLine());

T1 obj1 = new T1(n);

T2 obj2 = new T2(n);

T3 obj3 = new T3(n);

Thread obj4 = new Thread(obj1);

Thread obj5 = new Thread(obj2);

Thread obj6 = new Thread(obj3);

obj4.setPriority(3);

obj6.setPriority(10);

obj4.start();

obj5.start();

obj6.start();

}

}

**OUTPUT:**

Enter the Value for n 4

Second Thread Started

Third Thread Started

First Thread Started

Cube of 0 is 0

Square root of 0 is 0.0

Square root of 1 is 1.0

Square of 0 is 0

Cube of 1 is 1

Cube of 2 is 8

Cube of 3 is 27

Second Thread Ended

Square of 1 is 1

Square root of 2 is 1.4142135623730951

Square root of 3 is 1.7320508075688772

Third Thread Ended

Square of 2 is 4

Square of 3 is 9

First Thread Ended