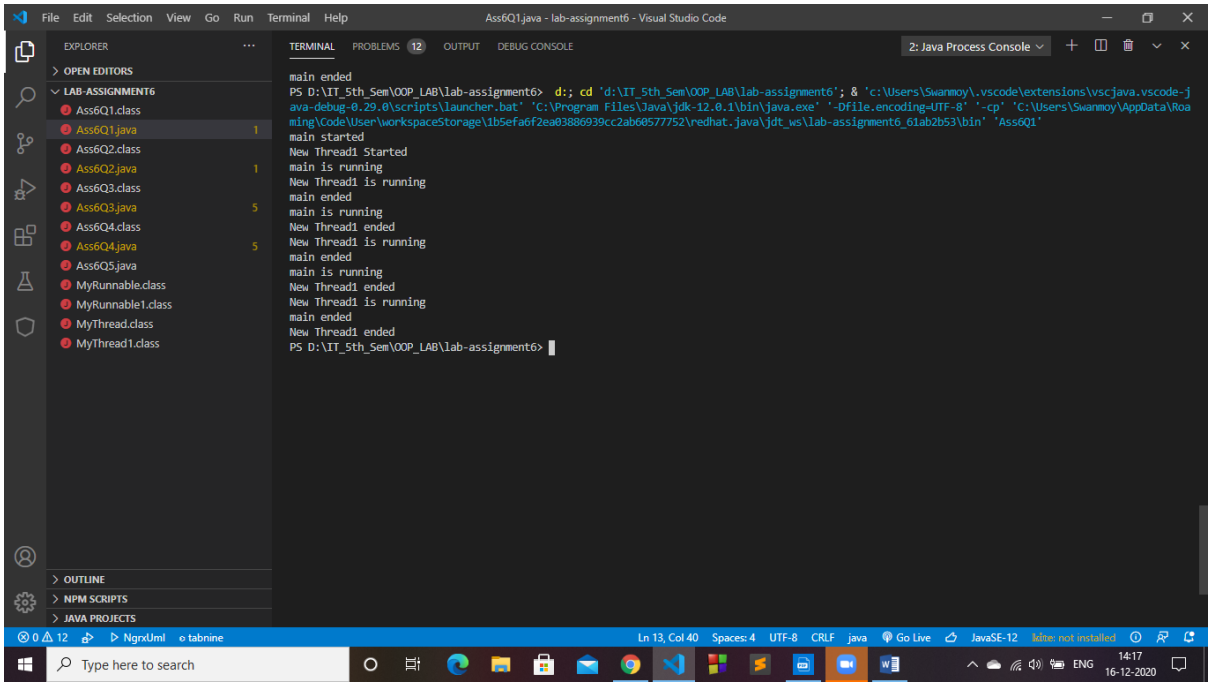
1.

Code

```
public class Ass6Q1
{
    public static void main(String[] args)
    {
        System.out.println(Thread.currentThread().getName()+" started");
        MyThread t=new MyThread("New Thread1");
        for(int i=0;i<3;i++)
        {
            System.out.println(Thread.currentThread().getName()+" is running")
;
            try {
                Thread.sleep(500);
            } catch (InterruptedException ex) {
                System.out.println(ex);
            }
            System.out.println(Thread.currentThread().getName()+" ended");
        }
    }
}
class MyThread extends Thread
{
    public MyThread(String name)
    {
        super(name);
        System.out.println(name+" Started");
        start();
    }
    public void run()
    {
        for(int i=0;i<3;i++)
        {
            System.out.println(Thread.currentThread().getName()+" is running")
;
            try {
                Thread.sleep(500);
            } catch (InterruptedException ex) {
                System.out.println(ex);;
            }
            System.out.println(Thread.currentThread().getName()+" ended");
        }
        return;
    }
}
```

Output



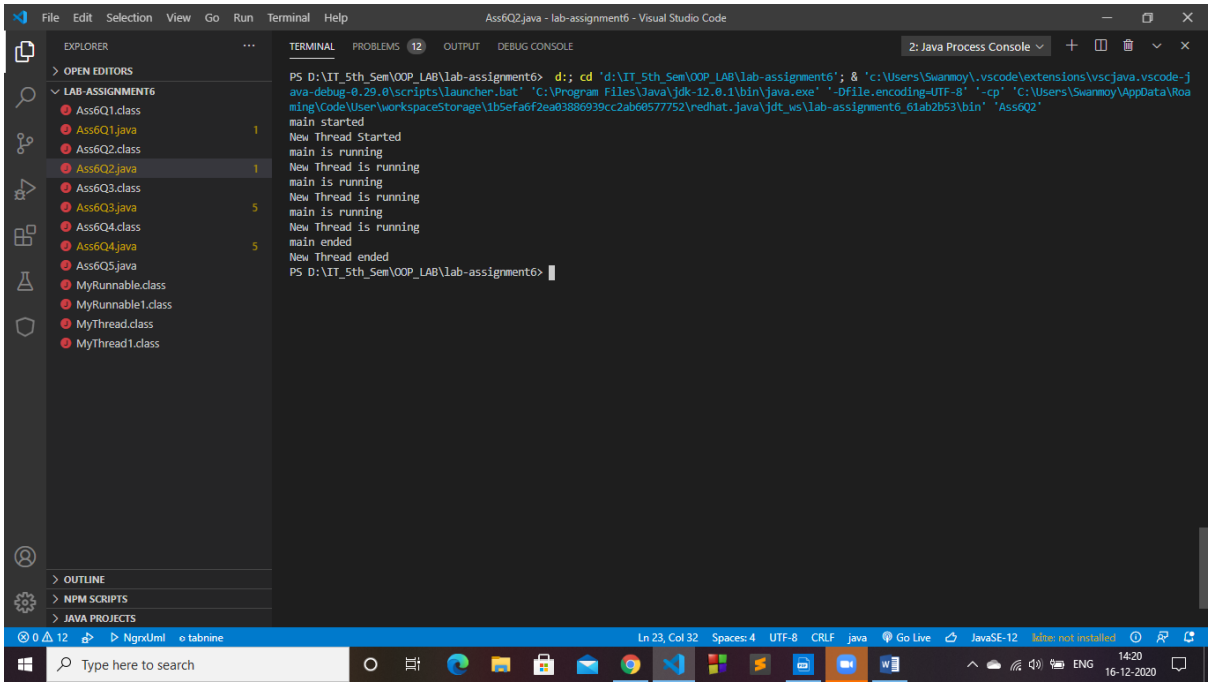
2.

Code

```
public class Ass6Q2 {
    public static void main(String[] args)
    {
        System.out.println(Thread.currentThread().getName()+" started");
        MyRunnable ob=new MyRunnable("New Thread");
        for(int i=0;i<3;i++)
        {
            System.out.println(Thread.currentThread().getName()+" is running")
;
            try {
                Thread.sleep(500);
            } catch (InterruptedException ex) {
                System.out.println(ex);;
            }
        }
        System.out.println(Thread.currentThread().getName() + " ended");
    }
}
class MyRunnable implements Runnable
{
    public MyRunnable(String name)
    {
        Thread t=new Thread(this, name);
        System.out.println(name+ " Started");
```

```
        t.start();
    }
    public void run()
    {
        for(int i=0;i<3;i++)
        {
            System.out.println(Thread.currentThread().getName()+" is running")
;
            try {
                Thread.sleep(500);
            } catch (InterruptedException ex) {
                System.out.println(ex);;
            }
        }
        System.out.println(Thread.currentThread().getName() + " ended");
        }
    }
```

Output



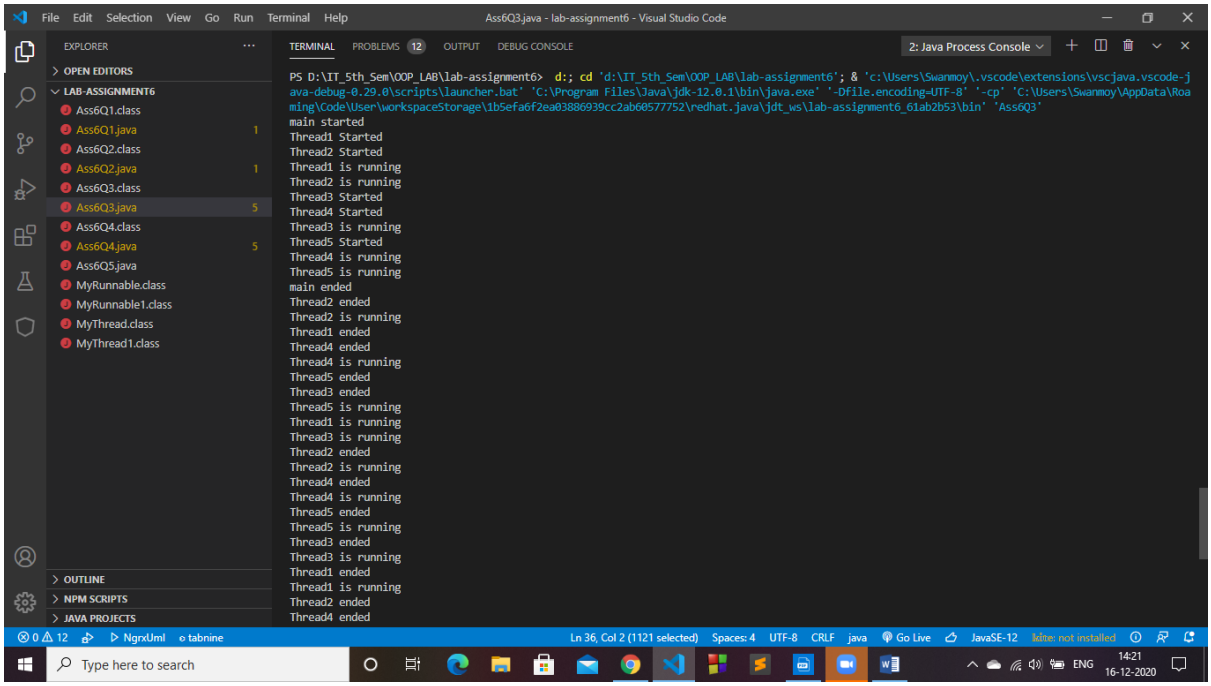
3.

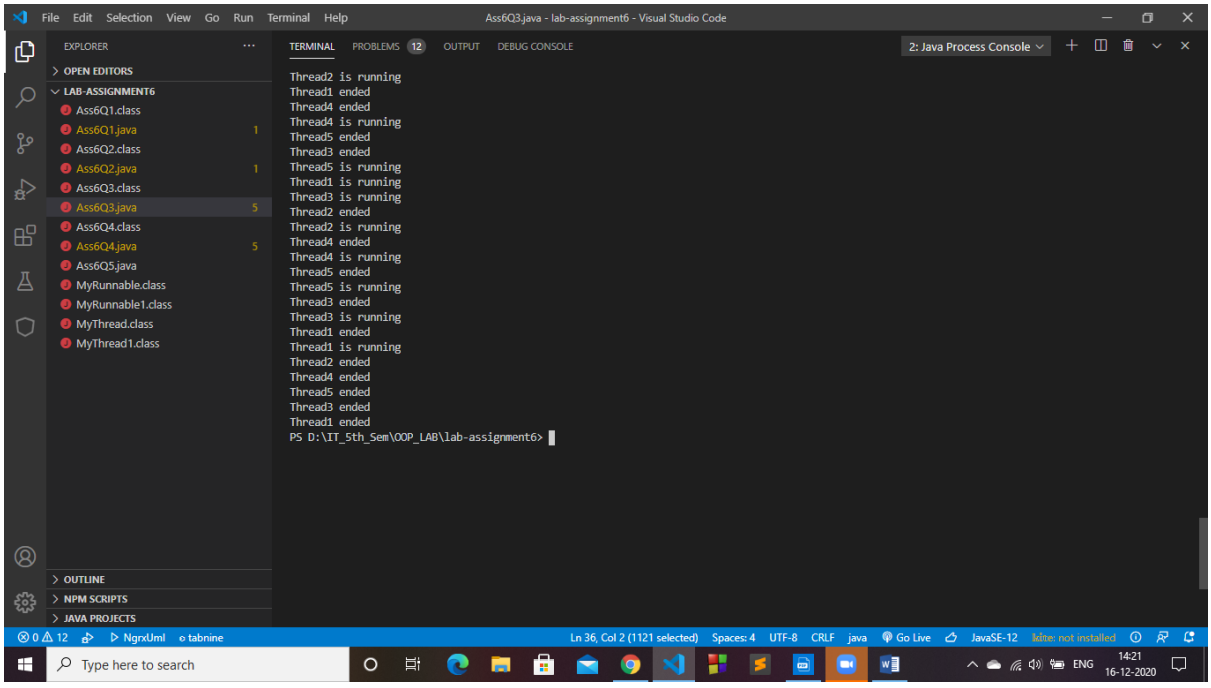
Code

```
public class Ass6Q3 {
    public static void main(String[] args)
    {
        System.out.println(Thread.currentThread().getName()+" started");
```

```
        MyThread1 t1=new MyThread1("Thread1", 2);
        MyThread1 t2=new MyThread1("Thread2", 4);
        MyThread1 t3=new MyThread1("Thread3", 1);
        MyThread1 t4=new MyThread1("Thread4", 5);
        MyThread1 t5=new MyThread1("Thread5", 3);
        System.out.println(Thread.currentThread().getName() + " ended");
    }
}
class MyThread1 extends Thread
{
    public MyThread1(String name, int pr)
    {
        super(name);
        System.out.println(name+" Started");
        this.setPriority(pr);
        start();
    }
    public void run()
    {
        for(int i=0;i<3;i++)
        {
            System.out.println(Thread.currentThread().getName()+" is running")
;
            try {
                Thread.sleep(500);
            } catch (InterruptedException ex) {
                System.out.println(ex);;
            }
            System.out.println(Thread.currentThread().getName()+" ended");
        }
        return;
    }
}
```

Output





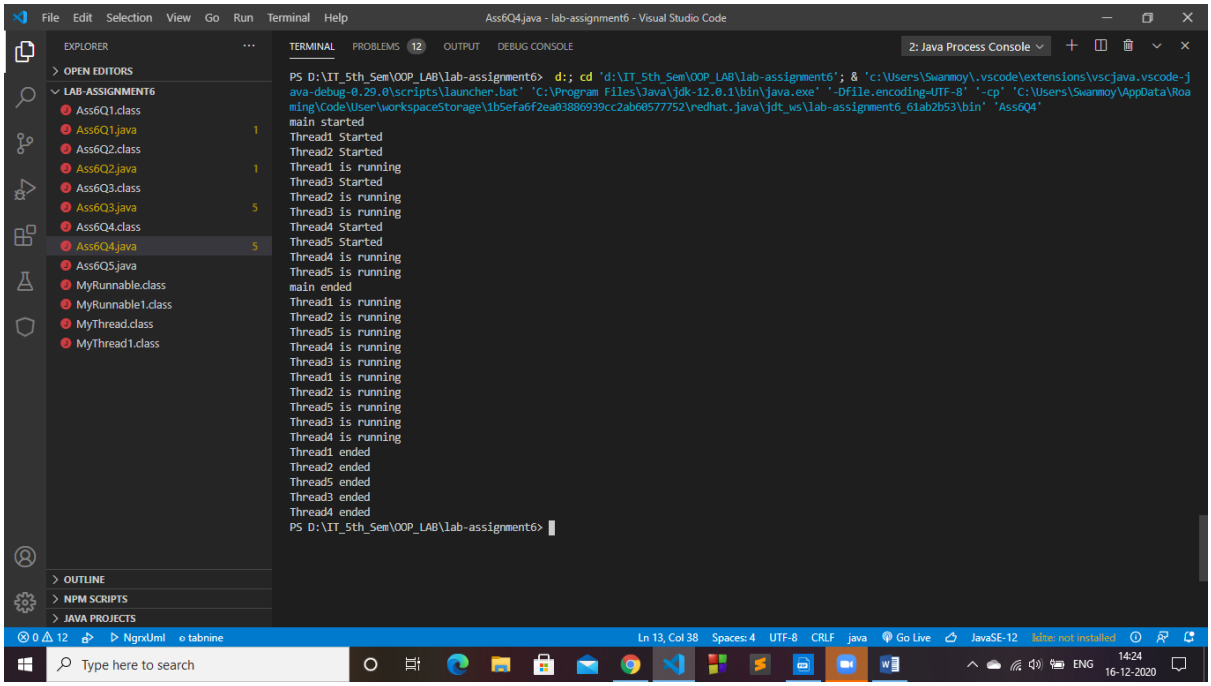
4.

Code

```
public class Ass6Q4 {
    public static void main(String[] args)
    {
        System.out.println(Thread.currentThread().getName()+" started");
        MyRunnable1 t1=new MyRunnable1("Thread1", 5);
        MyRunnable1 t2=new MyRunnable1("Thread2", 1);
        MyRunnable1 t3=new MyRunnable1("Thread3", 2);
```

```
        MyRunnable1 t4=new MyRunnable1("Thread4", 4);
        MyRunnable1 t5=new MyRunnable1("Thread5", 3);
        System.out.println(Thread.currentThread().getName() + " ended");
    }
}
class MyRunnable1 implements Runnable
{
    public MyRunnable1(String name, int pr)
    {
        Thread t=new Thread(this, name);
        System.out.println(name+ " Started");
        t.setPriority(pr);
        t.start();
    }
    public void run()
    {
        for(int i=0;i<3;i++)
        {
            System.out.println(Thread.currentThread().getName()+" is running")
;
            try {
                Thread.sleep(500);
            } catch (InterruptedException ex) {
                System.out.println(ex);;
            }
        }
        System.out.println(Thread.currentThread().getName() + " ended");
        }
    }
```

Output



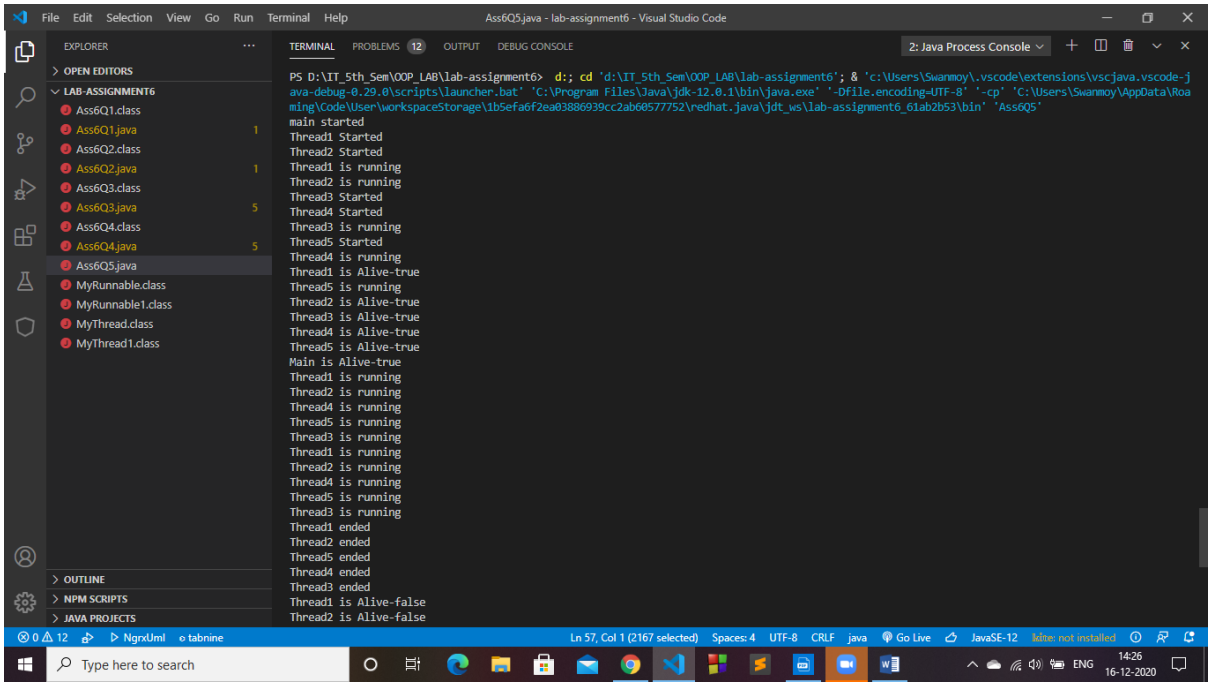
5.

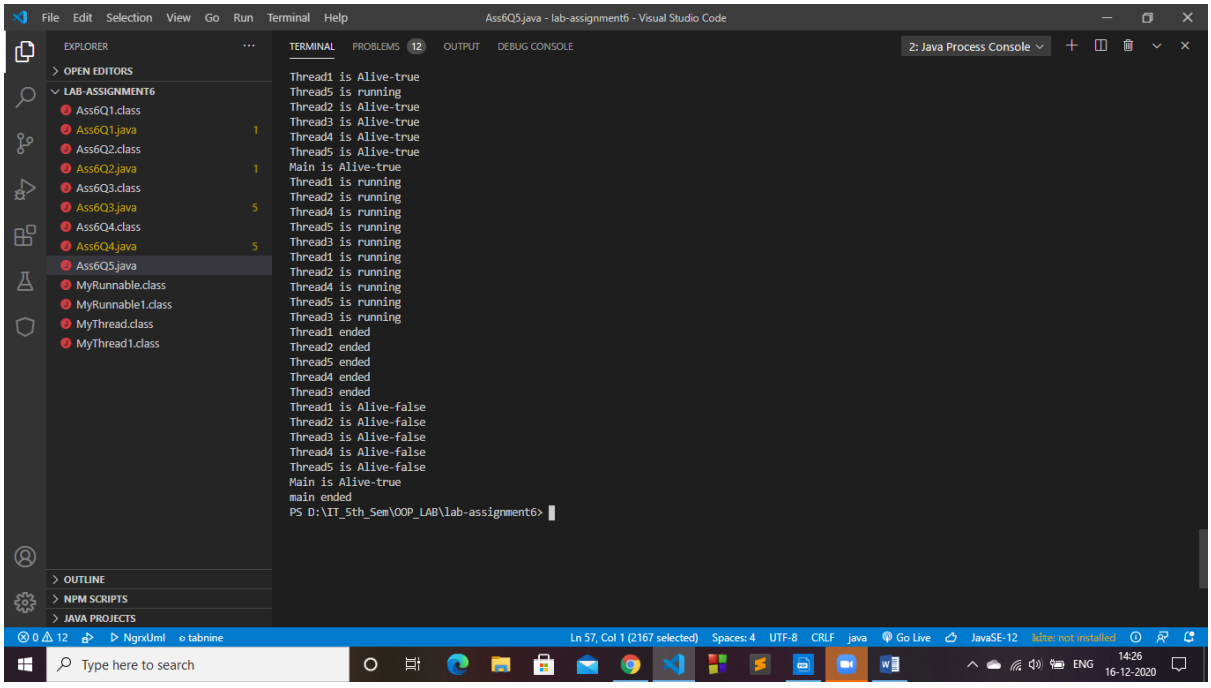
Code

```
public class Ass6Q5 {
    public static void main(String[] args)
    {
        System.out.println(Thread.currentThread().getName()+" started");
        MyRunnable2 t1=new MyRunnable2("Thread1", 5);
        MyRunnable2 t2=new MyRunnable2("Thread2", 1);
        MyRunnable2 t3=new MyRunnable2("Thread3", 2);
        MyRunnable2 t4=new MyRunnable2("Thread4", 4);
        MyRunnable2 t5=new MyRunnable2("Thread5", 3);
        System.out.println("Thread1 is Alive-"+ t1.isAlive());
        System.out.println("Thread2 is Alive-"+ t2.isAlive());
        System.out.println("Thread3 is Alive-"+ t3.isAlive());
        System.out.println("Thread4 is Alive-"+ t4.isAlive());
        System.out.println("Thread5 is Alive-"+ t5.isAlive());
        System.out.println("Main is Alive-
"+ Thread.currentThread().isAlive());
        try {
            t1.join();
            t2.join();
            t3.join();
            t4.join();
            t5.join();
        } catch (InterruptedException ex) {
            System.out.println(ex);
```

```
        }
        System.out.println("Thread1 is Alive-"+ t1.isAlive());
        System.out.println("Thread2 is Alive-"+ t2.isAlive());
        System.out.println("Thread3 is Alive-"+ t3.isAlive());
        System.out.println("Thread4 is Alive-"+ t4.isAlive());
        System.out.println("Thread5 is Alive-"+ t5.isAlive());
        System.out.println("Main is Alive-
"+ Thread.currentThread().isAlive());
        System.out.println(Thread.currentThread().getName() + " ended");
    }
}
class MyRunnable2 extends Thread
{
    public MyRunnable2(String name, int pr)
    {
        super(name);
        System.out.println(name+ " Started");
        this.setPriority(pr);
        start();
    }
    public void run()
    {
        for(int i=0;i<3;i++)
        {
            System.out.println(Thread.currentThread().getName()+" is running")
;
            try {
                Thread.sleep(500);
            } catch (InterruptedException ex) {
                System.out.println(ex);;
            }
        }
        System.out.println(Thread.currentThread().getName() + " ended");
        }
    }
```

Output


File Edit Selection View Go Run Terminal Help
Ass6Q5.java - lab-assignment6 - Visual Studio Code
EXPLORER
OPEN EDITORS
LAB-ASSIGNMENT6
Ass6Q1.class
Ass6Q1.java 1
Ass6Q2.class
Ass6Q2.java 1
Ass6Q3.class
Ass6Q3.java 5
Ass6Q4.class
Ass6Q4.java 5
Ass6Q5.java
MyRunnable.class
MyRunnable1.class
MyThread.class
MyThread1.class
OUTLINE
NPM SCRIPTS
JAVA PROJECTS
TERMINAL PROBLEMS 12 OUTPUT DEBUG CONSOLE
2: Java Process Console
PS D:\IT_5th_Sem\OOP_LAB\lab-assignment6> d:; cd 'd:\IT_5th_Sem\OOP_LAB\lab-assignment6'; & 'c:\Users\Swanmoy\.vscode\extensions\vscjava.vscode-java-debug-0.29.0\scripts\launcher.bat' 'C:\Program Files\Java\jdk-12.0.1\bin\java.exe' '-Dfile.encoding=UTF-8' '-cp' 'C:\Users\Swanmoy\AppData\Roaming\Code\User\workspaceStorage\1b5efa6f2ea03886939cc2ab60577752\redhat.java\jdt_ws\lab-assignment6_61ab2b53\bin' 'Ass6Q5'
main started
Thread1 Started
Thread2 Started
Thread1 is running
Thread2 is running
Thread3 Started
Thread4 Started
Thread3 is running
Thread5 Started
Thread4 is running
Thread1 is Alive-true
Thread5 is running
Thread2 is Alive-true
Thread3 is Alive-true
Thread4 is Alive-true
Thread5 is Alive-true
Main is Alive-true
Thread1 is running
Thread2 is running
Thread4 is running
Thread5 is running
Thread3 is running
Thread1 is running
Thread2 is running
Thread4 is running
Thread5 is running
Thread3 is running
Thread1 ended
Thread2 ended
Thread5 ended
Thread4 ended
Thread3 ended
Thread1 is Alive-false
Thread2 is Alive-false
Ln 57, Col 1 (2167 selected) Spaces: 4 UTF-8 CRLF java Go Live JavaSE-12
Type here to search
ENG 14:26 16-12-2020


File Edit Selection View Go Run Terminal Help
Ass6Q5.java - lab-assignment6 - Visual Studio Code
EXPLORER
OPEN EDITORS
LAB-ASSIGNMENT6
Ass6Q1.class
Ass6Q1.java 1
Ass6Q2.class
Ass6Q2.java 1
Ass6Q3.class
Ass6Q3.java 5
Ass6Q4.class
Ass6Q4.java 5
Ass6Q5.java
MyRunnable.class
MyRunnable1.class
MyThread.class
MyThread1.class
OUTLINE
NPM SCRIPTS
JAVA PROJECTS
TERMINAL PROBLEMS 12 OUTPUT DEBUG CONSOLE
2: Java Process Console
Thread1 is Alive-true
Thread5 is running
Thread2 is Alive-true
Thread3 is Alive-true
Thread4 is Alive-true
Thread5 is Alive-true
Main is Alive-true
Thread1 is running
Thread2 is running
Thread4 is running
Thread5 is running
Thread3 is running
Thread1 is running
Thread2 is running
Thread4 is running
Thread5 is running
Thread3 is running
Thread1 ended
Thread2 ended
Thread5 ended
Thread4 ended
Thread3 ended
Thread1 is Alive-false
Thread2 is Alive-false
Thread3 is Alive-false
Thread4 is Alive-false
Thread5 is Alive-false
Main is Alive-true
main ended
PS D:\IT_5th_Sem\OOP_LAB\lab-assignment6>
Ln 57, Col 1 (2167 selected) Spaces: 4 UTF-8 CRLF java Go Live JavaSE-12
Type here to search
ENG 14:26 16-12-2020