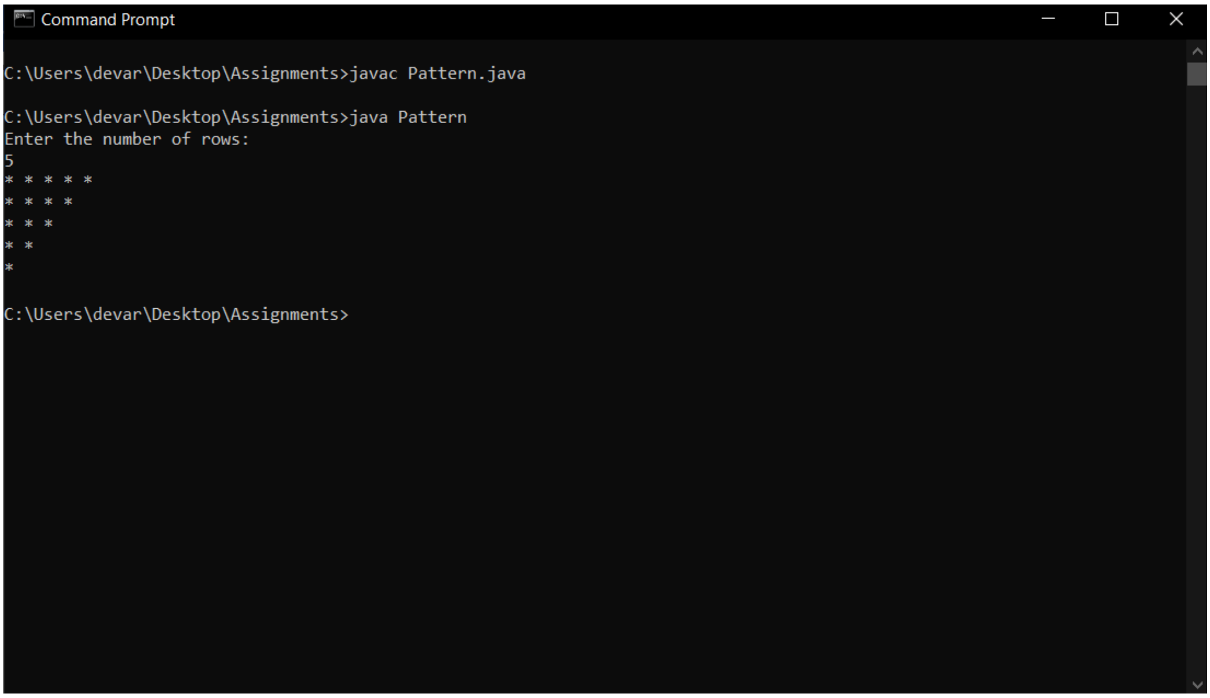
# ASSIGNMENT:-2,3

## 1.Write a program to show output like:

```
* * * * *

* * * *

* * *

* *

*
```

```
import java.util.Scanner;


public class Pattern {


    public static void main(String[] args)

    {

        Scanner sc = new Scanner(System.in);


        System.out.println("Enter the number of rows: ");


        int rows = sc.nextInt();
        for (int i= rows-1; i>=0 ; i--)
        {
            for (int j=0; j<=i; j++)
            {
                System.out.print("*" + " ");
            }
            System.out.println();
        }
        sc.close();
    }
}
```

OUTPUT:



```
C:\Users\devar\Desktop\Assignments>javac Pattern.java

C:\Users\devar\Desktop\Assignments>java Pattern
Enter the number of rows:
5
* * * * *
* * * *
* * *
* *
*


C:\Users\devar\Desktop\Assignments>
```

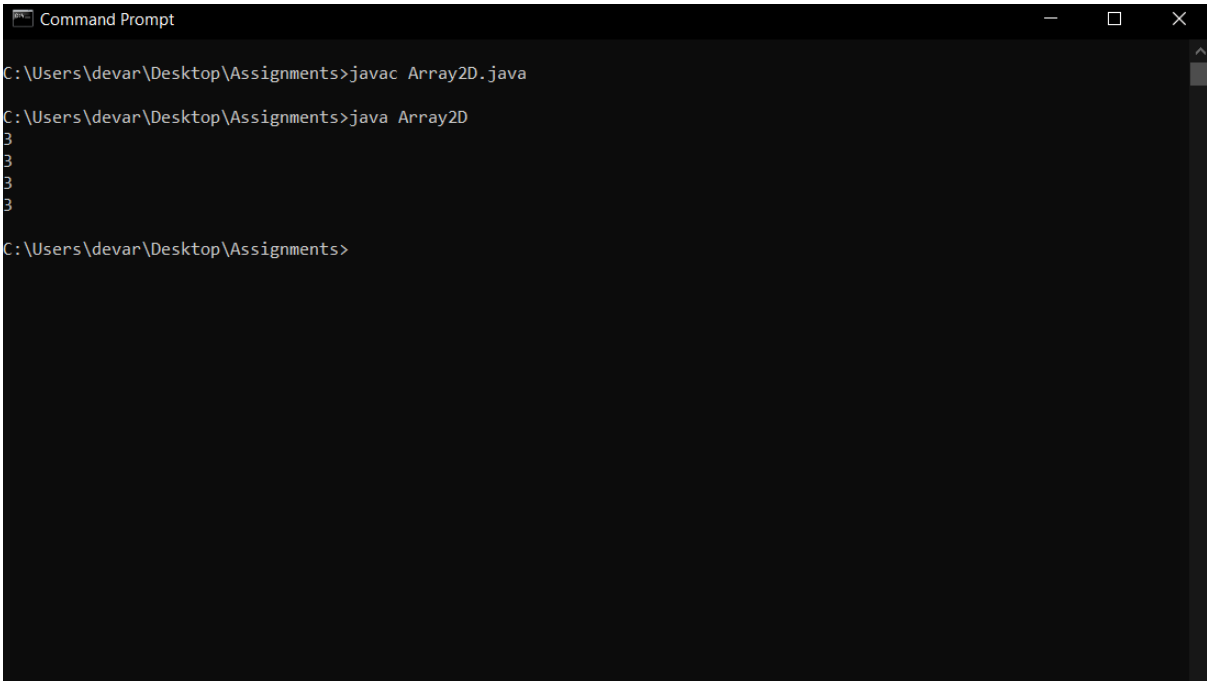
## 2. Write an application that creates a two dimension array with int values. The

## first, second and third elements should be arrays with one, two and three

## numbers respectively. Display the length of each dimension.

```
public class Array2D {
    public static void main(String[] args) {



        int[][] array2D = new int[3][3];


        int lengthOne = array2D.length;


        int lengthTwo = array2D[0].length;
        int lengthThree= array2D[1].length;
        int lengthfour = array2D[2].length;


       System.out.println(lengthOne);
       System.out.println(lengthTwo);
       System.out.println(lengthThree);
       System.out.println(lengthfour);


    }
  }
```

OUTPUT:



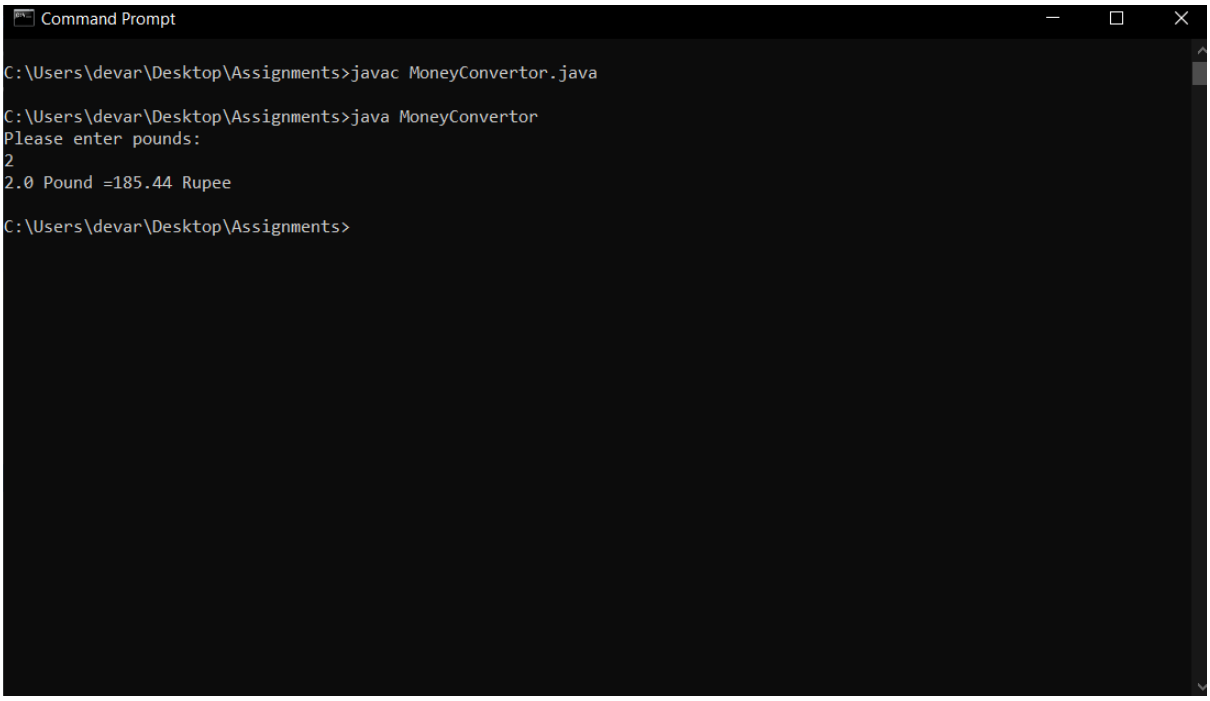
```
C:\Users\devar\Desktop\Assignments>javac Array2D.java

C:\Users\devar\Desktop\Assignments>java Array2D
3
3
3
3

C:\Users\devar\Desktop\Assignments>
```

## 3.Write a java program for converting Pound into Rupees. ( using scanner class

## also and take 1 Pound = 100 Rupees.)

```
import java.util.Scanner;

public class MoneyConvertor {
    public static void main(String[] args) {

      double pound;

      Scanner in = new Scanner(System.in);

      System.out.println("Please enter pounds:");

      pound = in.nextLong();

      double rupees = pound * 92.72;

      System.out.println(pound + " Pound ="+ rupees +"
Rupee");
    }
  }
```

OUTPUT:



```
C:\Users\devar\Desktop\Assignments>javac MoneyConvertor.java

C:\Users\devar\Desktop\Assignments>java MoneyConvertor
Please enter pounds:
2
2.0 Pound =185.44 Rupee

C:\Users\devar\Desktop\Assignments>
```

## 4. Create a class named 'Member' having the following members:

Data members

1 - Name

2 - Age

3 - Phone number

4 - Address

5 – Salary

It also has a method named 'printSalary' which prints the salary of the

members.

Two classes 'Employee' and 'Manager' inherits the 'Member' class. The

'Employee' and 'Manager' classes have data members 'specialization' and

'department' respectively. Now, assign name, age, phone number, address and

salary to an employee and a manager by making an object of both of these

classes and print the same.

```
class Member{

        String name;

        int age;

        String number;
        String address;
        int salary;

public void printSalary(){
        System.out.print(salary);
        }
}

class Employee extends Member {
    String specialization;
}

class Manager extends Member{
    String department;
}

class Ans{
    public static void main(String[] args){
        Employee e = new Employee();
        e.name = "xyz";
        e.age = 23;
        e.number = "888291211";
        e.address = "xyzxyz";
        e.salary = 21231;
        e.specialization = "xyzxyz";

        Manager m = new Manager();


        m.name = "ABC";
        m.age = 33;
        m.number = "9087653123";
        m.address = "NMKLOPSA";
        m.salary = 33903;
        m.department = "Department of science";

        System.out.println("Details of the
Employee:"+"\nName:"+e.name+"\nAge:"+e.age+"\nNumber
:"+e.number+"\nAddress:"+e.address+"\nSpecialization" +
e.specialization+"\nSalary:");
        e.printSalary();
```

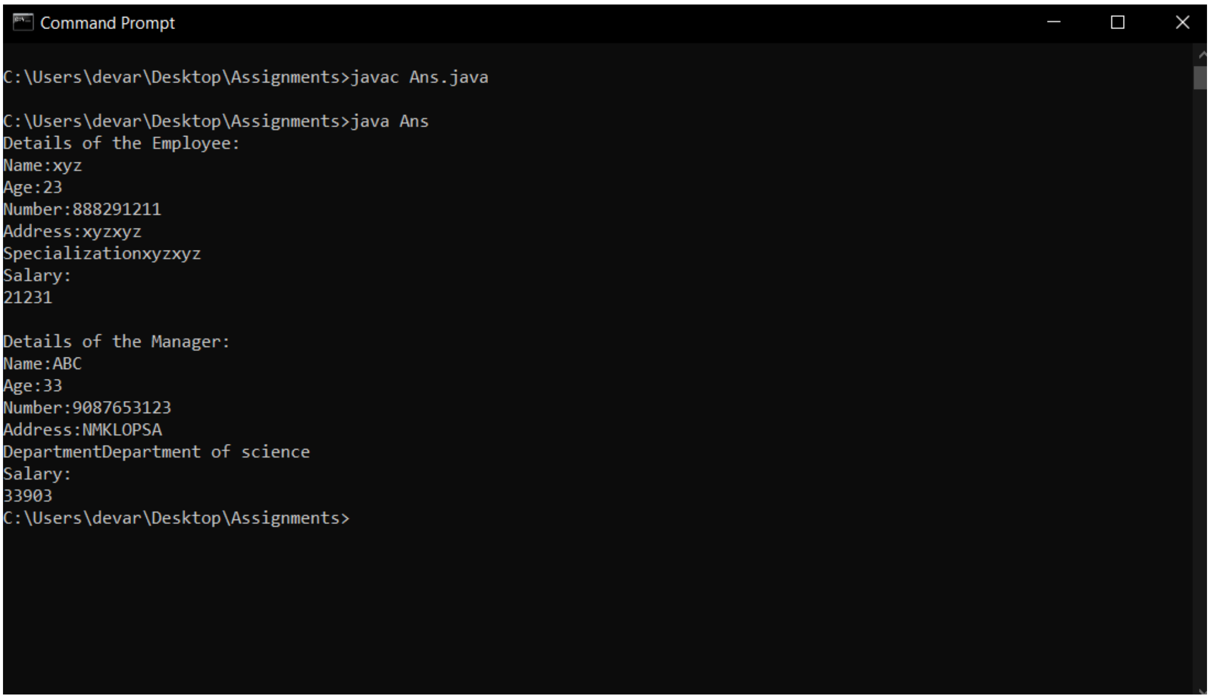
System.out.println("\n\nDetails of the Manager:"+"\nName:"+m.name+"\nAge:"+m.age+"\nNumber:"+m.number+"\nAddress:"+m.address+"\nDepartment" + m.department+"\nSalary:");

m.printSalary();

}

}

OUTPUT:

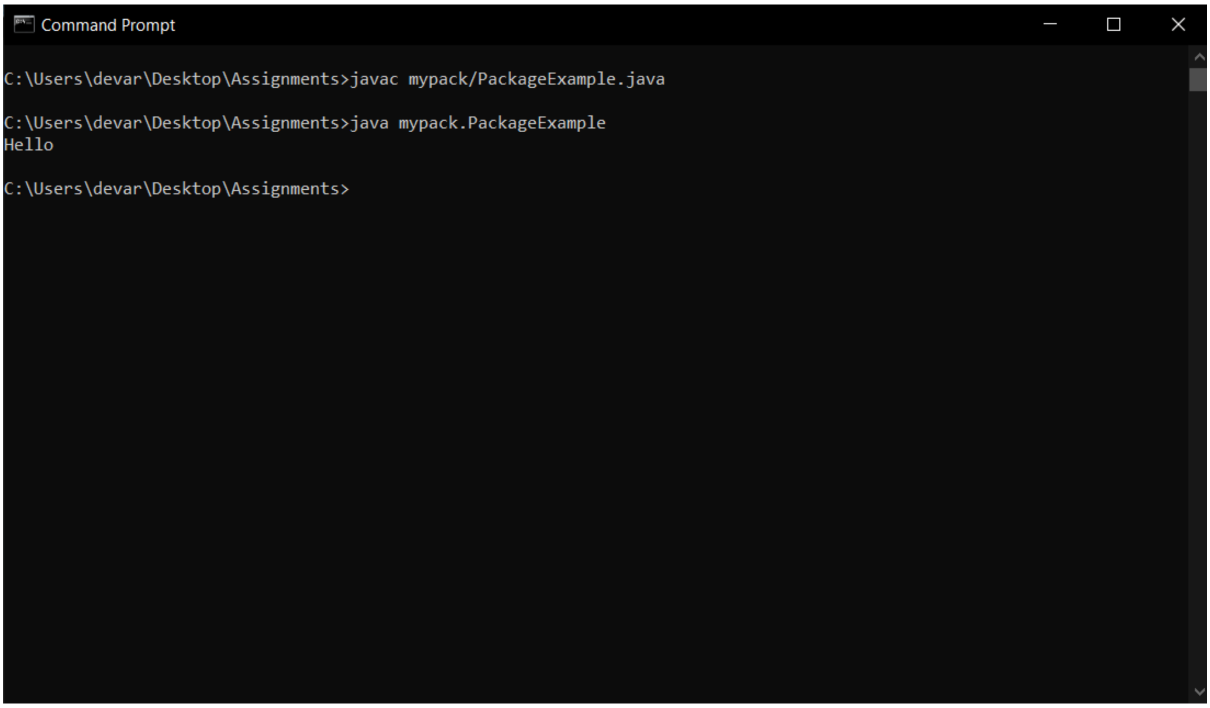


```
C:\Users\devar\Desktop\Assignments>javac Ans.java

C:\Users\devar\Desktop\Assignments>java Ans
Details of the Employee:
Name:xyz
Age:23
Number:888291211
Address:xyzxyz
Specializationxyzxyz
Salary:
21231

Details of the Manager:
Name:ABC
Age:33
Number:9087653123
Address:NMKLOPSA
DepartmentDepartment of science
Salary:
33903
C:\Users\devar\Desktop\Assignments>
```

## 5. Write a java program which shows importing of classes from other user define packages.

```
package pack;
//It is saved into pack folder



public class Demo {
    public void msg() {
        System.out.println("Hello");
    }
}

package mypack;
//It is saved into mypack folder

        import pack.Demo;
class PackageExample {
    public static void main(String args[]) {
        Demo obj = new Demo();
        obj.msg();
    }
}
```

OUTPUT:

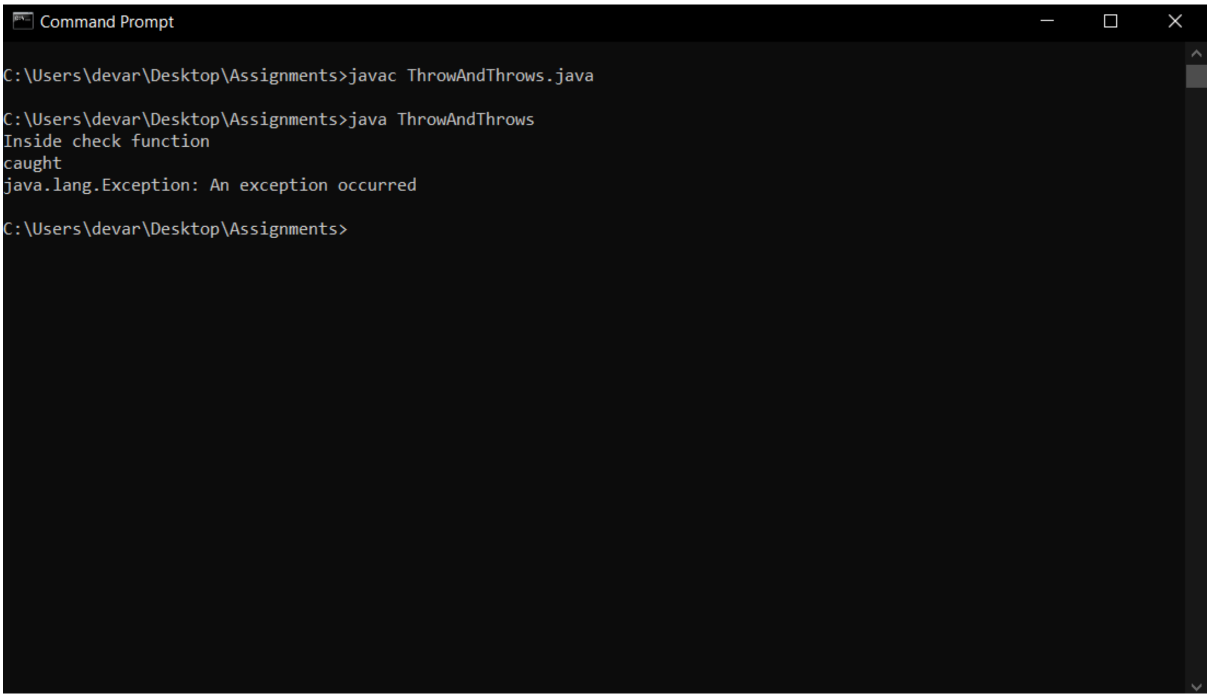


```
C:\Users\devar\Desktop\Assignments>javac mypack/PackageExample.java

C:\Users\devar\Desktop\Assignments>java mypack.PackageExample
Hello

C:\Users\devar\Desktop\Assignments>
```

## 6. Write a java program to generate user defined exception using “throw” and “throws” keyword.

```
class ThrowAndThrows
{
    static void check() throws Exception
    {
        System.out.println("Inside check function");
        throw new Exception("An exception occurred");
    }

    public static void main(String args[])
    {
        try
        {
            check();
        }
        catch(Exception e)
        {
            System.out.println("caught \n" + e);
        }
    }
}
```

OUTPUT:



```
C:\Users\devar\Desktop\Assignments>javac ThrowAndThrows.java

C:\Users\devar\Desktop\Assignments>java ThrowAndThrows
Inside check function
caught
java.lang.Exception: An exception occurred

C:\Users\devar\Desktop\Assignments>
```

## 7. Write a program to create thread which display “Hello World” message.

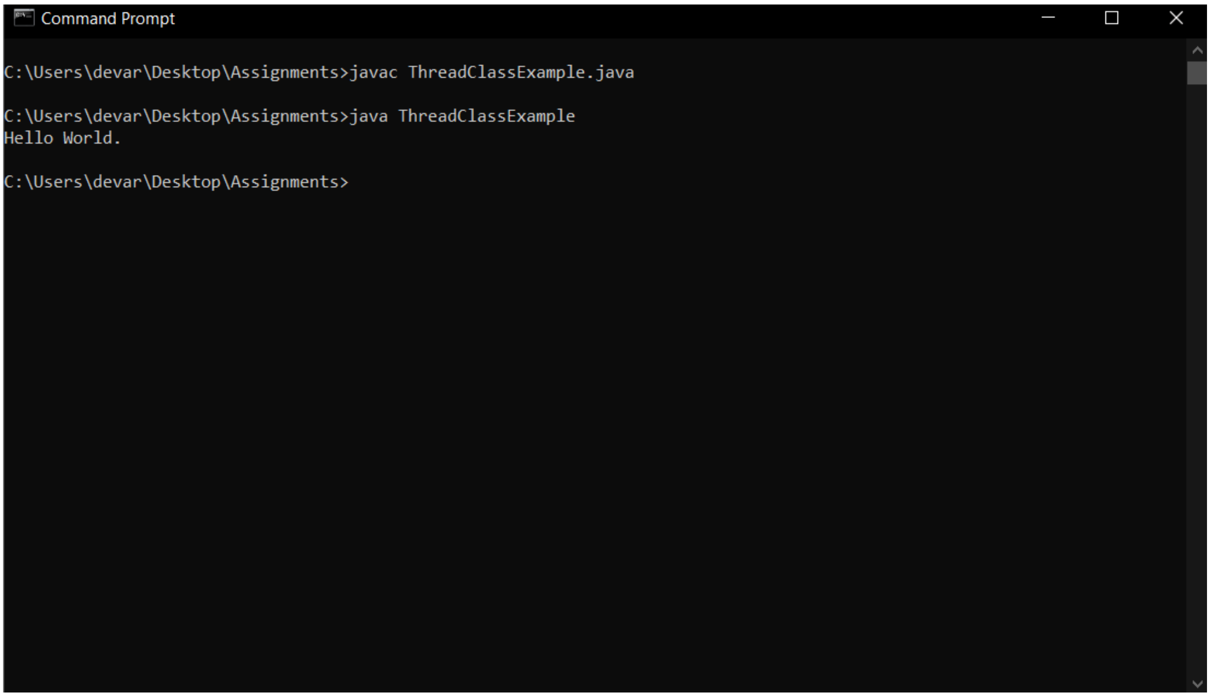
## A. by extending Thread class

## B. by using Runnable interface.

### **(i)by extending Thread class:**

```
class ThreadClassExample extends Thread{
    public void run(){
        System.out.println("Hello World.");
    }
    public static void main(String args[]){
        ThreadClassExample t1=new ThreadClassExample();
        t1.start();
    }
}
```

OUTPUT:

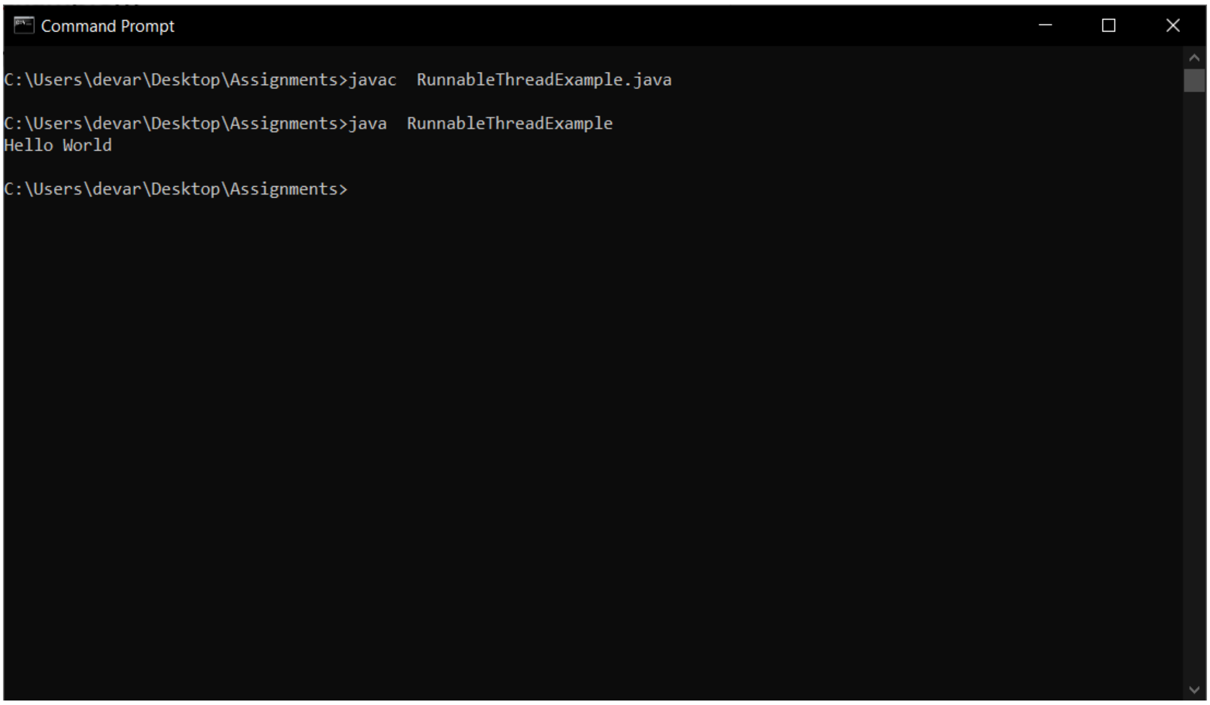


```
C:\Users\devar\Desktop\Assignments>javac ThreadClassExample.java

C:\Users\devar\Desktop\Assignments>java ThreadClassExample
Hello World.

C:\Users\devar\Desktop\Assignments>
```

### (ii)By using Runnable interface:

```
class RunnableThreadExample implements Runnable {

         public void run() {

         System.out.println("Hello World");

    }


    public static void main(String args[]) {

        RunnableThreadExample R = new
        RunnableThreadExample();

         Thread t1 = new Thread(R);

        t1.start();

    }

}
```

OUTPUT:

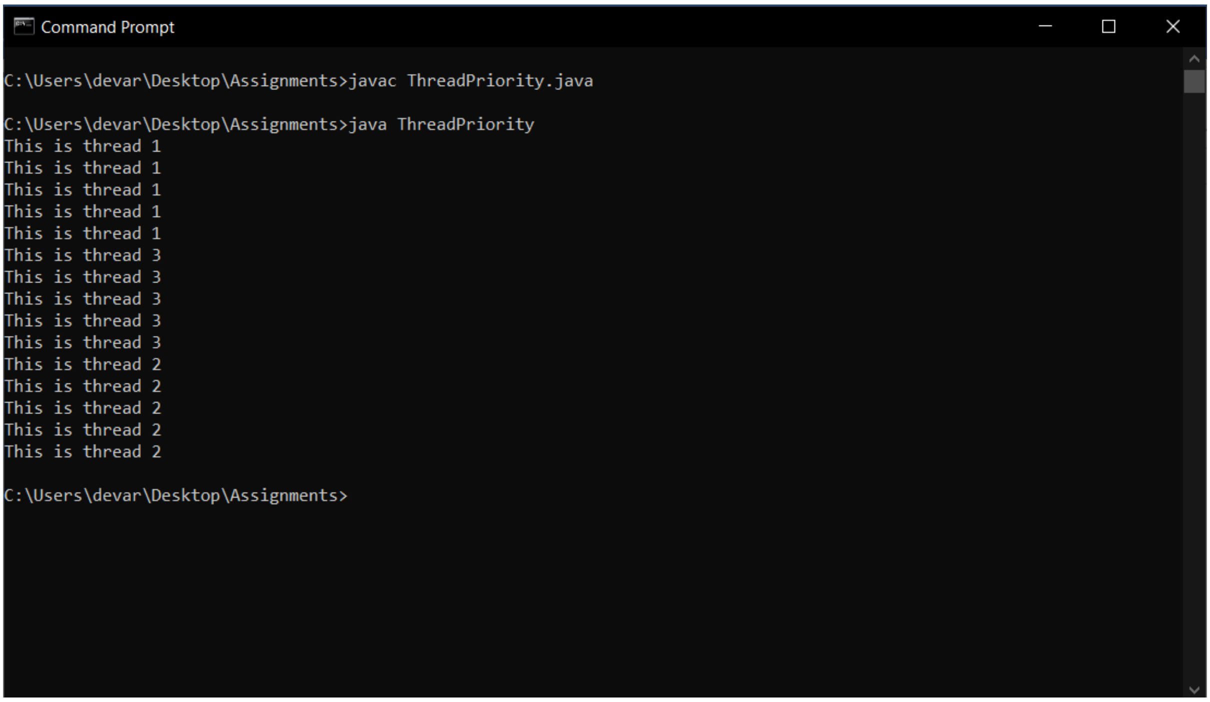


```
C:\Users\devar\Desktop\Assignments>javac  RunnableThreadExample.java

C:\Users\devar\Desktop\Assignments>java  RunnableThreadExample
Hello World

C:\Users\devar\Desktop\Assignments>
```

## 8. Write a program to create three threads ‘FIRST’, ‘SECOND’, ‘THIRD’. Set the priority of the ‘FIRST’ thread to 3, the ‘SECOND’ thread to 5(default) and the ‘THIRD’ thread to 7.

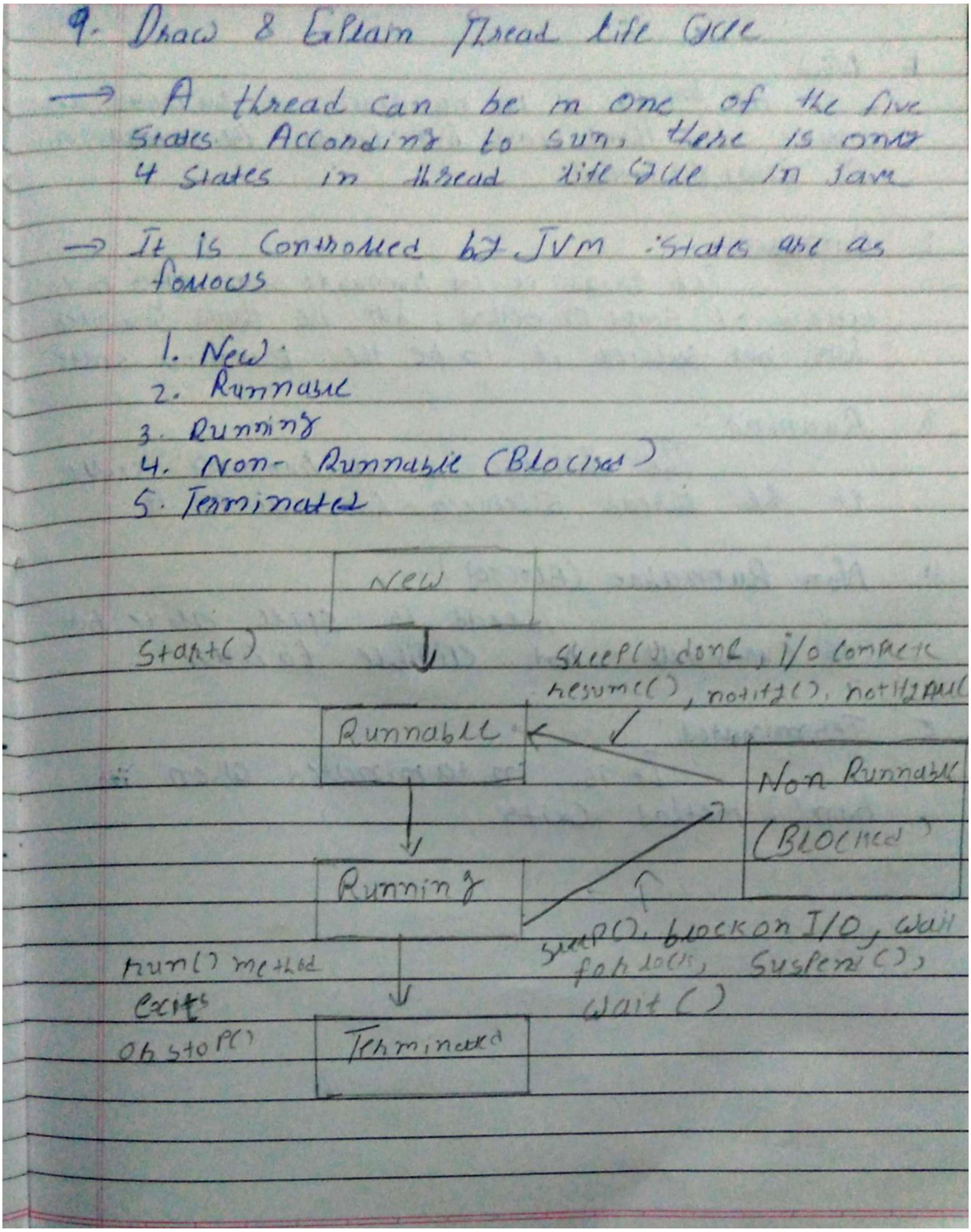
```
class Mythread1 extends Thread{
    public void run(){
        for (int i=0;i<5;i++) {


            System.out.println("This is thread 1");
        }

    }

}



class Mythread2 extends Thread{
    public void run(){
        for (int i=0;i<5;i++) {
            System.out.println("This is thread 2");
        }

    }

}

class Mythread3 extends Thread{
    public void run(){
        for (int i=0;i<5;i++) {
            System.out.println("This is thread 3");
        }


    }
}

class ThreadPriority{
    public static void main(String args[]) {
        Mythread1 t1 = new Mythread1();
        Mythread2 t2 = new Mythread2();
        Mythread3 t3= new Mythread3();
        t1.setPriority(3);
        t3.setPriority(7);
        t1.start();
        t2.start();
        t3.start();
    }
}
```

OUTPUT:



```
C:\Users\devar\Desktop\Assignments>javac ThreadPriority.java

C:\Users\devar\Desktop\Assignments>java ThreadPriority
This is thread 1
This is thread 1
This is thread 1
This is thread 1
This is thread 1
This is thread 3
This is thread 3
This is thread 3
This is thread 3
This is thread 3
This is thread 2
This is thread 2
This is thread 2
This is thread 2
This is thread 2

C:\Users\devar\Desktop\Assignments>
```

# 9. Draw & explain Thread life Cycle.



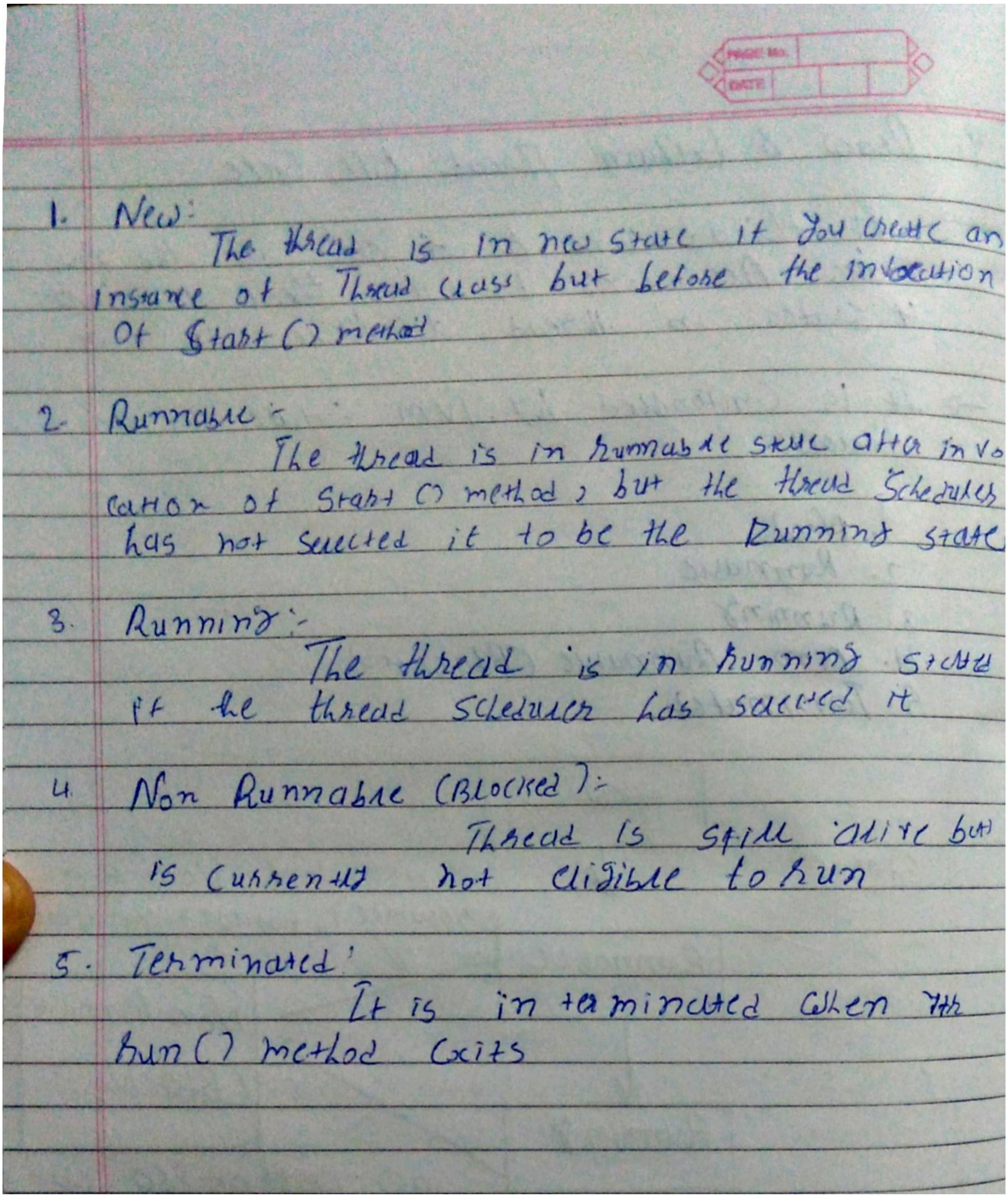
9. Draw & Explain Thread life Cycle

→ A thread can be in one of the five states. According to sun, there is only 4 states in thread life cycle in Java

→ It is Controlled by JVM. States are as follows

1. New.
2. Runnable
3. Running
4. Non- Runnable (Blocked)
5. Terminated

New
start()
sleep() done, i/o complete
resume(), notify(), notifyAll
Runnable
Non Runnable (Blocked)
Running
sleep(), block on I/O, wait for lock, suspend(), wait()
run() method exits or stop()
Terminated



1. New:
The thread is in new state if you create an instance of Thread class but before the invocation of start() method

2. Runnable:-
The thread is in runnable state after invocation of start() method, but the thread scheduler has not selected it to be the running state

3. Running:-
The thread is in running state if the thread scheduler has selected it

4. Non Runnable (Blocked):-
Thread is still alive but is currently not eligible to run

5. Terminated:
It is in terminated when the run() method exits

## 10. Write a java program that implements an interface AdvancedArithmetic which

contains a method signature int divisor_sum(int n).

You need to write a class called MyCalculator which implements the interface.

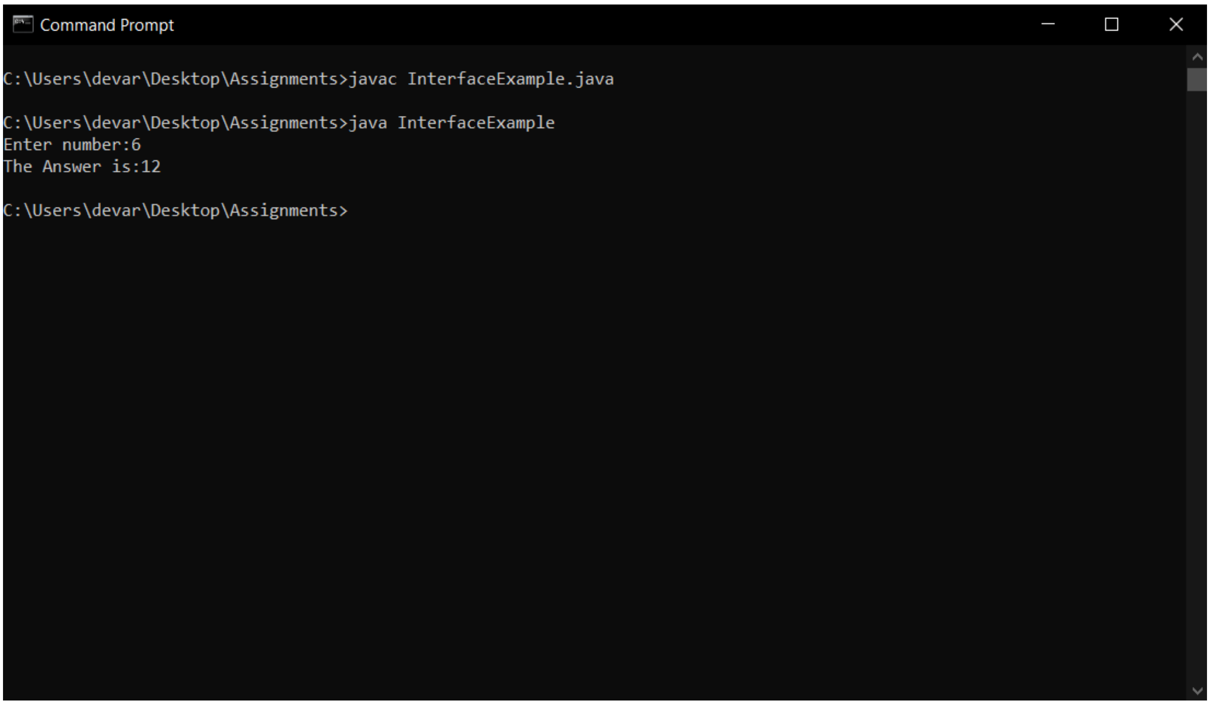
divisorSum function just takes an integer as input and return the sum of all its

divisors. For example divisors of 6 are 1, 2, 3 and 6, so divisor_sum should return 12

```
import java.util.Scanner;


interface  AdvancedArithmetic{
   int divisor_sum(int n);


}
class MyCalculator implements AdvancedArithmetic {


   public int divisor_sum(int n) {
      int sum=0;
```

```
        for(int i=1;i<=n;i++) {
            if(n%i==0)
                sum+=i;
        }
        return sum;
    }
}

public class InterfaceExample {
    public static void main(String[] args) {
        Scanner sc =new Scanner((System.in));
        System.out.print("Enter number:");
        int number=sc.nextInt();
        MyCalculator myCalculator=new MyCalculator();
        int ans=myCalculator.divisor_sum(number);
        System.out.println("The Answer is:"+ans);



    }
}
```

# OUTPUT:



```
C:\Users\devar\Desktop\Assignments>javac InterfaceExample.java

C:\Users\devar\Desktop\Assignments>java InterfaceExample
Enter number:6
The Answer is:12

C:\Users\devar\Desktop\Assignments>
```