Practical 6

Program 1:--

Write a program to create thread which display “Hello World” message.
A. by extending Thread class
B. by using Runnable interface.

➔ By Extednding Thread Class

CODE :--

```
//Created By 21CE105 VRAJ PATEL

// Write a program to create thread which display “Hello World” message.
// A. by extending Thread class
// B. by using Runnable interface.

//Github Link = https://github.com/PatelVraj10/java-practical-file-6

class MyThread extends Thread{

    public void run()
    {
        try {
                System.out.println("Hello World");
        } catch (Exception e) {
            System.out.println("Exception Occured");
        }
    }
}

public class PR_6_1_1 {
    public static void main(String[] args) {
        MyThread t1= new MyThread();
        t1.start();
    }
}
```

➔ By Implementing Runnable Interface

CODE :--

```
//Created By 21CE105 VRAJ PATEL

// Write a program to create thread which display “Hello World” message.
// A. by extending Thread class
// B. by using Runnable interface.

//Github Link = https://github.com/PatelVraj10/java-practical-file-6


class MyThread1 implements Runnable{

    public void run()
    {
        try {
                System.out.println("Hello World 1
"+Thread.currentThread().getName());
        } catch (Exception e) {
            System.out.println("Exception Occured");
        }
    }
}
public class PR_6_1_2 {
    public static void main(String[] args) {
        MyThread1 t1 = new MyThread1();
        Thread t = new Thread(t1);
        t.run();
    }
}
```

Program 2:--

Generate 15 random numbers from 1 to 100 and store it in an int array. Write a program to display the numbers stored at odd indexes by thread1 and display numbers stored at even indexes by thread2.

CODE :--

```
// Created By 21CE105 VRAJ PATEL

// Generate 15 random numbers from 1 to 100 and store it in an int array.
Write a program to display the numbers
// stored at odd indexes by thread1 and display numbers stored at even
indexes by thread2.

//Github Link = https://github.com/PatelVraj10/java-practical-file-6


import java.util.*;
import java.util.stream.IntStream;

        class TestEven extends Thread//TestEven class extends Thread class
        {
            int arr[]=new int[15];//Intializing an array arr
            TestEven(int arr1[])
            {
                this.arr=arr1;
            }
            public void run()//Creating a Thread t1
            {
                for(int i=0;i<15;i=i+1)//Determining if the element is even
                {
                    if(arr[i]%2==0)
                    {
                        System.out.println("This is an Even Element:
"+arr[i]);
                    }
                }
            }
        }
        class TestOdd extends Thread//TestOdd class extends Thread class
```

```
    {
      int arr[]=new int[15];//Intializing an array arr
      TestOdd(int arr1[])
      {
        this.arr=arr1;
      }
      public void run()//Creating a Thread t1
      {
        for (int i = 0; i < 15; i = i + 1)//Determining if the
element is odd
        {
          if(arr[i]%2!=0)
          {
            System.out.println("This is an Odd Element:
"+arr[i]);
          }
        }


      }
    }
    public class PR_6_2
    {
      public static void main(String[] args)
      {
        int[]  randomIntsArray = IntStream.generate(() -> new
Random().nextInt(15)).limit(100).toArray();//Generating a random Integer
Array of size 15 and limiit 100
        TestEven t1=new TestEven(randomIntsArray);//Creating a Thread
t1
        TestOdd t2=new TestOdd(randomIntsArray);//Creating a Thread
t2
        t1.start();//Executing Thread t1
        t2.start();//Executing Thread t2



      }
    }
```

Program 3:--

Write a program to increment the value of one variable by one and display it after one second using thread using sleep() method.

CODE :--

```
//Creatd By 21CE105 VRAJ PATEL

// Write a program to increment the value of one variable by one and display
it after one second using thread
// using sleep() method.

//Github Link = https://github.com/PatelVraj10/java-practical-file-6


import javax.swing.plaf.synth.SynthOptionPaneUI;

class Thread1 implements Runnable{

    public int a = 10;
    @Override
    public void run() {
        // TODO Auto-generated method stub

        // public int a = 10;
        try{
                a++;
                Thread.sleep(1000);
        }
        catch(Exception e){
                System.out.println("Exception Occured " + e);
        }
    }

}

class Thread2 extends Thread1{

   @Override
   public void run() {
       // TODO Auto-generated method stub
       super.run();
       System.out.println("Value After Incrementing a = " + a);
    }
}

public class PR_6_3 {
    public static void main(String[] args) {
        Thread1 t1 = new Thread1();
        Thread2 t2 = new Thread2();
        t1.run();
        t2.run();
    }
}
```

Program 4:--

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.

CODE :--

```
//Created By 21CE105 VRAJ PATEL

// 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.

//Github Link = https://github.com/PatelVraj10/java-practical-file-6

class TestPriority extends Thread//TestPriority class extends Thread class
{
    public void run()
    {
        System.out.println("Running Thread is:
"+Thread.currentThread().getName());//Printing Current Thread Name
        System.out.println("Priority of  "+Thread.currentThread().getName()+"
is: "+Thread.currentThread().getPriority());//Printing the Priority of the
current Thread
    }

}


public class PR_6_4
{
    public static void main(String[] args)
    {
        TestPriority t1=new TestPriority();//Creating a Thread t1
        TestPriority t2=new TestPriority();//Creating a Thread t2
        TestPriority t3=new TestPriority();//Creating a Thread t3
        t1.setName("FIRST");
        t2.setName("SECOND");
        t3.setName("THIRD");
        t1.setPriority(3);//Setting Priority for all the Threads
        t2.setPriority(Thread.NORM_PRIORITY);
        t3.setPriority(7);
        t1.start();//Executing Thread t1
        t2.start();//Executing Thread t2
        t3.start();//Executing Thread t2



    }
}
```

Program 5:--

Write a program to solve producer-consumer problem using thread Synchronization.

CODE :--

```
//Created By 21CE105 VRAJ PATEL

// Write a program to solve producer-consumer problem using thread
// Synchronization.

//Github Link = https://github.com/PatelVraj10/java-practical-file-6

class Wait extends Thread//wait class extends Thread class
{
    int sal=0;//Initializing the salary to 0
    public void run()
    {

        synchronized (this)//synchronized block
        {
            for(int i=0;i<12;i=i+1)//Incrementing salary for each month
            {
                sal=sal+10000;
            }
            this.notify();   //Waking up the user Thread or main Thread
        }

    }
}
public class PR_6_5
{
    public static void main(String[] args) {
        Wait w=new Wait();//Creating a Thread w
        Wait w1=new Wait();//Creating a Thread w1
        Wait w2=new Wait();//Creating a Thread w2
        w.setPriority(3);//Setting Priority for all the Threads
        w1.setPriority(Thread.NORM_PRIORITY);
        w2.setPriority(7);
        w.start();//Executing Thread w
        w1.start();//Executing Thread w1
        w2.start();//Executing Thread w2
        try//try block
        {
            synchronized (w) {
                w.wait(1000);//putting the main thread in waiting state for
1s
            }
            synchronized (w1)
            {
                w1.wait(2000);//putting the main thread in waiting state for
2s
            }
            System.out.println("Total Salary of first employee = "+
w.sal);//Printing salaries of all employees
            System.out.println("Total Salary of second employee = "+ w1.sal);
            System.out.println("Total Salary of third employee = "+ w2.sal);

        }
        catch(InterruptedException e)//catch block
        {
            System.out.println(e);
        }
    }
}
```