Sample Programs

Qu 1. Write a java program to demonstrate Simple Thread.

```
import java.io.*;
class tdemo extends Thread
{
        public static void main(String args[])
        {
        //used to create a new thread
                Thread t=new Thread(new tdemo());
                //starts a new thread
                t.start();
        }
        /*Never call run() directly — always use start() ,and  Executes run() in a
        separate thread,JVM allocates memory and resources*/

        public void run()
        {
                System.out.println("My first Thread prog");
        }
}
```

Qu 2. Write a java program to print 1 to 10 numbers after every 5 seconds.

```
import java.io.*;
class tdemo1 extends Thread
{
        public static void main(String args[])
        {     tdemo1 t1= new tdemo1();

                Thread t1=new Thread(new tdemo1());
                t1.start();
        }
        public void run()
        {
        try
        {
                for(int i=1;i<=10;i++)
                {
                        System.out.println(i);
//   sleep() is a static method Of the Thread classs
Pauses the current thread for a specified time

                        sleep(5000);
                }
        }
```

```
        catch(Exception e)
        {
                System.out.println(e);
        }
        }
}
```

Qu 3. Write a java program to print character A to Z numbers after every 2 seconds.

```
import java.io.*;
class tdemo2 extends Thread
{
        public static void main(String args[])
        {
                Thread t1=new Thread(new tdemo2());
                t1.start();
        }
        public void run(){
                try
                {
    for(char ch='A';ch<='Z';ch++)
    {
                                System.out.println(ch);
                                sleep(2000);

                        }
                }
                        catch(Exception e)
                        {
                                System.out.println(e);
                        }
        }
}
```

Qu 4. Write a java program to print given message n times using thread.

```
import java.io.*;
import java.util.*;
public class tdemo3 extends Thread
{
        int n;
        String msg;
                tdemo3(int n,String msg)
        {
                this.n=n;
```

```
		this.msg=msg;
	}
	public void run()
	{
		for(int i=1;i<=n;i++)
		{
			System.out.println(msg);
		}
	}
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		System.out.println("Enter the value of n");
		int n=sc.nextInt();
		System.out.println("Enter the message");
		String msg=sc.next();
		Thread t1=new Thread(new tdemo3(n,msg));
		t1.start();
	}
}
```

Qu 5. Write a java program to display following details

Print Message “COVID19” 10 times
Print Message “LOCKDOWN2020” 20 times
Print Message “VACCINATED2021” 30 times

```
import java.io.*;
import java.util.*;
public class tdemo4 extends Thread
{
	int n;
	String msg;
	tdemo4(int n,String msg)
	{
		this.n=n;
		this.msg=msg;
	}
	public void run()
	{
		try
		{
			for(int i=1;i<=n;i++)
			{
				System.out.println(msg);
```

```
                }
            }
            catch(Exception e)
            {
                System.out.println(e);
            }
        }
        public static void main(String args[])
        {
            Thread t1=new Thread(new tdemo4(10,"COVID19"));
            t1.start();
            Thread t2=new Thread(new tdemo4(20,"LOCKDOWN2020"));
            t2.start();
            Thread t3=new Thread(new tdemo4(30,"VACCINATED2021"));
            t3.start();
        }
}
```

Qu 6. Write a java program to generate 10 random numbers using thread.

```
import java.io.*;
import java.util.*;
class tdemo5 extends Thread
{
        public static void main(String args[])
        {
            Thread t=new Thread(new tdemo5());
            t.start();
        }
        public void run()
        {
            for(int i=1;i<10;i++)
            {
            //  Random()  which can generate random numbers using various
            methods. Random is a class in the java.util package.

            Random r=new Random();
            int n=r.nextInt(50);
            System.out.println(i);
            }
        }
}
```

Qu 7. Write a java program to demonstrate thread synchronisation.

```
import java.io.*;
class mythread extends Thread
{
	String msg[]={"Java","Supports","Multithreading","concept"};
	mythread(String name)
	{
		super(name);
	}
	public void run()
	{
		display(getName());
		System.out.println("Exit from"+grtName());
	}

	// Synchronization: synchronized ensures that display() is accessed by only
	one thread at a time, preventing mixed outputs.

	synchronized void display(String name)
	{
		for(int i=0;i<msg;i++)
		{
			System.out.println(name+msg[i]);
		}
	}
}
class mysynchro
{
	public static void main(String args[])
	{
		mythread t1=new mythread("Thread 1:");
		mythread t2=new mythread("Thread 2:");
		t1.start();
		t2.start();
		System.out.println("Main thread exited");
	}
}
```

Qu 8. Write a java program to demonstrate thread priority.

```
import java.io.*;
class prioritydemo extends Thread
{
	public void run()
	{
```

```
		System.out.println("running thread name is:"+Thread.currentThread().getName());
		System.out.println("running thread priority is:"+Thread.currentThread().getPriority());
	}
	public static void main(String args[])
	{
		prioritydemo m1=new prioritydemo();
		prioritydemo m2=new prioritydemo();
// Set priorities Minimum priority = 1
		m1.setPriority(Thread.MIN_PRIORITY);
// Maximum  priority = 10
		m2.setPriority(Thread.MAX_PRIORITY);
		m1.start();
		m2.start();
	}
}
```

Qu 9. Write a java program which generate any random number using first thread , if number is even, second thread will calculate its square and if number is odd third thread will calculate cube of number.

```
import java.io.*;

import java.util.*;
class square extends Thread
{
	int n;
	square(int n)
	{
		this.n=n;
	}
	public void run()
	{
		System.out.println("Square:"+n*n);
	}
}
class cube extends Thread
{
	int n;
	cube(int n)
	{
		this.n=n;
	}
	public void run()
	{
```

```
                System.out.println("Cube:"+n*n*n);
        }
}
class number extends Thread
{
        int n;
        public void run()
        {
                try
                {
                        Random r=new Random();
                        for(int i=1;i<=10;i++)
                        {
                                n=r.nextInt(20);
                                System.out.println("generated number is:"+n);
                                if(n%2==0)
                                {
                                        square s=new square(n);
                                        s.start();
                                }
                                else
                                {
                                        cube c=new cube(n);
                                        c.start();
                                }

                                sleep(2000);
                        }
                }
                catch(Exception e)
                {
                        System.out.println(e);
                }
        }
}
class tdemo7
{
        public static void main(String args[])
        {
                number n=new number();
                n.start();
        }
}
```