

CS261

ASSIGNMENT 8

NAME:

ARCHIT AGRAWAL

ROLL NO. :

202051213

SECTION:

A

1. Illustrate both way of implementing thread. (Write a program)

Approach

Create a public class 'Multithreading' which contains the 'main' method only. Create two other classes,

- class Thread1: this extends the java.lang.Thread class to implement threads
- class Thread2: this class implements thread by implementing the Runnable interface.

Simply, write the run method in both the classes.

To implement thread using class Thread1, create an object of Thread1 class and call the run method.

To implement thread by implementing the Runnable interface, create an object of Thread2 class first. Then, create an object of java.lang.Thread class and pass the object of Thread2 class in the constructor of java.lang.Thread class (copy constructor).

Code

```
import java.util.*;
import java.lang.*;

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

        //creating object of Thread1 class
        //Thread1 class implements thread by extending thread class
        Thread1 obj1 = new Thread1();
        obj1.start();
        //This class implements thread by implementing Runnable
        interface
        //creating object of Thread2 class
        Thread2 obj2 = new Thread2();
        //passing object of Thread2 class to object of java.lang.Thread
        class
        Thread object = new Thread(obj2);
        object.start();

    }
```

```
}

class Thread1 extends Thread{
    public void run(){
        System.out.println("Thread is running, implemented by extending
Thread class.");
    }
}

class Thread2 implements Runnable{

    public void run(){
        System.out.println("Thread is running, implemented using
Runnable interface.");
    }
}
```

Output

```
PS C:\Users\Archit\Desktop\cprog> cd "c:\Users\Archit\Desktop\cprog\" ; if ($?) {
Thread is running, implemented by extending Thread class.
Thread is running, implemented using Runnable interface.
PS C:\Users\Archit\Desktop\cprog> }
```

2. Write a program create a class 'simpleobject'. Using constructor display the message.

Approach

Write a class 'simpleobject' and write two constructors for this class. One non-parameterised and the other parameterised. Write a message inside these constructors.

Write the main method and call both the constructors from this main method.

Code

```
import java.util.*;

public class simpleobject{

    //non-parameterised constructor
    public simpleobject(){
```

```
        System.out.println("Non-parameterised constructor called..");
    }

    //parameterised constructor
    public simpleobject(int x){
        System.out.println("Parameterised constructor called...");
    }

    public static void main(String[] args) {
        //calling Non-parameterised constructor
        simpleobject obj1 = new simpleobject();

        //calling Parameterised constructor
        simpleobject obj2 = new simpleobject(5);

    }
}
```

Output

```
PS C:\Users\Archit\Desktop\cprog> cd "c:\Users\Archit\Desktop\cprog\" ; if ($?) { java
Non-parameterised constructor called..
Parameterised constructor called...
PS C:\Users\Archit\Desktop\cprog> █
```