Practical No. 13

Aim : Create, debug, and run java programs based on Threads by implementing Runnable interface.

Practical No. 13

PAGE NO.
DATE

Aim: Create, debug and run java programs based on Threads by extending implementing Runnable interface.

Theory:

What is runnable interface?

- java.lang.Runnable is an interface that is to be treated by a class whose instances are intended to be executed by a thread.
- Runnable interface has only one function it in it, which has to be overidden in the class that implements Runnable interface. This function is the run public void run() function.
- Runnable interface is used in conditions where a class has to extend another class, as java doesn't support extending multiple classes, so in such cases Runnable interface is used.

What are the steps to create a Runnable class by implementing Runnable interface?

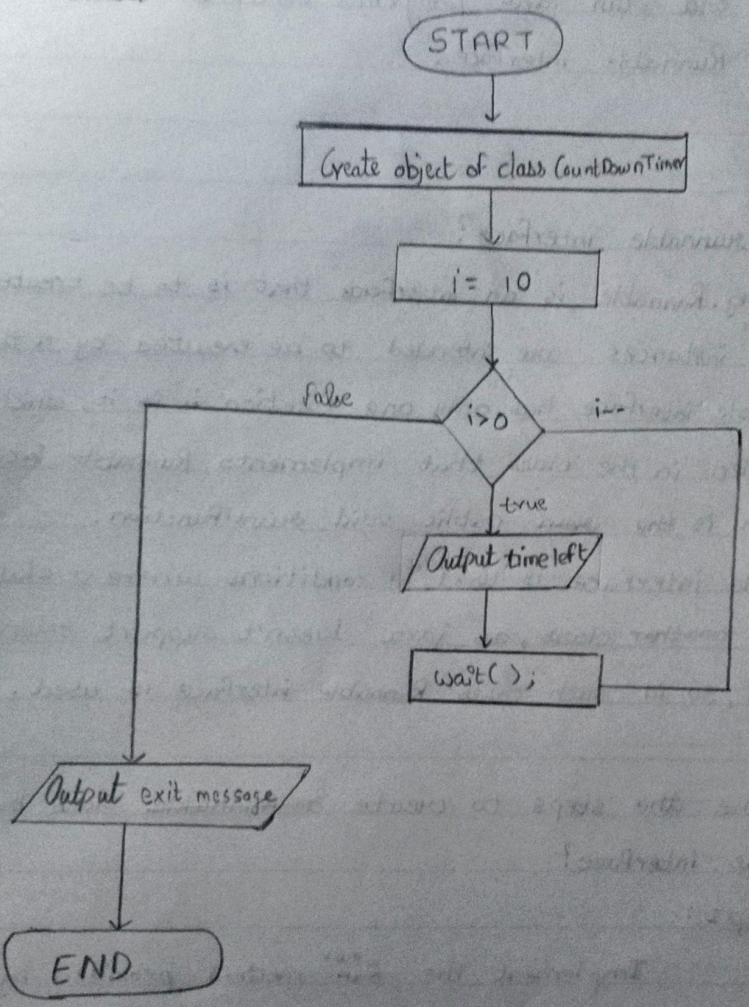
→ Step 1:

Implement the run method provided by the Runnable interface. This method provides entry point for the thread and you will the code part for the thread goes in it.

Syntax:

```
public void run(){
....
}
```

Flowchart :



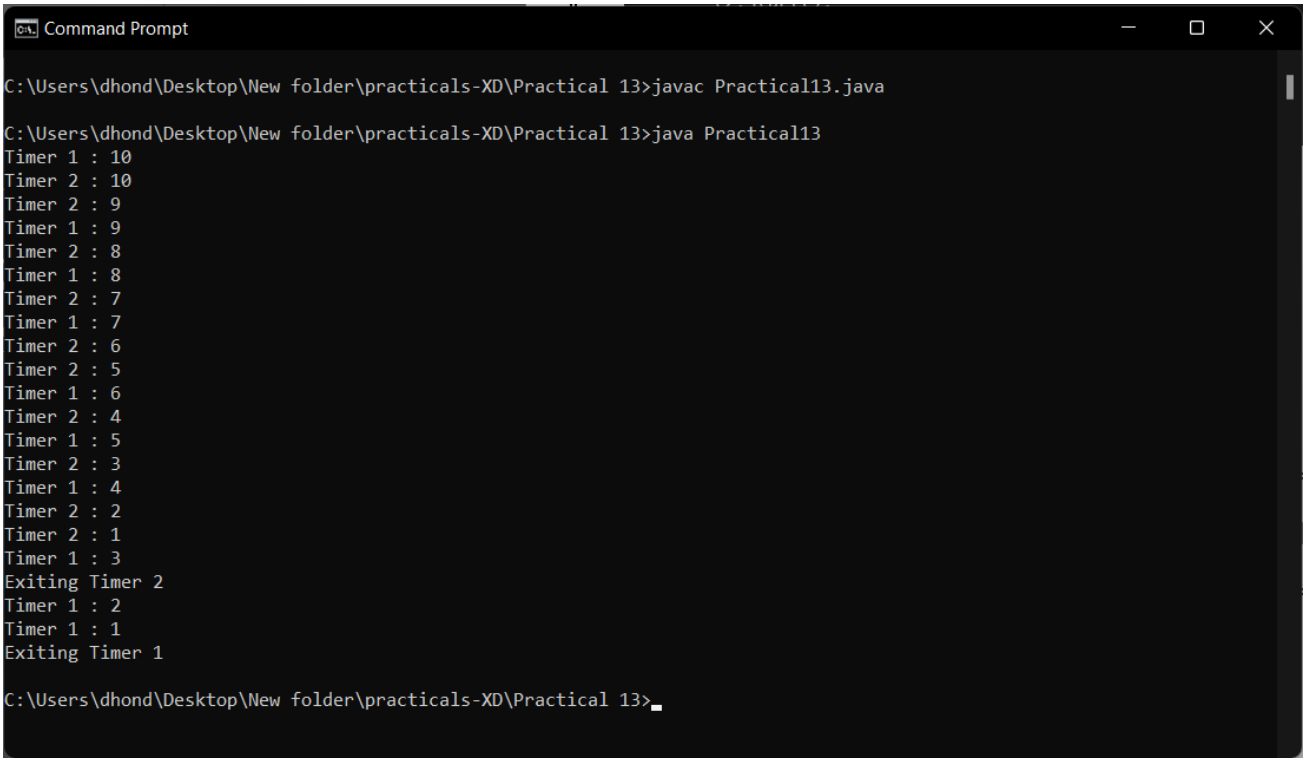
Conclusion :

Hence, by performing this practical I learnt about the concept of creating threads by using Runnable interface. I also created, developed and executed java programs based on threads by implementing Runnable interface.

Code:

```
class CountdownTimer implements Runnable{
        String name;
        int ms;

        CountdownTimer(String name, int ms){
            this.name = name;
            this.ms = ms;
        }

        synchronized public void run() {
            for(int i = 10; i > 0; i--){
                System.out.println(name + " : " + i);
                try {
                    wait(ms);
                }catch (Exception e){
                    System.out.println(e);
                }
            }
            System.out.println("Exiting " + name);
        }
    }

class Practical13{

    public static void main(String[] args) {
        // 250, 500 are milliseconds for wait
        CountdownTimer c1 = new CountdownTimer("Timer 1",1000);
        CountdownTimer c2 = new CountdownTimer("Timer 2",750);

        Thread t1 = new Thread(c1);
        Thread t2 = new Thread(c2);
        t1.start();
        t2.start();

    }

}
```

Output:


Command Prompt
C:\Users\dhond\Desktop\New folder\practicals-XD\Practical 13>javac Practical13.java
C:\Users\dhond\Desktop\New folder\practicals-XD\Practical 13>java Practical13
Timer 1 : 10
Timer 2 : 10
Timer 2 : 9
Timer 1 : 9
Timer 2 : 8
Timer 1 : 8
Timer 2 : 7
Timer 1 : 7
Timer 2 : 6
Timer 2 : 5
Timer 1 : 6
Timer 2 : 4
Timer 1 : 5
Timer 2 : 3
Timer 1 : 4
Timer 2 : 2
Timer 2 : 1
Timer 1 : 3
Exiting Timer 2
Timer 1 : 2
Timer 1 : 1
Exiting Timer 1
C:\Users\dhond\Desktop\New folder\practicals-XD\Practical 13>

PAGE NO.
DATE

Step 2:

Instantiate the object of thread using the following constructor.

Thread (Runnable threadObj);

example,

Thread t = new Thread (new ClassRunnable());

~~Class Run~~

Step 3 :

Once the thread is created, you can start it by calling the start method, which executes call to run method.

example.

t.start();

Conclusion:

Hence, by performing this practical I learnt about the concept of creating threads & by using Runnable interface. I also created, developed and executed java program based on Threads by implementing Runnable interface.