

## Object Oriented Programming Lab 9

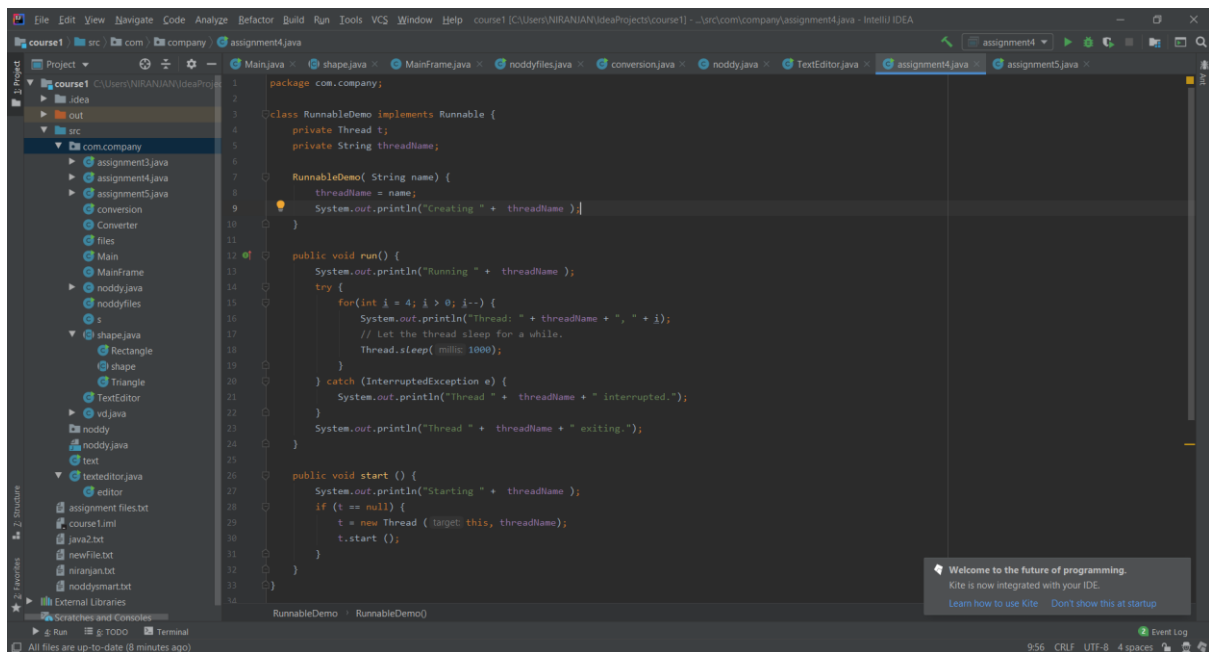
Niranjan Dhokarikar 65

ME-A Batch 3

### Problem Statement ->

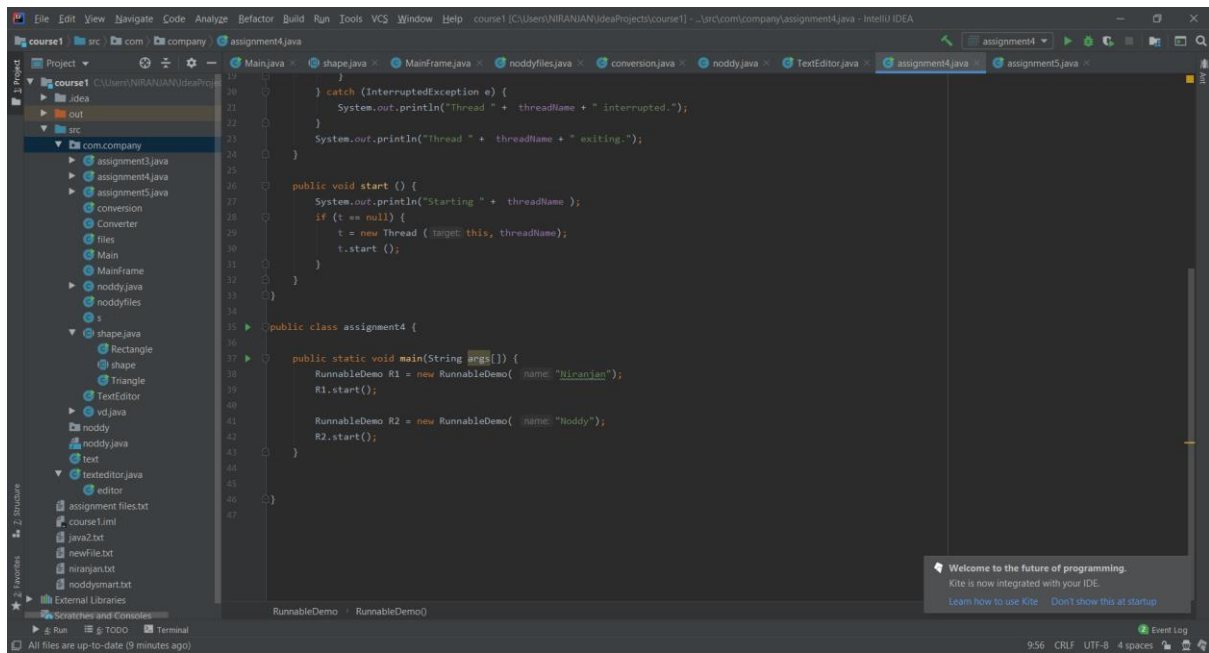
Write a java program which implements interface to create multiple threads and synchronize their working.

Code ->



The screenshot shows an IDE window with a Java project named 'course1'. The file explorer on the left shows a package 'com.company' containing several files, including 'assignment4.java'. The main editor displays the code for 'RunnableDemo' in 'assignment4.java'. The code implements the 'Runnable' interface and uses a 'Thread' object to create and manage multiple threads. The 'run()' method contains a loop that prints the thread name and a sleep duration. The 'start()' method creates a new thread and starts it.

```
1 package com.company;
2
3 class RunnableDemo implements Runnable {
4     private Thread t;
5     private String threadName;
6
7     RunnableDemo( String name) {
8         threadName = name;
9         System.out.println("Creating " + threadName );
10    }
11
12    public void run() {
13        System.out.println("Running " + threadName );
14        try {
15            for(int i = 4; i > 0; i--) {
16                System.out.println("Thread: " + threadName + ", " + i);
17                // Let the thread sleep for a while.
18                Thread.sleep( millis: 1000);
19            }
20        } catch (InterruptedException e) {
21            System.out.println("Thread " + threadName + " interrupted.");
22        }
23        System.out.println("Thread " + threadName + " exiting.");
24    }
25
26    public void start () {
27        System.out.println("Starting " + threadName );
28        if (t == null) {
29            t = new Thread ( target: this, threadName);
30            t.start ();
31        }
32    }
33 }
```



## Output of the Following Program ->

