

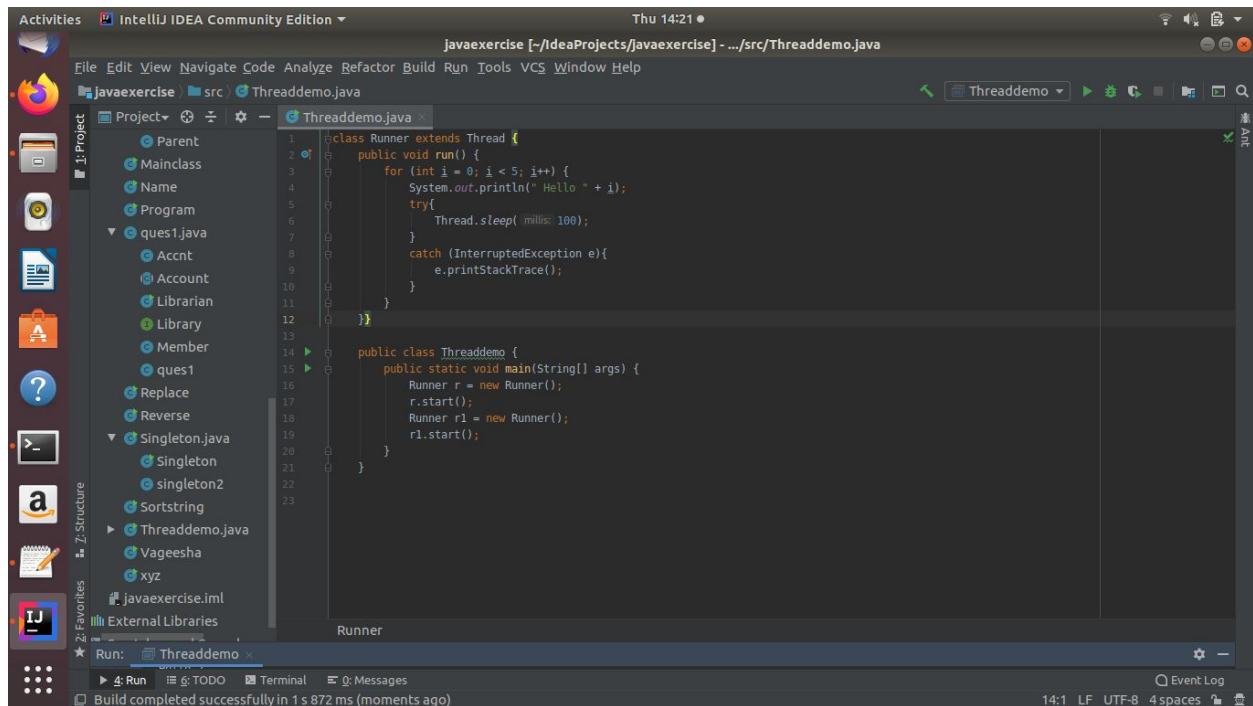
### 1. Name of method in Thread class to pause execution?

Thread.Sleep(millis) is used to pause current execution for a specific time in milliseconds.

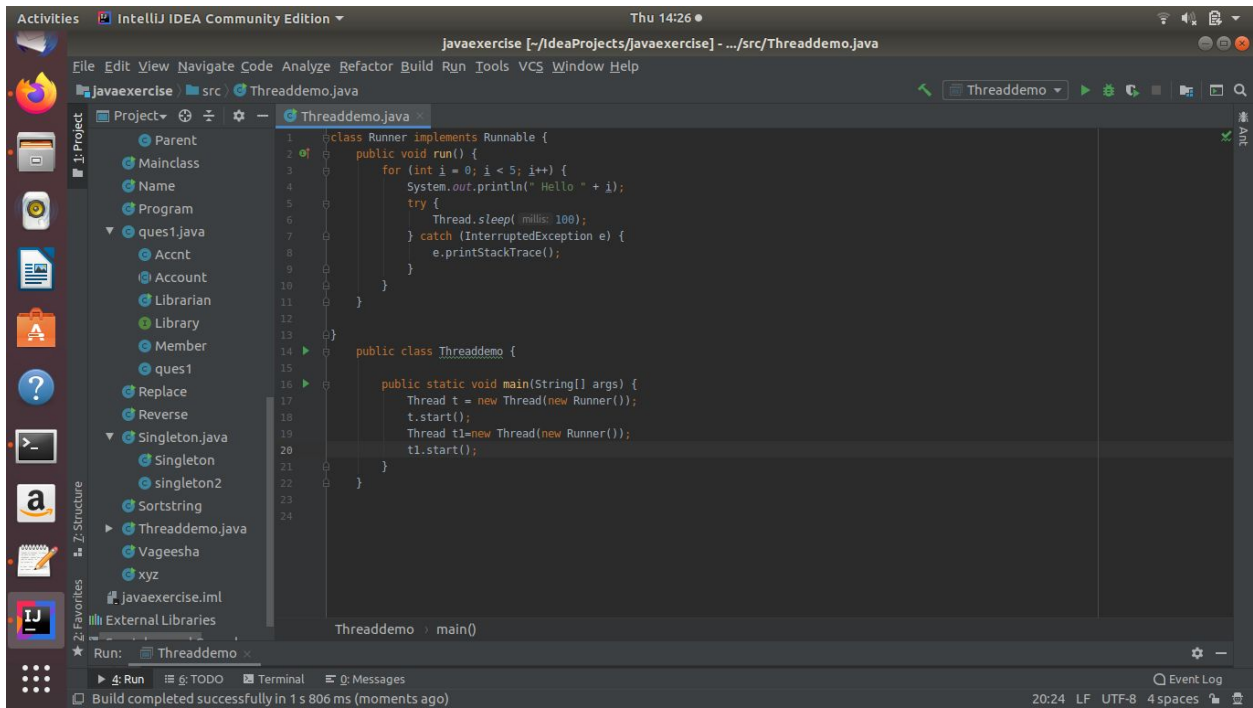
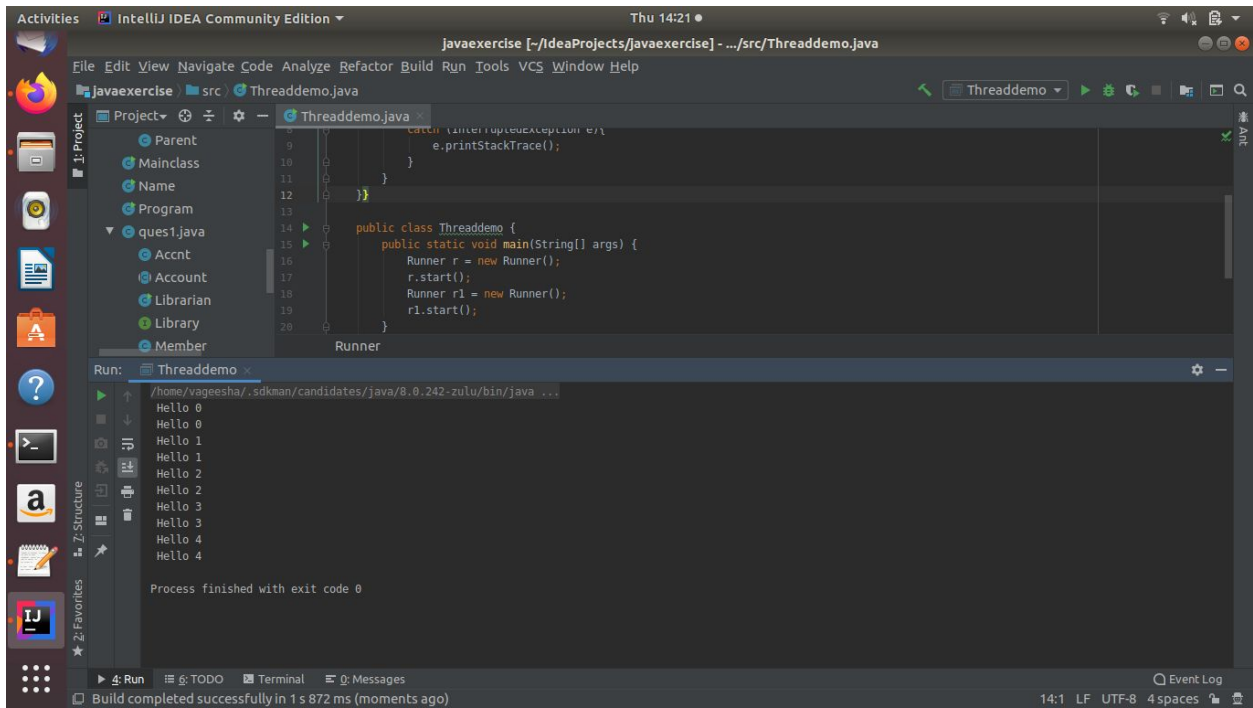
### 2. Role of "volatile" keyword.

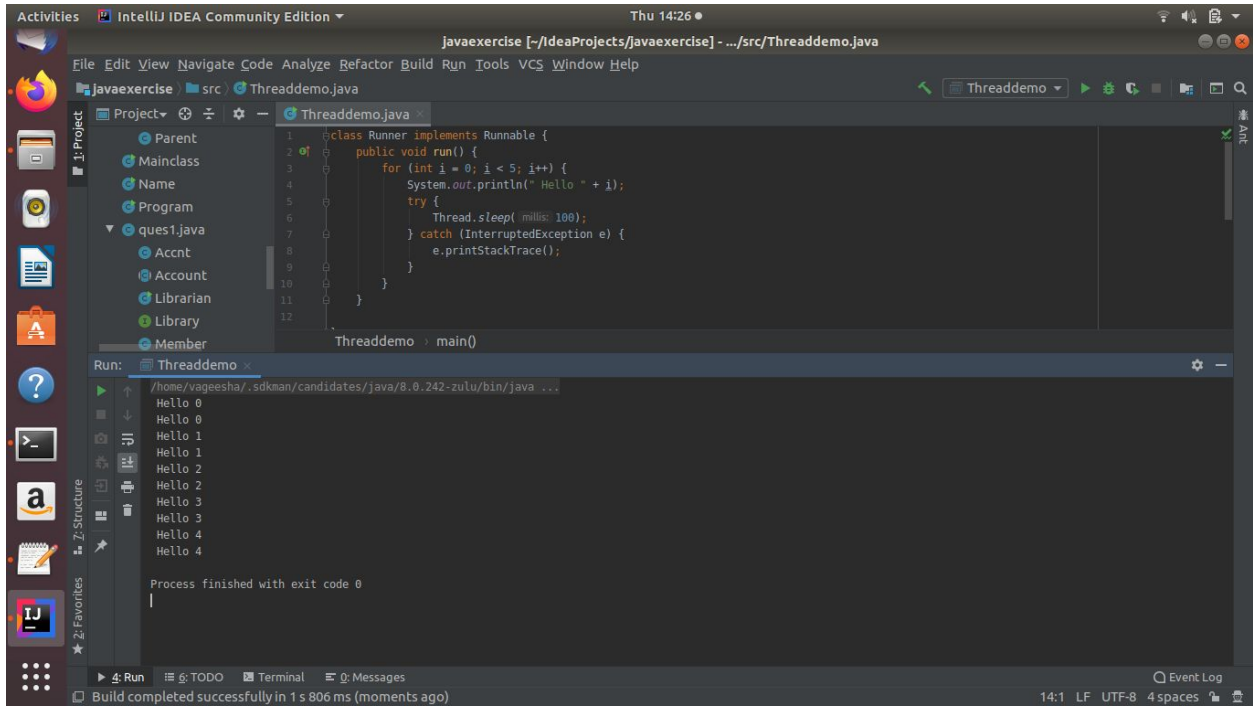
The Java volatile keyword is used to mark a Java variable as "being stored in main memory". That means, that every read of a volatile variable will be read from the computer's main memory, and not from the CPU cache, and that every write to a volatile variable will be written to main memory, and not just to the CPU cache.

### 3. Write a program to create a thread using Thread class and Runnable interface each.

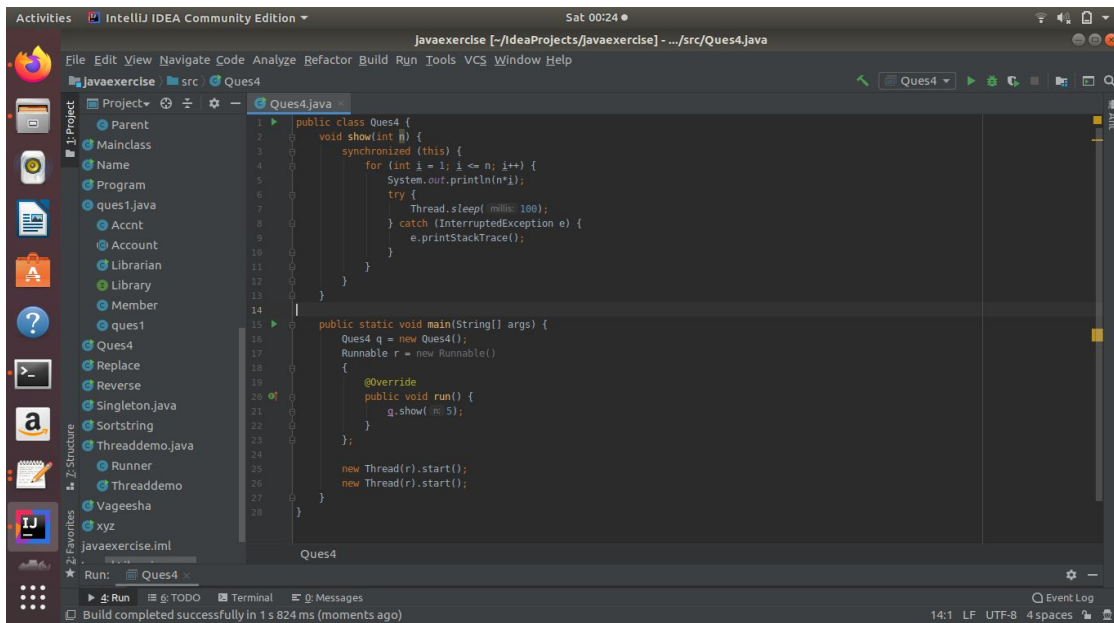


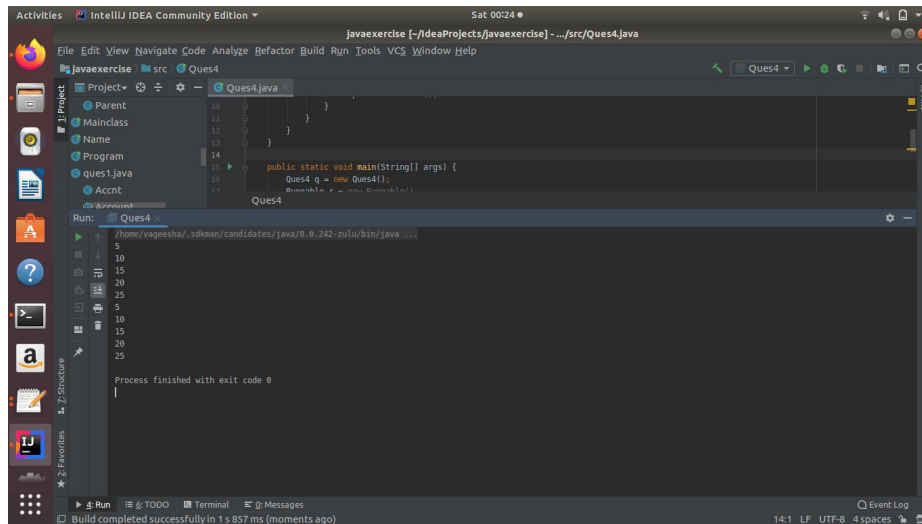
```
1 class Runner extends Thread {
2     public void run() {
3         for (int i = 0; i < 5; i++) {
4             System.out.println(" Hello " + i);
5             try {
6                 Thread.sleep(100);
7             } catch (InterruptedException e) {
8                 e.printStackTrace();
9             }
10        }
11    }
12 }
13
14 public class Threaddemo {
15     public static void main(String[] args) {
16         Runner r = new Runner();
17         r.start();
18         Runner r1 = new Runner();
19         r1.start();
20     }
21 }
22
23 }
```



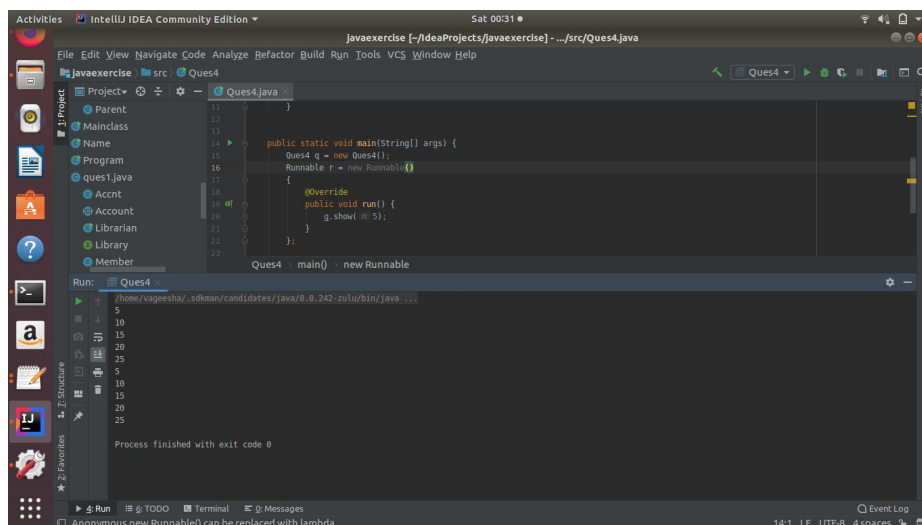
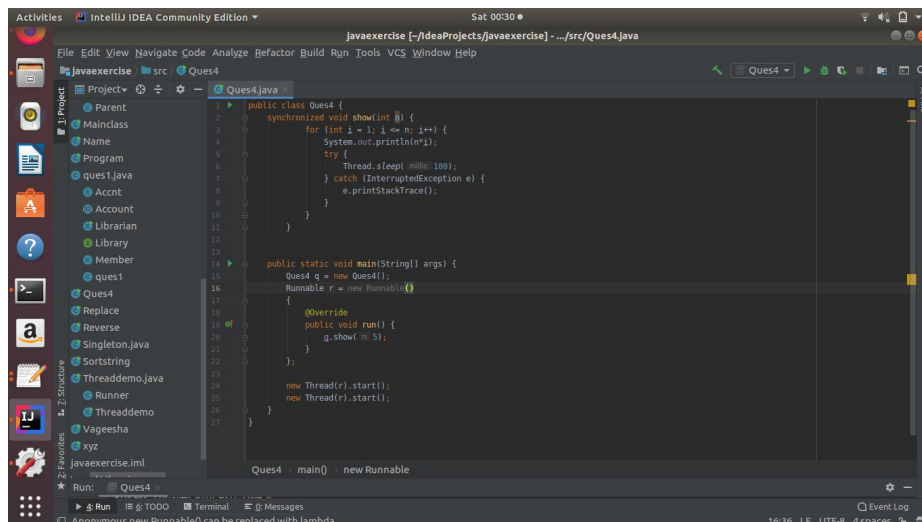


#### 4. Write a program using synchronization block

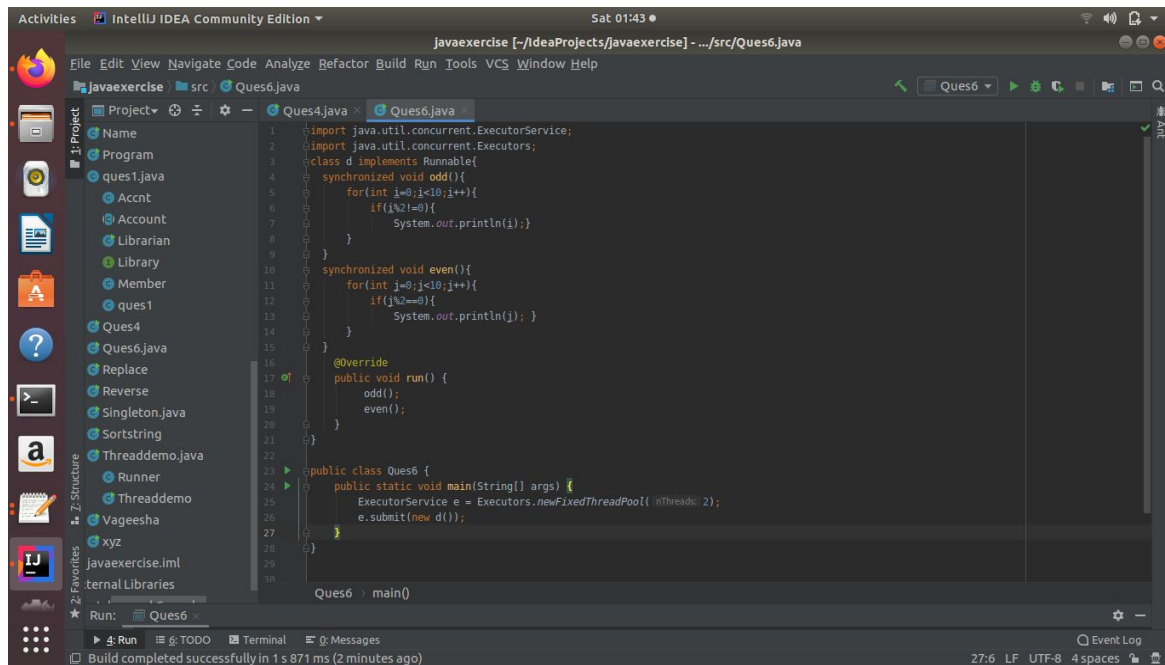




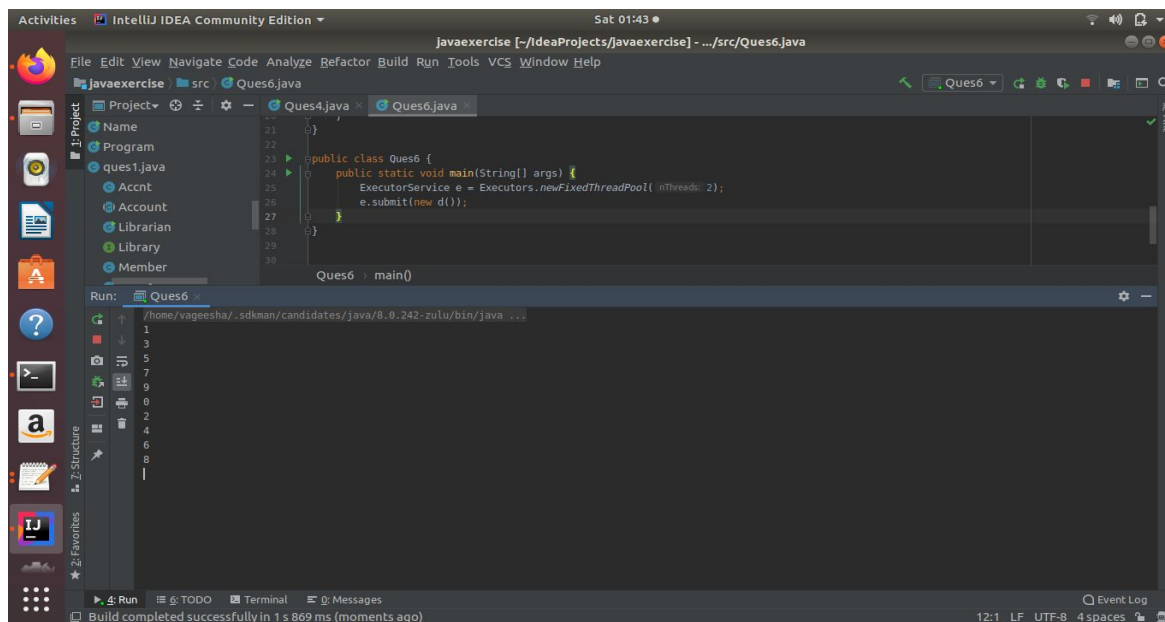
## 5. Write a program using synchronization method



6. Write a program to create a Thread pool of 2 threads where one Thread will print even numbers and other will print odd numbers.

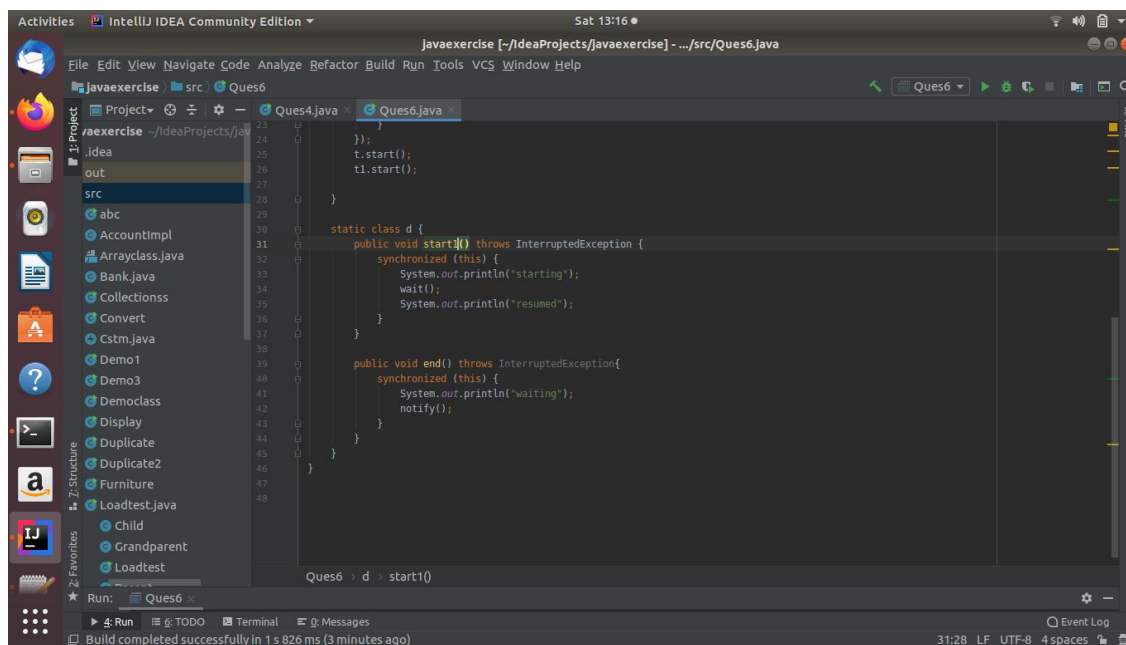
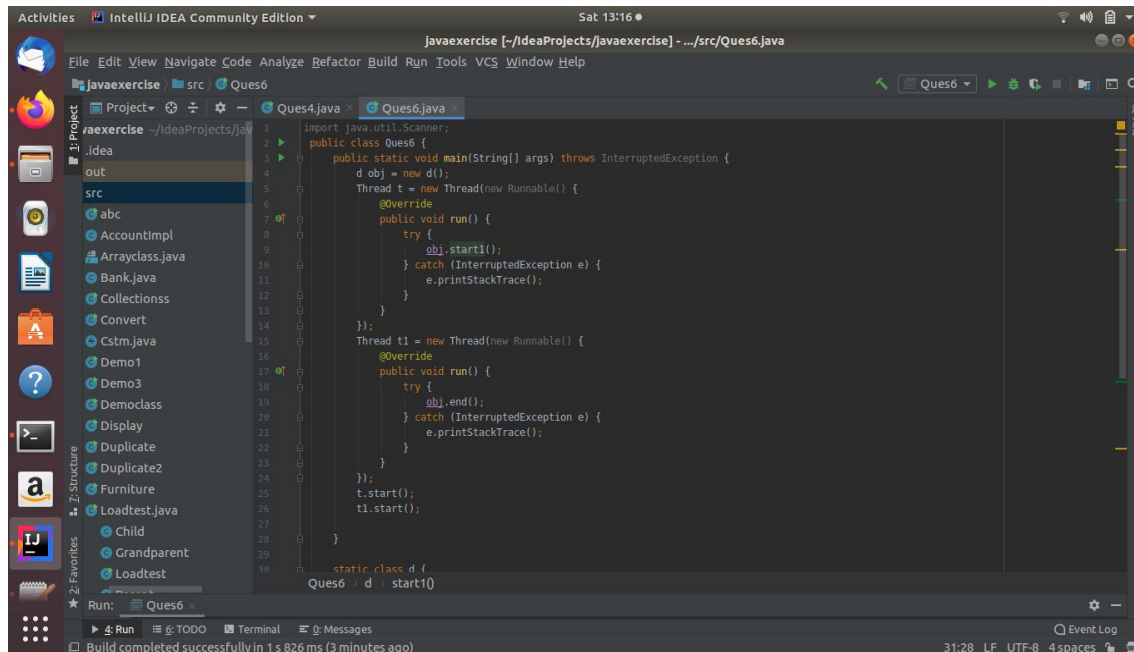


```
1 import java.util.concurrent.ExecutorService;
2 import java.util.concurrent.Executors;
3 class d implements Runnable{
4     synchronized void odd(){
5         for(int i=0;i<10;i++){
6             if(i%2!=0){
7                 System.out.println(i);
8             }
9         }
10    synchronized void even(){
11        for(int j=0;j<10;j++){
12            if(j%2==0){
13                System.out.println(j);
14            }
15        }
16    }
17    @Override
18    public void run() {
19        odd();
20        even();
21    }
22 }
23 public class Ques6 {
24     public static void main(String[] args) {
25         ExecutorService e = Executors.newFixedThreadPool( 2);
26         e.submit(new d());
27     }
28 }
29
30 Ques6 > main()
```

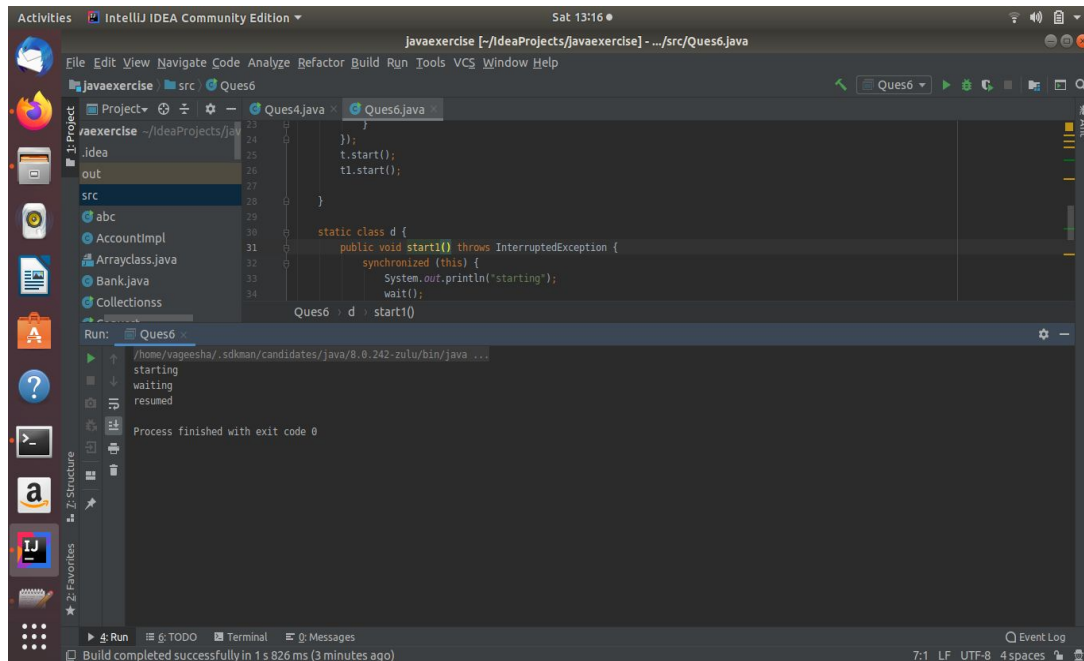


```
Run: Ques6
/home/vageesha/.sdkman/candidates/java/8.0.242-zulu/bin/java ...
1
3
5
7
9
2
4
6
8
10
Build completed successfully in 1 s 869 ms (moments ago)
```

7. Write a program to demonstrate wait and notify methods.







## 8. Write a program to demonstrate sleep and join methods.

