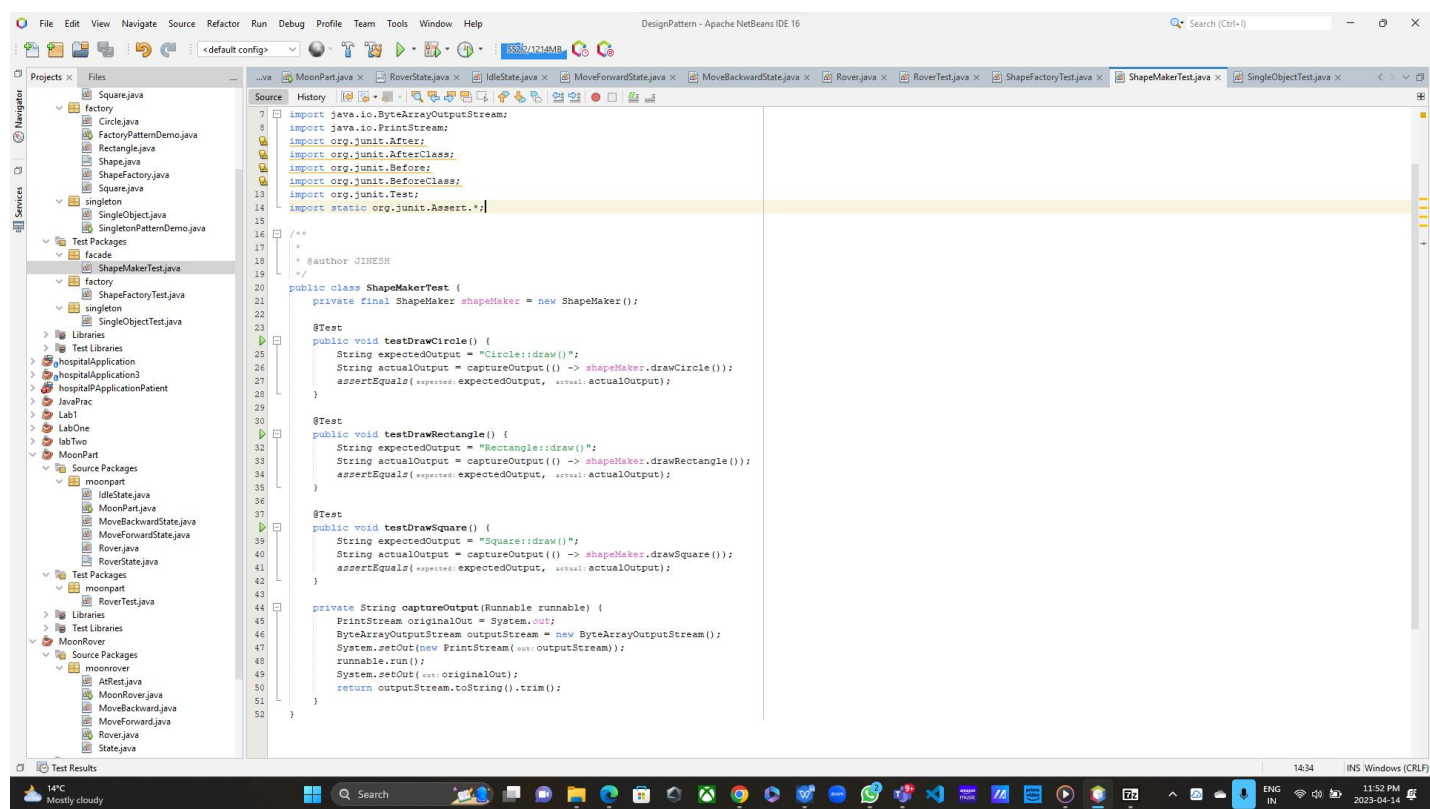
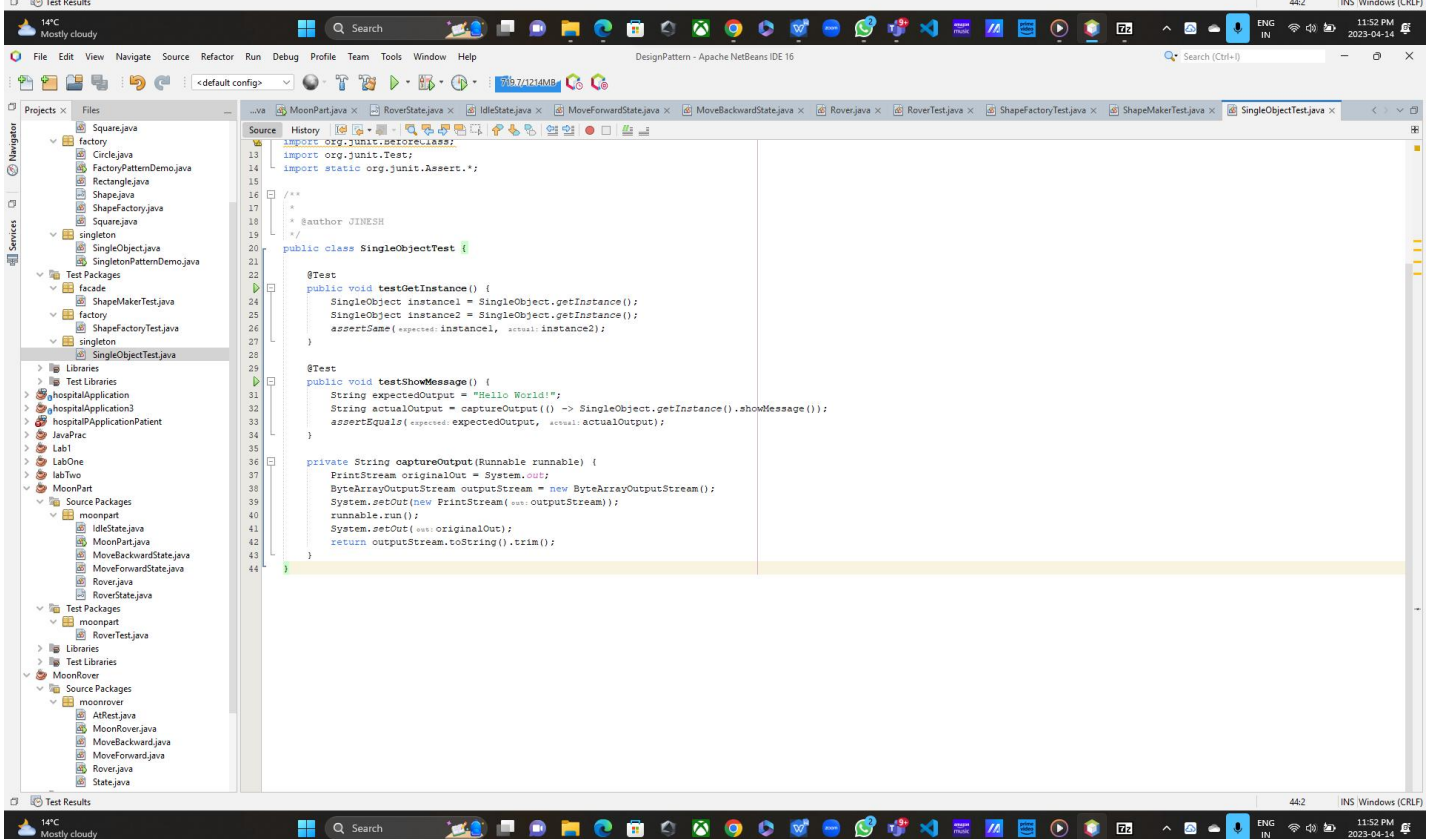
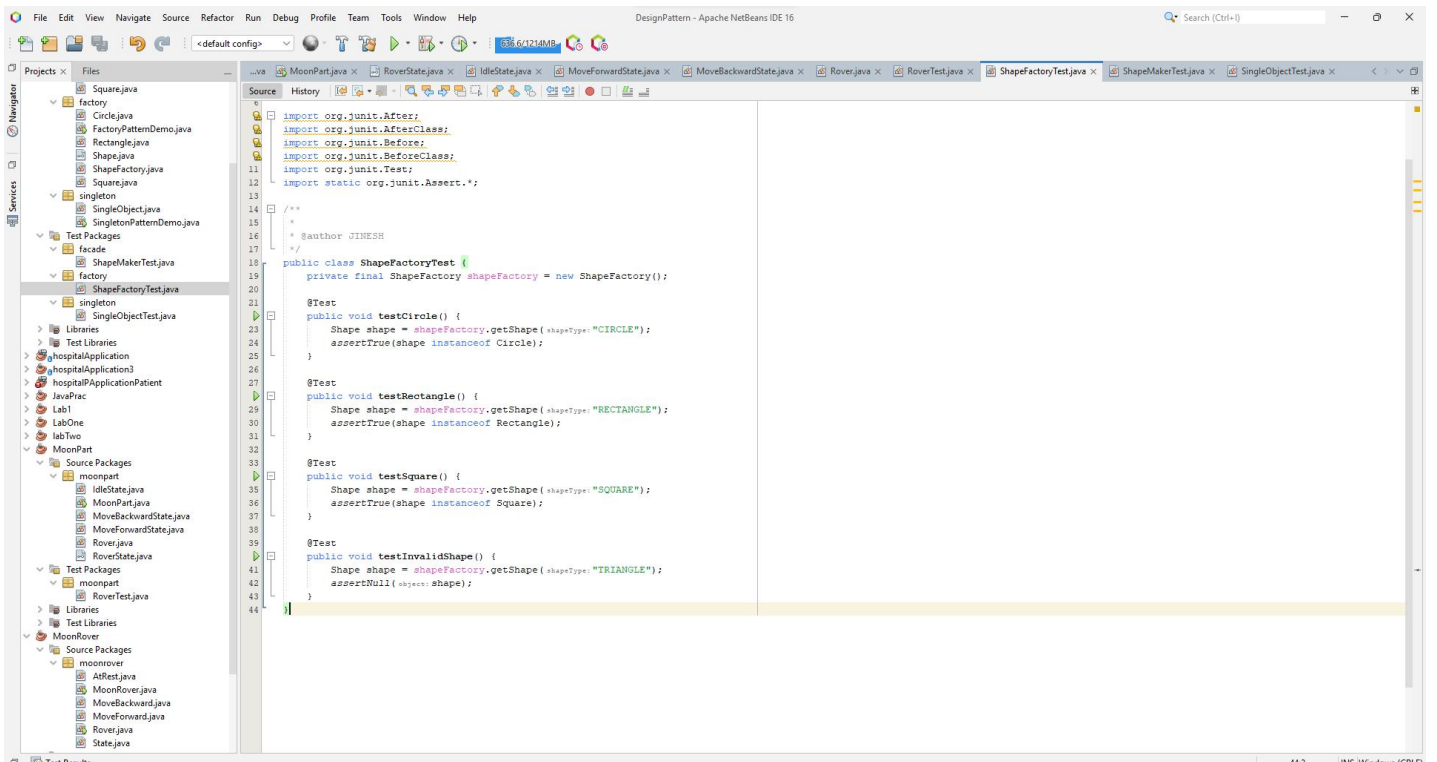
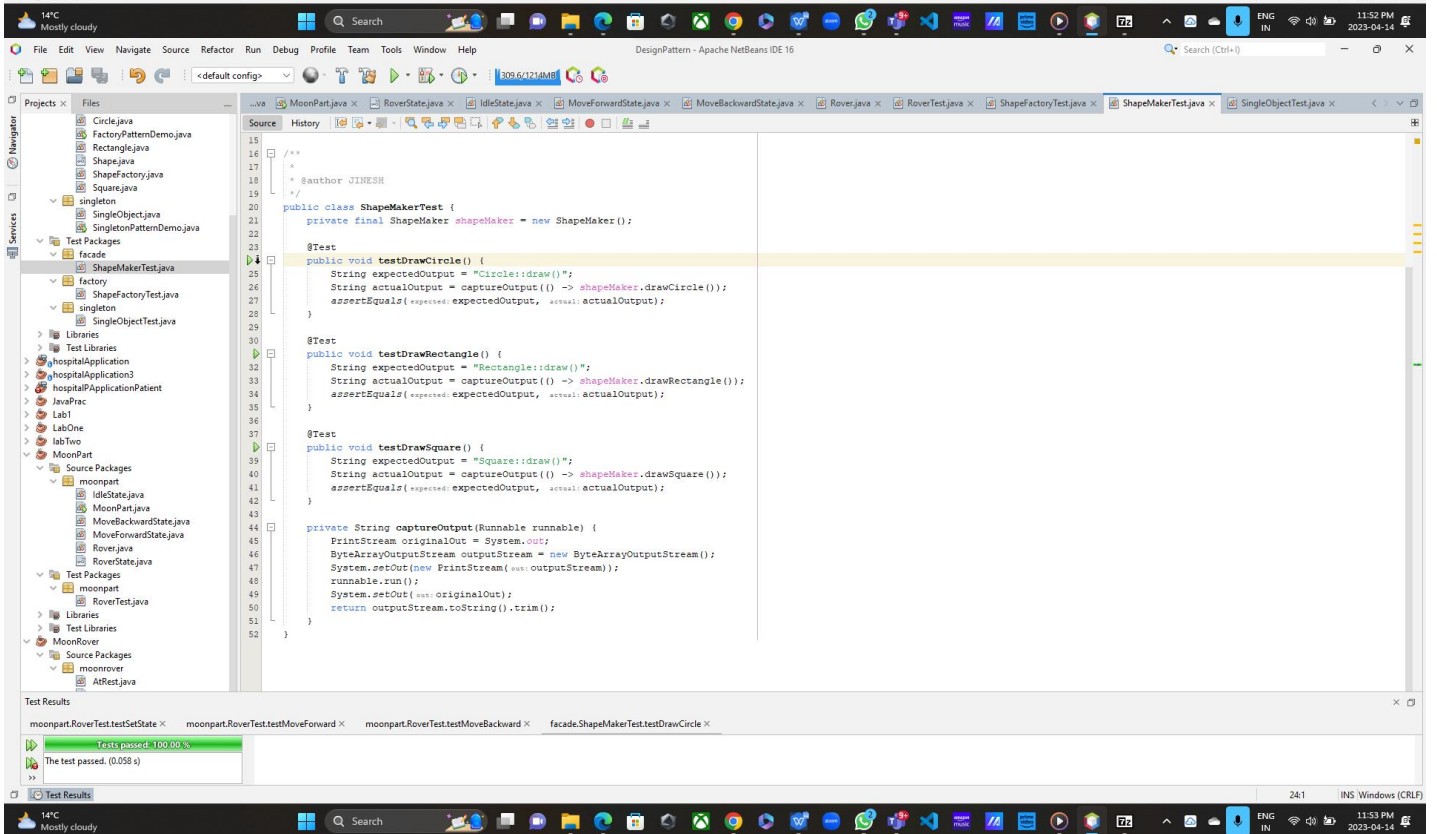
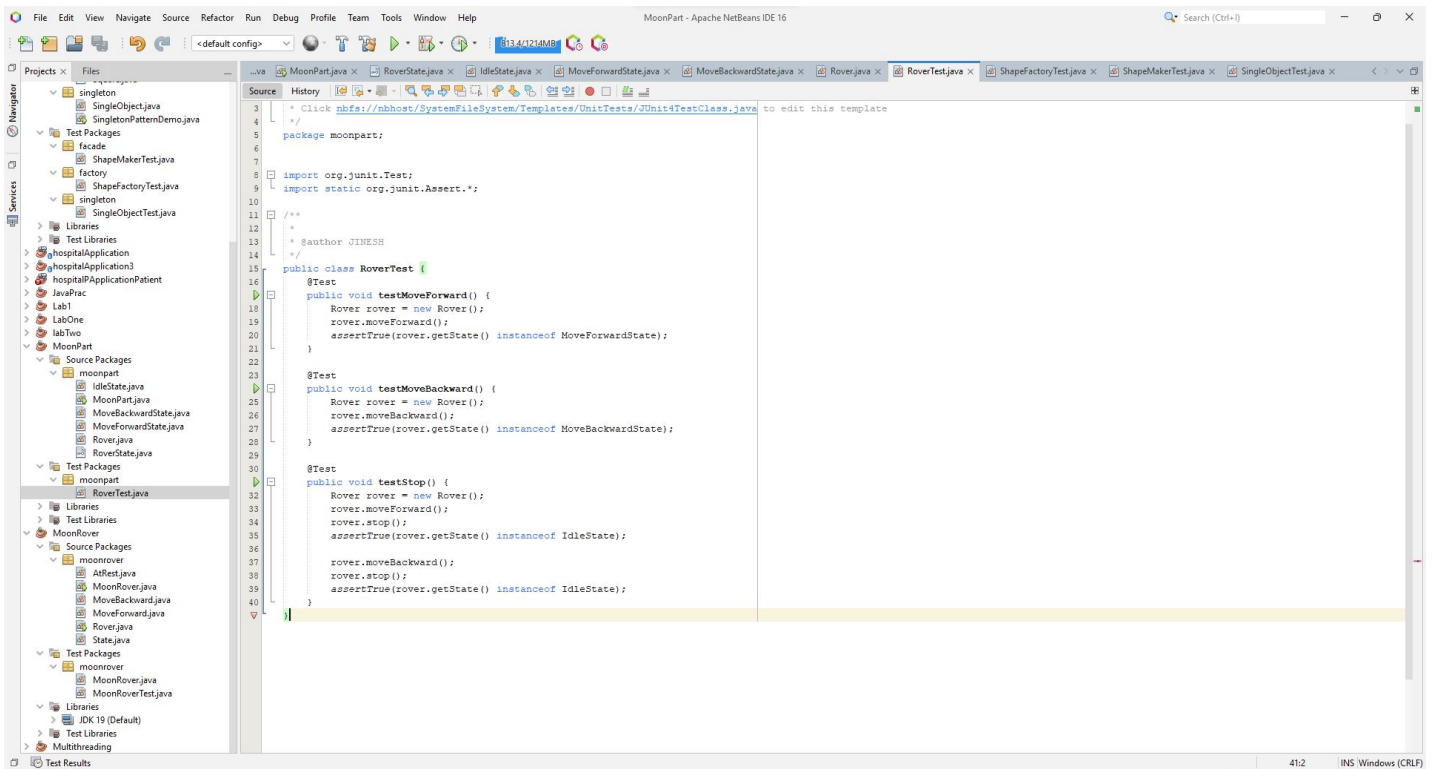


Assignment 4: Design Pattern

ScreenShots:



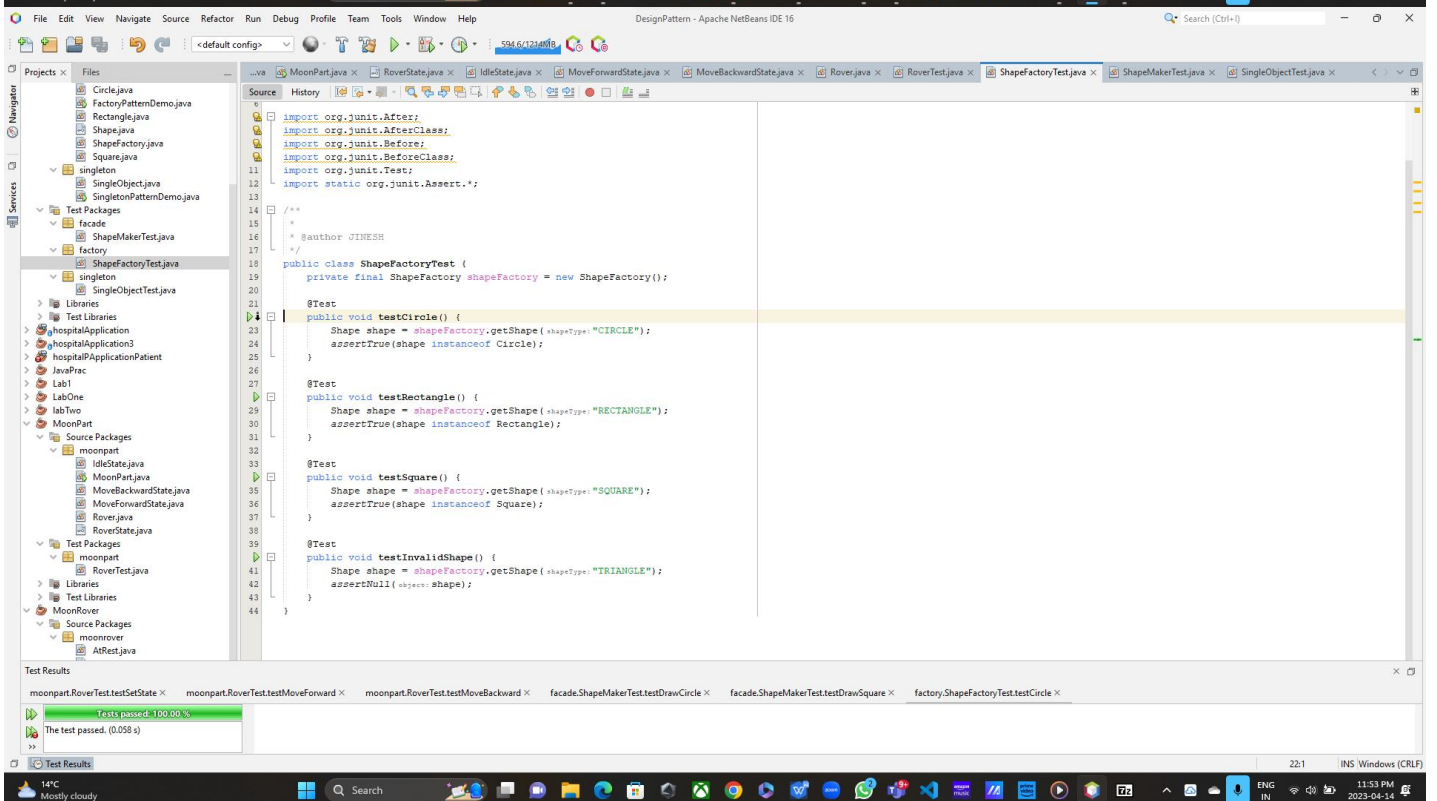
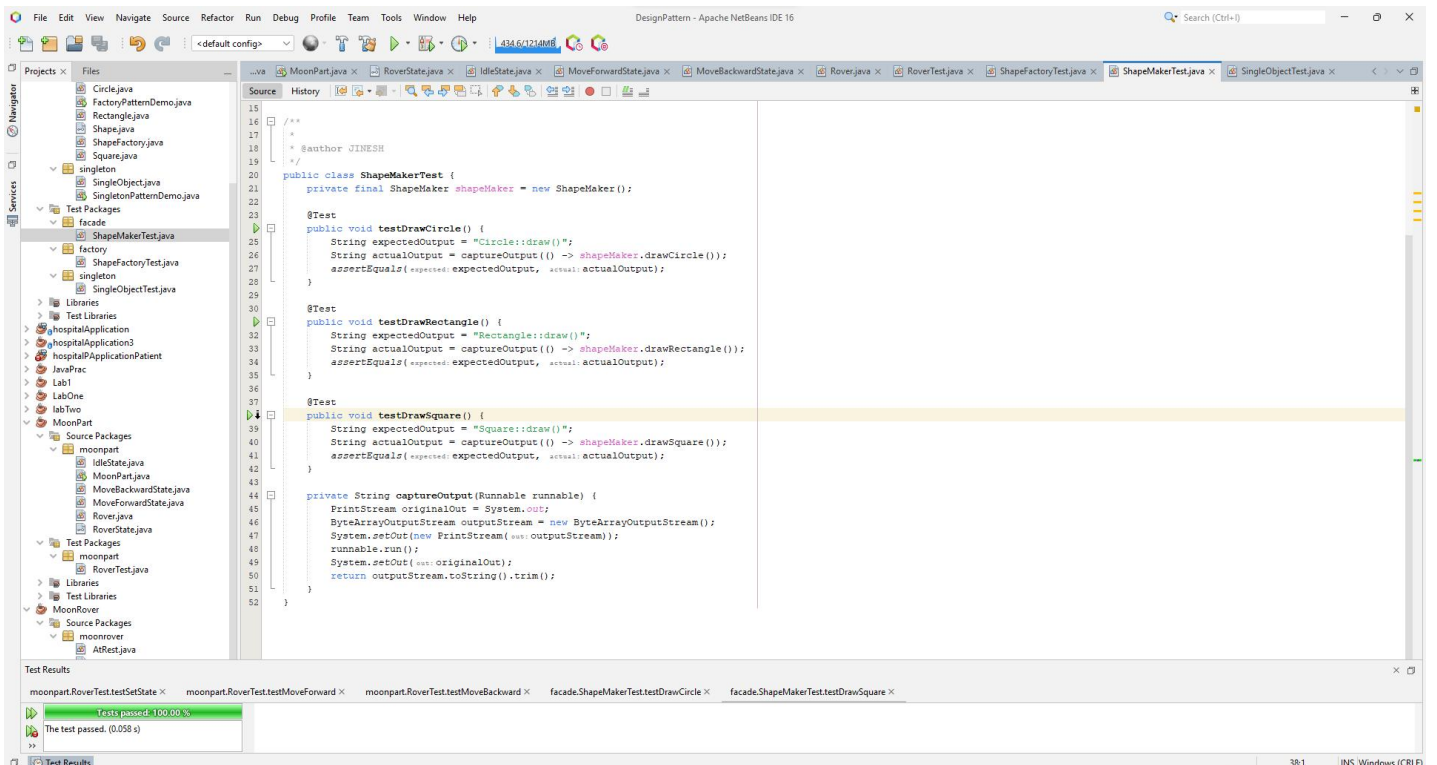


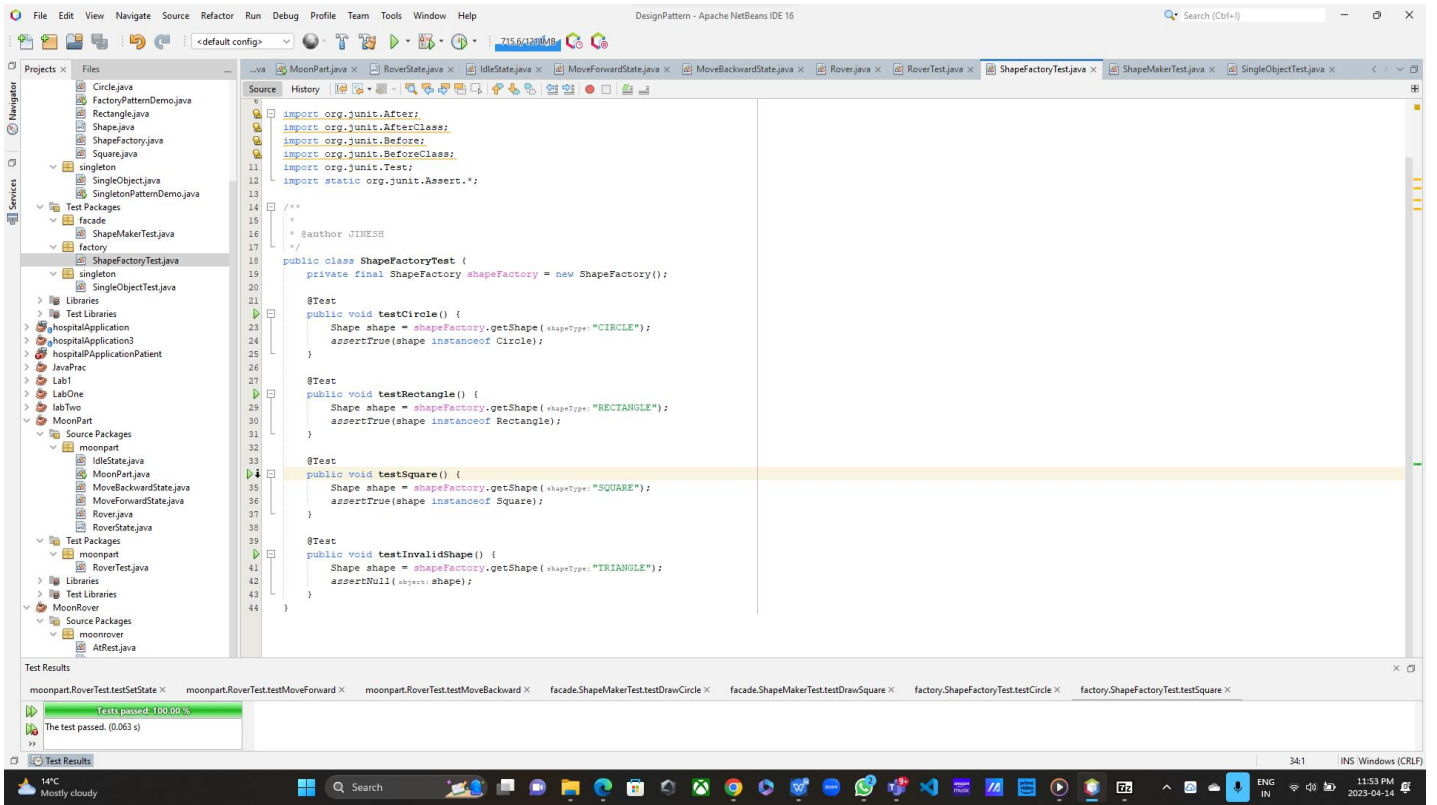


Test Results

Test Results
moonpart.RoverTest.testSetState
moonpart.RoverTest.testMoveForward
moonpart.RoverTest.testMoveBackward
facade.ShapeMakerTest.testDrawCircle

Tests passed: 100.00 %
The test passed. (0.058 s)





DesignPattern - Apache NetBeans IDE 16

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+F)

Projects Files

- Circle.java
- FactoryPatternDemo.java
- Rectangle.java
- Shape.java
- ShapeFactory.java
- ShapeFactoryTest.java
- Square.java
- singleton
- Singleton.java
- SingletonPatternDemo.java
- Test Packages
- facade
- ShapeMakerTest.java
- factory
- ShapeFactoryTest.java
- singleton
- SingletonTest.java

Test Results

moonpart.RoverTest.testSetState × moonpart.RoverTest.testMoveForward × moonpart.RoverTest.testMoveBackward × facade.ShapeMakerTest.testDrawCircle × facade.ShapeMakerTest.testDrawSquare × factory.ShapeFactoryTest.testCircle × factory.ShapeFactoryTest.testSquare × singleton.SingletonTest.testGetIn

Tests passed: 100.00 %

The test passed. (0.058 s)

```
Source History  
13 import org.junit.BeforeClass;  
14 import org.junit.Test;  
15 import static org.junit.Assert.*;  
16  
17 /**  
18  * @author JINESH  
19  */  
20 public class SingletonTest {  
21  
22     @Test  
23     public void testGetInstance() {  
24         Singleton instance1 = Singleton.getInstance();  
25         Singleton instance2 = Singleton.getInstance();  
26         assertEquals("expected: instance1, actual: instance2");  
27     }  
28  
29     @Test  
30     public void testShowMessage() {  
31         String expectedOutput = "Hello World!";  
32         String actualOutput = captureOutput(() -> Singleton.getInstance().showMessage());  
33         assertEquals("expected: expectedOutput, actual: actualOutput");  
34     }  
35  
36     private String captureOutput(Runnable runnable) {  
37         PrintStream originalOut = System.out;  
38         ByteArrayOutputStream outputStream = new ByteArrayOutputStream();  
39         System.setOut(new PrintStream(outputStream));  
40         runnable.run();  
41         System.setOut(originalOut);  
42         return outputStream.toString().trim();  
43     }  
44 }
```

DesignPattern - Apache NetBeans IDE 16

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+F)

Projects Files

- Circle.java
- FactoryPatternDemo.java
- Rectangle.java
- Shape.java
- ShapeFactory.java
- ShapeFactoryTest.java
- Square.java
- singleton
- Singleton.java
- SingletonPatternDemo.java
- Test Packages
- facade
- ShapeMakerTest.java
- factory
- ShapeFactoryTest.java
- singleton
- SingletonTest.java

Test Results

moonpart.RoverTest.testSetState × moonpart.RoverTest.testMoveForward × moonpart.RoverTest.testMoveBackward × facade.ShapeMakerTest.testDrawCircle × facade.ShapeMakerTest.testDrawSquare × factory.ShapeFactoryTest.testCircle × factory.ShapeFactoryTest.testSquare × singleton.SingletonTest.testGetIn

Tests passed: 0.00 %

No test passed, 1 test caused an error. (0.06 s)

moonpart.RoverTest Failed

```
Source History  
1 /**  
2  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license  
3  * Click nbfs://nbhost/SystemFileSystem/Templates/UnitTests/JUnit4TestClass.java to edit this template  
4  */  
5 package moonpart;  
6  
7 import org.junit.Test;  
8 import static org.junit.Assert.*;  
9  
10 /**  
11  * @author JINESH  
12  */  
13 public class RoverTest {  
14  
15     @Test  
16     public void testMoveForward() {  
17         Rover rover = new Rover();  
18         rover.moveForward();  
19         assertTrue(rover.getState() instanceof MoveForwardState);  
20     }  
21  
22     @Test  
23     public void testMoveBackward() {  
24         Rover rover = new Rover();  
25         rover.moveBackward();  
26         assertTrue(rover.getState() instanceof MoveBackwardState);  
27     }  
28  
29     @Test  
30     public void testStop() {  
31         Rover rover = new Rover();  
32         rover.moveForward();  
33         rover.stop();  
34         assertTrue(rover.getState() instanceof IdleState);  
35  
36         rover.moveBackward();  
37         rover.stop();  
38         assertTrue(rover.getState() instanceof IdleState);  
39     }  
40 }
```