

Exercise:

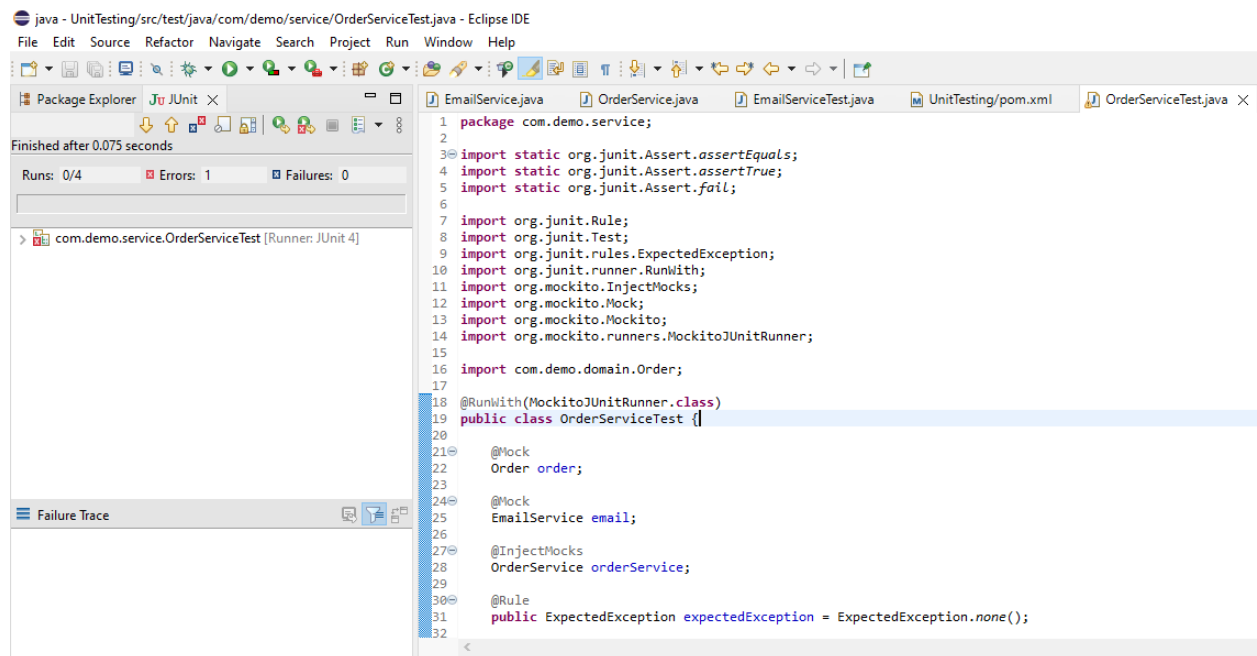
Write the unit tests for following:

- OrderService.java
- EmailService.java

Find the code base here.

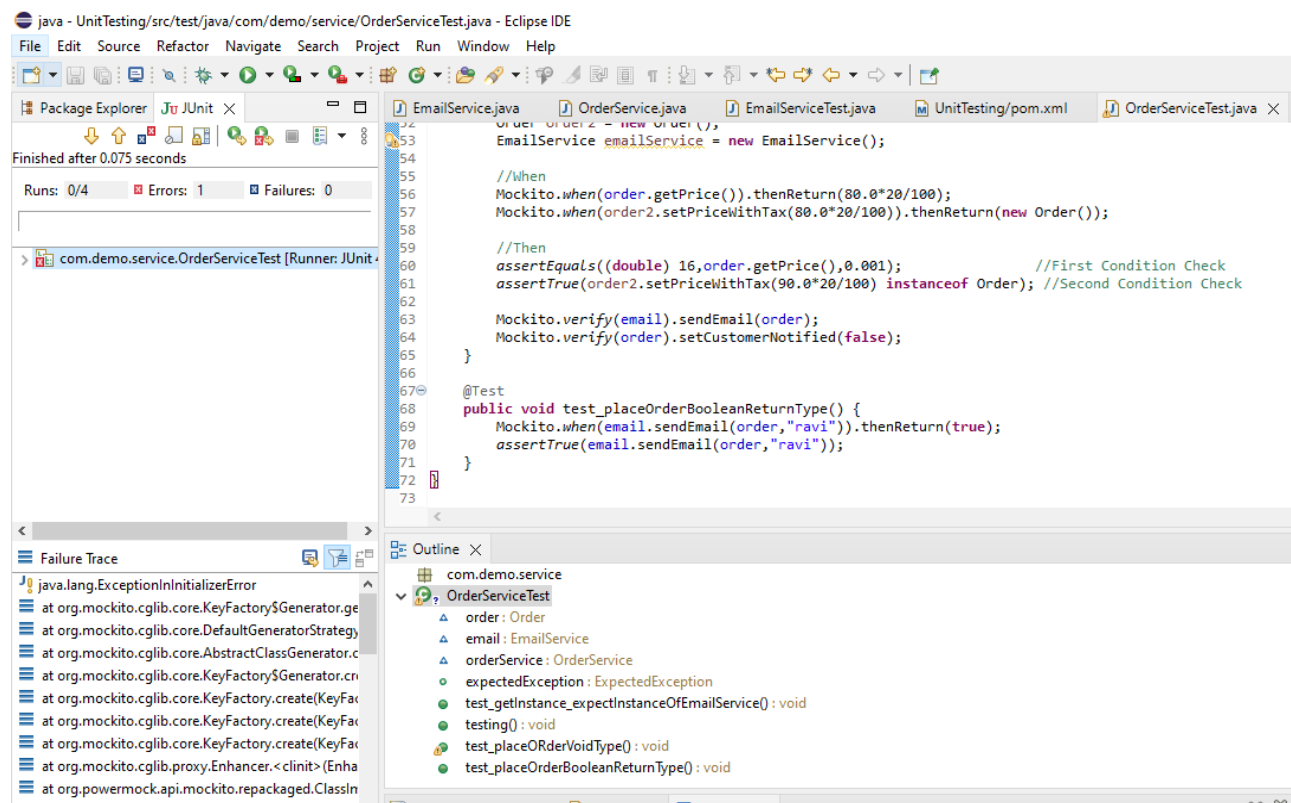
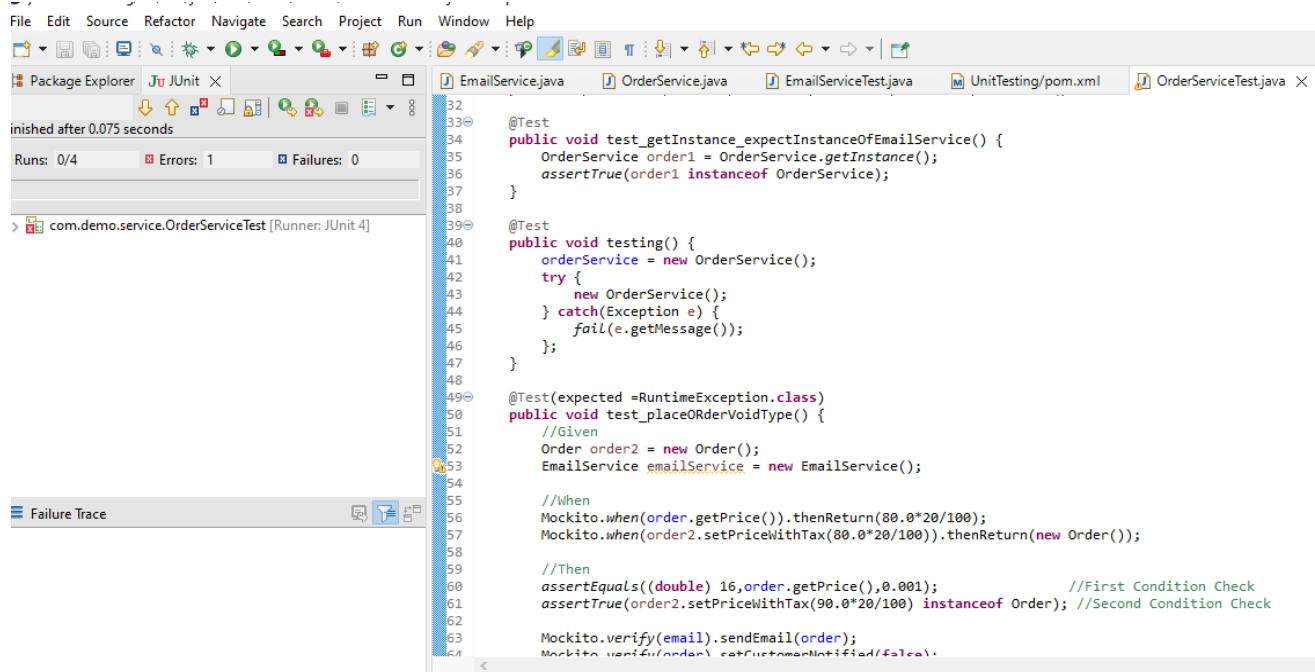
<https://github.com/nitin-singh-0208/unit-testing>

- OrderService.java



The screenshot shows the Eclipse IDE interface. The top toolbar includes icons for File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The Package Explorer on the left shows the project structure with 'JUnit' selected. The Run and Debug console shows 'Finished after 0.075 seconds' and 'Runs: 0/4', 'Errors: 1', 'Failures: 0'. The main editor displays the source code for 'OrderServiceTest.java'.

```
1 package com.demo.service;
2
3 import static org.junit.Assert.assertEquals;
4 import static org.junit.Assert.assertTrue;
5 import static org.junit.Assert.fail;
6
7 import org.junit.Rule;
8 import org.junit.Test;
9 import org.junit.rules.ExpectedException;
10 import org.junit.runner.RunWith;
11 import org.mockito.InjectMocks;
12 import org.mockito.Mock;
13 import org.mockito.Mockito;
14 import org.mockito.runners.MockitoJUnitRunner;
15
16 import com.demo.domain.Order;
17
18 @RunWith(MockitoJUnitRunner.class)
19 public class OrderServiceTest {
20
21     @Mock
22     Order order;
23
24     @Mock
25     EmailService email;
26
27     @InjectMocks
28     OrderService orderService;
29
30     @Rule
31     public ExpectedException expectedException = ExpectedException.none();
32 }
```



- EmailService.java

The screenshot shows the Eclipse IDE interface. The top menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The Package Explorer on the left shows the project structure, with the test class `com.demo.service.EmailServiceTest` selected. The JUnit runner indicates the test finished after 0.092 seconds, with 2/2 runs, 0 errors, and 0 failures. The main editor displays the source code for `EmailServiceTest.java`, which includes imports for JUnit and Mockito, and two test methods: `sendEmailTest()` and `setcheckmailException()`. The Outline view at the bottom right shows the class structure, including the `emailService` field and the two test methods.

```
1 package com.demo.service;
2
3 import static org.junit.Assert.assertTrue;
4
5 import org.junit.Test;
6 import org.junit.runner.RunWith;
7 import org.mockito.InjectMocks;
8 import org.mockito.runners.MockitoJUnitRunner;
9 import com.demo.domain.Order;
10
11 @RunWith(MockitoJUnitRunner.class)
12 public class EmailServiceTest {
13
14     @InjectMocks
15     EmailService emailService;
16
17     @Test
18     public void sendEmailTest() {
19         Order o1 = new Order(5, "biscuits", 34 );
20         boolean b = emailService.sendEmail(o1,"ravi gupta");
21         assertTrue(b);
22     }
23
24     @Test(expected = RuntimeException.class)
25     public void setcheckmailException() {
26         emailService.sendEmail(new Order());
27     }
28 }
```