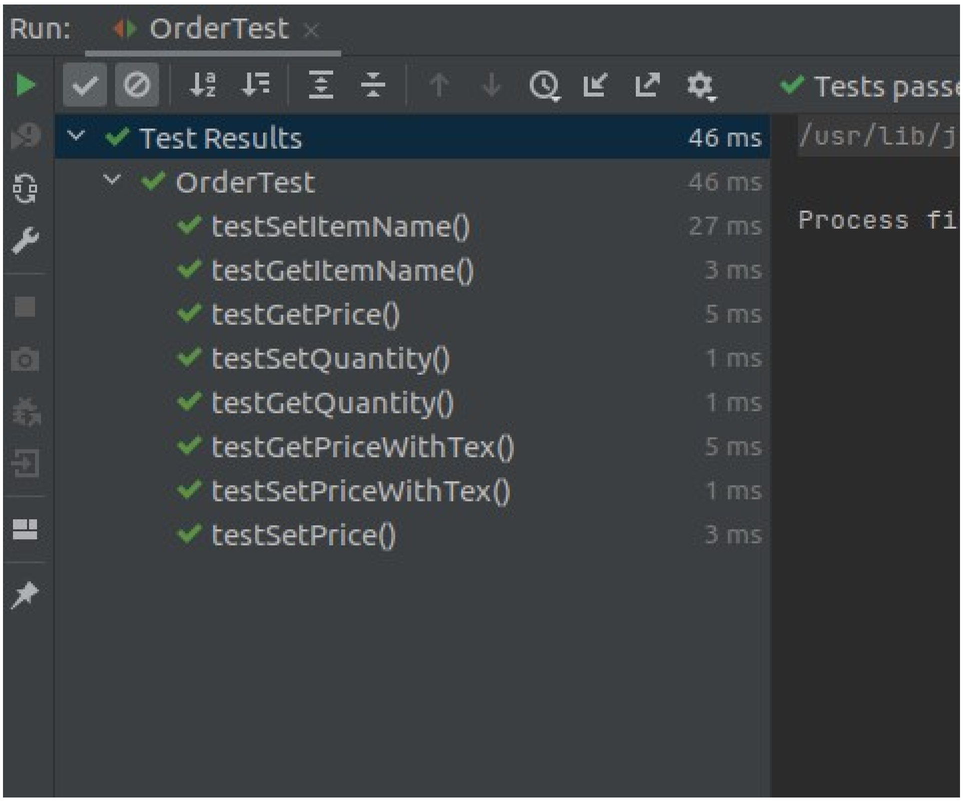
Java: Unit testing & Clean Code Exercise

OrderTest.java

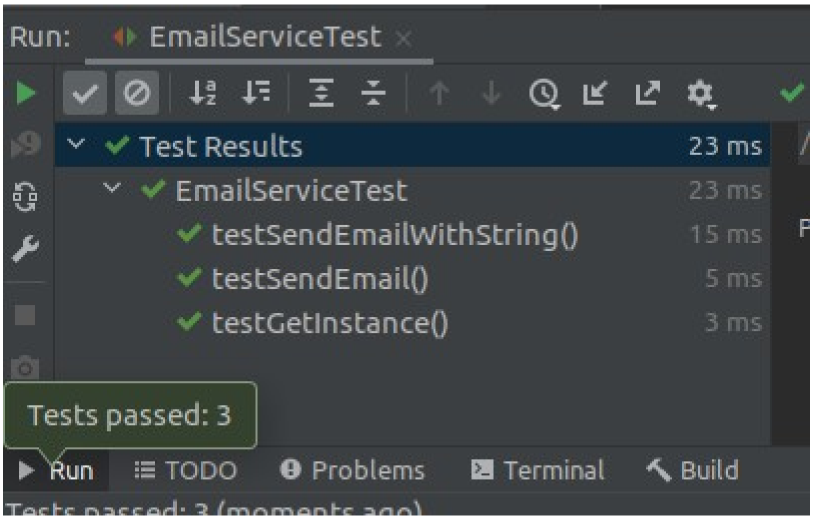
```
import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.*;

class OrderTest {

    private Order order = new Order(2, "Pen", 200.50);
    @Test
    void testGetQuantity() {
        int expected = 2;
        assertTrue(order.getQuantity() == expected);
    }

    @Test
    void testSetQuantity() {
        Order returnObj = order.setQuantity(2);
        assertEquals(returnObj, order);
    }

    @Test
    void testSetItemName() {
        Order returnObj = order.setItemName("Pen");
        assertEquals(returnObj, order);
    }

    @Test
    void testGetItemName() {
        String expected = "Pen";
        String actual = order.getItemName();
        assertTrue(expected.equals(actual));
    }

    @Test
    void testGetPrice() {
        Double expected = 200.5;
        Double actual = order.getPrice();
        assertEquals(expected, actual);
    }

    @Test
    void testSetPrice() {
        Order returnObj = order.setPrice(200.5);
        assertEquals(returnObj, order);
    }


    @Test
    void testGetPriceWithTex() {
        Double expected = 307.75;
        Double actual = 300.75;
        assertNotEquals(expected, actual);
    }

    @Test
    void testSetPriceWithTex() {
        Double expected = 300.75;
        Double actual = 300.75;
        assertTrue(expected.equals(actual));
    }
}
```



```
✓ Tests passe
✓ Test Results                   46 ms    /usr/lib/j
  ✓ OrderTest                    46 ms
    ✓ testSetItemName()          27 ms    Process fi
    ✓ testGetItemName()           3 ms
    ✓ testGetPrice()              5 ms
    ✓ testSetQuantity()           1 ms
    ✓ testGetQuantity()           1 ms
    ✓ testGetPriceWithTex()       5 ms
    ✓ testSetPriceWithTex()       1 ms
    ✓ testSetPrice()              3 ms
```

EmailServiceTest.java

```
import org.junit.jupiter.api.Test;
import org.junit.jupiter.api.function.Executable;

import static org.junit.jupiter.api.Assertions.*;

class EmailServiceTest {

    Order order = new Order();
    EmailService email=new EmailService();

    @Test
    void testGetInstance()
    {
        assertTrue(EmailService.getInstance() instanceof EmailService);
    }

    @Test
    void testSendEmail()
    {

        Executable executable=()->email.sendEmail(order);
        assertThrows(RuntimeException.class,executable);
    }

    @Test
    void testSendEmailWithString()
    {
        assertTrue(email.sendEmail(order,"Mobile"));
    }


}
```



```
Test Results                    23 ms
  EmailServiceTest              23 ms
    testSendEmailWithString()   15 ms
    testSendEmail()              5 ms
    testGetInstance()            3 ms
Tests passed: 3
Tests passed: 3 (moments ago)
```