**Finish the incomplete test cases for following:**

* **Order.java**

package test.java.demo.domain;

import demo.domain.Order;

import org.junit.jupiter.api.BeforeEach;

import org.junit.jupiter.api.Test;

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

public class OrderTest {

private Order order;

@BeforeEach

void setup(){

this.order = new Order(20,"Item1",500.0);

}

@Test

public void testSetQuantity() {

//given

int expectedQuantity = 20;

//when

Order recommendedOrder=order.setQuantity(expectedQuantity);

//then

assertEquals(order,recommendedOrder);

}

@Test

public void testGetQuantity() {

//given

int expectedQuantity=20;

//when

int actual=order.getQuantity();

//then

assertEquals(expectedQuantity,actual);

}

@Test

public void testSetItemName() {

//given

String expectedName = "Item2";

//when

Order recommendedOrder = order.setItemName(expectedName);

//then

assertEquals(order,recommendedOrder);

}

@Test

public void testGetItem() {

//given

String expectedName= "Item1";

//when

String actual = order.getItemName();

//then

assertEquals(expectedName,actual);

}

@Test

public void testGetPrice() {

//given

double expectedPrice = 500.0;

//when

double actualPrice=order.getPrice();

//then

assertEquals(expectedPrice,actualPrice);

}

@Test

public void testSetPrice() {

//given

double expectedPrice = 300.0;

//when

Order actualOrder = order.setPrice(expectedPrice);

//then

assertEquals(order,actualOrder);

}

@Test

public void testGetPriceWithTex() {

//given

double expectedPriceWithTax=0.0;

//when

double actualPriceWithTax = new Order().getPriceWithTex();

//then

assertEquals(expectedPriceWithTax,actualPriceWithTax);

}

@Test

public void testSetPriceWithTex() {

//given

double expectedPriceWithTax=0.0;

//when

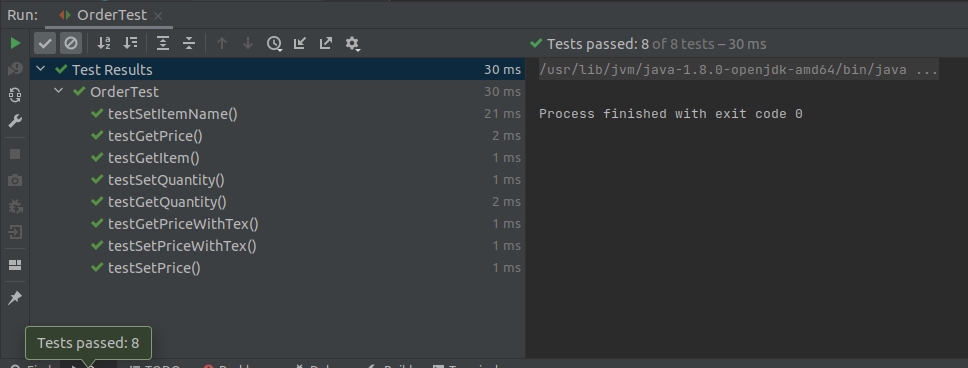
Order actualPriceWithTax=order.setPriceWithTex(expectedPriceWithTax);

//then

assertEquals(order,actualPriceWithTax);

}

}



* **EmailService.java**

package test.java.demo.service;

import main.java.demo.service.EmailService;

import org.junit.jupiter.api.Test;

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

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

public class EmailServiceTest {

@Test

public void testSingleton() {

EmailService instance1 = EmailService.getInstance();

EmailService instance2 = EmailService.getInstance();

System.out.println("checking singleton objects equality");

boolean actualValue = (instance1 == instance2);

assertEquals(true, actualValue);

}

@Test

public void whenExceptionThrown\_thenAssertionSucceeds(){

demo.domain.Order order = new demo.domain.Order();

Executable executable = () -> EmailService.getInstance().sendEmail(order);

Exception exception = assertThrows(RuntimeException.class, executable);

}

@Test

void isEmailSent(){

String emailStatus = "Email sent successfully";

demo.domain.Order order = new demo.domain.Order();

assertTrue(EmailService.getInstance().sendEmail((demo.domain.Order) order,emailStatus));

}

}

