DTS JUnit-JMeter Capstone Assignment

1. Small e-commerce application testing using Eclipse IDE:
2. Product.java:

package Shop;

public class Product {

// Attributes

private String name; // The name of the product (e.g., "Laptop")

private double price; // The price of the product (e.g., 1000.0)

private int quantity; // The quantity of the product (e.g., 2)

// Constructor

public Product(String name, double price, int quantity) {

if (price < 0 || quantity < 0) {

throw new IllegalArgumentException("Price and quantity must be non-negative.");

}

this.name = name;

this.price = price;

this.quantity = quantity;

}

// Methods

public double getTotalPrice() {

return price \* quantity; // Returns the total price (price \* quantity)

}

public String getName() {

return name; // Returns the name of the product

}

public double getPrice() {

return price; // Returns the price of the product

}

public int getQuantity() {

return quantity; // Returns the quantity of the product

}

}

1. ShoppingCart.java:

package Shop;

import java.util.List;

import java.util.ArrayList;

public class ShoppingCart {

// Attribute

private List<Product> products; // A list to store products in the cart

// Constructor

public ShoppingCart() {

this.products = new ArrayList<>();

}

// Methods

public void addProduct(Product product) {

this.products.add(product); // Adds a product to the cart

}

public double getTotalPrice() {

double total = 0;

for (Product product : products) {

total += product.getTotalPrice(); // Calculates the total price

}

return total;

}

public int getProductCount() {

return products.size(); // Returns the number of products in the cart

}

public List<Product> getProducts() {

return products; // Returns the list of products in the cart

}

}

1. ProductTests.java:

package Tests;

import org.junit.jupiter.api.Assertions;

import org.junit.jupiter.api.Test;

import Shop.Product;

public class ProductTests {

@Test

void testProductTotalPrice() {

Product product = new Product("Laptop", 1000.0, 2);

double totalPrice = product.getTotalPrice();

Assertions.*assertEquals*(2000.0, totalPrice, 0.001);

}

}

1. ShoppingCartTests.java:

package Tests;

import org.junit.jupiter.api.Assertions;

import org.junit.jupiter.api.Test;

import Shop.Product;

import Shop.ShoppingCart;

public class ShoppingCartTests {

@Test

void testAddProductToShoppingCart() {

ShoppingCart cart = new ShoppingCart();

Product product = new Product("Laptop", 1000.0, 1);

cart.addProduct(product);

Assertions.*assertTrue*(cart.getProducts().contains(product));

}

@Test

void testGetTotalPriceOfShoppingCart() {

ShoppingCart cart = new ShoppingCart();

Product product1 = new Product("Laptop", 1000.0, 1);

Product product2 = new Product("Mouse", 50.0, 2);

cart.addProduct(product1);

cart.addProduct(product2);

double totalPrice = cart.getTotalPrice();

Assertions.*assertEquals*(1100.0, totalPrice, 0.001);

}

@Test

void testGetProductCountInShoppingCart() {

ShoppingCart cart = new ShoppingCart();

Product product1 = new Product("Laptop", 1000.0, 1);

Product product2 = new Product("Mouse", 50.0, 2);

cart.addProduct(product1);

cart.addProduct(product2);

int productCount = cart.getProductCount();

Assertions.*assertEquals*(2, productCount);

}

@Test

void testAddProductWithInvalidData() {

ShoppingCart cart = new ShoppingCart();

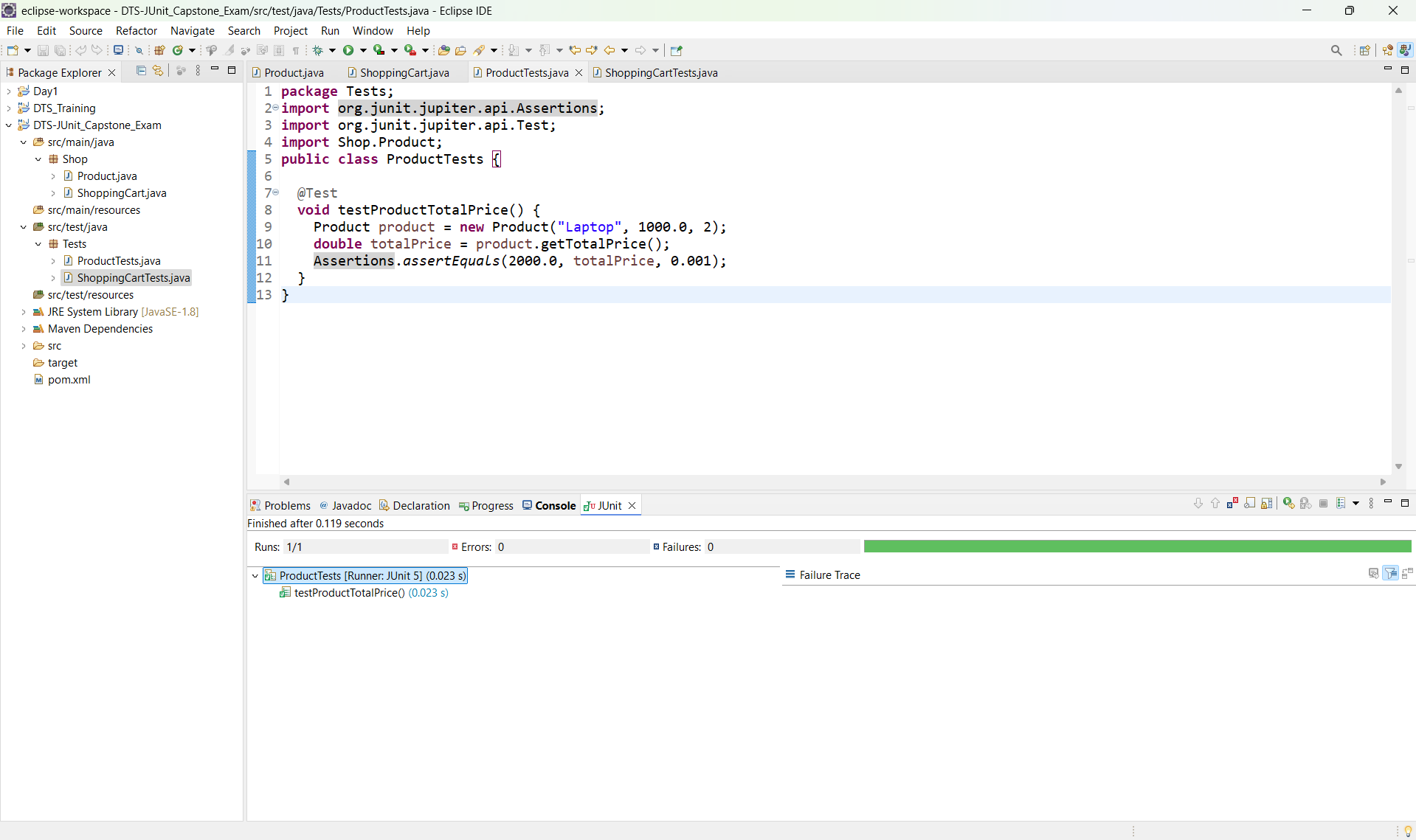
Assertions.*assertThrows*(IllegalArgumentException.class, () -> cart.addProduct(new Product("Laptop", -1000.0, 1)));

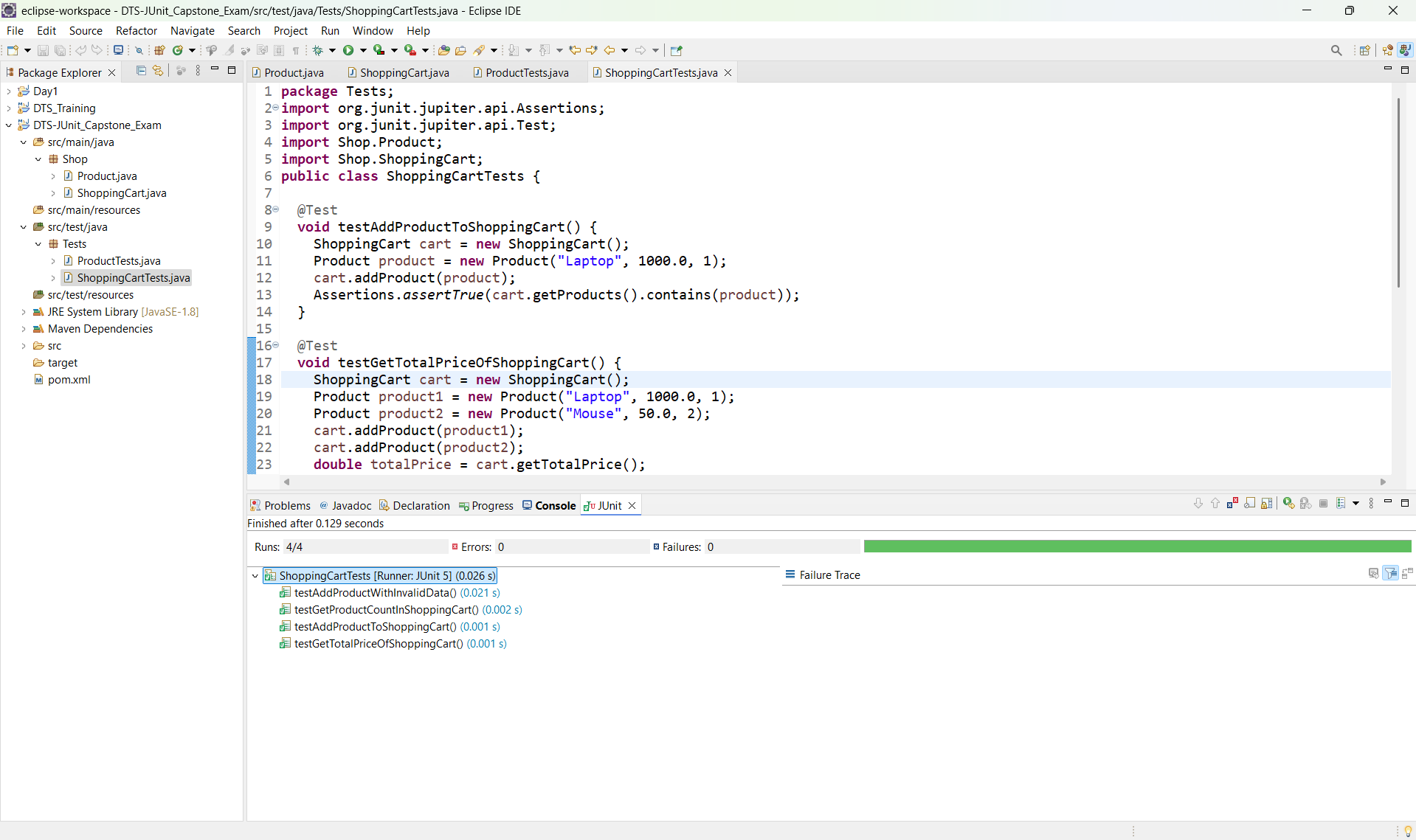
Assertions.*assertThrows*(IllegalArgumentException.class, () -> cart.addProduct(new Product("Mouse", 50.0, -2)));

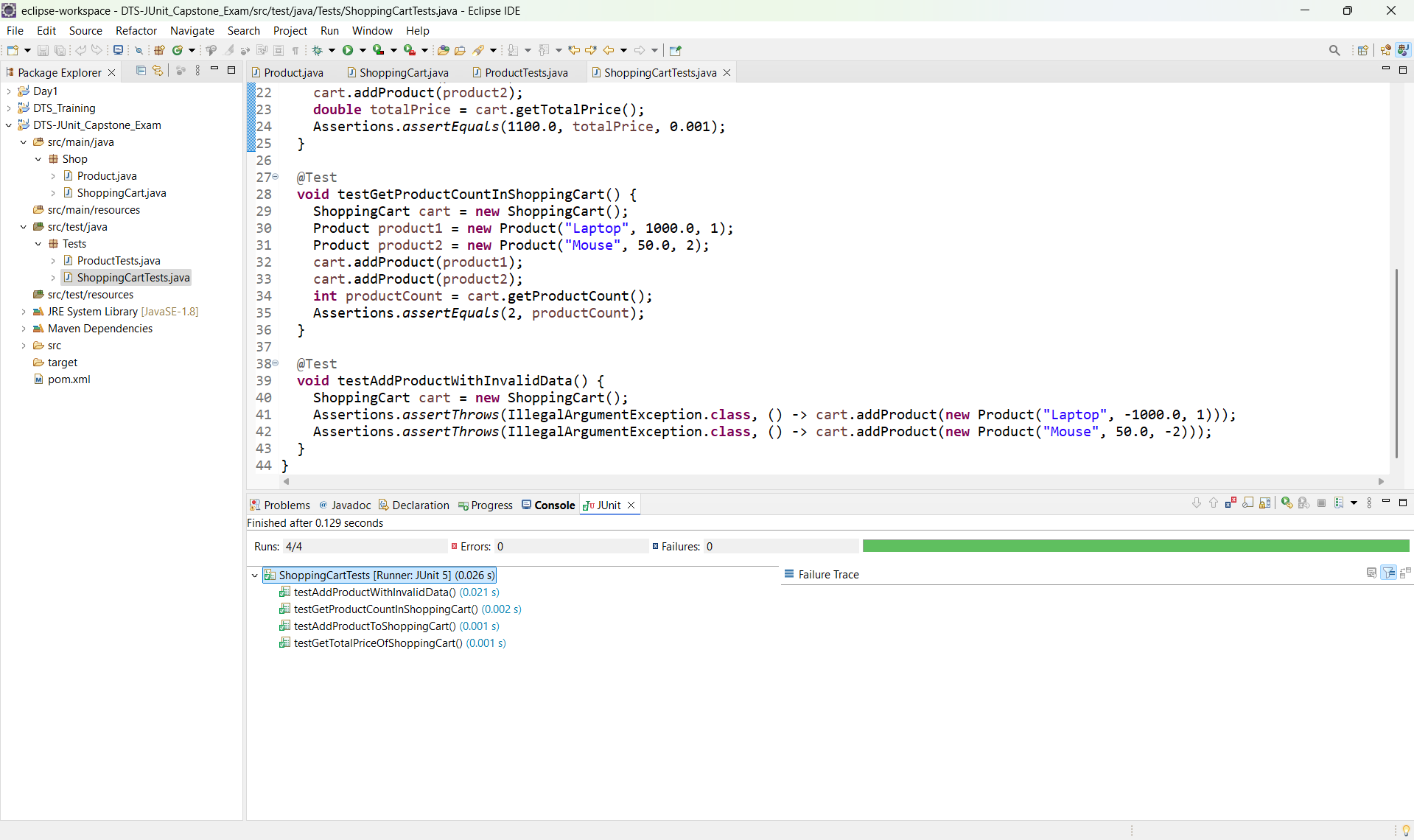
}

}

OUTPUT:







1. Create a Test Plan in JMeter for simulating traffic to the rest api use appropriate listeners and samplers:

