**Exercise 4: Arrange-Act-Assert (AAA) Pattern, Test Fixtures, Setup and Teardown Methods in JUnit**

**CODE**

**OrderProcessor.java**

package com.example;

public class OrderProcessor {

private boolean orderPlaced;

public boolean placeOrder() {

orderPlaced = true;

return orderPlaced;

}

public boolean cancelOrder() {

if (orderPlaced) {

orderPlaced = false;

return true;

}

else {

return false;

}

}

public boolean isOrderPlaced() {

return orderPlaced;

}

}

**OrderProcessorTest.java**

package com.example;

import static org.junit.Assert.\*;

import org.junit.Before;

import org.junit.After;

import org.junit.Test;

public class OrderProcessorTest {

private OrderProcessor orderProcessor;

@Before

public void setUp() {

orderProcessor = new OrderProcessor();

System.out.println("Setup: New OrderProcessor created.");

}

@After

public void tearDown() {

orderProcessor = null;

System.out.println("Teardown: OrderProcessor destroyed.");

}

@Test

public void testPlaceOrder() {

boolean result = orderProcessor.placeOrder();

assertTrue(result);

assertTrue(orderProcessor.isOrderPlaced());

}

@Test

public void testCancelOrder() {

orderProcessor.placeOrder();

boolean result = orderProcessor.cancelOrder();

assertTrue(result);

assertFalse(orderProcessor.isOrderPlaced());

}

@Test

public void testCancelWithoutPlacing() {

boolean result = orderProcessor.cancelOrder();

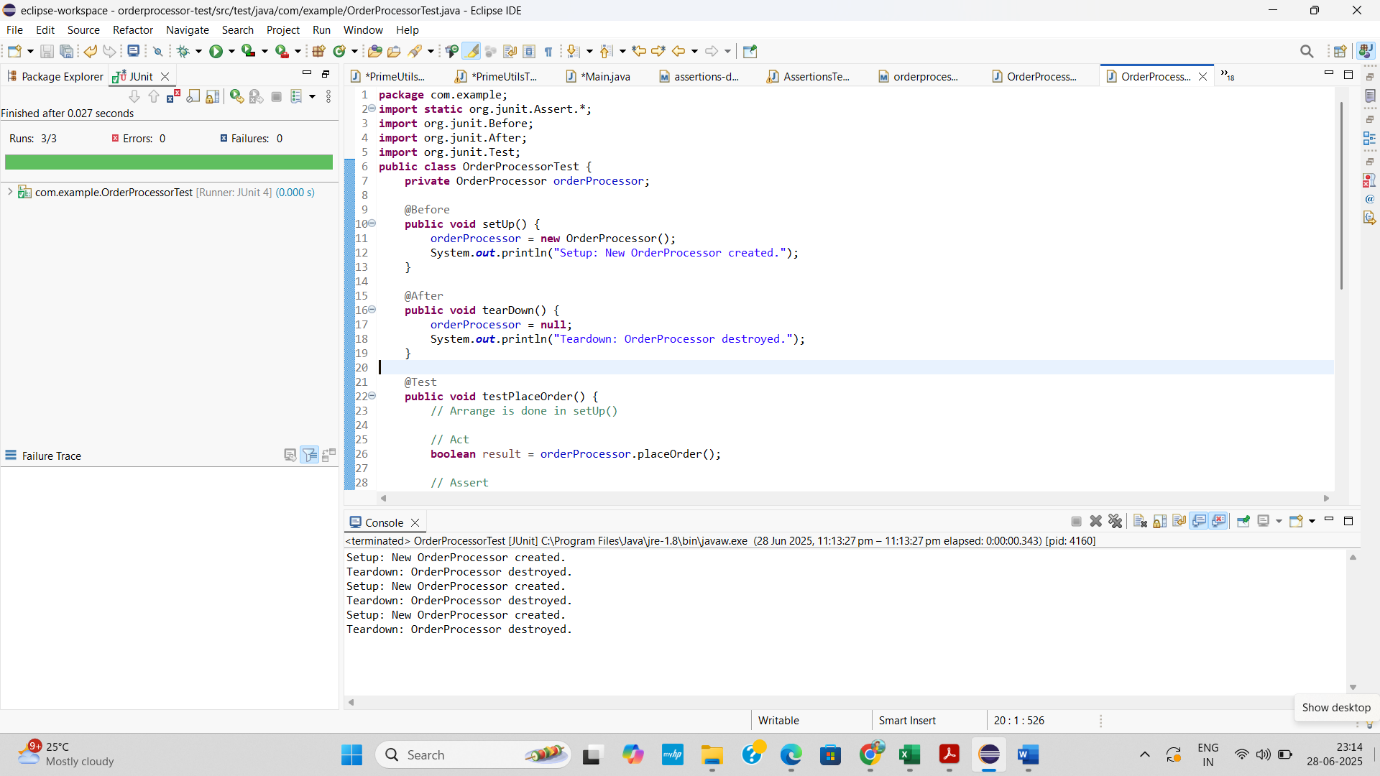
assertFalse(result);

assertFalse(orderProcessor.isOrderPlaced());

}

}

**OUTPUT**

****