**Exercise 1: Setting Up JUnit in Java using Eclipse (Without Maven)**

Calculator.java

**package** javv;

**public** **class** Calculator {

**public** **static** **int** calculate(**int** a , **int** b) {

**return** a\*b;

}

**public** String concat(String a ,String b) {

**return** a + b;

}

}

Concat.java

package javv;

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

import org.junit.jupiter.api.Test;

public class concat {

@Test

public void concatTest() {

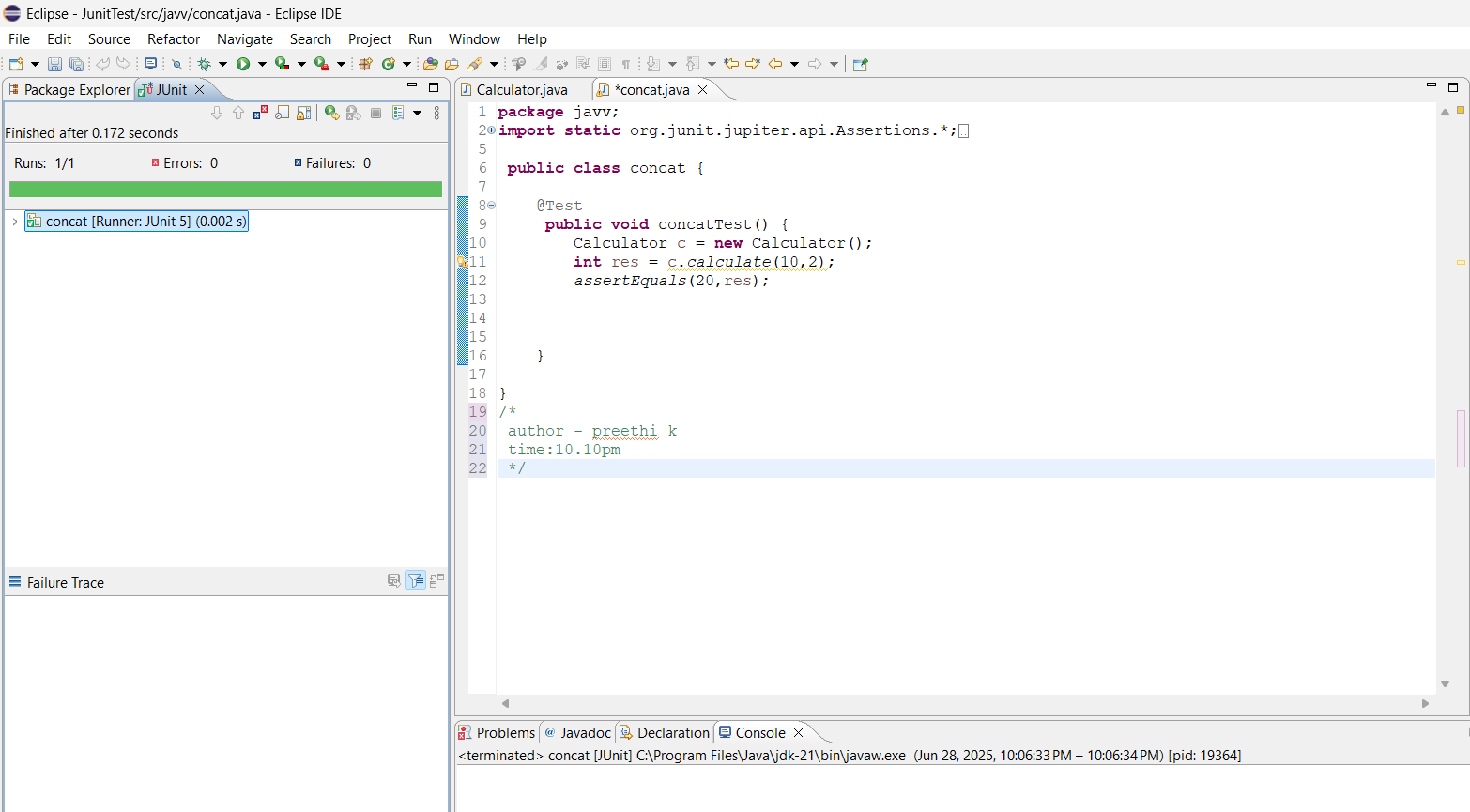
Calculator c = new Calculator();

int res = c.calculate(10,2);

assertEquals(20,res);

}

}



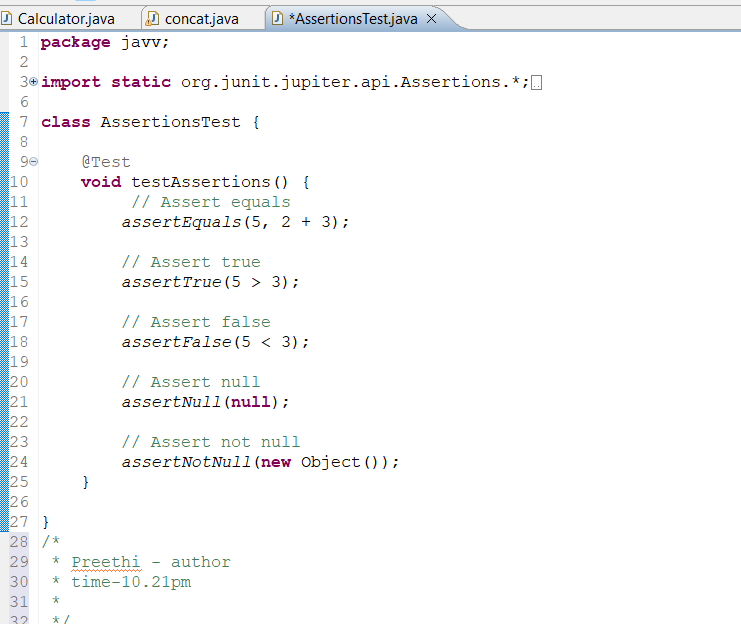
Error in Junit testing

A screenshot of a computer

AI-generated content may be incorrect.

**Exercise 3: Assertions in Junit**

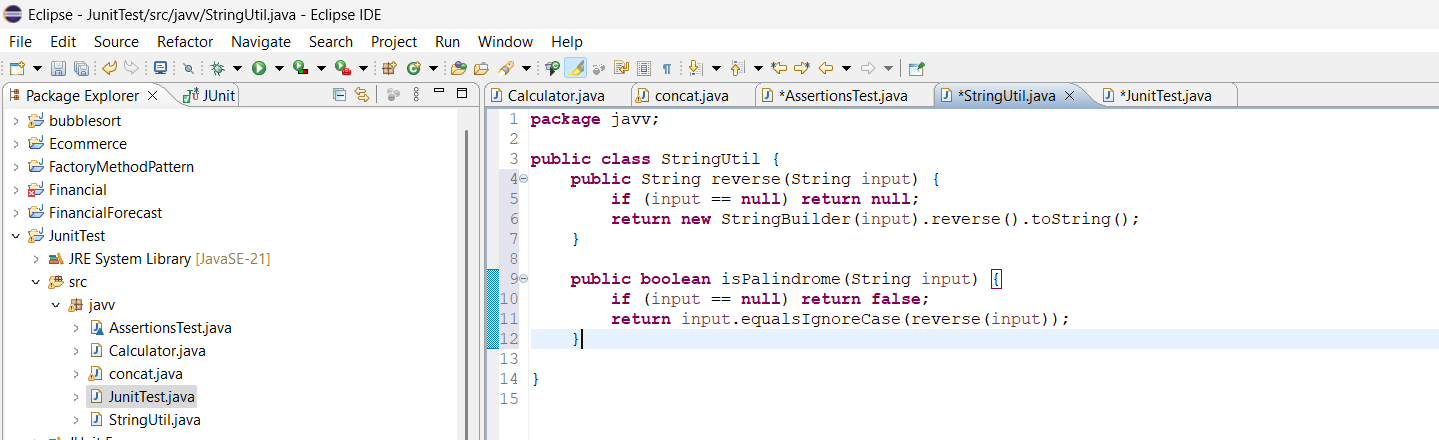
**AssertionsTest.java**

****

**A screenshot of a computer

AI-generated content may be incorrect.**

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

****

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screen shot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer program

AI-generated content may be incorrect.**

**Error :**

**A screenshot of a computer

AI-generated content may be incorrect.**

**. Mockito exercises**

**Exercise 1: Mocking and Stubbing**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screen shot of a computer

AI-generated content may be incorrect.**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**Exercise 2: Verifying Interactions**

**Pom.xml**

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0

http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.example</groupId>

<artifactId>mockitotest</artifactId>

<version>1.0-SNAPSHOT</version>

<packaging>jar</packaging>

<dependencies>

<!-- JUnit 5 -->

<dependency>

<groupId>org.junit.jupiter</groupId>

<artifactId>junit-jupiter</artifactId>

<version>5.10.0</version>

<scope>test</scope>

</dependency>

<!-- Mockito -->

<dependency>

<groupId>org.mockito</groupId>

<artifactId>mockito-core</artifactId>

<version>5.7.0</version>

<scope>test</scope>

</dependency>

</dependencies>

<build>

<plugins>

<!-- Enable JUnit 5 with Maven -->

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-surefire-plugin</artifactId>

<version>3.0.0-M9</version>

</plugin>

</plugins>

</build>

</project>

**MockitoTest.java**

**package** com.example;

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

**import** **static** org.mockito.Mockito.\*;

**class** ExternalApi {

**void** getData() {

// mock method

}

}

**class** MyService {

**private** ExternalApi api;

// Constructor that accepts a mock or real object

**public** MyService(ExternalApi api) {

**this**.api = api;

}

**public** **void** fetchData() {

api.getData(); // This will throw NullPointerException if api is null

}

}

**public** **class** mockitoTest {

@Test

**void** testVerifyInteraction() {

// Step 1: Create mock

ExternalApi mockApi = *mock*(ExternalApi.**class**);

// Step 2: Inject the mock into MyService

MyService service = **new** MyService(mockApi);

// Step 3: Call the method that should use the mock

service.fetchData();

// Step 4: Verify that getData() was called on the mock

*verify*(mockApi).getData();

}

}

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**6. SL4J Logging exercises**

**Exercise 1: Logging Error Messages and Warning Levels**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Pom.xml**

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0

http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.example</groupId>

<artifactId>logging-example</artifactId>

<version>1.0-SNAPSHOT</version>

<dependencies>

<!-- SLF4J API -->

<dependency>

<groupId>org.slf4j</groupId>

<artifactId>slf4j-api</artifactId>

<version>1.7.30</version>

</dependency>

<!-- Logback Classic -->

<dependency>

<groupId>ch.qos.logback</groupId>

<artifactId>logback-classic</artifactId>

<version>1.2.3</version>

</dependency>

</dependencies>

<!-- Optional: Plugin for running main class -->

<build>

<plugins>

<plugin>

<groupId>org.codehaus.mojo</groupId>

<artifactId>exec-maven-plugin</artifactId>

<version>1.6.0</version>

<configuration>

<mainClass>com.example.LoggingExample</mainClass>

</configuration>

</plugin>

</plugins>

</build>

</project>

**OUTPUT:**

**A screenshot of a computer

AI-generated content may be incorrect.**