**Exercise1:Setting up Junit Test**

**JUnitDemo/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>JUnitDemo</artifactId>

<version>0.0.1-SNAPSHOT</version>

<dependencies>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.13.2</version>

<scope>test</scope>

</dependency>

</dependencies>

</project>

**FactorialCalculator.java**

**package** com.example;

**public** **class** FactorialCalculator {

**public** **int** factorial(**int** n) {

**if** (n < 0) {

**throw** **new** IllegalArgumentException("Factorial is undefined for negative numbers.");

}

**int** result = 1;

**for** (**int** i = 2; i <= n; i++) {

result \*= i;

}

**return** result;

}

}

**FactorialCalculatorTest.java**

**package** com.example;

**import** org.junit.Test;

**import** **static** org.junit.Assert.*assertEquals*;

**public** **class** FactorialCalculatorTest {

@Test

**public** **void** testFactorialOfZero() {

FactorialCalculator calc = **new** FactorialCalculator();

*assertEquals*(1, calc.factorial(0)); // 0! = 1

}

@Test

**public** **void** testFactorialOfPositiveNumber() {

FactorialCalculator calc = **new** FactorialCalculator();

*assertEquals*(120, calc.factorial(5)); // 5! = 120

*assertEquals*(6, calc.factorial(3)); // 3! = 6

}

@Test(expected = IllegalArgumentException.**class**)

**public** **void** testFactorialOfNegativeNumber() {

FactorialCalculator calc = **new** FactorialCalculator();

calc.factorial(-4); // Should throw exception

}

}





