

1.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>DISPLAYRESULT_MAVEN</artifactId>
  <version>1.0-SNAPSHOT</version>

  <name>Student Result Application</name>

  <properties>
    <maven.compiler.source>17</maven.compiler.source>
    <maven.compiler.target>17</maven.compiler.target>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  </properties>

  <dependencies>

    <dependency>
      <groupId>org.junit.jupiter</groupId>
      <artifactId>junit-jupiter</artifactId>
      <version>5.10.0</version>
      <scope>test</scope>
    </dependency>
  </dependencies>

  <build>
    <plugins>

      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.11.0</version>
      </plugin>

      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-surefire-plugin</artifactId>
        <version>3.1.2</version>
      </plugin>
    </plugins>
  </build>
</project>
```

</project>

2. Create StudentResult.java

```
package com.example;

public class StudentResult {

    public int calculateTotal(int s1, int s2, int s3, int s4, int s5) {
        return s1 + s2 + s3 + s4 + s5;
    }

    public double calculateAverage(int total) {
        return total / 5.0;
    }

    public String calculateGrade(double average) {
        if (average >= 90)
            return "A+";
        else if (average >= 75)
            return "A";
        else if (average >= 60)
            return "B";
        else if (average >= 50)
            return "C";
        else
            return "Fail";
    }
}
```

3. Create StudentResultTest.java for JUNIT testing

```
package com.example;

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

class StudentResultTest {

    StudentResult sr = new StudentResult();

    @Test
    void testCalculateTotal() {
        int total = sr.calculateTotal(80, 85, 90, 75, 70);
        assertEquals(400, total);
    }
}
```

```
@Test
void testCalculateAverage() {
    double avg = sr.calculateAverage(400);
    assertEquals(80.0, avg);
}

@Test
void testCalculateGradeAPlus() {
    assertEquals("A+", sr.calculateGrade(95));
}

@Test
void testCalculateGradeB() {
    assertEquals("B", sr.calculateGrade(65));
}

@Test
void testCalculateGradeFail() {
    assertEquals("Fail", sr.calculateGrade(40));
}
}
```