# Run C001BasicExampleTest as a New Maven Project in IntelliJ

This document contains step-by-step instructions to create a new Maven project in IntelliJ, add the provided test, implement the production class, and run the test. Follow each numbered step in order.

## 1. Create a new Maven project

1. Open IntelliJ IDEA → File → New → Project...

2. Choose "Maven" on the left. Uncheck "Create from archetype". Click Next.

3. Fill:

• GroupId: org.p001

• ArtifactId: c001-basic-example

• Version: 1.0-SNAPSHOT (default)

Click Next → choose project location → Finish.

4. Wait for IntelliJ to create and import the Maven project and finish indexing.

## 2. Set Project SDK to Java 17

1. File → Project Structure → Project

2. Project SDK → Add if needed → point to your JDK 17 (Corretto/OpenJDK).

3. Ensure Project language level is 17. Click OK.

## 3. Replace pom.xml with minimal JUnit5 configuration

Open the generated pom.xml and replace its content (or merge relevant parts) with the following minimal configuration:

<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>org.p001</groupId>  
 <artifactId>c001-basic-example</artifactId>  
 <version>1.0-SNAPSHOT</version>  
  
 <properties>  
 <maven.compiler.source>17</maven.compiler.source>  
 <maven.compiler.target>17</maven.compiler.target>  
 <junit.jupiter.version>5.9.3</junit.jupiter.version>  
 </properties>  
  
 <dependencies>  
 <dependency>  
 <groupId>org.junit.jupiter</groupId>  
 <artifactId>junit-jupiter</artifactId>  
 <version>${junit.jupiter.version}</version>  
 <scope>test</scope>  
 </dependency>  
 </dependencies>  
  
 <build>  
 <plugins>  
 <plugin>  
 <groupId>org.apache.maven.plugins</groupId>  
 <artifactId>maven-surefire-plugin</artifactId>  
 <version>3.0.0-M8</version>  
 </plugin>  
 </plugins>  
 </build>  
</project>

## 4. Create package structure & files

1. In Project view expand src.

2. Right-click src/main/java → New → Package → enter org.p001.

3. Right-click the package → New → Java Class → C001BasicExample.

4. Right-click src/test/java → New → Package → enter org.p001.

5. Right-click the test package → New → Java Class → C001BasicExampleTest.

6. Paste the test code into C001BasicExampleTest.java (ensure package org.p001; is at the top).

## 5. Production class (C001BasicExample)

Create src/main/java/org/p001/C001BasicExample.java with the following content:

package org.p001;  
  
public class C001BasicExample {  
 public boolean isValidUserName(String userName) {  
 if (userName == null) return false;  
 String trimmed = userName.trim();  
 if (trimmed.isEmpty()) return false;  
 return trimmed.matches("^[A-Za-z0-9]+$");  
 }  
}

## 6. Test class (C001BasicExampleTest)

Create src/test/java/org/p001/C001BasicExampleTest.java and paste your test code (example below):

package org.p001;  
  
import org.junit.jupiter.api.Test;  
  
import static org.junit.jupiter.api.Assertions.assertFalse;  
import static org.junit.jupiter.api.Assertions.assertTrue;  
  
public class C001BasicExampleTest {  
  
 @Test  
 public void testIsValidUserName() {  
 // Arrange  
 C001BasicExample cut = new C001BasicExample();  
 String userName = "john1234";  
 // Act  
 boolean result = cut.isValidUserName(userName);  
 // Assert  
 assertTrue(result);  
  
  
 String userNameInvalid1 = "john\_1234";  
 String userNameInvalid2 = "john@1234";  
 String userNameInvalid3 = "";  
 String userNameInvalid4 = " ";  
 String userNameInvalid5 = null;  
  
 assertFalse(cut.isValidUserName(userNameInvalid1));  
 assertFalse(cut.isValidUserName(userNameInvalid2));  
 assertFalse(cut.isValidUserName(userNameInvalid3));  
 assertFalse(cut.isValidUserName(userNameInvalid4));  
 assertFalse(cut.isValidUserName(userNameInvalid5));  
 }  
}

## 7. Run the test in IntelliJ

1. Open C001BasicExampleTest.java.

2. Click the green Run icon in the left gutter next to the test method or class → Run 'C001BasicExampleTest'.

3. The Run window shows test results (green = pass). If it fails, inspect stack trace and fix the code or package mismatch.

## 8. Run tests from command line

Open terminal at the project root and run:

• All tests: mvn test

• Single test class: mvn -Dtest=org.p001.C001BasicExampleTest test

## 9. TDD-friendly IntelliJ tips

• Paste the test first and run it (RED). If the production class is missing, put the caret on the red class name in the test → Alt+Enter → Create class 'C001BasicExample'.

• Use the gutter Run/Debug icons to iterate quickly (Red → Green → Refactor).

• Create a JUnit run configuration for repeated runs: Run → Edit Configurations → + → JUnit → choose test class.

## 10. Troubleshooting checklist

• Package mismatch / ClassNotFound — ensure package line matches folder path (src/main/java/org/p001 etc.).

• JUnit not on classpath — ensure pom.xml saved and IntelliJ finished Maven import (Maven tool window → Reimport).

• Java version mismatch — set Project SDK to JDK 17 and ensure pom.xml compiler version is 17.

End of instructions.