



Res.MEN 014915 - 02 AGO 2022
RENOVACIÓN ACREDITACIÓN

Creando un nuevo proyecto desde la terminal

Programación Orientada a Objetos

Área de programación
Programa de Ingeniería de Sistemas y Computación
Facultad de Ingeniería

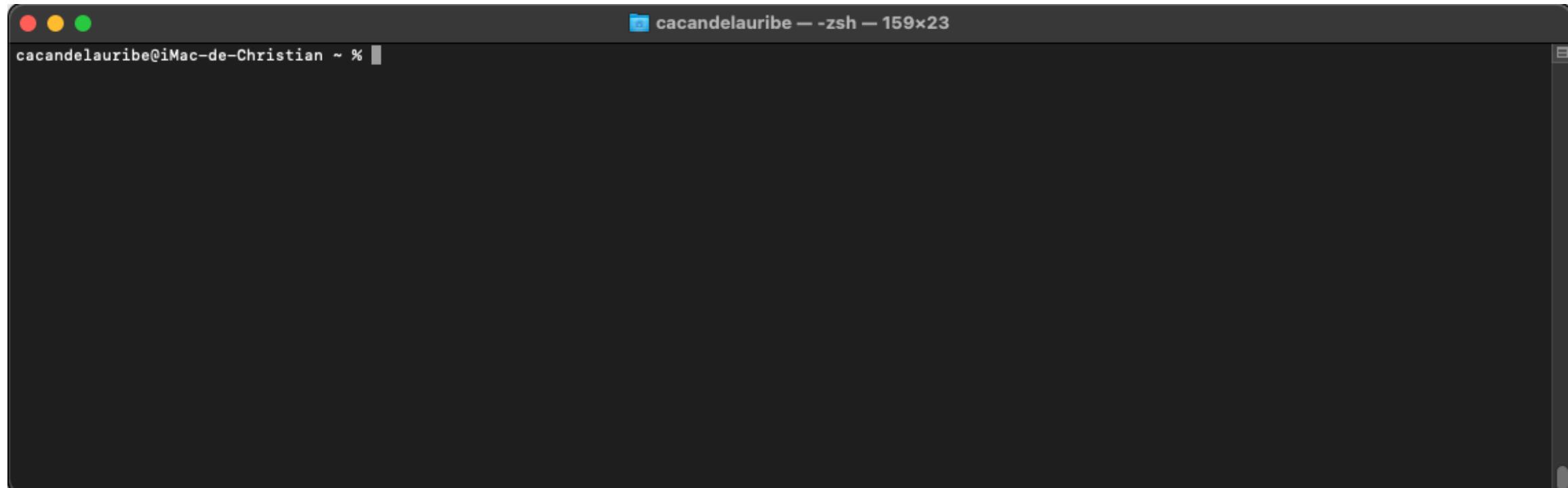
UNIQUINDÍO
en conexión territorial

www.uniquindio.edu.co





Abra una terminal (power shell en windows)





Listar las plantillas

```
mvn archetype:generate
```

```
cacandelaureibe — java -classpath /usr/local/apache-maven-3.8.6/boot/plexus-classworlds-2.6.0.jar -Dclassworlds.conf=/usr/local/apache-maven-3.8.6/bin/m2.con...
3292: remote -> tr.com.obss.sdlc.archetype:obss-archetype-webapp (This archetype provides a skelton for the Java Web Application packages.)
3293: remote -> ua.co.gravy.archetype:single-project-with-junit-and-slf4j (Create a single project with jUnit, Mockito and slf4j dependencies.)
3294: remote -> uk.ac.ebi.gxa:atlas-archetype (Archetype for generating a custom Atlas webapp)
3295: remote -> uk.ac.gate:gate-plugin-archetype (Maven archetype to create a new GATE plugin project.)
3296: remote -> uk.ac.gate:gate-pr-archetype (Maven archetype to create a new GATE plugin project including a sample PR class (an empty LanguageAnalyser).)
3297: remote -> uk.ac.nactem.argo:argo-analysis-engine-archetype (An archetype which contains a sample Argo (UIMA) Analysis Engine)
3298: remote -> uk.ac.nactem.argo:argo-reader-archetype (An archetype which contains a sample Argo (UIMA) Reader)
3299: remote -> uk.ac.rdg.resc:edal-ncwms-based-webapp (-)
3300: remote -> uk.co.markg.archetypes:java11-junit5 (An archetype for generate java 11 projects with junit 5.)
3301: remote -> uk.co.nemstix:basic-javaee7-archetype (A basic Java EE7 Maven archetype)
3302: remote -> uk.co.solong:angular-spring-archetype (So Long archetype for RESTful spring services with an AngularJS frontend. Includes debian deployment)
3303: remote -> us.fatehi:schemacrawler-archetype-maven-project (-)
3304: remote -> us.fatehi:schemacrawler-archetype-plugin-command (-)
3305: remote -> us.fatehi:schemacrawler-archetype-plugin-dbconnector (-)
3306: remote -> us.fatehi:schemacrawler-archetype-plugin-lint (-)
3307: remote -> ws.nzen.format.maven:zaftig_offal_hamisha (A maven archetype for my style of project.)
3308: remote -> ws.osiris:osiris-archetype (Maven Archetype for Osiris)
3309: remote -> xyz.luan.generator:xyz-gae-generator (-)
3310: remote -> xyz.luan.generator:xyz-generator (-)
3311: remote -> za.co.absa.hyperdrive:component-archetype (-)
3312: remote -> za.co.absa.hyperdrive:component-archetype_2.11 (-)
3313: remote -> za.co.absa.hyperdrive:component-archetype_2.12 (-)
Choose a number or apply filter (format: [groupId:]artifactId, case sensitive contains): 2072: ■
```

Al presionar enter seleccionará el archetype por defecto 2072 (quick start).



Seleccione la Versión más actual, en este caso 1.4

```
202302 -- java -classpath /usr/local/apache-maven-3.8.6/boot/plexus-classworlds-2.6.0.jar -Dclassworlds.conf=/usr/local/apache-maven-3.8.6/bin/m2.conf -Dmav...  
3302: remote -> uk.co.solong:angular-spring-archetype (So Long archetype for RESTful spring services with an AngularJS frontend. Includes debian deployment)  
3303: remote -> us.fatehi:schemacrawler-archetype-maven-project (-)  
3304: remote -> us.fatehi:schemacrawler-archetype-plugin-command (-)  
3305: remote -> us.fatehi:schemacrawler-archetype-plugin-dbconnector (-)  
3306: remote -> us.fatehi:schemacrawler-archetype-plugin-lint (-)  
3307: remote -> ws.nzen.format.maven:zaftig_offal_hamisha (A maven archetype for my style of project.)  
3308: remote -> ws.osiris:osiris-archetype (Maven Archetype for Osiris)  
3309: remote -> xyz.luan.generator:xyz-gae-generator (-)  
3310: remote -> xyz.luan.generator:xyz-generator (-)  
3311: remote -> za.co.absa.hyperdrive:component-archetype (-)  
3312: remote -> za.co.absa.hyperdrive:component-archetype_2.11 (-)  
3313: remote -> za.co.absa.hyperdrive:component-archetype_2.12 (-)  
Choose a number or apply filter (format: [groupId:]artifactId, case sensitive contains): 2072:  
Choose org.apache.maven.archetypes:maven-archetype-quickstart version:  
1: 1.0-alpha-1  
2: 1.0-alpha-2  
3: 1.0-alpha-3  
4: 1.0-alpha-4  
5: 1.0  
6: 1.1  
7: 1.3  
8: 1.4  
Choose a number: 8: █
```

Al presionar enter seleccionará la versión por defecto 8 (1.4)



Escriba el "group id" para el caso co.edu.uniquindio.poo

```
202302 — java -classpath /usr/local/apache-maven-3.8.6/boot/plexus-classworlds-2.6.0.jar -Dclassworlds.conf=/usr/local/apache-maven-3.8.6/bin/m2.conf -Dmav...  
3309: remote -> xyz.luan.generator:xyz-gae-generator (-)  
3310: remote -> xyz.luan.generator:xyz-generator (-)  
3311: remote -> za.co.absa.hyperdrive:component-archetype (-)  
3312: remote -> za.co.absa.hyperdrive:component-archetype_2.11 (-)  
3313: remote -> za.co.absa.hyperdrive:component-archetype_2.12 (-)  
Choose a number or apply filter (format: [groupId:]artifactId, case sensitive contains): 2072:  
Choose org.apache.maven.archetypes:maven-archetype-quickstart version:  
1: 1.0-alpha-1  
2: 1.0-alpha-2  
3: 1.0-alpha-3  
4: 1.0-alpha-4  
5: 1.0  
6: 1.1  
7: 1.3  
8: 1.4  
Choose a number: 8:  
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom  
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom (1.  
6 kB at 21 kB/s)  
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar  
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar (7.  
1 kB at 178 kB/s)  
Define value for property 'groupId': co.edu.uniquindio.poo
```

Tras ingresar el group id presione enter.



Escriba el nombre del proyecto, por ejemplo
"nombreproyecto"

```
202302 -- java -classpath /usr/local/apache-maven-3.8.6/boot/plexus-classworlds-2.6.0.jar -Dclassworlds.conf=/usr/local/apache-maven-3.8.6/bin/m2.conf -Dmav...  
  
3310: remote -> xyz.luan.generator:xyz-generator (-)  
3311: remote -> za.co.absa.hyperdrive:component-archetype (-)  
3312: remote -> za.co.absa.hyperdrive:component-archetype_2.11 (-)  
3313: remote -> za.co.absa.hyperdrive:component-archetype_2.12 (-)  
Choose a number or apply filter (format: [groupId:]artifactId, case sensitive contains): 2072:  
Choose org.apache.maven.archetypes:maven-archetype-quickstart version:  
1: 1.0-alpha-1  
2: 1.0-alpha-2  
3: 1.0-alpha-3  
4: 1.0-alpha-4  
5: 1.0  
6: 1.1  
7: 1.3  
8: 1.4  
Choose a number: 8:  
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom  
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom (1.  
6 kB at 21 kB/s)  
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar  
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar (7.  
1 kB at 178 kB/s)  
Define value for property 'groupId': co.edu.uniquindio.poo  
Define value for property 'artifactId': nombreproyecto
```

Tras ingresar el nombre del proyecto (artifact id) presione enter.



Ingrese la versión del proyecto, por ejemplo 1.0

```
202302 — java -classpath /usr/local/apache-maven-3.8.6/boot/plexus-classworlds-2.6.0.jar -Dclassworlds.conf=/usr/local/apache-maven-3.8.6/bin/m2.conf -Dmav...  
3311: remote -> za.co.absa.hyperdrive:component-archetype (-)  
3312: remote -> za.co.absa.hyperdrive:component-archetype_2.11 (-)  
3313: remote -> za.co.absa.hyperdrive:component-archetype_2.12 (-)  
Choose a number or apply filter (format: [groupId:]artifactId, case sensitive contains): 2072:  
Choose org.apache.maven.archetypes:maven-archetype-quickstart version:  
1: 1.0-alpha-1  
2: 1.0-alpha-2  
3: 1.0-alpha-3  
4: 1.0-alpha-4  
5: 1.0  
6: 1.1  
7: 1.3  
8: 1.4  
Choose a number: 8:  
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom  
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom (1.  
6 kB at 21 kB/s)  
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar  
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar (7.  
1 kB at 178 kB/s)  
Define value for property 'groupId': co.edu.uniquindio.poo  
Define value for property 'artifactId': nombreproyecto  
Define value for property 'version' 1.0-SNAPSHOT: : 1.0
```

Tras ingresar la versión del proyecto presione enter.



Acepte la creación del paquete "co.edu.uniquindio.poo"

```
202302 — java -classpath /usr/local/apache-maven-3.8.6/boot/plexus-classworlds-2.6.0.jar -Dclassworlds.conf=/usr/local/apache-maven-3.8.6/bin/m2.conf -Dmav...  
3312: remote -> za.co.absa.hyperdrive:component-archetype_2.11 (-)  
3313: remote -> za.co.absa.hyperdrive:component-archetype_2.12 (-)  
Choose a number or apply filter (format: [groupId:]artifactId, case sensitive contains): 2072:  
Choose org.apache.maven.archetypes:maven-archetype-quickstart version:  
1: 1.0-alpha-1  
2: 1.0-alpha-2  
3: 1.0-alpha-3  
4: 1.0-alpha-4  
5: 1.0  
6: 1.1  
7: 1.3  
8: 1.4  
Choose a number: 8:  
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom  
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom (1.  
6 kB at 21 kB/s)  
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar  
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar (7.  
1 kB at 178 kB/s)  
Define value for property 'groupId': co.edu.uniquindio.poo  
Define value for property 'artifactId': nombreproyecto  
Define value for property 'version' 1.0-SNAPSHOT: : 1.0  
Define value for property 'package' co.edu.uniquindio.poo: :
```

Al presionar enter aceptara el paquete por defecto (co.edu.uniquindio.poo)



Confirme los datos de creación

```
202302 — java -classpath /usr/local/apache-maven-3.8.6/boot/plexus-classworlds-2.6.0.jar -Dclassworlds.conf=/usr/local/apache-maven-3.8.6/bin/m2.conf -Dmav...  
3: 1.0-alpha-3  
4: 1.0-alpha-4  
5: 1.0  
6: 1.1  
7: 1.3  
8: 1.4  
Choose a number: 8:  
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom  
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom (1.  
6 kB at 21 kB/s)  
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar  
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar (7.  
1 kB at 178 kB/s)  
Define value for property 'groupId': co.edu.uniquindio.poo  
Define value for property 'artifactId': nombreproyecto  
Define value for property 'version' 1.0-SNAPSHOT: : 1.0  
Define value for property 'package' co.edu.uniquindio.poo: :  
Confirm properties configuration:  
groupId: co.edu.uniquindio.poo  
artifactId: nombreproyecto  
version: 1.0  
package: co.edu.uniquindio.poo  
Y: :
```

Al presionar enter confirmará los datos.

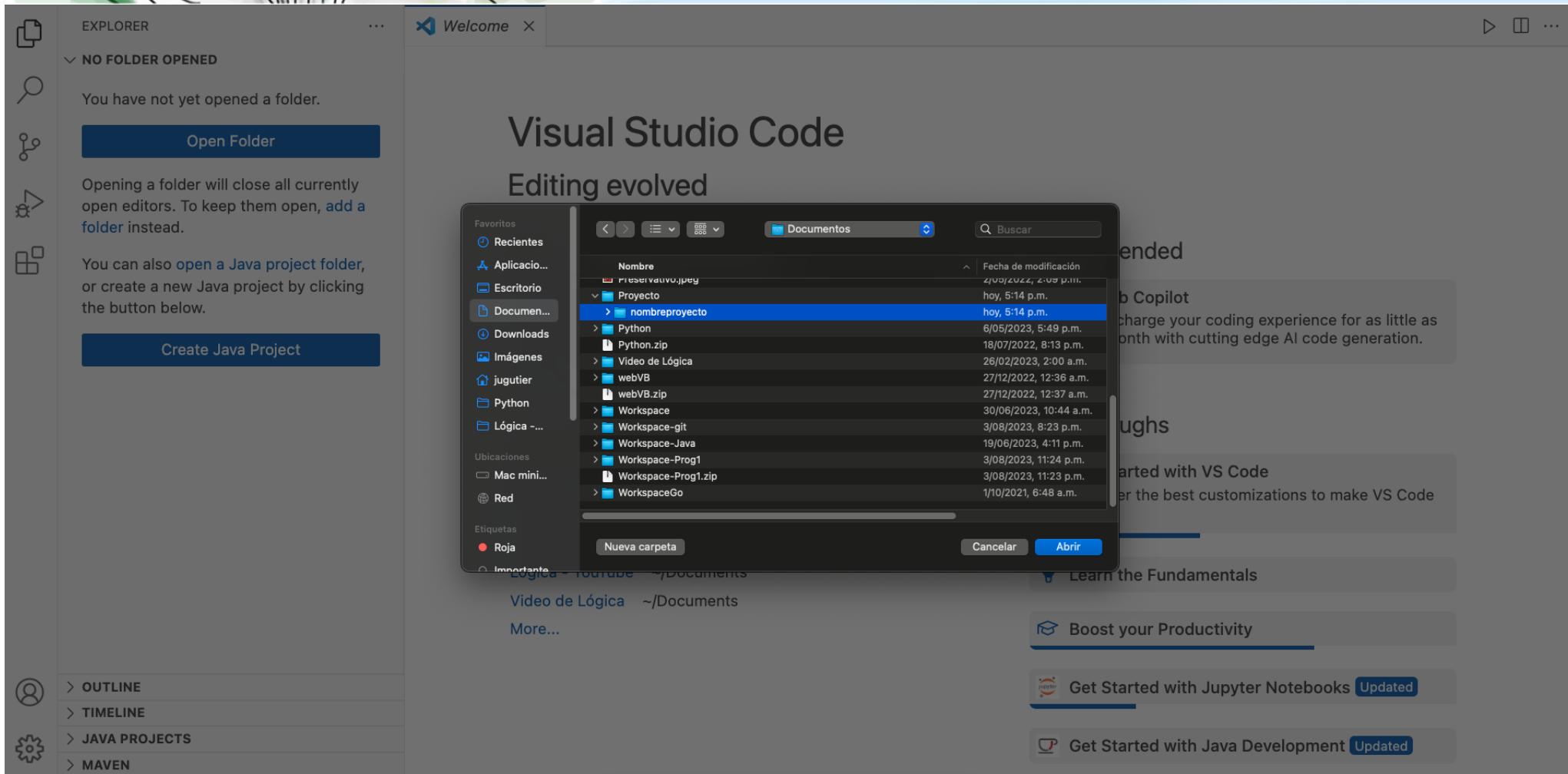


Proyecto creado

```
version: 1.0
package: co.edu.uniquindio.poo
Y: :
[INFO] -----
[INFO] Using following parameters for creating project from Archetype: maven-archetype-quickstart:1.4
[INFO] -----
[INFO] Parameter: groupId, Value: co.edu.uniquindio.poo
[INFO] Parameter: artifactId, Value: nombreproyecto
[INFO] Parameter: version, Value: 1.0
[INFO] Parameter: package, Value: co.edu.uniquindio.poo
[INFO] Parameter: packageInPathFormat, Value: co/edu/uniquindio/poo
[INFO] Parameter: package, Value: co.edu.uniquindio.poo
[INFO] Parameter: groupId, Value: co.edu.uniquindio.poo
[INFO] Parameter: artifactId, Value: nombreproyecto
[INFO] Parameter: version, Value: 1.0
[INFO] Project created from Archetype in dir: /Users/cacandelaurebe/workspaces/202302/nombreproyecto
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 16:27 min
[INFO] Finished at: 2023-08-24T08:18:43-05:00
[INFO] -----
cacandelaurebe@iMac-de-Christian 202302 %
```

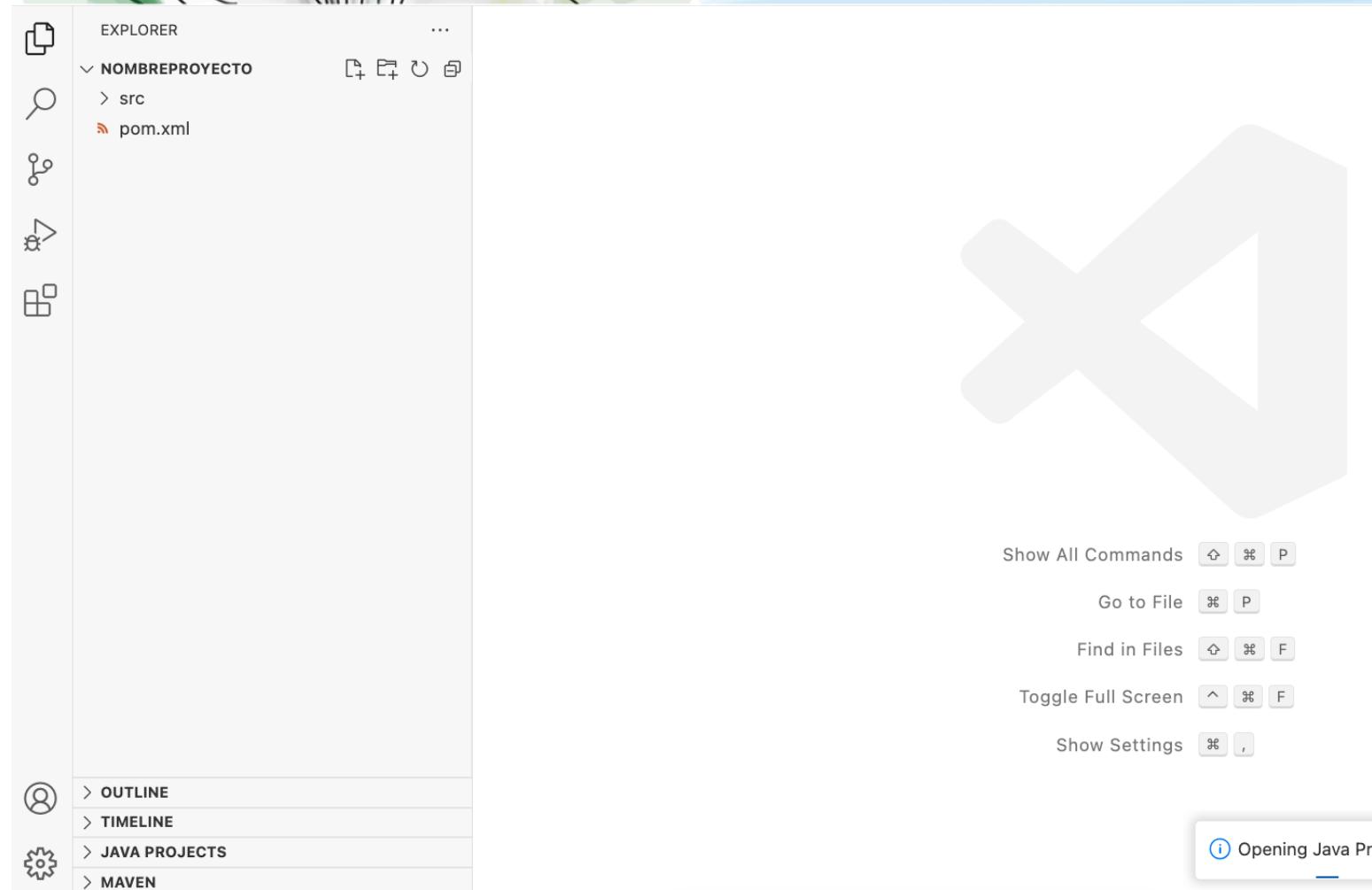
El proyecto ha sido creado, abralo en el Visual Studio Code.

Clic "Open Folder" en la opción "EXPLORER" y seleccione la carpeta del proyecto



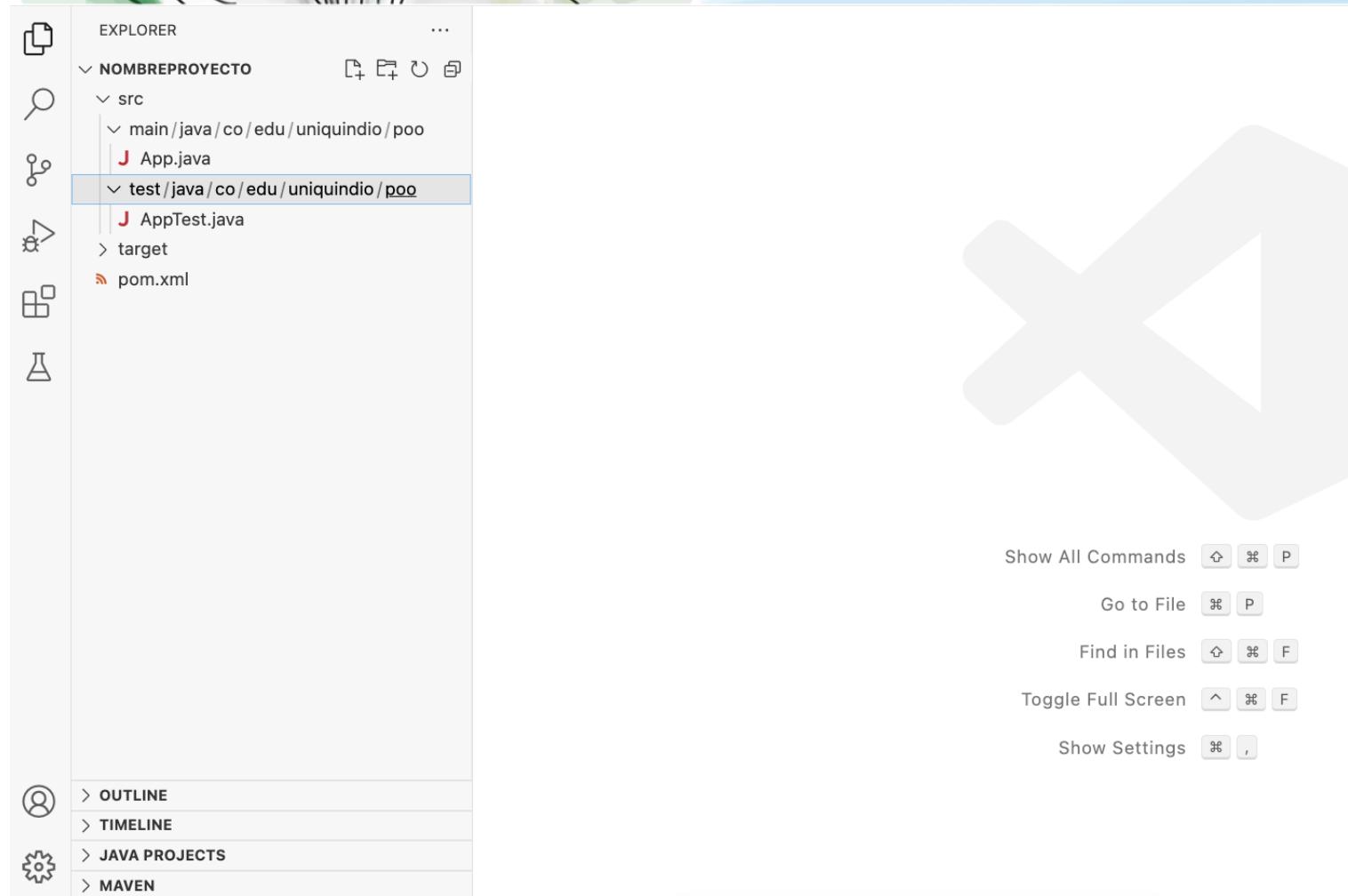


Observe al lado izquierdo (nombre del proyecto, carpeta src y un archivo pom.xml)



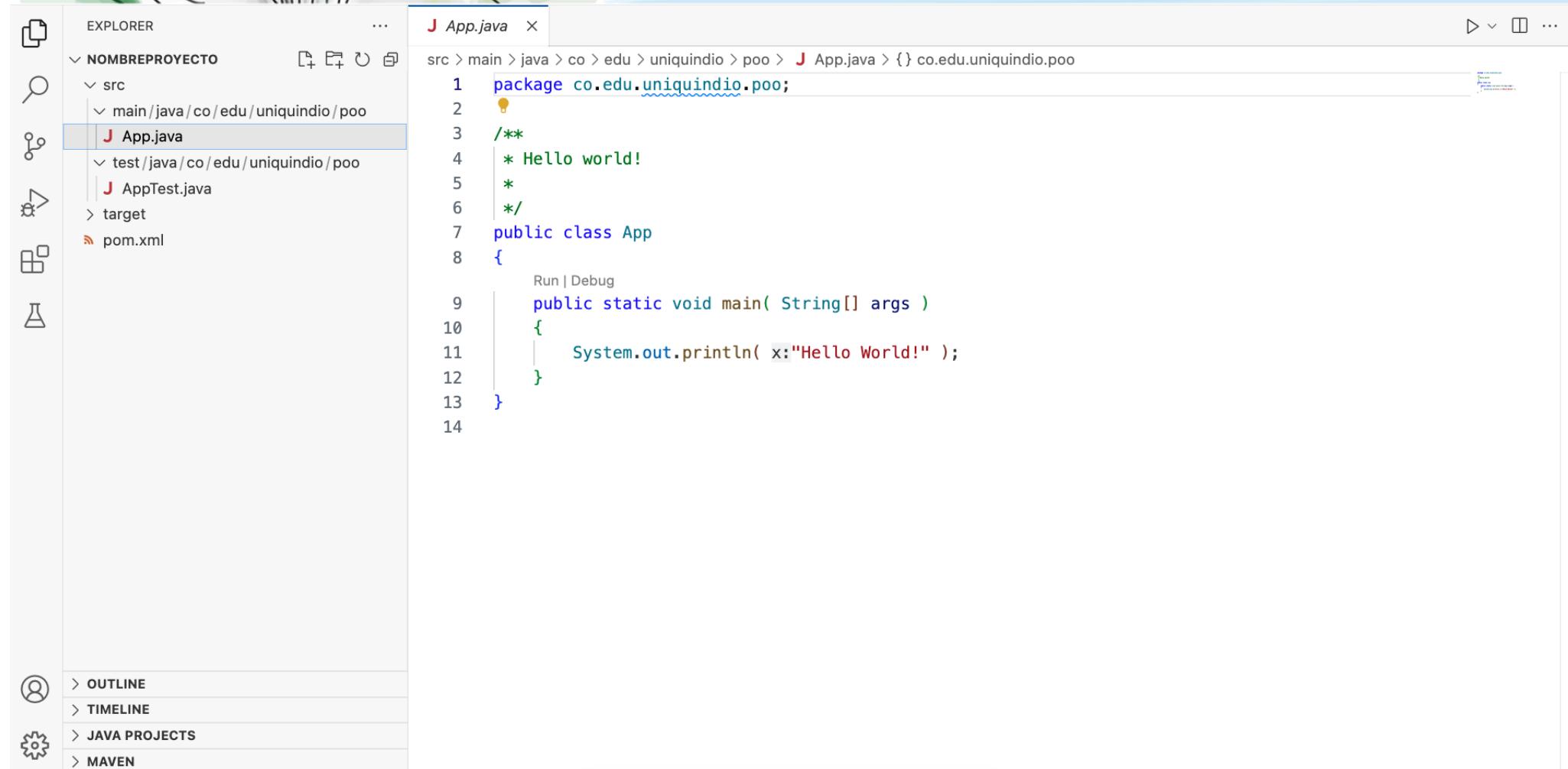


Dentro de la carpeta src está un ejemplo "App.java" y "AppTest.java"





"App.java" es un ejemplo que imprime en pantalla "Hello World!"



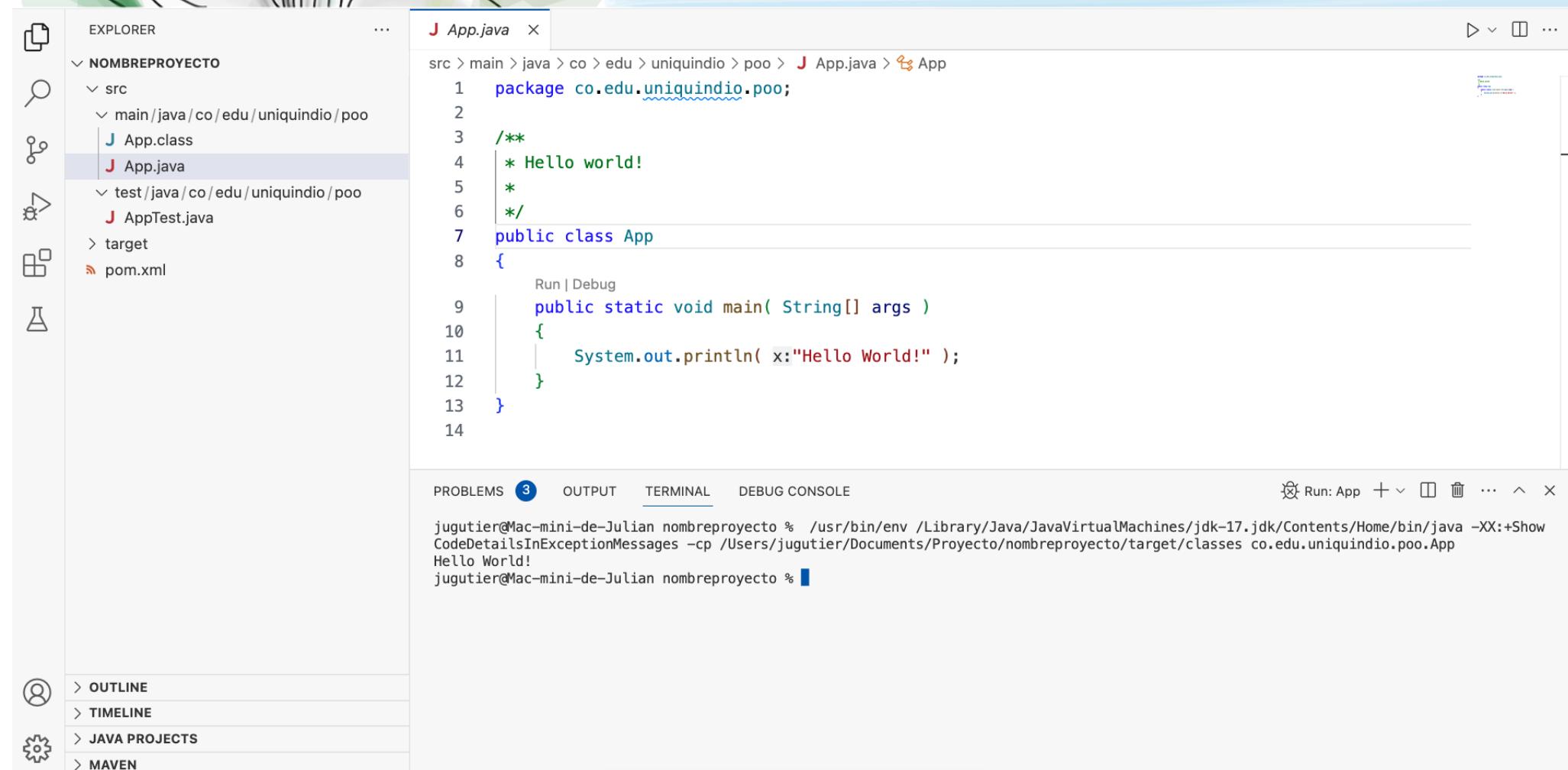
The screenshot shows a Java development environment with the following details:

- EXPLORER** view: Shows the project structure under "NOMBREPROYECTO". The "src" folder contains "main/java/co/edu/uniquindio/poo" which has "App.java" selected. Other files include "AppTest.java" and "pom.xml".
- EDITOR** view: The "App.java" file is open. The code is as follows:

```
1 package co.edu.uniquindio.poo;
2
3 /**
4  * Hello world!
5  *
6  */
7 public class App
8 {
9     public static void main( String[] args )
10    {
11        System.out.println( "Hello World!" );
12    }
13 }
14
```



Si lo desea lo puede ejecutar el ejemplo con el icono en forma triángulo parte superior derecha



The screenshot shows the Visual Studio Code (VS Code) interface. On the left is the Explorer sidebar with a tree view of the project structure:

- NOMBREPROYECTO**
 - src**
 - main/java/co/edu/uniquindio/poo**
 - App.class**
 - App.java** (selected)
 - test/java/co/edu/uniquindio/poo**
 - AppTest.java**
 - target**
 - pom.xml**

The main editor area shows the content of **App.java**:

```
1 package co.edu.uniquindio.poo;
2
3 /**
4  * Hello world!
5  *
6  */
7 public class App
8 {
9     public static void main( String[] args )
10    {
11        System.out.println( "Hello World!" );
12    }
13 }
14
```

The terminal at the bottom shows the output of running the application:

```
jugutier@Mac-mini-de-Julian nombreproyecto % /usr/bin/env /Library/Java/JavaVirtualMachines/jdk-17.jdk/Contents/Home/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /Users/jugutier/Documents/Proyecto/nombreproyecto/target/classes co.edu.uniquindio.poo.App
Hello World!
jugutier@Mac-mini-de-Julian nombreproyecto %
```

"AppTest.java" es un ejemplo de una prueba



The screenshot shows a Java code editor interface with a sidebar and a main workspace.

EXPLORER sidebar:

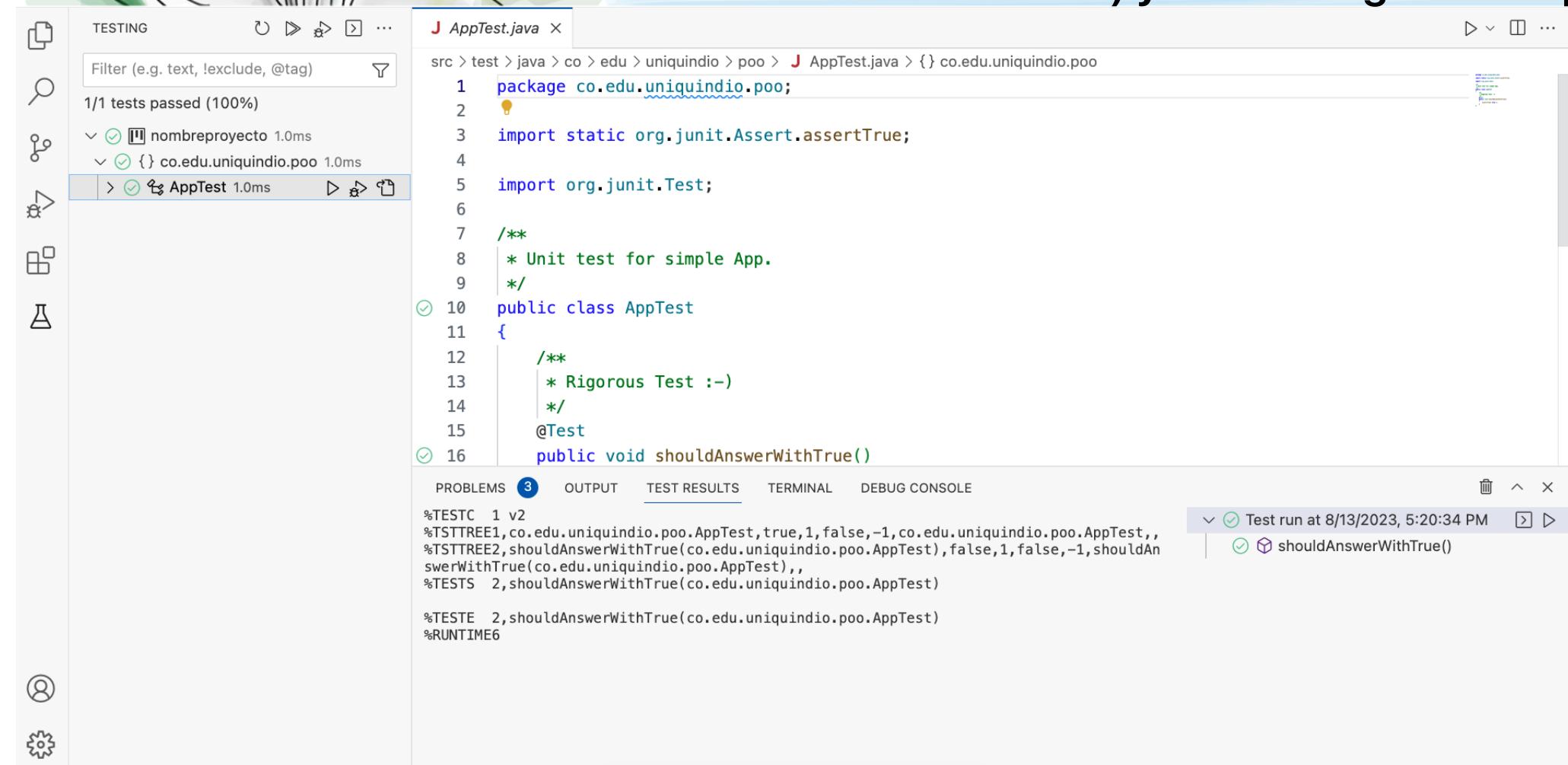
- NOMBREPROYECTO
 - src
 - main/java/co/edu/uniquindio/poo
 - App.class
 - App.java
 - test/java/co/edu/uniquindio/poo
 - AppTest.java
 - target
 - pom.xml

AppTest.java code editor:

```
src > test > java > co > edu > uniquindio > poo > J AppTest.java > {} co.edu.uniquindio.poo
1 package co.edu.uniquindio.poo;
2
3 import static org.junit.Assert.assertTrue;
4
5 import org.junit.Test;
6
7 /**
8 * Unit test for simple App.
9 */
10 public class AppTest
11 {
12     /**
13      * Rigorous Test :-)
14      */
15     @Test
16     public void shouldAnswerWithTrue()
17     {
18         assertTrue( true );
19     }
20 }
21
```



Puede ejecutar la prueba usando el icono en forma de botella de laboratorio (a la izquierda y en el medio) y en el triángulo de la prueba



TESTING Filter (e.g. text, !exclude, @tag)

1/1 tests passed (100%)

- ✓ (I) nombreproyecto 1.0ms
- ✓ (I) {} co.edu.uniquindio.poo 1.0ms

> (I) AppTest 1.0ms

J AppTest.java

```
src > test > java > co > edu > uniquindio > poo > J AppTest.java > {} co.edu.uniquindio.poo
1 package co.edu.uniquindio.poo;
2
3 import static org.junit.Assert.assertTrue;
4
5 import org.junit.Test;
6
7 /**
8 * Unit test for simple App.
9 */
10 public class AppTest
11 {
12     /**
13      * Rigorous Test :-)
14      */
15     @Test
16     public void shouldAnswerWithTrue()
```

PROBLEMS 3 OUTPUT TEST RESULTS TERMINAL DEBUG CONSOLE

```
%TESTC 1 v2
%TSTTREE1,co.edu.uniquindio.poo.AppTest,true,1,false,-1,co.edu.uniquindio.poo.AppTest,
%TSTTREE2,shouldAnswerWithTrue(co.edu.uniquindio.poo.AppTest),false,1,false,-1,shouldAn
swerWithTrue(co.edu.uniquindio.poo.AppTest),
%TESTS 2,shouldAnswerWithTrue(co.edu.uniquindio.poo.AppTest)

%TESTE 2,shouldAnswerWithTrue(co.edu.uniquindio.poo.AppTest)
%RUNTIME6
```

Test run at 8/13/2023, 5:20:34 PM

shouldAnswerWithTrue()

Observe el archivo de configuración del proyecto Maven (pom.xml)



The screenshot shows a Java development environment with the following interface elements:

- EXPLORER** panel on the left showing project structure: NOMBREPROYECTO (src: main/java/co/edu/uniquindio/poo, test/java/co/edu/uniquindio/poo; target; pom.xml).
- pom.xml** editor panel on the right displaying the XML configuration file.
- A floating window in the bottom right corner provides information about the "Language Support for Java(TM) by Red Hat" extension.

```
<?xml version="1.0" encoding="UTF-8"?>
<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>co.edu.uniquindio.poo</groupId>
  <artifactId>nombrefreproyecto</artifactId>
  <version>1.0</version>
  <name>nombrefreproyecto</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>
  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>
  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.11</version>
      <scope>test</scope>
    </dependency>
  </dependencies>

```

The floating window contains the following message:

"Language Support for Java(TM) by Red Hat" recommends you install "Dependency Analytics": "The Dependency Analytics extension helps you to stay informed about vulnerable dependencies in pom.xml files."

Source: Language Support for Java(TM) ... [Install](#) [Never](#) [Later](#)



Actualizando el archivo de configuración **pom.xml**

- Número de la versión a de Java que se utilizará, para nuestro caso, 17.
- Cambiar la versión del componente de pruebas a **JUnit Jupiter**.
 - Actualizar el ejemplo de la prueba a la nueva versión
- Elimine los *plugins* que no usaremos, dejando solamente:
 - *maven-compiler-plugin* (utilizado para compilar el proyecto) v. 3.11.0
<https://mvnrepository.com/artifact/org.apache.maven.plugins/maven-compiler-plugin>
 - *maven-surefire-plugin* (utilizado para ejecutar las pruebas) v. 3.1.2
<https://mvnrepository.com/artifact/org.apache.maven.plugins/maven-surefire-plugin>

Actualice el número de la versión de Java a utilizar, para nosotros será 17

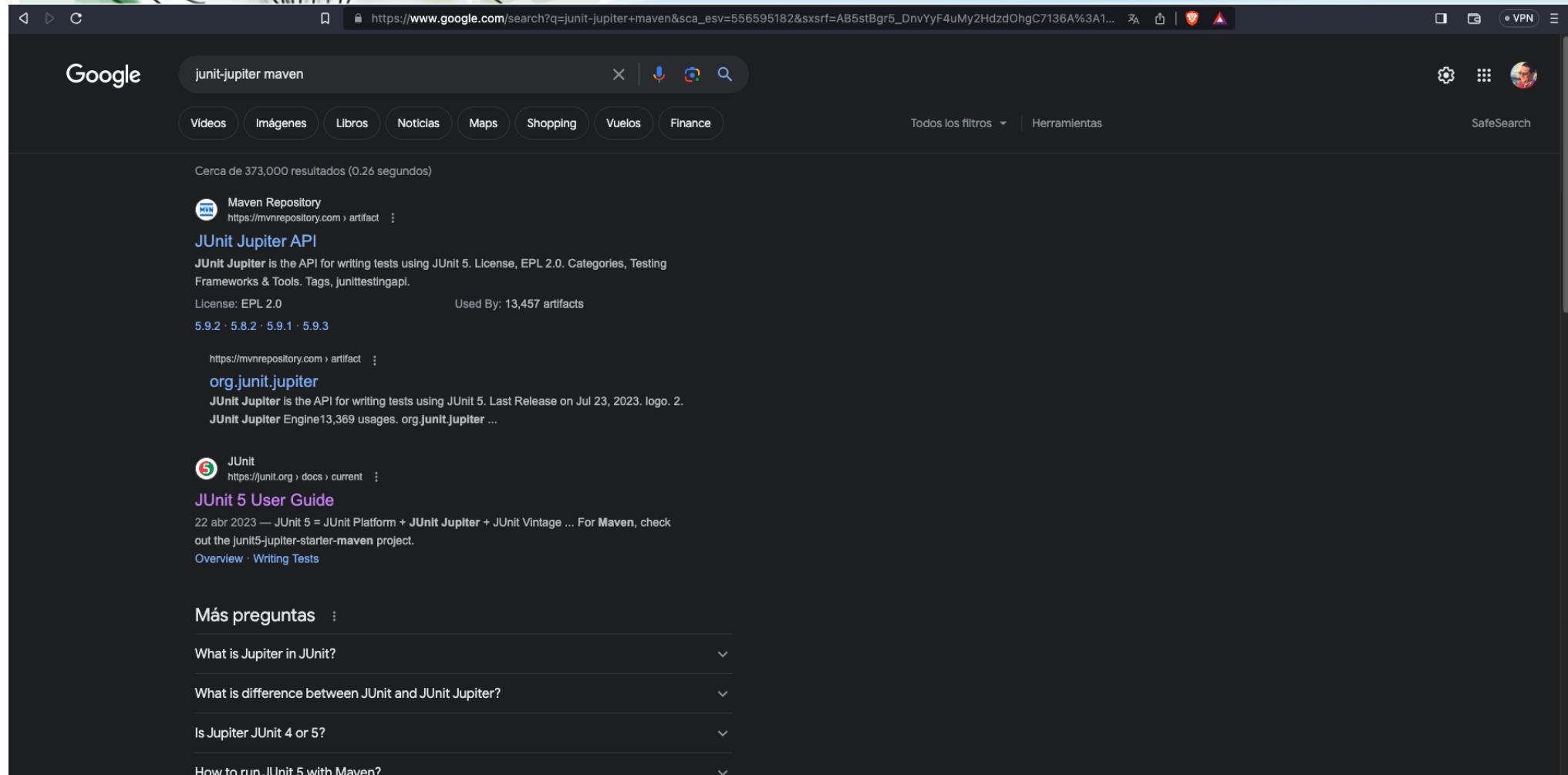


The screenshot shows the VS Code interface with a decorative floral header. The left sidebar contains icons for Explorer, Search, Source Control, Problems, Tasks, and Maven. The Explorer view shows a project structure with a .vscode folder, a src folder containing main/java and test/java subfolders, and target and pom.xml files. The pom.xml file is selected in the Explorer. The main editor area shows the content of the pom.xml file:

```
<?xml version="1.0" encoding="UTF-8"?>
<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>co.edu.uniquindio.poo</groupId>
  <artifactId>nombreproyecto</artifactId>
  <version>1.0</version>
  <name>nombreproyecto</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>
  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>17</maven.compiler.source>
    <maven.compiler.target>17</maven.compiler.target>
  </properties>
  <dependencies>
    <!-- https://mvnrepository.com/artifact/org.junit.jupiter/junit-jupiter-api -->
    <dependency>
      <groupId>org.junit.jupiter</groupId>
      <artifactId>junit-jupiter-api</artifactId>
      <version>5.10.0</version>
    
```



Busque en Internet "junit-jupiter maven"



Cerca de 373,000 resultados (0.26 segundos)

Maven Repository
<https://mvnrepository.com/artifact> :

JUnit Jupiter API
JUnit Jupiter is the API for writing tests using JUnit 5. License, EPL 2.0. Categories, Testing Frameworks & Tools. Tags, junittestingapi.

License: EPL 2.0 Used By: 13,457 artifacts
5.9.2 · 5.8.2 · 5.9.1 · 5.9.3
<https://mvnrepository.com/artifact> :

org.junit.jupiter
JUnit Jupiter is the API for writing tests using JUnit 5. Last Release on Jul 23, 2023. logo. 2.
JUnit Jupiter Engine13,369 usages. org.junit.jupiter ...

JUnit
<https://junit.org/docs/current> :

JUnit 5 User Guide
22 abr 2023 — JUnit 5 = JUnit Platform + JUnit Jupiter + JUnit Vintage ... For Maven, check out the [junit5-jupiter-starter-maven](#) project.
[Overview](#) · [Writing Tests](#)

Más preguntas :

What is Jupiter in JUnit?
What is difference between JUnit and JUnit Jupiter?
Is Jupiter JUnit 4 or 5?
How to run JUnit 5 with Maven?

Clic en la última versión, para el caso 5.10.0



MVN REPOSITORY

Indexed Artifacts (34.9M)

Search for groups, artifacts, categories

Home » org.junit.jupiter » junit-jupiter-api

JUnit Jupiter API

JUnit Jupiter is the API for writing tests using JUnit 5.

License: EPL 2.0

Categories: Testing Frameworks & Tools

Tags: junit, testing, api

Ranking: #38 in MvnRepository (See Top Artifacts)
#3 in Testing Frameworks & Tools

Used By: 13,459 artifacts

Central (64) Redhat GA (1) ICM (2)

Version	Vulnerabilities	Repository	Usages	Date
5.10.0		Central	343	Jul 23, 2023
5.10.0-RC2		Central	10	Jul 19, 2023
5.10.0-RC1		Central	14	Jul 06, 2023
5.10.0-M1		Central	94	May 13, 2023
5.9.3		Central	1,354	Apr 26, 2023
5.9.2		Central	1,809	Jan 10, 2023
5.9.1		Central	1,782	Sep 20, 2022
5.9.0		Central	1,163	Jul 26, 2022
5.9.0-RC1		Central	79	Jul 04, 2022
5.9.0-M1		Central	108	May 15, 2022
5.8.2		Central	2,608	Nov 28, 2021
5.8.1		Central	1,179	Sep 22, 2021
5.8.0		Central	298	Sep 12, 2021
5.8.0-RC1		Central	35	Aug 17, 2021
5.8.0-M1		Central	286	Feb 11, 2021

Indexed Repositories (1921)

- Central
- Atlassian
- Sonatype
- Hortonworks
- Spring Plugins
- Spring Lib M
- JCenter
- JBossEA
- Atlassian Public
- KtorEAP

Popular Tags

aar amazon android apache api application arm assets atlassian aws build build-system client clojure cloud config cran data database eclipse example extension github gradle groovy http io jboss kotlin library logging maven module npm persistence platform plugin rest rlang sdk security server service spring starter testing tools ui web webapp

About

Web site developed by @frodriguez

Powered by: Scala, Play, Spark, Akka and Cassandra

Seleccione y copie las líneas para la configuración Maven



MVN REPOSITORY

Indexed Artifacts (34.9M)

Search for groups, artifacts, categories

Home » org.junit.jupiter » junit-jupiter-api » 5.10.0

JUnit Jupiter API » 5.10.0

JUnit Jupiter is the API for writing tests using JUnit 5.

License: EPL 2.0

Categories: Testing Frameworks & Tools

Tags: junit, testing, api

HomePage: <https://junit.org/junit5/>

Date: Jul 23, 2023

Files: pom (3 KB) jar (206 KB) View All

Repositories: Central

Ranking: #38 in MvnRepository (See Top Artifacts)
#3 in Testing Frameworks & Tools

Used By: 13,459 artifacts

Maven Gradle Gradle (Short) Gradle (Kotlin) SBT Ivy Grape Leiningen Buildr

```
<!-- https://mvnrepository.com/artifact/org.junit.jupiter/junit-jupiter-api -->
<dependency>
    <groupId>org.junit.jupiter</groupId>
    <artifactId>junit-jupiter-api</artifactId>
    <version>5.10.0</version>
    <scope>test</scope>
</dependency>
```

Include comment with link to declaration

Copied to clipboard!

Compile Dependencies (3)

Category/License	Group / Artifact	Version	Updates
Annotation Lib Apache 2.0	org.apiguardian » apiguardian-api	1.1.2	✓

Indexed Repositories (1921)

- Central
- Atlassian
- Sonatype
- Hortonworks
- Spring Plugins
- Spring Lib M
- JCenter
- JBossEA
- Atlassian Public
- KtorEAP

Popular Tags

- aar
- amazon
- android
- apache
- api
- application
- arm
- assets
- atlassian
- aws
- build
- build-system
- client
- clojure
- cloud
- config
- cran
- data
- database
- eclipse
- example
- extension
- github
- gradle
- groovy
- http
- io
- jboss
- kotlin
- library
- logging
- maven
- module
- npm
- persistence
- platform
- plugin
- rest
- rlang
- sdk
- security
- server
- service
- spring
- starter
- testing
- tools
- ui
- web
- webapp

About

Web site developed by @frodriguez

Powered by: Scala, Play, Spark, Akka and Cassandra

Elimine del archivo pom.xml la dependencia junit 4.11



The image shows a screenshot of a Microsoft Visual Studio Code (VS Code) interface. The left sidebar is the Explorer view, showing a project structure named 'NOMBREPROYECTO' containing '.vscode', 'src' (with 'main/java/co/edu/uniquindio/poo' and 'test/java/co/edu/uniquindio/poo' subfolders), and 'target'. A file named 'pom.xml' is selected in the Explorer. The main editor area shows the XML content of 'pom.xml'. A specific dependency block for 'junit' version 4.11 is highlighted with a blue selection rectangle. The code is as follows:

```
<name>nombrefreproyecto</name>
<!-- FIXME change it to the project's website -->
<url>http://www.example.com</url>

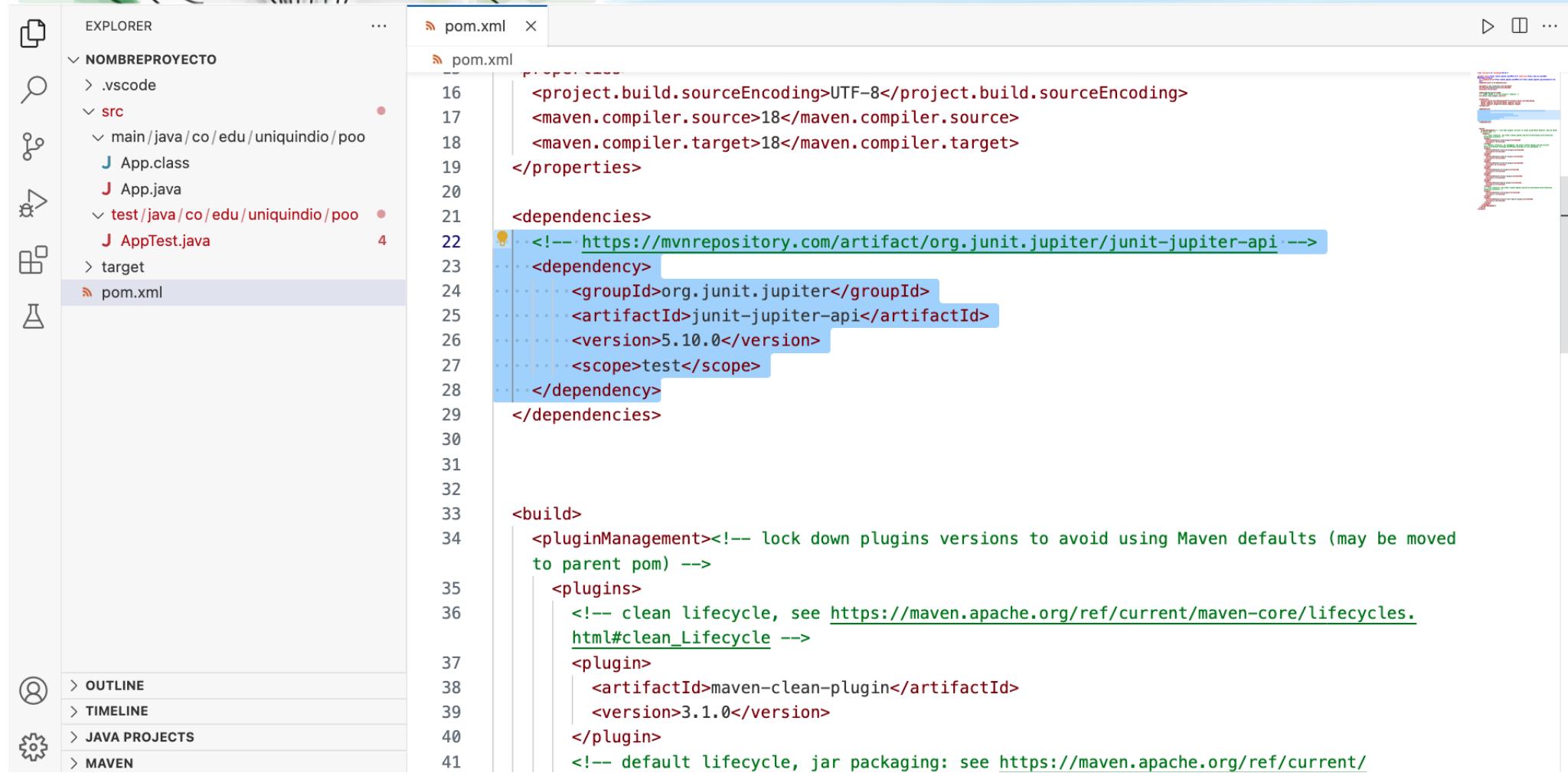
<properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>18</maven.compiler.source>
    <maven.compiler.target>18</maven.compiler.target>
</properties>

<dependencies>
    <dependency>
        <groupId>junit</groupId>
        <artifactId>junit</artifactId>
        <version>4.11</version>
        <scope>test</scope>
    </dependency>
</dependencies>

<build>
    <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->
        <plugins>
            <!-- clean lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#clean\_Lifecycle -->
            <plugin>
```



Pegue en el mismo lugar del archivo pom.xml
las líneas que copió de Internet junit-jupiter

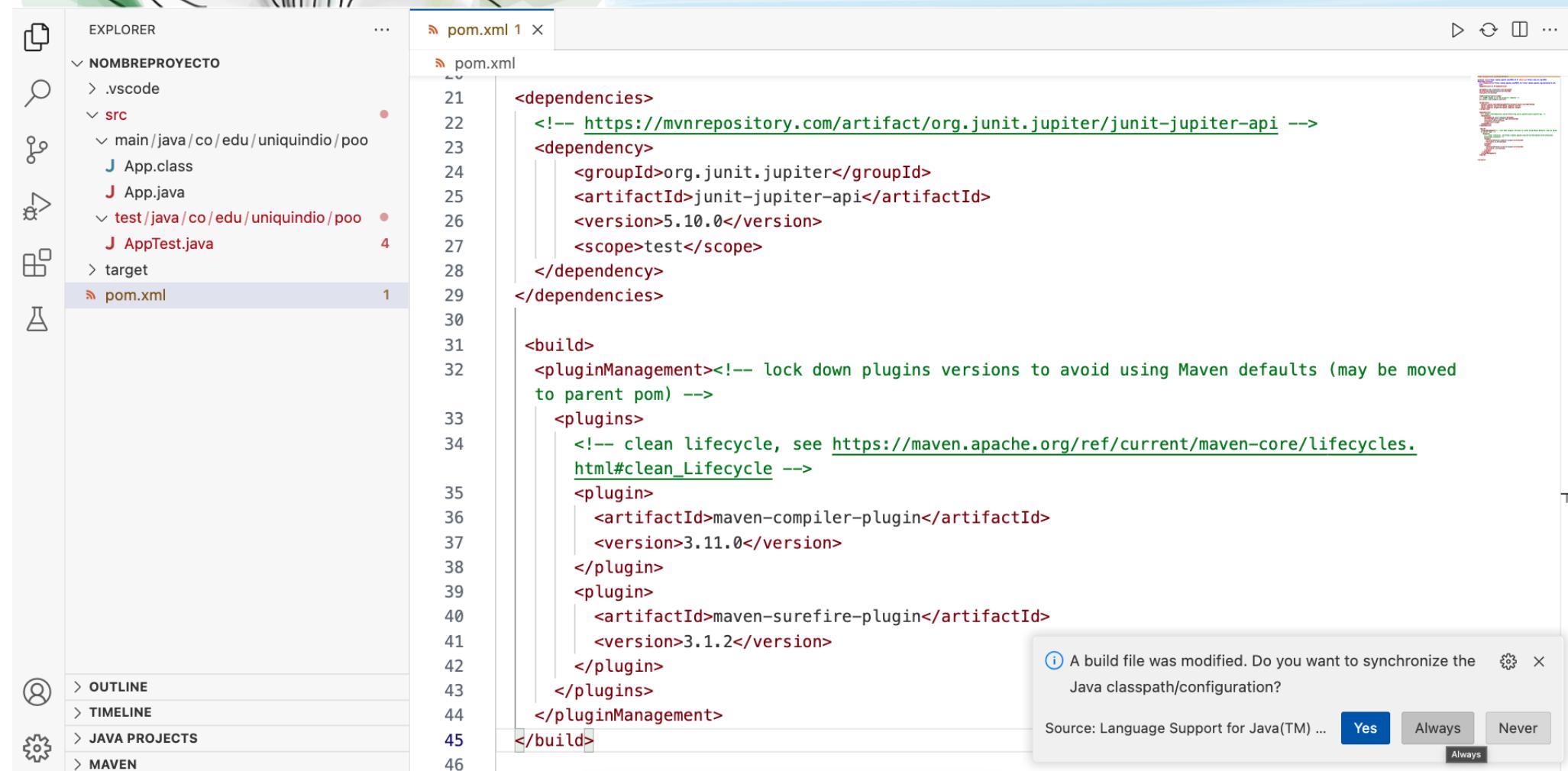


```
<!-- https://mvnrepository.com/artifact/org.junit.jupiter/junit-jupiter-api -->
<dependency>
    <groupId>org.junit.jupiter</groupId>
    <artifactId>junit-jupiter-api</artifactId>
    <version>5.10.0</version>
    <scope>test</scope>
</dependency>
```

The image shows a screenshot of the Visual Studio Code (VS Code) interface. On the left is the Explorer sidebar, which lists the project structure: .vscode, src (containing main/java/co/edu/uniquindio/poo with App.class and App.java), test/java/co/edu/uniquindio/poo with AppTest.java, target, and pom.xml. The pom.xml file is selected in the Explorer. The main area is the Editor, showing the XML code for a Maven project's pom.xml file. The code includes sections for properties, dependencies, build, and pluginManagement. A specific dependency for 'junit-jupiter-api' is highlighted with a blue selection bar, indicating it has been copied from the internet. The code is as follows:

```
<!-- https://mvnrepository.com/artifact/org.junit.jupiter/junit-jupiter-api -->
<dependency>
    <groupId>org.junit.jupiter</groupId>
    <artifactId>junit-jupiter-api</artifactId>
    <version>5.10.0</version>
    <scope>test</scope>
</dependency>
```

Seleccione "Always" para sincronizar los cambios en el proyecto



Deje los plugins que utilizaremos y actualice la versión según la imagen



The image shows a screenshot of the Visual Studio Code (VS Code) interface. The left sidebar displays a project structure for a Java application named 'NOMBREPROYECTO'. The 'src' folder contains 'main/java/co/edu/uniquindio/poo' with files 'App.class', 'App.java', and 'test/java/co/edu/uniquindio/poo' with file 'AppTest.java'. The 'pom.xml' file is selected in the Explorer view. The main editor window shows the contents of the 'pom.xml' file, which defines Maven dependencies and build plugins. The 'maven-compiler-plugin' and 'maven-surefire-plugin' are highlighted with blue selection bars. The bottom right corner of the editor shows a preview of the code in a browser. The bottom navigation bar includes tabs for PROBLEMS, OUTPUT, TEST RESULTS, TERMINAL (which is active), and DEBUG CONSOLE. The terminal shows the command 'jugutier@Mac-mini-de-Julian nombreproyecto %'. The bottom status bar includes icons for zsh, terminal control, and file operations.

```
<dependency>
    <scope>runtime</scope>
</dependency>
</dependencies>

<build>
<pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->
<plugins>
    <!-- clean lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#clean\_Lifecycle -->
<plugin>
    <artifactId>maven-compiler-plugin</artifactId>
    <version>3.11.0</version>
</plugin>
<plugin>
    <artifactId>maven-surefire-plugin</artifactId>
    <version>3.1.2</version>
</plugin>
</plugins>
</pluginManagement>
</build>
fi
</project>
```

Ajuste el archivo de la prueba para nueva versión



The image shows a screenshot of a Java development environment in Visual Studio Code. The Explorer sidebar on the left shows a project structure with files like .vscode, pom.xml, App.class, App.java, and AppTest.java. The AppTest.java file is open in the main editor, displaying the following code:

```
src > test > java > co > edu > unquindio > poo > J AppTest.java > ...
1 package co.edu.unquindio.poo;
2
3 import static org.junit.jupiter.api.Assertions.assertTrue;
4
5 import org.junit.jupiter.api.Test;
6
7 /**
8 * Unit test for simple App.
9 */
10 public class AppTest
11 {
12     /**
13      * Rigorous Test :-)
14      */
15     @Test
16     public void shouldAnswerWithTrue()
17     {
18         assertTrue( true );
19     }
20 }
21
```

Ejecute nuevamente la prueba



The screenshot shows a Java test file named `J AppTest.java` in an IDE. The code is a JUnit test for a class named `AppTest`. The test method is `shouldAnswerWithTrue()`. The test passes, as indicated by the green checkmark in the test results tree on the left.

```
src > test > java > co > edu > uniquindio > poo > J AppTest.java > AppTest
1 package co.edu.uniquindio.poo;
2
3 import static org.junit.jupiter.api.Assertions.assertEquals;
4
5 import org.junit.jupiter.api.Test;
6
7 /**
8 * Unit test for simple App.
9 */
10 public class AppTest
11 {
12     /**
13      * Rigorous Test :-)
14      */
15     @Test
16     public void shouldAnswerWithTrue()
```

The IDE interface includes a sidebar with icons for file, search, and other tools. Below the editor are tabs for PROBLEMS, OUTPUT, TEST RESULTS (which is selected), TERMINAL, and DEBUG CONSOLE. A status bar at the bottom shows test counts and execution details.

TESTC 1 v2
%TSTTREE2,co.edu.uniquindio.poo.AppTest,true,1,false,1,AppTest,,[engine:junit-jupiter]/
[class:co.edu.uniquindio.poo.AppTest]
%TSTTREE3,shouldAnswerWithTrue(co.edu.uniquindio.poo.AppTest),false,1,false,2,shouldAns
werWithTrue(),,[engine:junit-jupiter]/[class:co.edu.uniquindio.poo.AppTest]/[method:sho

Modifique la prueba con este ejemplo

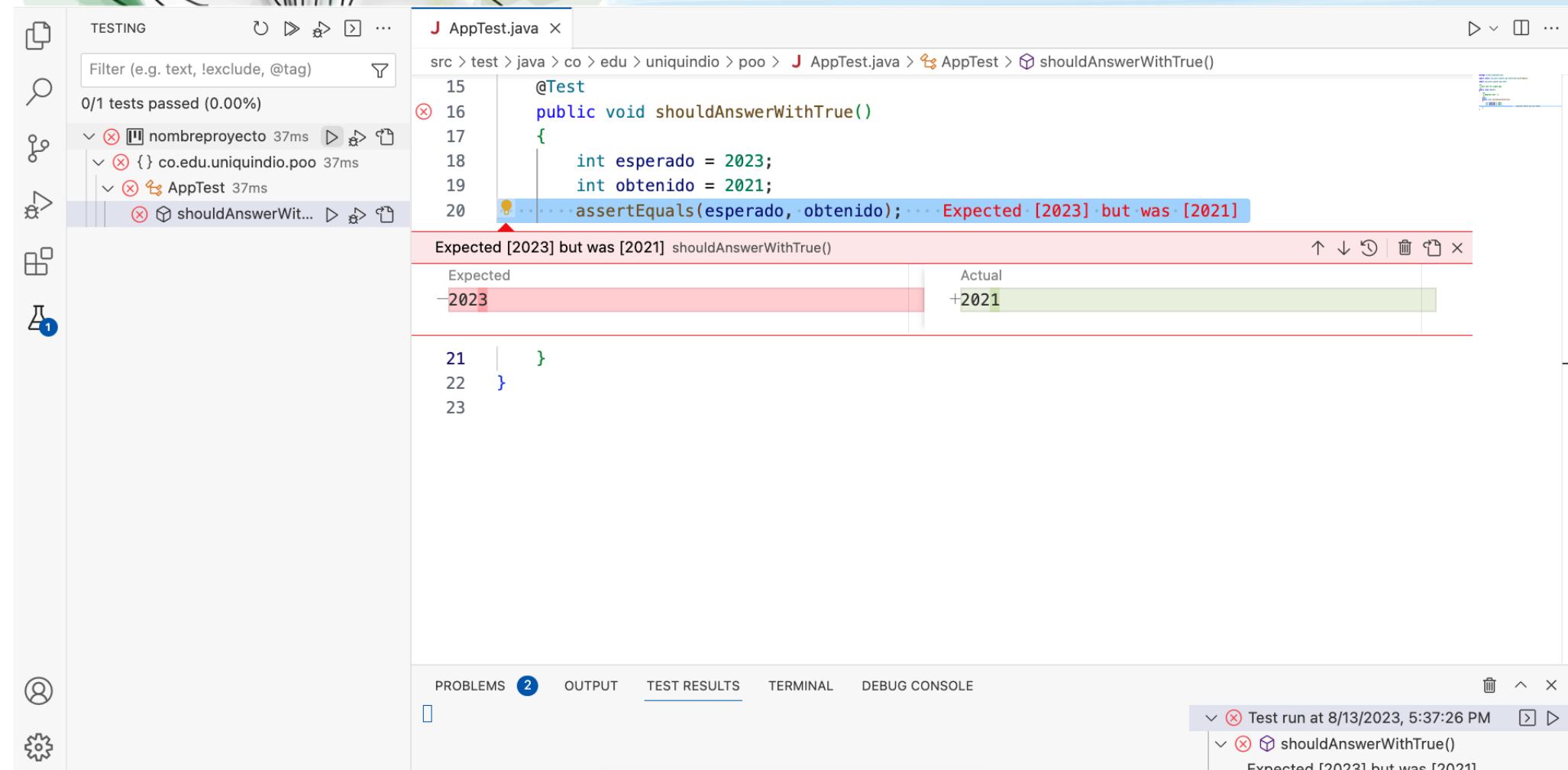


The screenshot shows a Java code editor in an IDE, specifically IntelliJ IDEA, displaying a unit test named `AppTest.java`. The code uses JUnit Jupiter assertions to check if a value is equal to 2023.

```
TESTING      ⏪ ⏴ ⏵ ⏶ ⏷ ...  
Filter (e.g. text, !exclude, @tag)   
1/1 tests passed (100%)  
  ✓ nombrereproyecto 16ms  
    ✓ {} co.edu.uniquindio.poo 16ms  
      ✓ AppTest 16ms  
        ✓ shouldAnswerWithTrue() 16ms  
        ⏪ ⏵ ⏶ ⏷ ...  
  
J AppTest.java x ⏪ ⏴ ⏵ ⏶ ⏷ ...  
src > test > java > co > edu > uniquindio > poo > J AppTest.java > AppTest  
1 package co.edu.uniquindio.poo;  
2  
3 import static org.junit.jupiter.api.Assertions.assertEquals;  
4  
5 import org.junit.jupiter.api.Test;  
6  
7 /**  
8  * Unit test for simple App.  
9 */  
10 public class AppTest  
11 {  
12     /**  
13      * Rigorous Test :-)  
14      */  
15     @Test  
16     public void shouldAnswerWithTrue()  
17     {  
18         int esperado = 2023;  
19         int obtenido = 2023;  
20         assertEquals(esperado, obtenido);  
21     }  
22 }  
23  
PROBLEMS 2 OUTPUT TEST RESULTS TERMINAL DEBUG CONSOLE ⏪ ⏴ ⏵ ⏶ ⏷ ...  
  ✓ Test run at 8/13/2023, 5:37:07 PM ⏪ ⏵ ⏶ ⏷ ...  
  ✓ shouldAnswerWithTrue()
```



Finalmente haga que la prueba falle usando un valor diferente para la variable obtenido



The screenshot shows the Visual Studio Code interface with the following details:

- TESTING View:** Shows 0/1 tests passed (0.00%).
- AppTest.java:** A Java test class with one test method, `shouldAnswerWithTrue()`.

```
15  @Test
16  public void shouldAnswerWithTrue()
17  {
18      int esperado = 2023;
19      int obtenido = 2021;
20      assertEquals(esperado, obtenido); ... Expected [2023] but was [2021]
```
- Test Results:** A modal window displays the failure message: "Expected [2023] but was [2021] shouldAnswerWithTrue()". It shows the **Expected** value as 2023 in a red bar and the **Actual** value as +2021 in a green bar.
- Bottom Status Bar:** Shows tabs for PROBLEMS, OUTPUT, TEST RESULTS (selected), TERMINAL, and DEBUG CONSOLE. The TEST RESULTS tab has a count of 2.
- Bottom Right:** A sidebar shows the test run details: "Test run at 8/13/2023, 5:37:26 PM" and the failed test: "shouldAnswerWithTrue()". The message "Expected [2023] but was [2021]" is also present.

