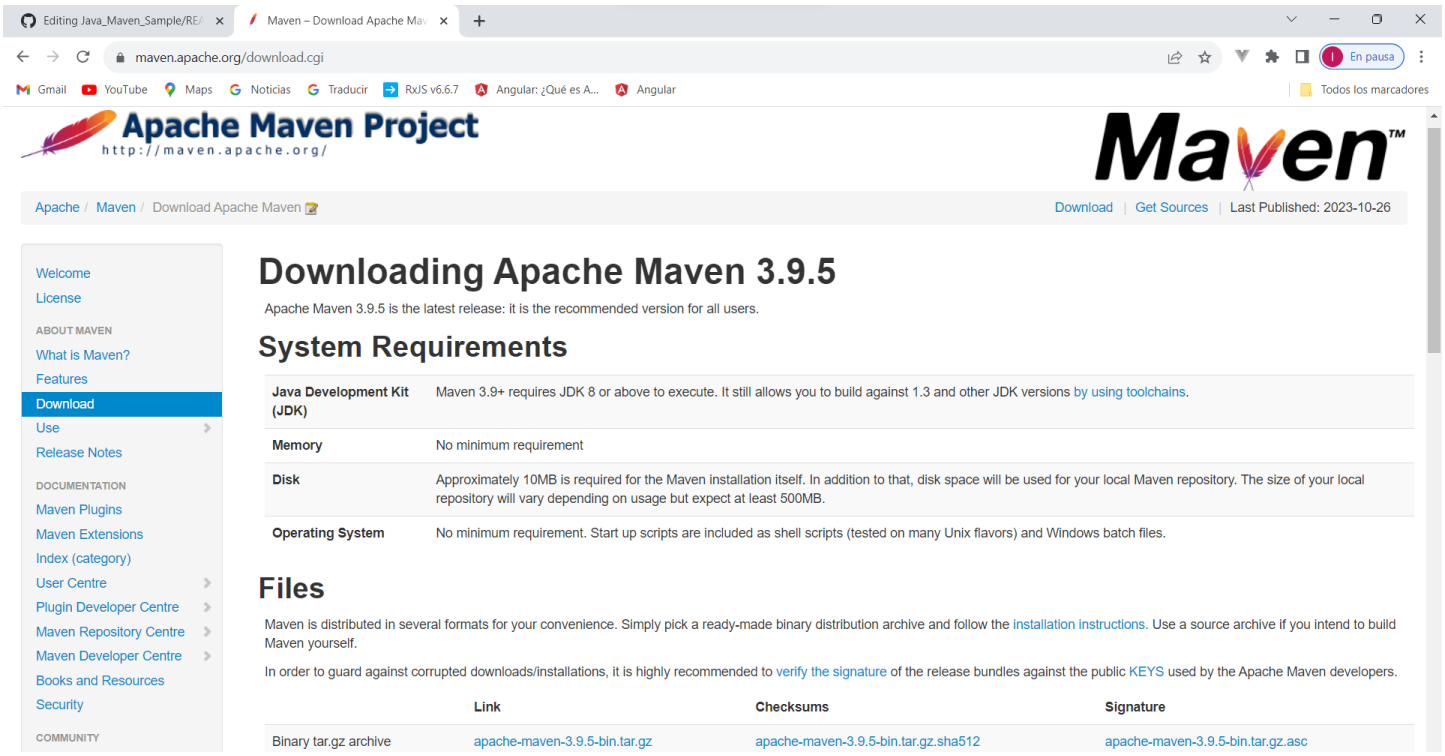
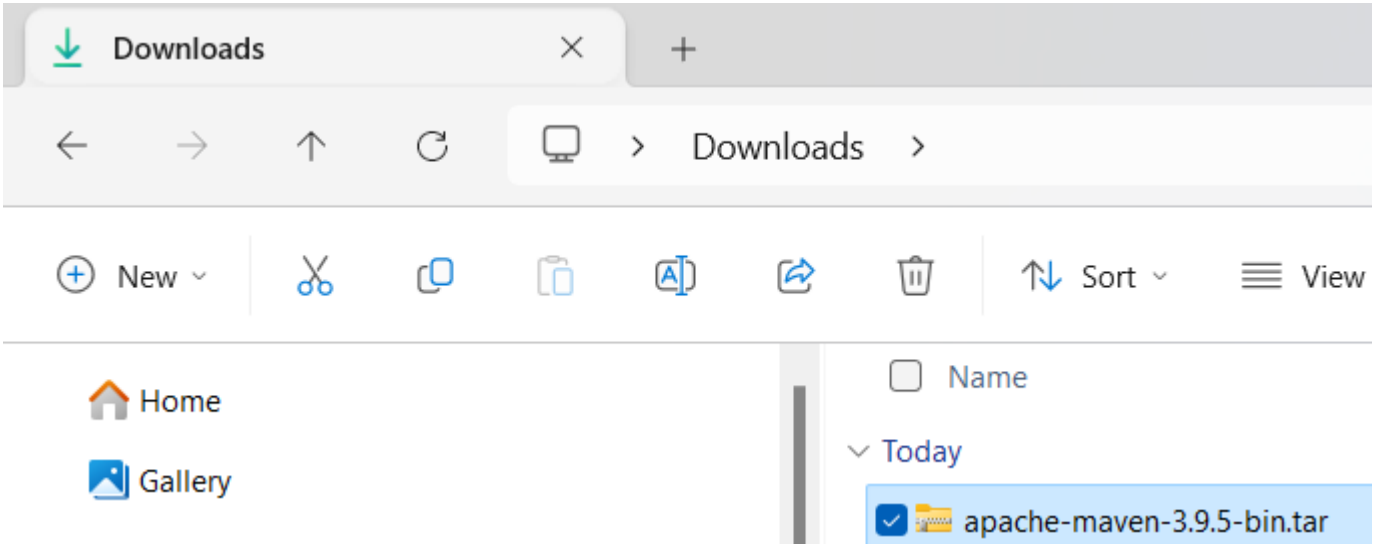
Java Maven Sample

1. Download and install Apache Maven

Download Apache Maven from the web page: https://maven.apache.org/download.cgi

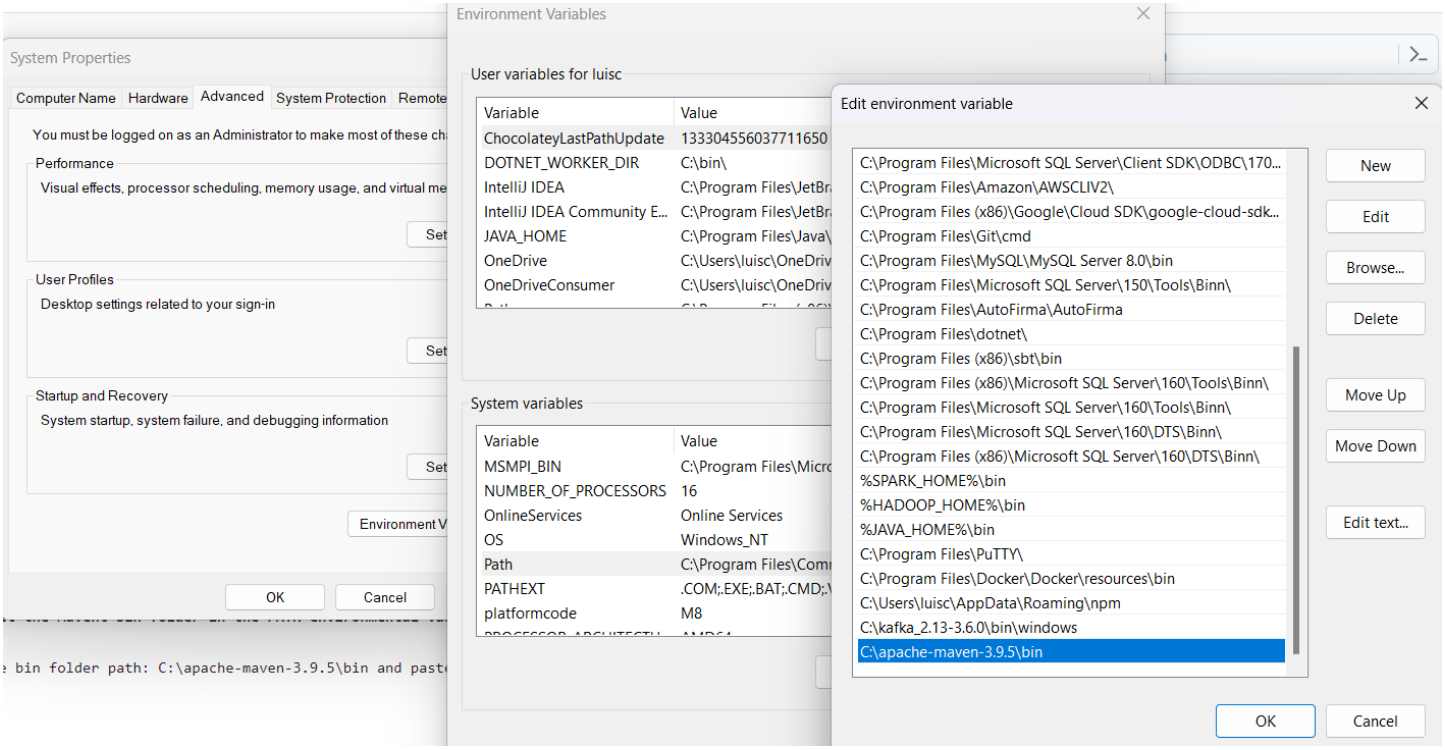


Unzig the **apache-maven-3.9.5-bin.tar** file and place the folder in the C: root hard disk



2. Set the Mavent bin folder in the PATH environmental variable

Copy the bin folder path: **C:\apache-maven-3.9.5\bin** and paste in the **PATH** environmental variable



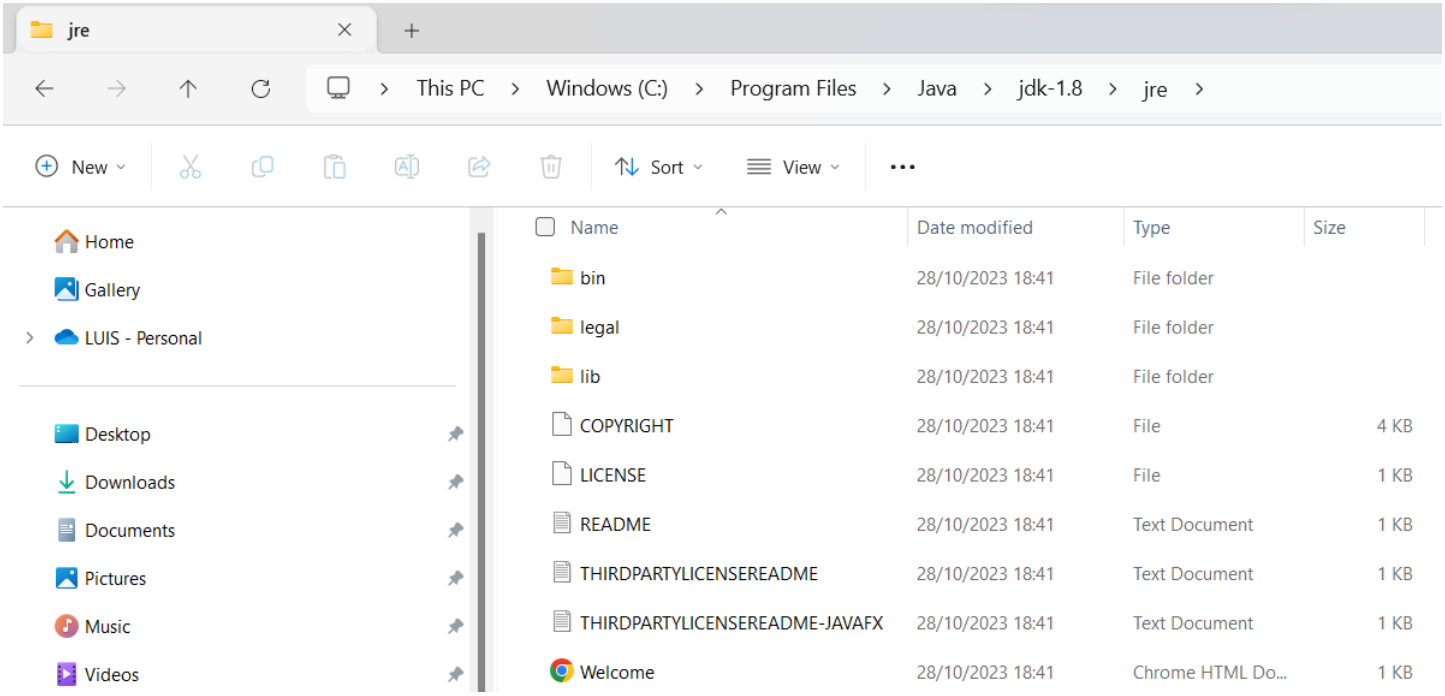
3. Verify Maven installation

Run this command to verify the Maven installation

```
mvn -v
```

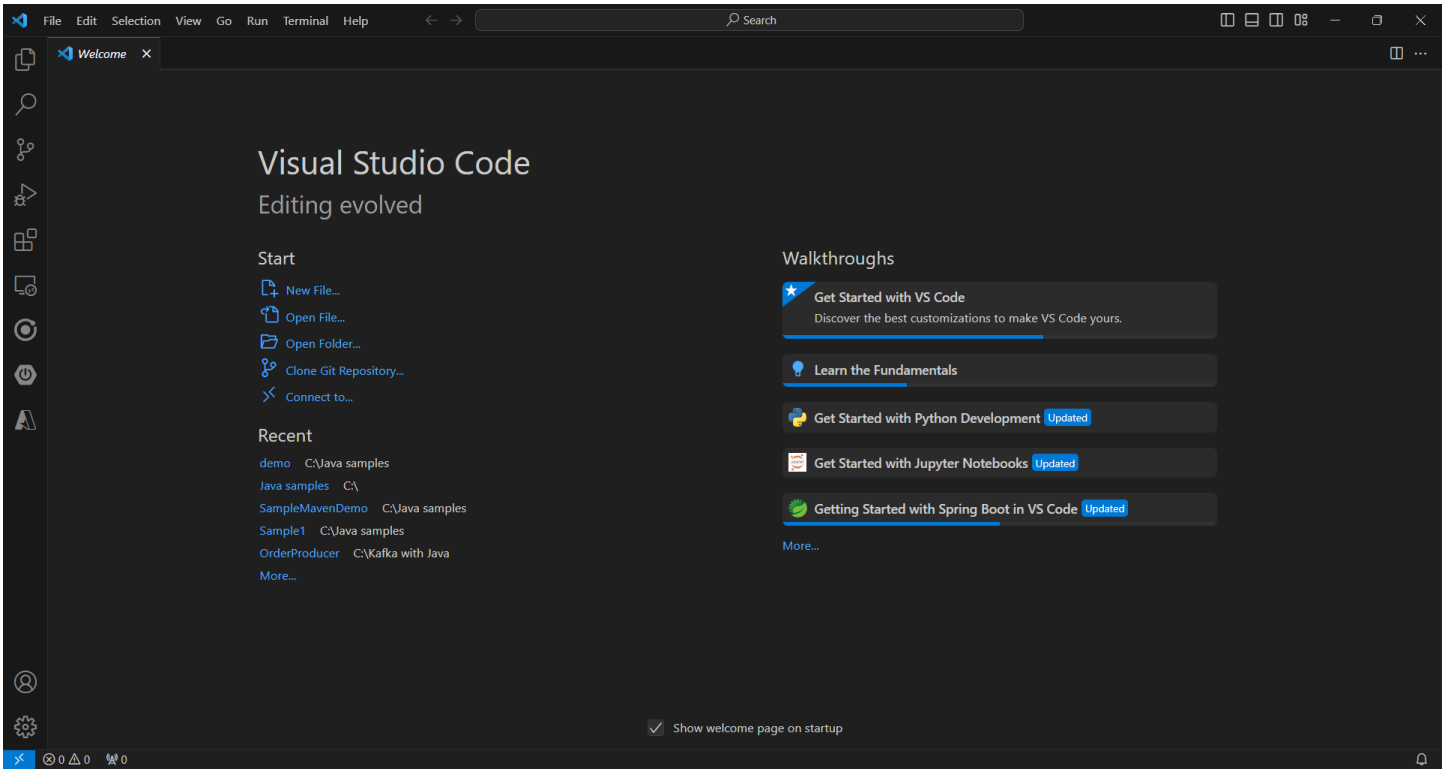
```
C:\Users\luisc>mvn -v
Apache Maven 3.9.5 (57804ffe001d7215b5e7bcb531cf83df38f93546)
Maven home: C:\apache-maven-3.9.5
Java version: 1.8.0_391, vendor: Oracle Corporation, runtime: C:\Program Files\Java\jdk-1.8\jre
Default locale: en_US, platform encoding: Cp1252
OS name: "windows 11", version: "10.0", arch: "amd64", family: "windows"
```

IMPORTANT NOTE: We already installed before "C:\Program Files\Java\jdk-1.8\jre"

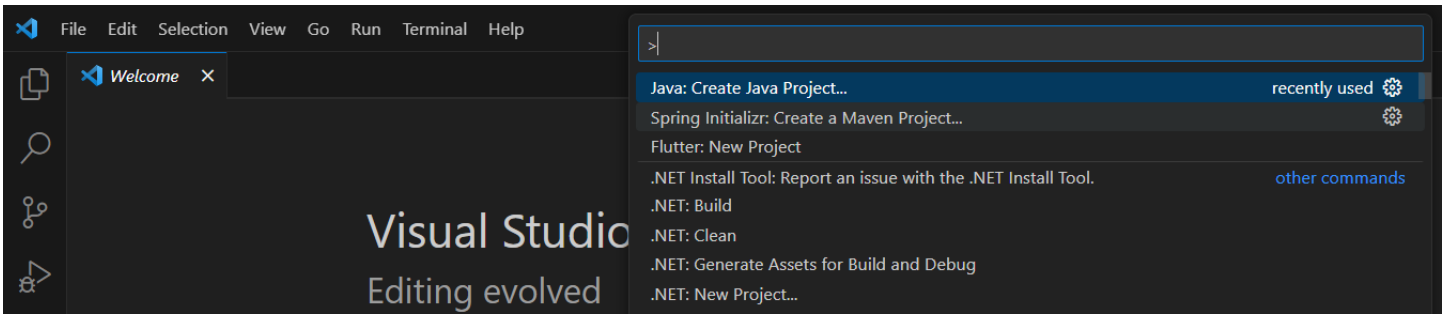


4. Create the Java application in VSCode

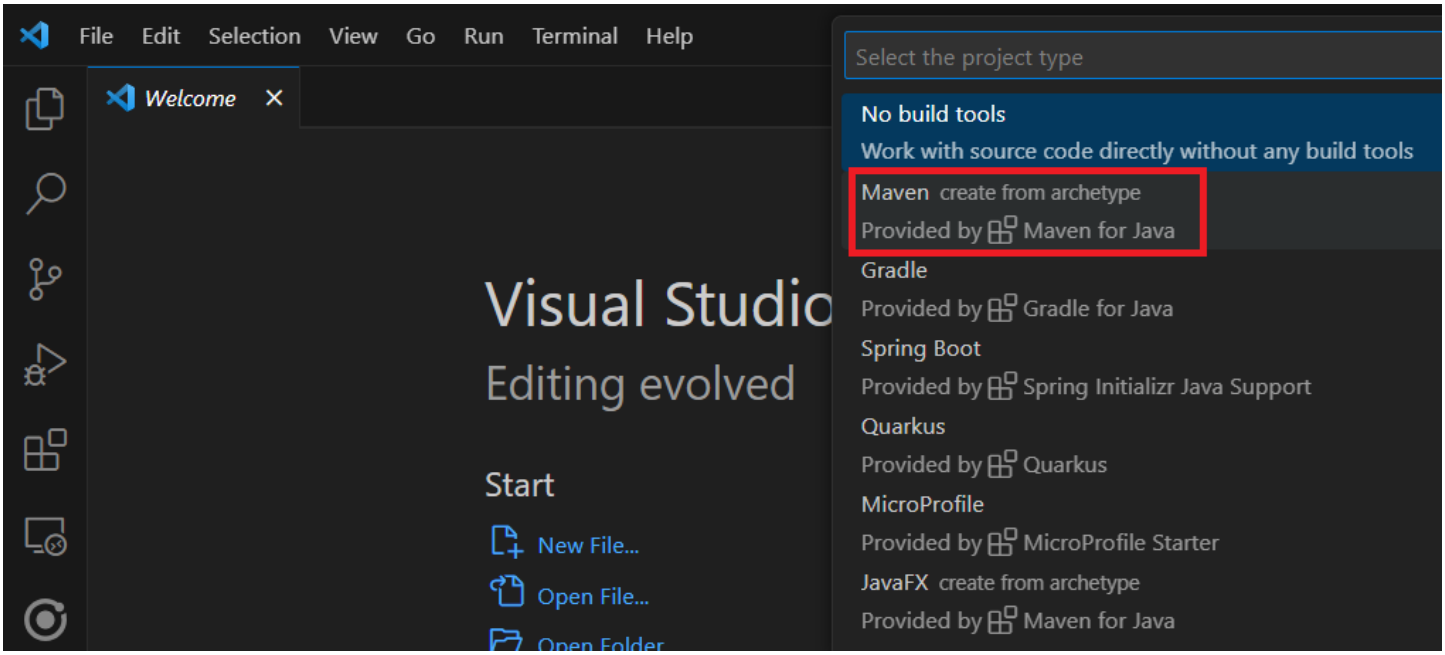
Open VSCode



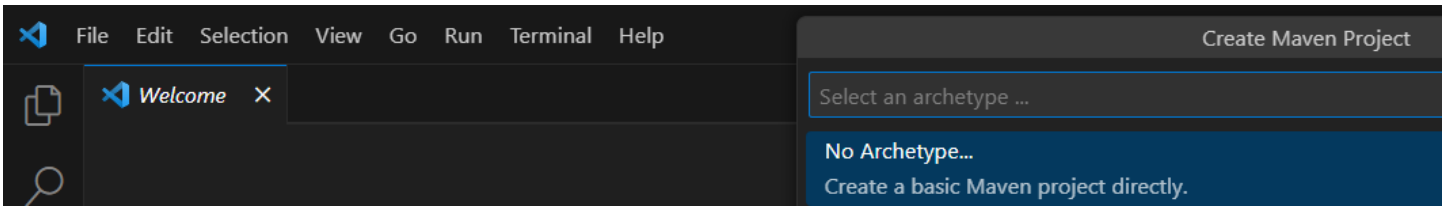
Press the keys **Ctrl+Shift+P** for creating a **new Java application**



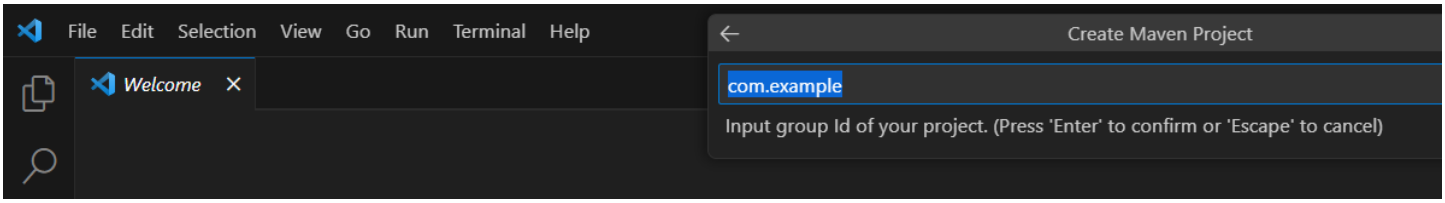
We select **Maven** project type



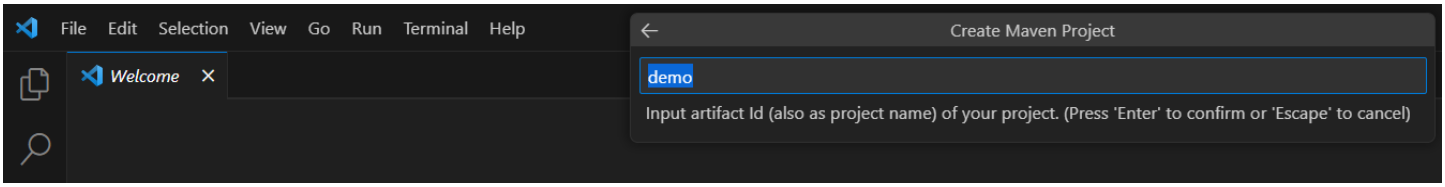
We select **No Archetype**



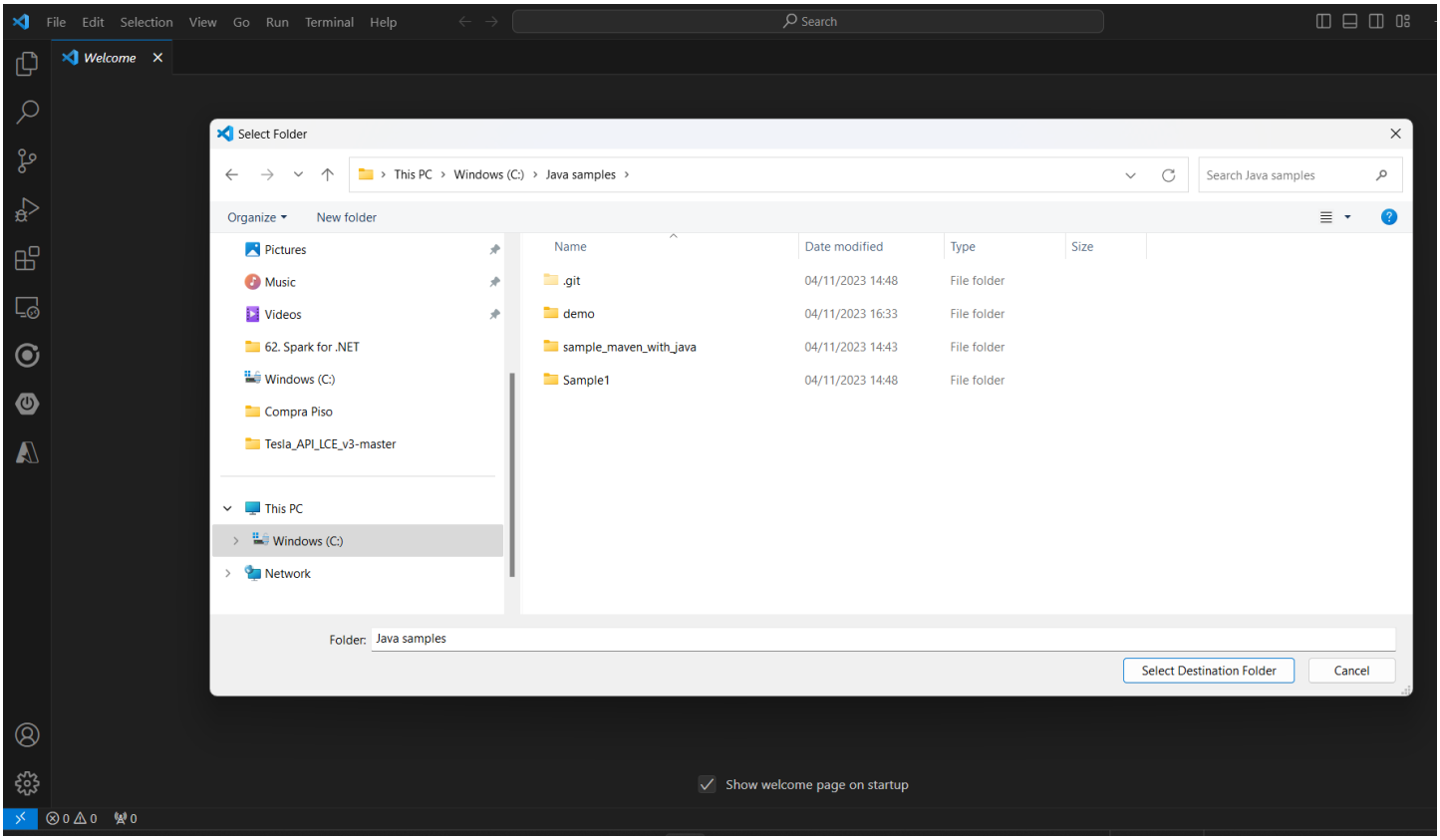
We set the package name **com.example**



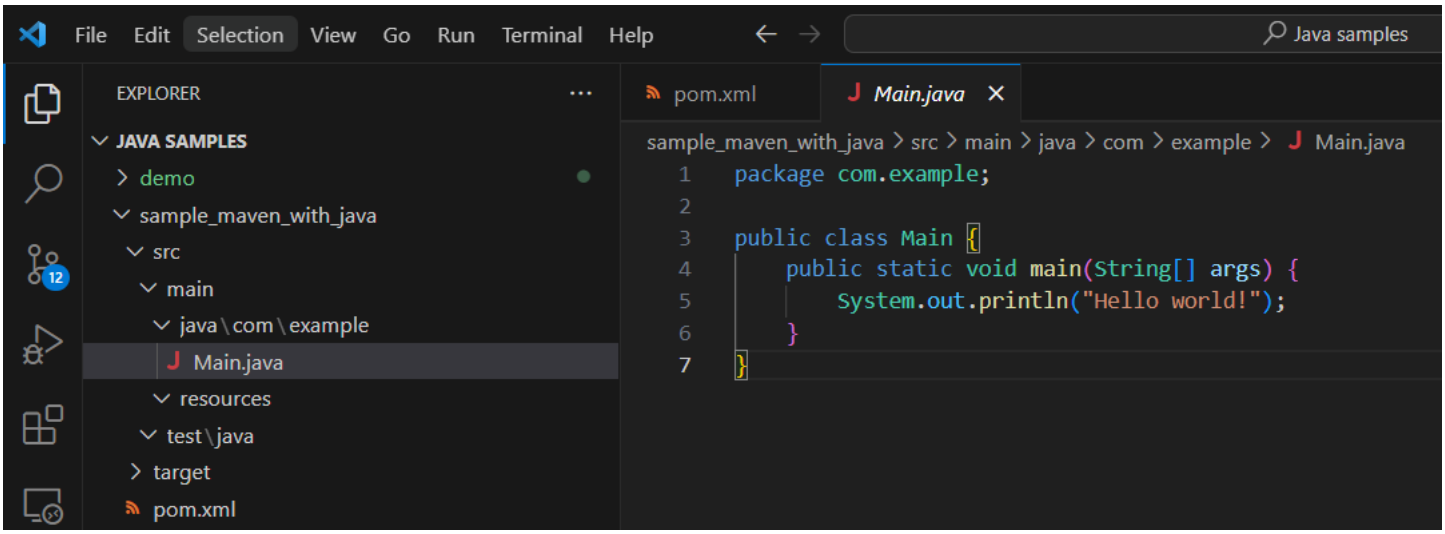
We set the application name



We select the folder where to place the new Java application



We open the new Java application



The original **pom.xml** file not include the tag. We will add this tag in the following sections below

```
<?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/mav
    <modelVersion>4.0.0</modelVersion>

    <groupId>com.example</groupId>
    <artifactId>sample_maven_with_java</artifactId>
    <version>1.0-SNAPSHOT</version>
```

```
    <properties>
        <maven.compiler.source>1.8</maven.compiler.source>
        <maven.compiler.target>1.8</maven.compiler.target>
    </properties>
</project>
```

5. Java application source code

```
package com.example;

public class Main {
    public static void main(String[] args) {
        System.out.println("Hello world!");
    }
}
```

6. Java application 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/mav
    <modelVersion>4.0.0</modelVersion>

    <groupId>com.example</groupId>
    <artifactId>sample_maven_with_java</artifactId>
    <version>1.0-SNAPSHOT</version>

    <properties>
        <maven.compiler.source>1.8</maven.compiler.source>
        <maven.compiler.target>1.8</maven.compiler.target>
    </properties>

    <build>
        <plugins>
            <plugin>
                <!-- Maven JAR Plugin Configuration -->
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-jar-plugin</artifactId>
                <version>3.1.0</version>
                <configuration>
                    <archive>
                        <manifest>
                            <mainClass>com.example.Main</mainClass>
                        </manifest>
                    </archive>
                </configuration>
```

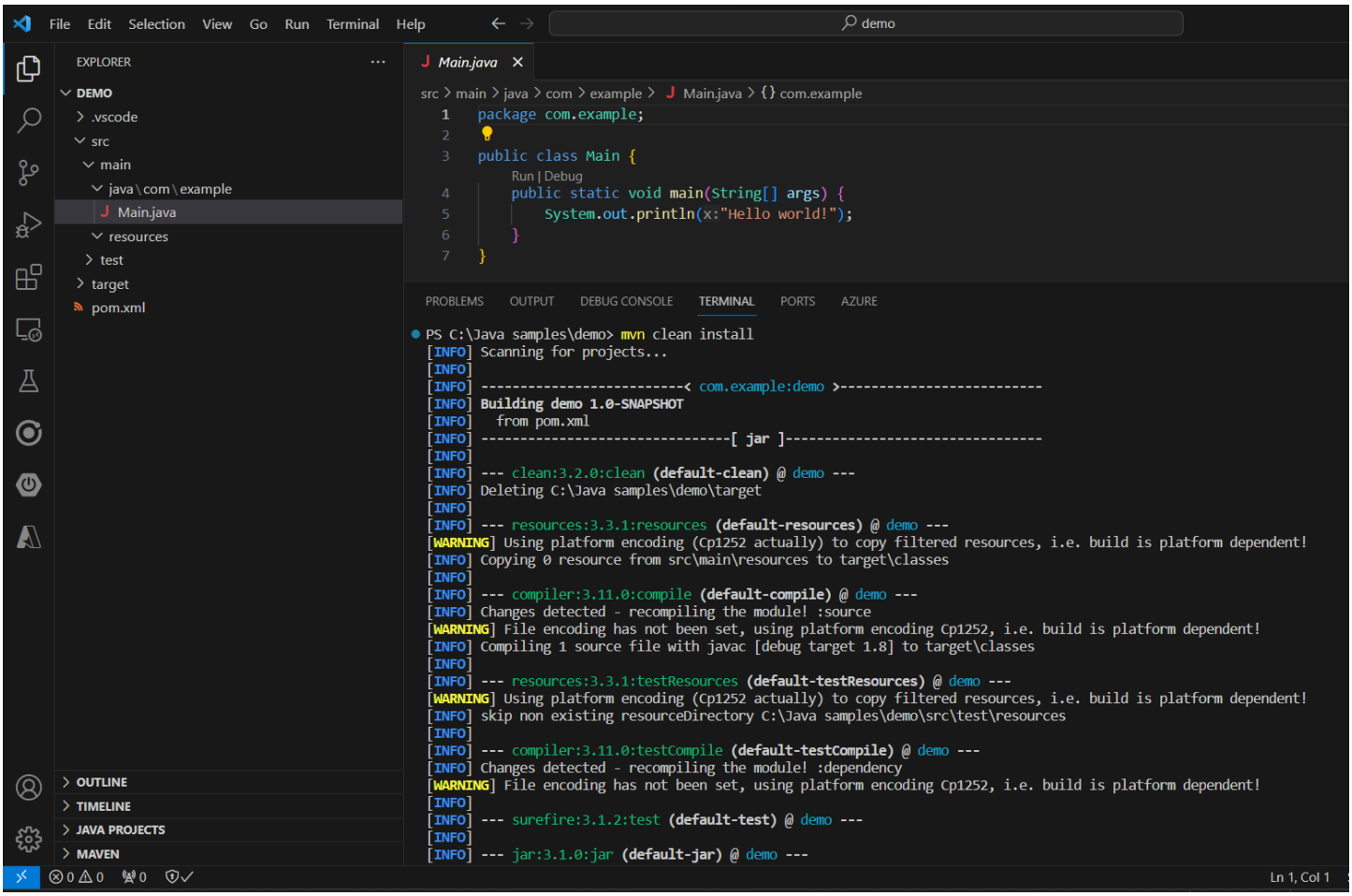
```
            </plugin>

            <plugin>
                <!-- Exec Maven Plugin Configuration -->
                <groupId>org.codehaus.mojo</groupId>
                <artifactId>exec-maven-plugin</artifactId>
                <version>3.1.0</version>
                <configuration>
                    <mainClass>com.example.Main</mainClass>
                </configuration>
                <executions>
                    <execution>
                        <goals>
                            <goal>java</goal>
                        </goals>
                    </execution>
                </executions>
            </plugin>
        </plugins>
    </build>
</project>
```

7. Compile and Run the Java with Maven application

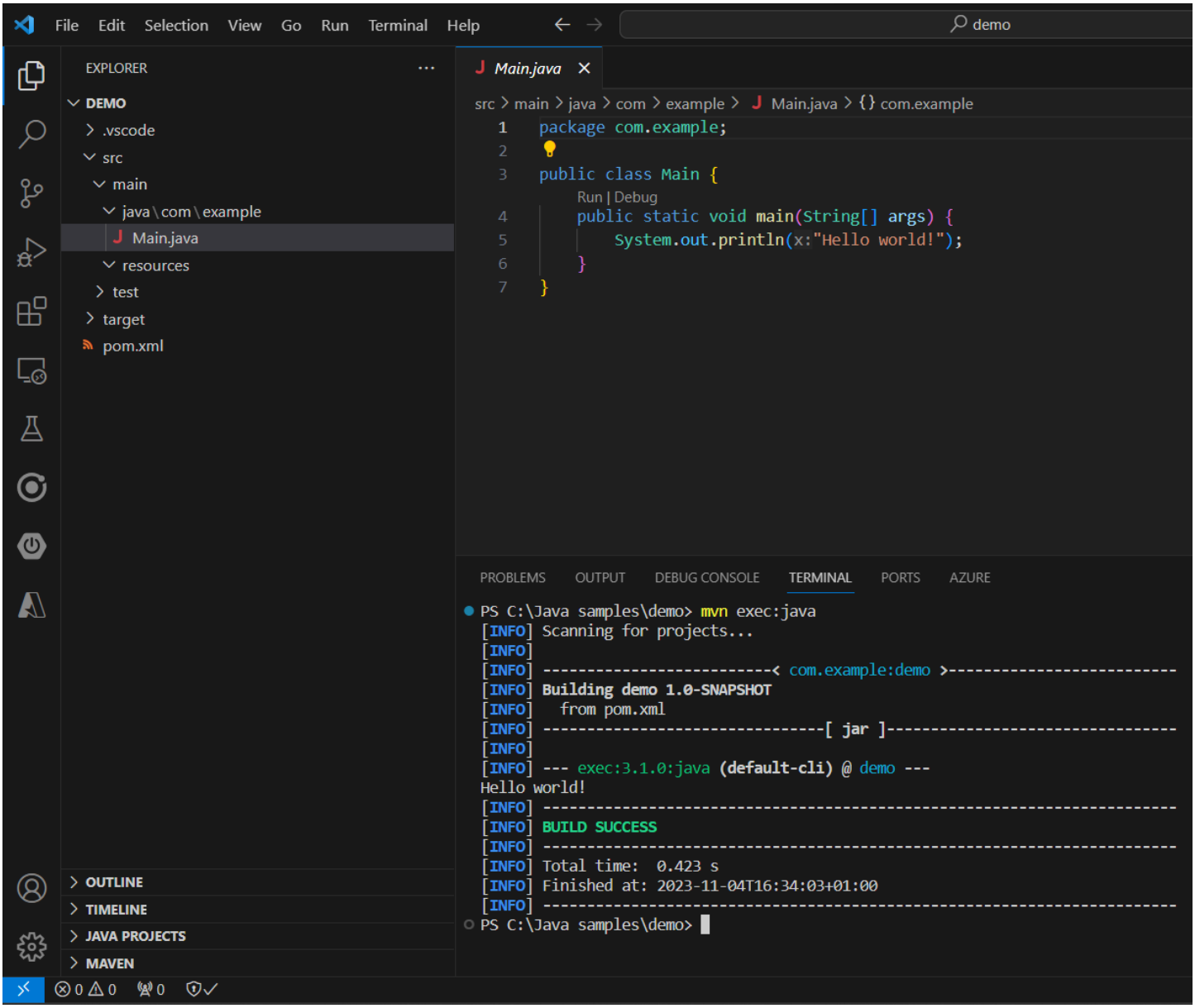
To compile the application run the command:

```
mvn clean install
```



To run the application execute the command:

```
mvn exec:java
```


File Edit Selection View Go Run Terminal Help
demo
EXPLORER
DEMO
.vscode
src
main
java \ com \ example
Main.java
resources
test
target
pom.xml
Main.java
src > main > java > com > example > Main.java > {} com.example
1 package com.example;
2
3 public class Main {
Run | Debug
4 public static void main(String[] args) {
5 System.out.println(x:"Hello world!");
6 }
7 }
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE
PS C:\Java samples\demo> mvn exec:java
[INFO] Scanning for projects...
[INFO]
[INFO] ----------------------< com.example:demo >----------------------
[INFO] Building demo 1.0-SNAPSHOT
[INFO] from pom.xml
[INFO] --------------------------------[jar]---------------------------------
[INFO]
[INFO] --- exec:3.1.0:java (default-cli) @ demo ---
Hello world!
[INFO] --
[INFO] BUILD SUCCESS
[INFO] --
[INFO] Total time: 0.423 s
[INFO] Finished at: 2023-11-04T16:34:03+01:00
[INFO] --
PS C:\Java samples\demo>
OUTLINE
TIMELINE
JAVA PROJECTS
MAVEN