



2.1 - Tecnologías de Servicios de Web

Tema 2 - Aplicaciones Web



Tema 2 - Aplicaciones Web

Tema 2.1 – Spring y Maven

Java EE y Spring

- **Java Enterprise Edition (Java EE)** apareció a finales de los 90 para el desarrollo de aplicaciones web con **Java**
- Soportada por **Oracle, IBM, RedHat**, etc..
- Este estándar ha evolucionado mucho en los últimos años y actualmente se desarrolla bajo la fundación Eclipse y se llama **Jakarta EE**



JAKARTA EE

<https://jakarta.ee/>

Spring

- Spring es un framework de desarrollo de **aplicaciones empresariales** basado en tecnologías Java
- Partes del framework están implementadas sobre **Java Enterprise Edition**
- Actualmente pertenece a **VMWare**

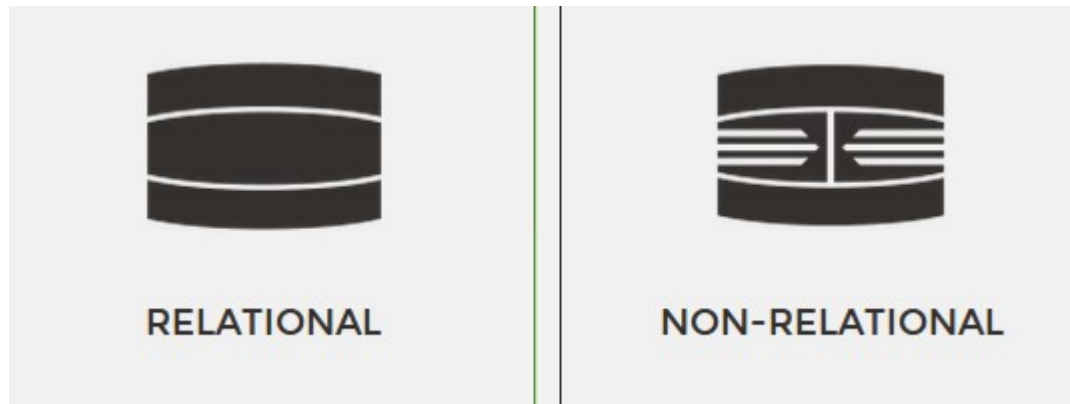


<http://spring.io/>

- Spring permite el desarrollo de **aplicaciones de servidor**:
 - **Aplicaciones web, servicios REST y websockets**
 - Análisis de datos big data
 - Procesado de tareas por lotes (Batch)
 - Integración de sistemas



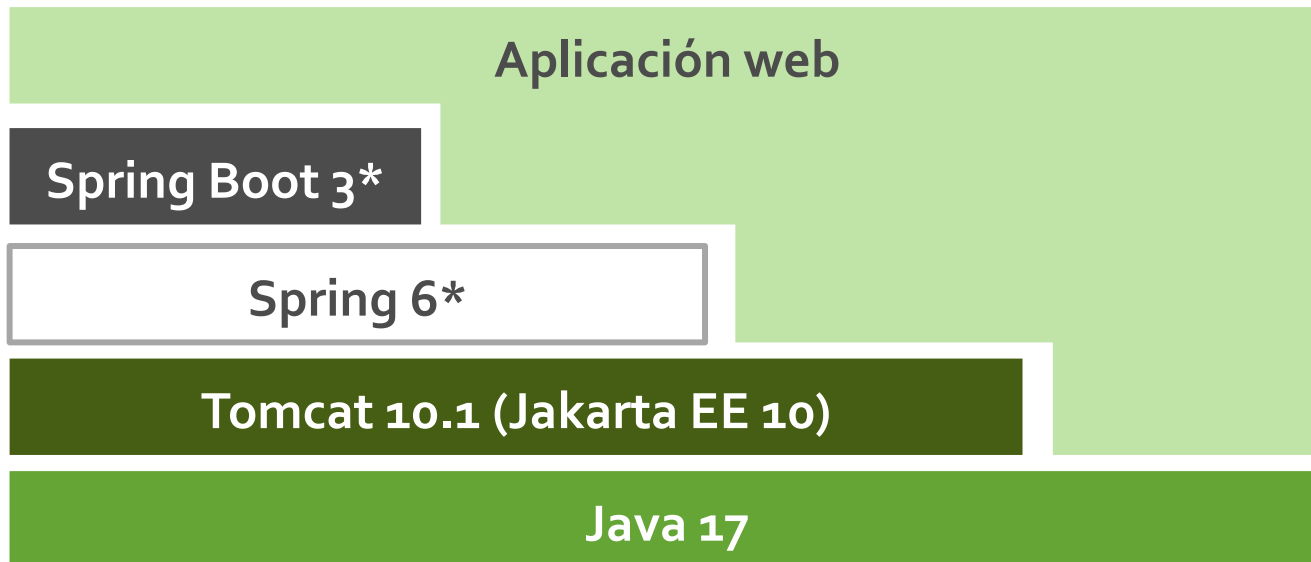
- Tiene un completo soporte de **acceso a bases de datos** de diferentes tipos:
 - Relacionales (SQL)
 - No relacionales (No SQL)



- ## Spring Boot

- Es una librería basada en Spring que **facilita** el desarrollo de aplicaciones
- **Integra el servidor web** (Tomcat, Jetty, etc.)
- **Simplifica la configuración** y acelera el desarrollo
- Se pueden desarrollar aplicaciones **directamente Spring**

- Spring Boot



Para implementar la **aplicación web** se pueden usar librerías de **Java SE 17**, las librerías de **Jakarta EE 10** proporcionadas por **Tomcat 10.1**, la librería **Spring 6*** y la librería **Spring Boot 3***
(* se publicará la release el 24 Nov 2022)

Maven

- Sistema de **gestión de dependencias** (*librerías*) con descarga automática
- Sistema de **construcción de proyectos** (de código a paquete entregable y ejecutable)
- Estructura única de proyecto compatible con todos los **entornos de desarrollo** y sistemas de CI



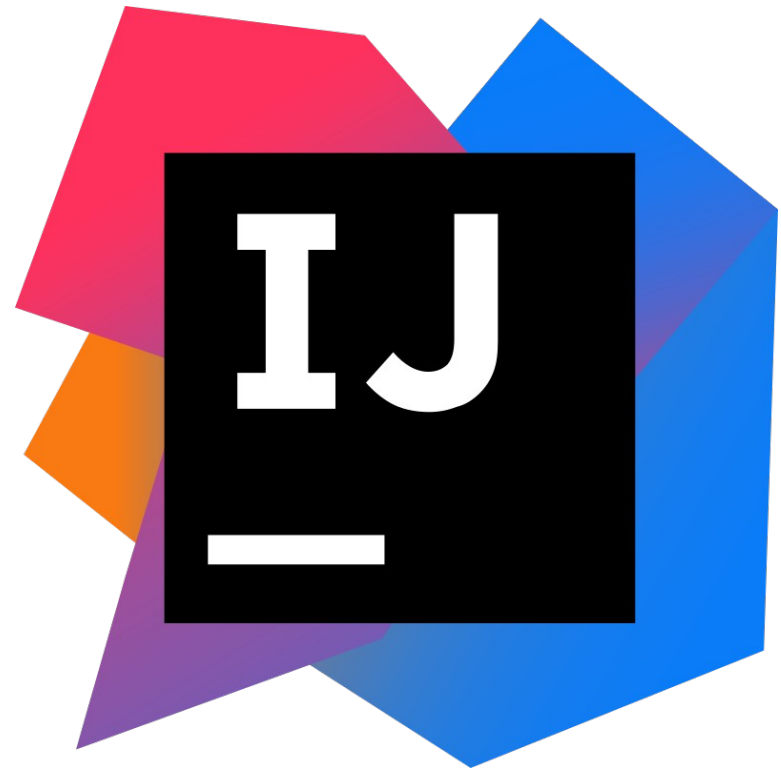
Maven

- Maven gestiona muchos aspectos de la **construcción** de la aplicación
 - Descarga automática de dependencias
 - Compilación
 - Ejecución de tests
 - Publicación de la aplicación construida (binarios)
 - Documentación
 - ...

- ¿Cómo usar Maven?
 - **Desde el entorno de desarrollo:** Maven está integrado en los IDEs Java más importantes (Eclipse, IntelliJ, Visual Studio Code...)
 - **Desde la línea de comandos:** Sin necesidad de usar un IDE. Ideal para construcción de proyectos de forma automatizada

- **IntelliJ IDEA**

- Uno de los IDEs Java más usados
- Comercial (gratuito para estudiantes)
- Desarrollado por JetBrains



<https://www.jetbrains.com/es-es/idea/>

Maven

- New > Project...

New Project

Search:

New Project

Empty Project

Generators

- Maven Archetype
- Jakarta EE
- Spring Initializr
- JavaFX
- Quarkus
- Micronaut
- Ktor
- Kotlin Multiplatform
- Compose Multiplatform
- HTML
- React
- Express
- Angular CLI
- IDE Plugin
- Android

Name:

Location:

Project will be created in: ~/IdeaProjects/Sample

☐ Create Git repository

Language:

Build system:

JDK:

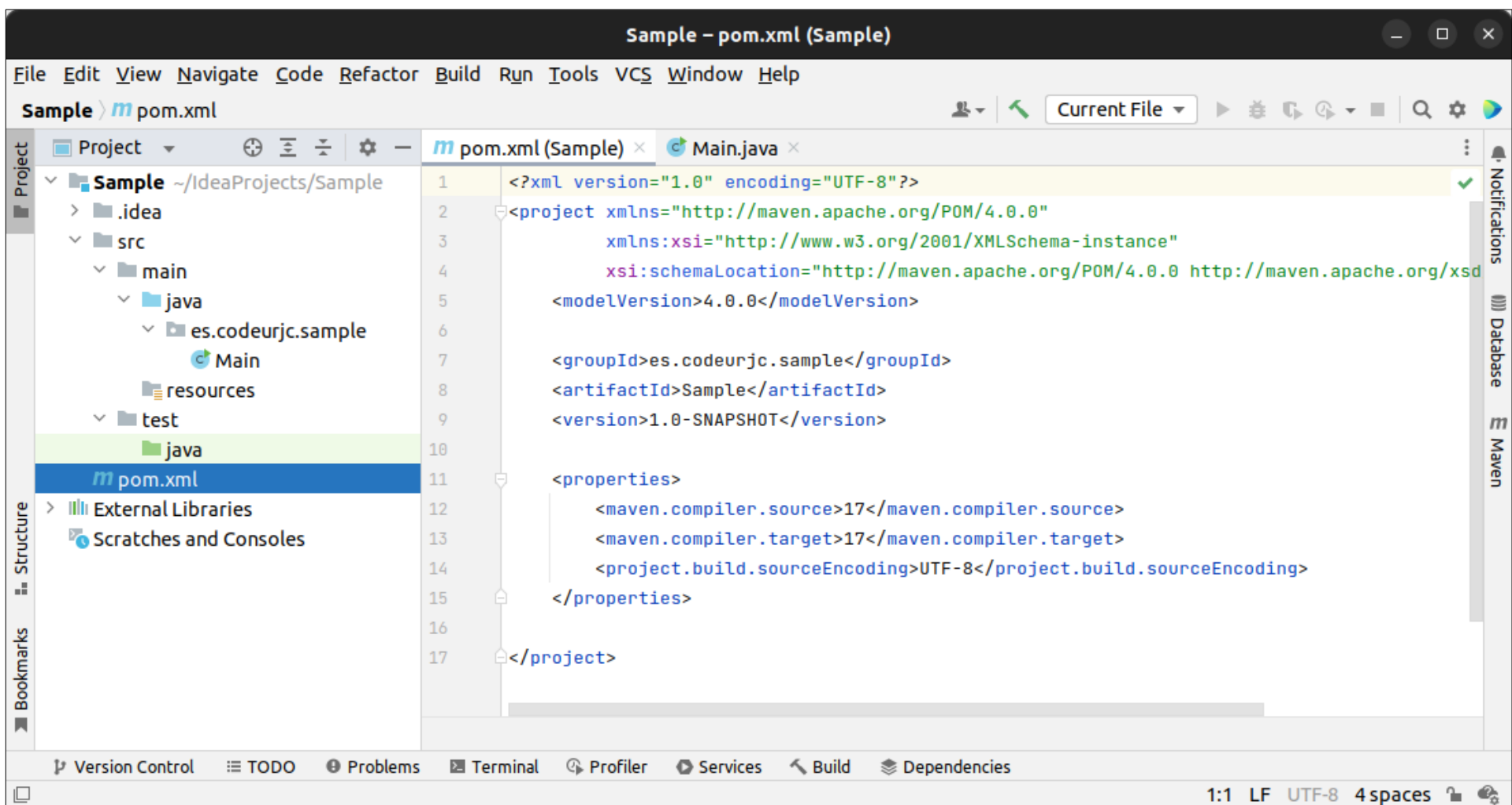
☒ Add sample code

Advanced Settings

GroupId:

ArtifactId:

Maven



- **pom.xml:** Configuración del proyecto
 - **groupId:** Organización, familia
 - **artifactId:** Nombre del proyecto
 - **version:** Versión del proyecto (especialmente útil para librerías)

```
<groupId>es.codeurjc</groupId>  
<artifactId>holamundo</artifactId>  
<version>0.0.1-SNAPSHOT</version>
```

- **pom.xml:** Configuración del proyecto
 - **properties:**
 - Configuraciones generales del proyecto
 - Versión de Java (Si no se especifica es la 1.5)
 - Codificación de los ficheros fuente (UTF-8 asegura compatibilidad linux, win, mac)

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


- **pom.xml: Configuración del proyecto**
 - **dependencies:**
 - ▮ Dependencias (**librerías**)
 - ▮ Cada librería está identificada por su **groupId**, **artifactId** y **versión** (coordenadas)
 - ▮ Se pueden poner tantas dependencias como se quiera

```
<dependencies>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>4.12</version>
    <scope>test</scope>
  </dependency>
</dependencies>
```

- **pom.xml:** Configuración del proyecto
 - **Cuidado!** Algunos cambios en el fichero pom.xml no se reflejan en eclipse de forma automática
 - Cuando se hace un cambio el IDE no se actualiza de forma automática, se tiene que indicar **explícitamente**
 - ▮ Botón derecho proyecto > Maven > Reload Project...

- **Compilar un proyecto Maven**
 - **De forma automática:**
 - ▮ El IDE se descarga las librerías y compila el código.
 - ▮ El IDE no empaqueta la aplicación en un .jar
 - **De forma manual:**
 - ▮ Se pueden ejecutar operaciones Maven desde el IDE o desde la línea de comandos (si está maven instalado en el sistema)

Crear proyecto Spring

- **IntelliJ IDEA**

- File > New project > **Spring Initializr**
- Indicar el nombre del proyecto:
 - Name: **helloworld-spring**
 - Group: **es.codeurjc**
 - Artifact: **helloworld-spring**
 - Package: **es.codeurjc.helloworld**
- Versión de Java: 17
- Se seleccionan las librerías que se quieren usar

Crear proyecto Spring

The screenshot shows the 'New Project' dialog in IntelliJ IDEA. On the left, a list of project generators is shown, with 'Spring Initializr' selected and highlighted in blue. Other generators include 'New Project', 'Empty Project', 'Maven Archetype', 'Jakarta EE', 'JavaFX', 'Quarkus', 'Micronaut', 'Ktor', 'Kotlin Multiplatform', 'Compose Multiplatform', 'HTML', 'React', 'Express', 'Angular CLI', 'IDE Plugin', and 'Android'. The right side of the dialog contains configuration fields for the new project:

- Server URL: start.spring.io (with a settings gear icon)
- Name:
- Location: (with a folder icon)
Project will be created in: ~/IdeaProjects/demo
- ☐ Create Git repository
- Language: ☒ Java ☐ Kotlin ☐ Groovy
- Type: ☒ Gradle ☐ Maven
- Group:
- Artifact:
- Package name:
- JDK: (with a dropdown arrow)
- Java: (with a dropdown arrow)
- Packaging: ☒ Jar ☐ War

At the bottom right, there are 'Next' and 'Cancel' buttons. A help icon (?) is located at the bottom left.

Crear proyecto Spring

New Project

Spring Boot: 3.0.0 (RC2)

☒ Download pre-built shared indexes for JDK and Maven libraries

Dependencies:

Search

Developer Tools

- ☒ GraalVM Native Support
- ☐ Spring Boot DevTools
- ☐ Lombok
- ☐ Spring Configuration Processor

> Web

> Template Engines

> Security

> SQL

> NoSQL

> Messaging

> I/O

> Ops

> Observability

> Testing

GraalVM Native Support

Support for compiling Spring applications to native executables using the GraalVM native-image compiler.

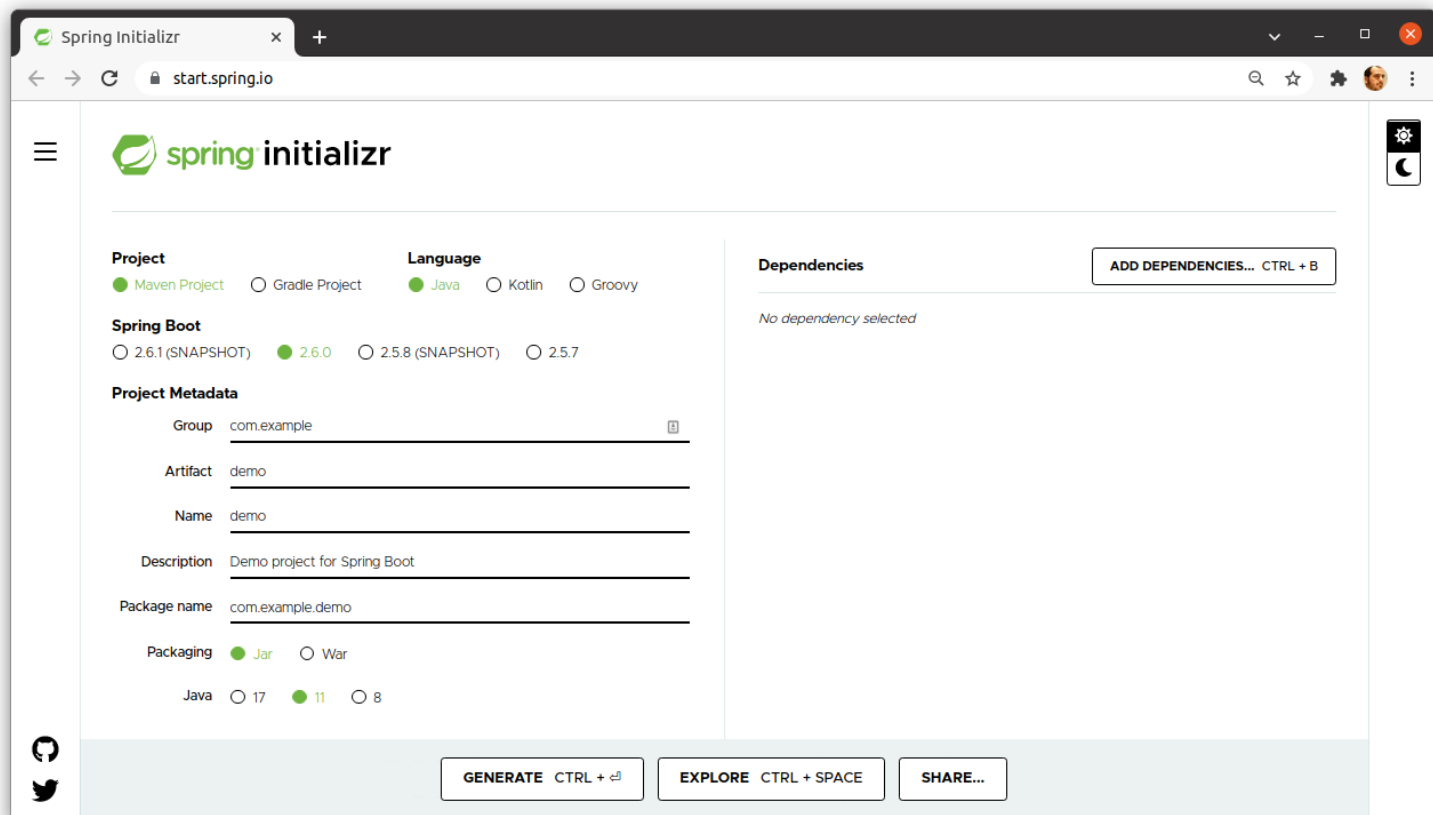
Added dependencies:

No dependencies added

? Previous Create Cancel

Crear proyecto Spring

- Desde la web



The screenshot shows the Spring Initializr web application in a browser window. The URL is start.spring.io. The interface is divided into several sections:

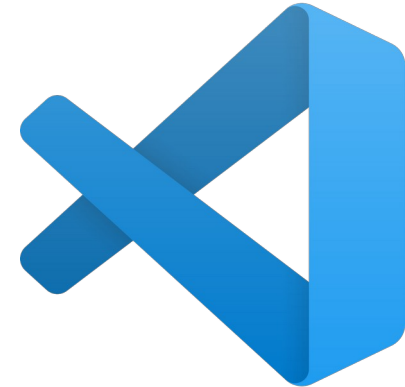
- Project:** Includes radio buttons for **Maven Project** (selected), **Gradle Project**, and **Language** options: **Java** (selected), **Kotlin**, and **Groovy**.
- Spring Boot:** Includes radio buttons for versions: **2.6.1 (SNAPSHOT)**, **2.6.0** (selected), **2.5.8 (SNAPSHOT)**, and **2.5.7**.
- Project Metadata:** A form with fields for:
 - Group:** `com.example`
 - Artifact:** `demo`
 - Name:** `demo`
 - Description:** `Demo project for Spring Boot`
 - Package name:** `com.example.demo`
 - Packaging:** **Jar** (selected), **War**
 - Java:** **11** (selected), **17**, **8**
- Dependencies:** A section with the text "No dependency selected" and a button labeled "ADD DEPENDENCIES... CTRL + B".

At the bottom, there are three buttons: **GENERATE CTRL + G**, **EXPLORE CTRL + SPACE**, and **SHARE...**

<https://start.spring.io>

Crear proyecto Spring

- Visual Studio Code



Extension Pack for Java v0.18.6 Preview

Microsoft | 10,281,031 | ★★★★★ (44)

Popular extensions for Java development that provides Java IntelliSense, debugging, testing, Maven/Gradle support, project ...

[Disable](#) [Uninstall](#) [Settings](#)

This extension is enabled globally.

[Details](#) [Feature Contributions](#) [Changelog](#) [Runtime Status](#)

Extension Pack (6)



Language Support for Java(TM)... ⌚ 37ms
Java Linting, Intellisense, formatting, refac...
Red Hat



Debugger for Java ⌚ 12ms
A lightweight Java debugger for Visual Stu...
Microsoft



Test Runner for Java ⌚ 28ms
Run and debug JUnit or TestNG test cases
Microsoft



Maven for Java ⌚ 24ms
Manage Maven projects, execute goals, ge...
Microsoft

Categories

[Programming Languages](#) [Linters](#)
[Debuggers](#) [Formatters](#)
[Snippets](#) [Extension Packs](#)

Resources

[Marketplace](#)
[Repository](#)

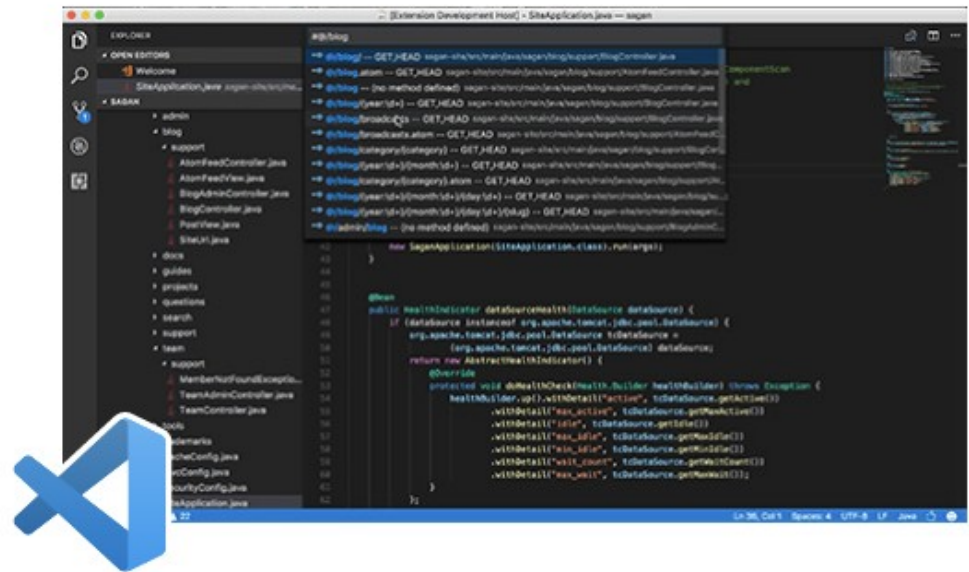
Crear proyecto Spring

- Visual Studio Code

Spring Tools 4 for Visual Studio Code

Free. Open source.

SPRING TOOLS 4
VSCode Marketplace



Instalamos las extensiones de Spring en nuestro VSCode

<https://marketplace.visualstudio.com/items?itemName=Pivotal.vscode-boot-dev-pack>

Crear proyecto Spring

- Visual Studio Code



Spring Boot Extension Pack v0.1.0

Pivotal | 819,964 | ★★★★★ (5)

A collection of extensions for developing Spring Boot applications

[Install](#)

[Details](#) [Feature Contributions](#)

Extension Pack (3)



Spring Boot Tools

Provides validation and content assist for S...
Pivotal



Spring Boot Dashboard

Spring Boot Dashboard for VS Code
Microsoft

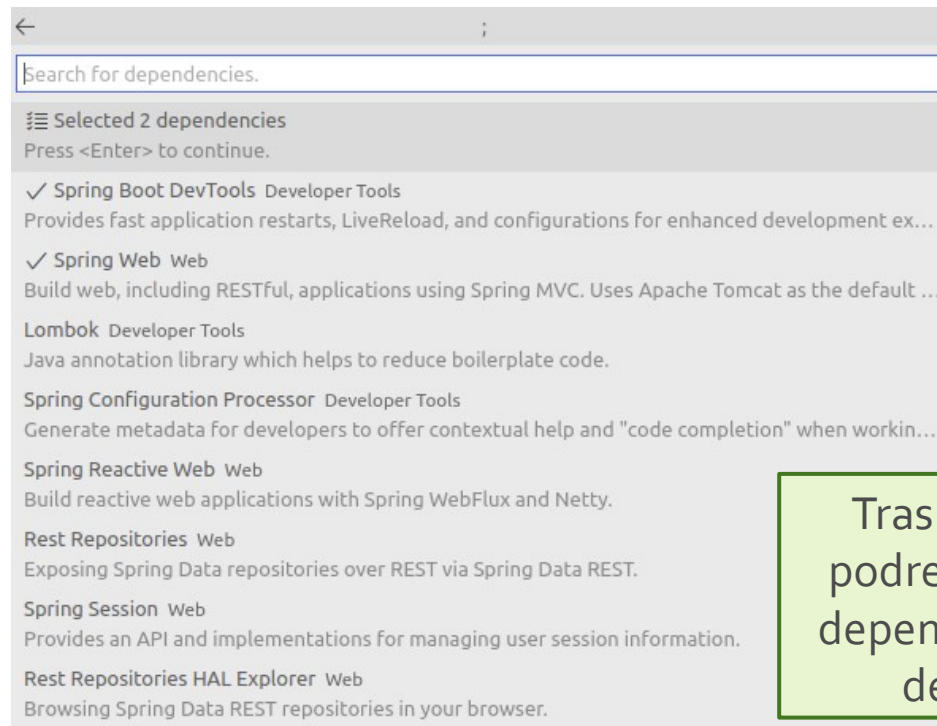
Crear proyecto Spring

- **Visual Studio Code**

- View > Command Palette (o Ctrl + Shift + P)
- Introducimos “Spring” y seleccionamos “Create a Maven Project”
- Nos pedirá incluir información adicional:
 - Versión de Spring Boot: 2.6
 - Lenguaje: *Java*
 - GroupId: *es.codeurjc* y ArtifactId: *helloworld-vscode*
 - Empaquetamiento: *JAR*
 - Versión: *17*

Crear proyecto Spring

- **Visual Studio Code**
 - Nos permite, además, seleccionar las dependencias básicas



Tras crear el proyecto,
podremos añadir nuevas
dependencias al pom.xml
de forma manual

Crear proyecto Spring

- **Visual Studio Code**
 - El pack de extensiones incluye numerosas funcionalidades, como **Spring Dashboard**, que nos permite visualizar las aplicaciones en el entorno y lanzarlas de forma sencilla.

Crear proyecto Spring

- Visual Studio Code

