

# Manual de instalación de IntelliJ para Windows 10 (Mayo 2018)

Manual elaborado por David Amancio Gil Méndez

## Descargamos los paquetes de software necesarios

1. Java Development Kit: [JDK](#)
  1. Seleccionamos la última versión disponible y la arquitectura (32 o 64 bits)
  2. En el momento de la creación de este manual se utilizó la versión [8u172-x64](#)
2. Binarios de Maven: [Maven](#)
  1. En el momento de la creación de este manual se utilizó la versión [3.3.9](#)
3. IntelliJ IDEA : [IntelliJ](#)
  1. Nos descargamos la versión [Community](#)

## Instalación de Java Development Kit

Instalamos el ejecutable de forma rutinaria. No es necesario cambiar ningún parámetro o configuración. Es importante fijarse en que carpeta se instala el software. Normalmente es la carpeta C:\Archivos de Programa\Java\jdk...

## Creación de variables de entorno

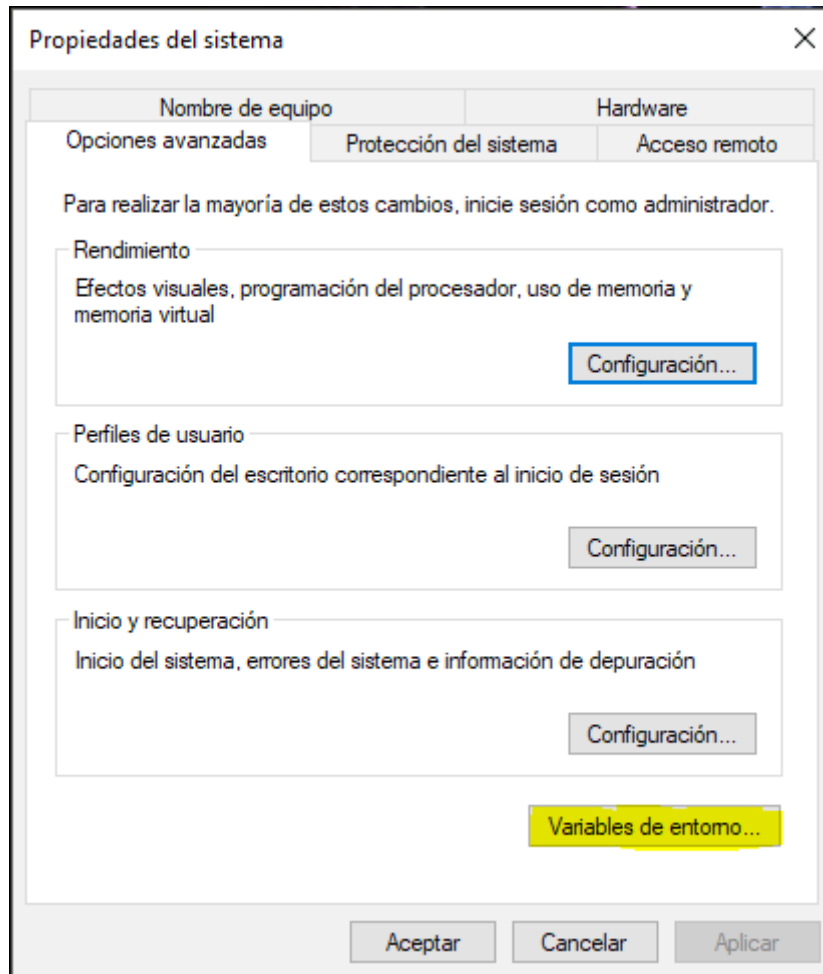
En el menú de inicio de Windows, escribimos “variables”. Aparecerán dos opciones. Seleccionamos “Editar las variables de Entorno de este sistema”

Otra forma de llegar al menú es haciendo click secundario sobre el icono de “Este Equipo” en el escritorio. Seleccionamos propiedades. Se abrirá la siguiente ventana:

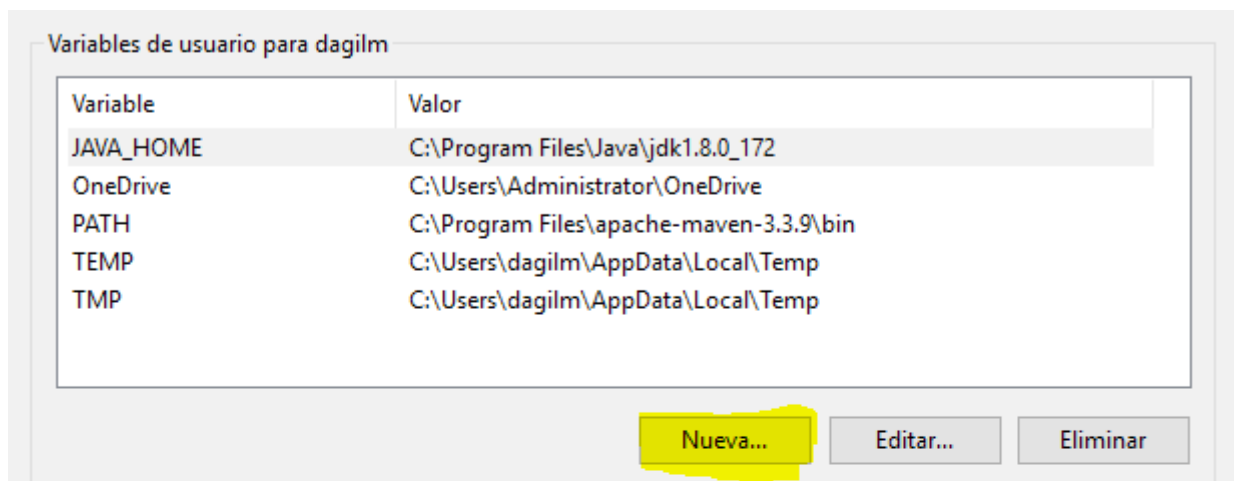


## MANUAL DE INSTALACIÓN DE INTELLIJ – MAVEN

Clicamos en “Configuración avanzada del sistema” y aparecerá una nueva ventana que es equivalente a la que aparece al realizar la búsqueda en el menú de inicio. En esta nueva ventana (Propiedades del Sistema), clicamos en el botón de “Variables de entorno...” que se encuentra en la parte inferior de la pestaña de “Opciones Avanzadas”.

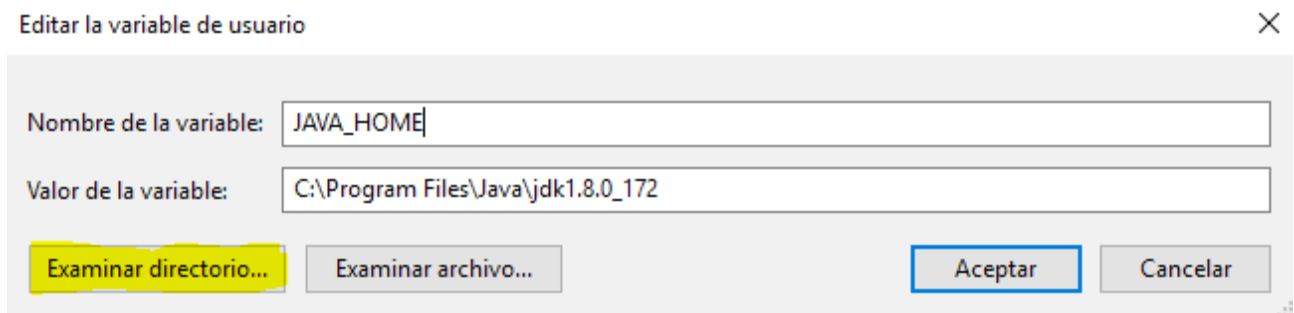


En el cuadro que aparece, seleccionamos el botón “Nueva...”:

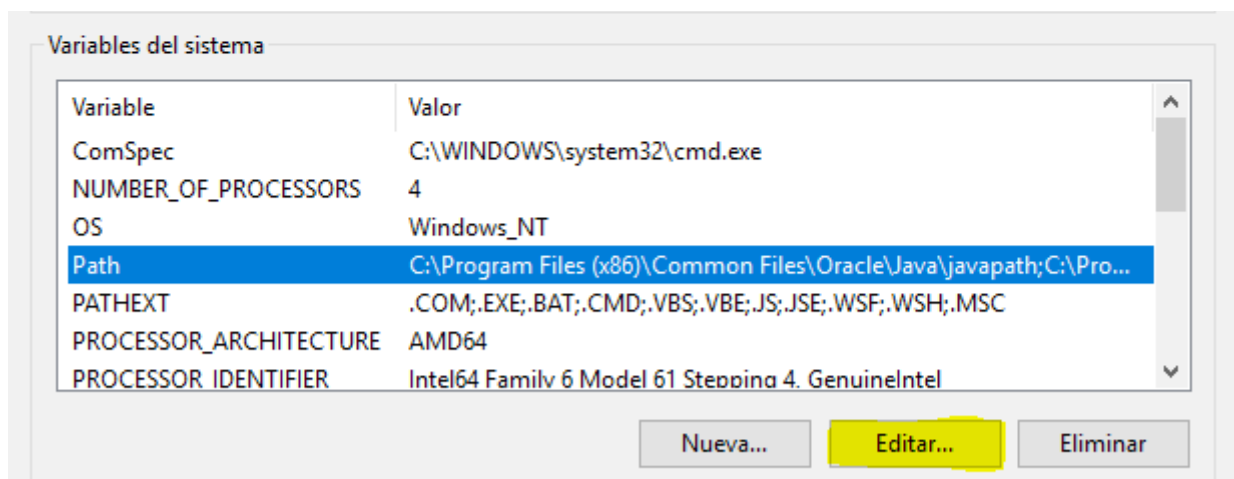


## MANUAL DE INSTALACIÓN DE INTELLIJ – MAVEN

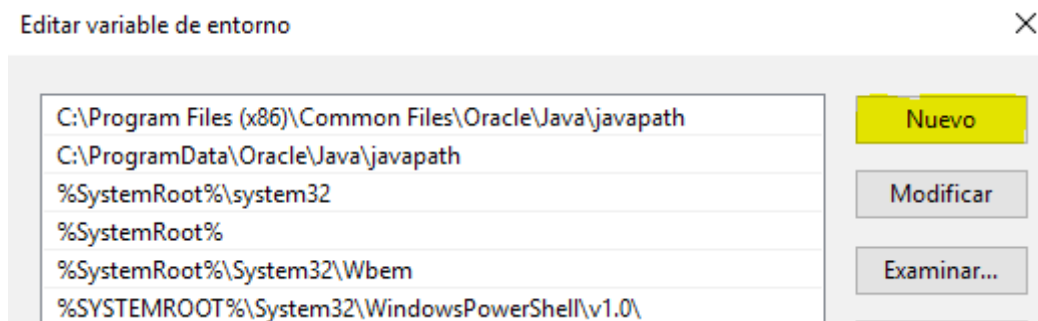
Creamos una nueva variable de usuario llamada “JAVA\_HOME” y establecemos como directorio de destino el directorio donde se ha instalado el JDK:



Una vez creada la variable, la hemos de añadir al Path del sistema. Para ello, clicamos sobre “Path” y “Editar...”



En la nueva ventana “Editar variable de Entorno, clicamos en “Nuevo”:



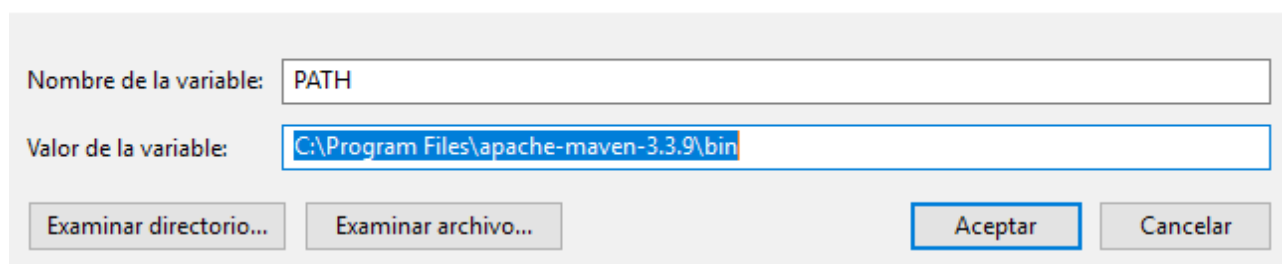
Se habilitará una caja de texto al final de la lista donde escribiremos “%JAVA\_HOME%” (sin las comillas dobles). De esta forma, queda configurado el JDK.

## Instalación de los binarios de Maven

Descargamos el binario de Maven y lo descomprimos. Se generará una carpeta llamada “apache-maven-3.3.9” (o cualquier otro número dependiendo de la versión). Dentro de esta carpeta existe un fichero README que explica el proceso de instalación.

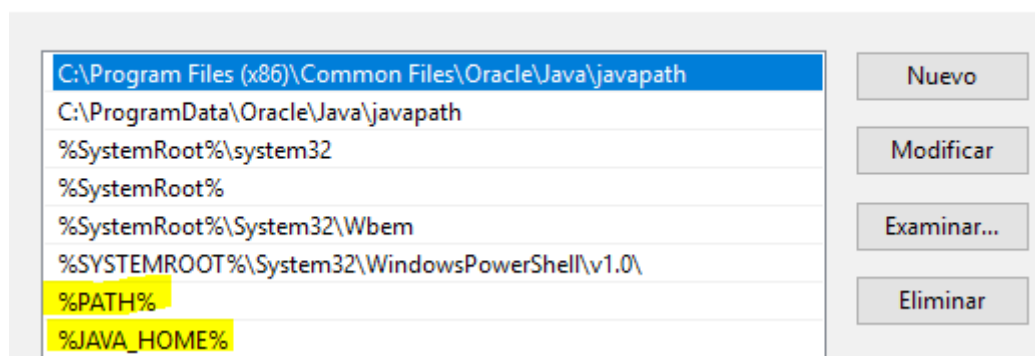
1. Copiamos la carpeta “apache-maven-x.x.x” en C:\Archivos de Programa\apache-maven-x.x.x
2. Creamos una nueva variable de usuario llamada PATH y que apunte al directorio C:\Archivos de Programa\apache-maven-3.3.9\bin

Editar la variable de usuario



3. Añadimos la variable de usuario “PATH” al path de las variables del sistema. Seguimos los mismos pasos que para la variable de JAVA\_HOME. Una vez terminado, deberíamos poder ver las siguientes variables en el path del sistema:

Editar variable de entorno



Para comprobar si los binarios de Maven se han instalado correctamente, abriremos una terminal. Para ello buscamos en el inicio de windows “símbolo del sistema” o “cmd”. Una vez dentro, ejecutaremos el comando “mvn -v”.

```

C:\> Símbolo del sistema

Microsoft Windows [Versión 10.0.15063]
(c) 2017 Microsoft Corporation. Todos los derechos reservados.

C:\Users\dagilm>mvn -v
Apache Maven 3.3.9 (bb52d8502b132ec0a5a3f4c09453c07478323dc5; 2015-11-10T17:41:47+01:00)
Maven home: C:\Program Files\apache-maven-3.3.9\bin\..
Java version: 1.8.0_172, vendor: Oracle Corporation
Java home: C:\Program Files\Java\jdk1.8.0_172\jre
Default locale: es_ES, platform encoding: Cp1252
OS name: "windows 10", version: "10.0", arch: "amd64", family: "dos"
  
```

## Configuración de Maven

En C:\Archivos de Programa\apache-maven-x.x.x\conf existe el archivo “settings.xml”. Hacemos una copia y lo renombramos como “settingsOld.xml”. Editamos “settings.xml” con algún editor. Se recomienda utilizar el [Notepad++](#) en caso de que no se disponga de ningún otro editor más allá del bloc de notas. Borramos el contenido del archivo y lo sustituimos por el siguiente:

Nota: Es importante que configuremos los campos marcados con amarillo de acuerdo a nuestro centro de trabajo y con nuestras credenciales de usuario. Si cambiamos la contraseña, tendremos que volver a editar el archivo “settings.xml”.

```
<?xml version="1.0" encoding="UTF-8"?>
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0 http://maven.apache.org/xsd/settings-
1.0.0.xsd">
  <proxies>
    <proxy>
      <active>true</active>
      <protocol>http</protocol>
      <host>proxy.indra.es</host>
      <port>8080</port>
      <username>dagilm</username>
      <password>*****</password>
    </proxy>
  </proxies>
  <profiles>
    <profile>
      <id>defaultRepo</id>
      <repositories>
        <repository>
          <id>central</id>
          <!--<url>https://repo1.maven.org/maven2</url> -->
          <url> http://central.maven.org/maven2/</url>
          <releases>
            <enabled>true</enabled>
          </releases>
        </repository>
      </repositories>
      <pluginRepositories>
        <pluginRepository>
          <id>central</id>
          <url>https://repo1.maven.org/maven2</url>
          <releases>
            <enabled>true</enabled>
          </releases>
        </pluginRepository>
      </pluginRepositories>
    </profile>
  </profiles>
  <activeProfiles>
    <activeProfile>defaultRepo</activeProfile>
  </activeProfiles>
</settings>
```

## Instalación de IntelliJ IDEA con plugin de Scala

Instalaremos la última versión disponible de IntelliJ IDEA de acuerdo a la versión de nuestro sistema operativo (32bits o 64bits).

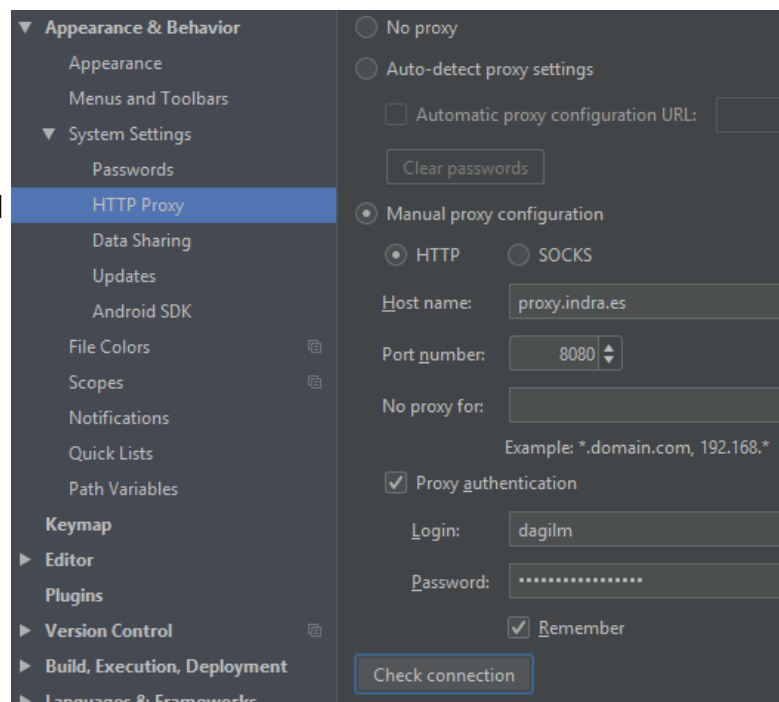
Una vez arranquemos por primera vez el IntelliJ IDEA, seguiremos los siguientes pasos:

1. En la pantalla de inicio, clicamos en “Configure”. Se desplegará un menú y en el seleccionamos “Settings”.



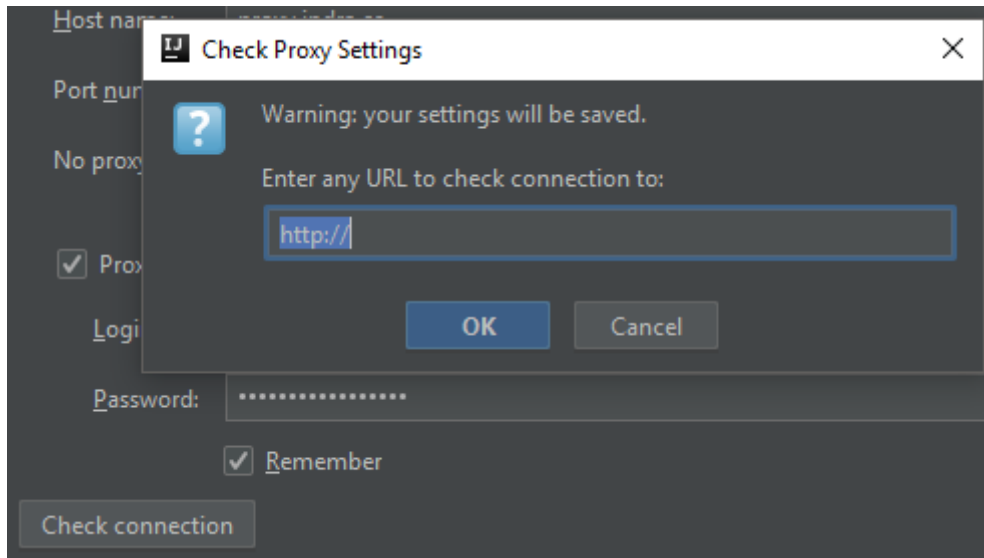
2. Dentro de Settings, expandimos la pestaña de “Appearance & Behaviour”. Dentro de ella, expandimos “System Settings”. Finalmente, accedemos a “HTTP Proxy”.

- 2.1. En este menú, hemos de introducir los datos del proxy de Indra o de nuestro centro de trabajo manualmente así como nuestras credenciales de usuario. De esta forma, IntelliJ será capaz de conectarse a la red.

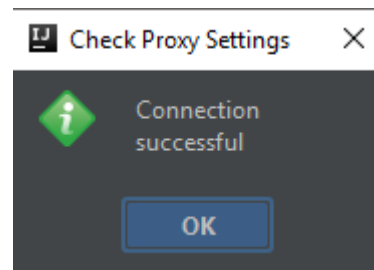


## MANUAL DE INSTALACIÓN DE INTELLIJ – MAVEN

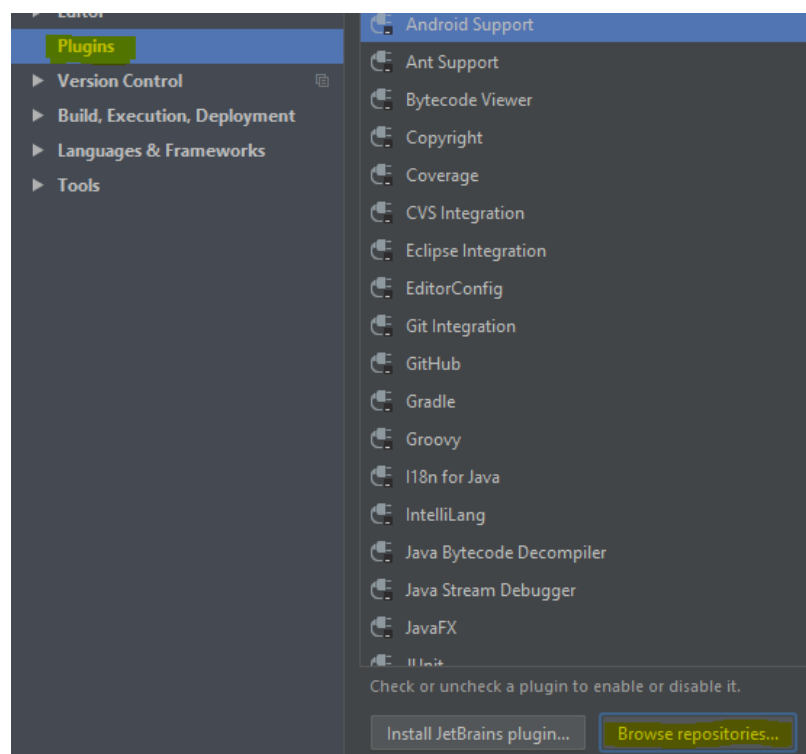
- 2.2. Una vez hemos rellenado los campos, debemos comprobar que la conexión es exitosa. Para ello clicamos en “Check connection”.



- 2.3. Después de introducir la dirección web a comprobar, debería aparecer “Connection successful”



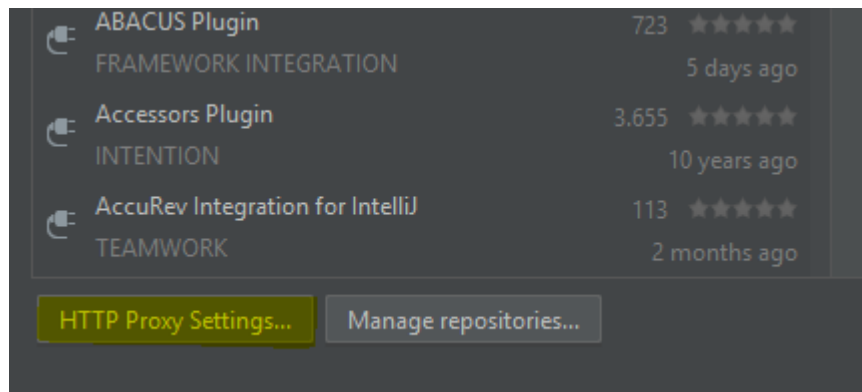
3. Volvemos al menú inicial y esta vez clicamos en “Plugins” y en la parte inferior clicamos en “Browse repositories...”



## MANUAL DE INSTALACIÓN DE INTELLIJ – MAVEN

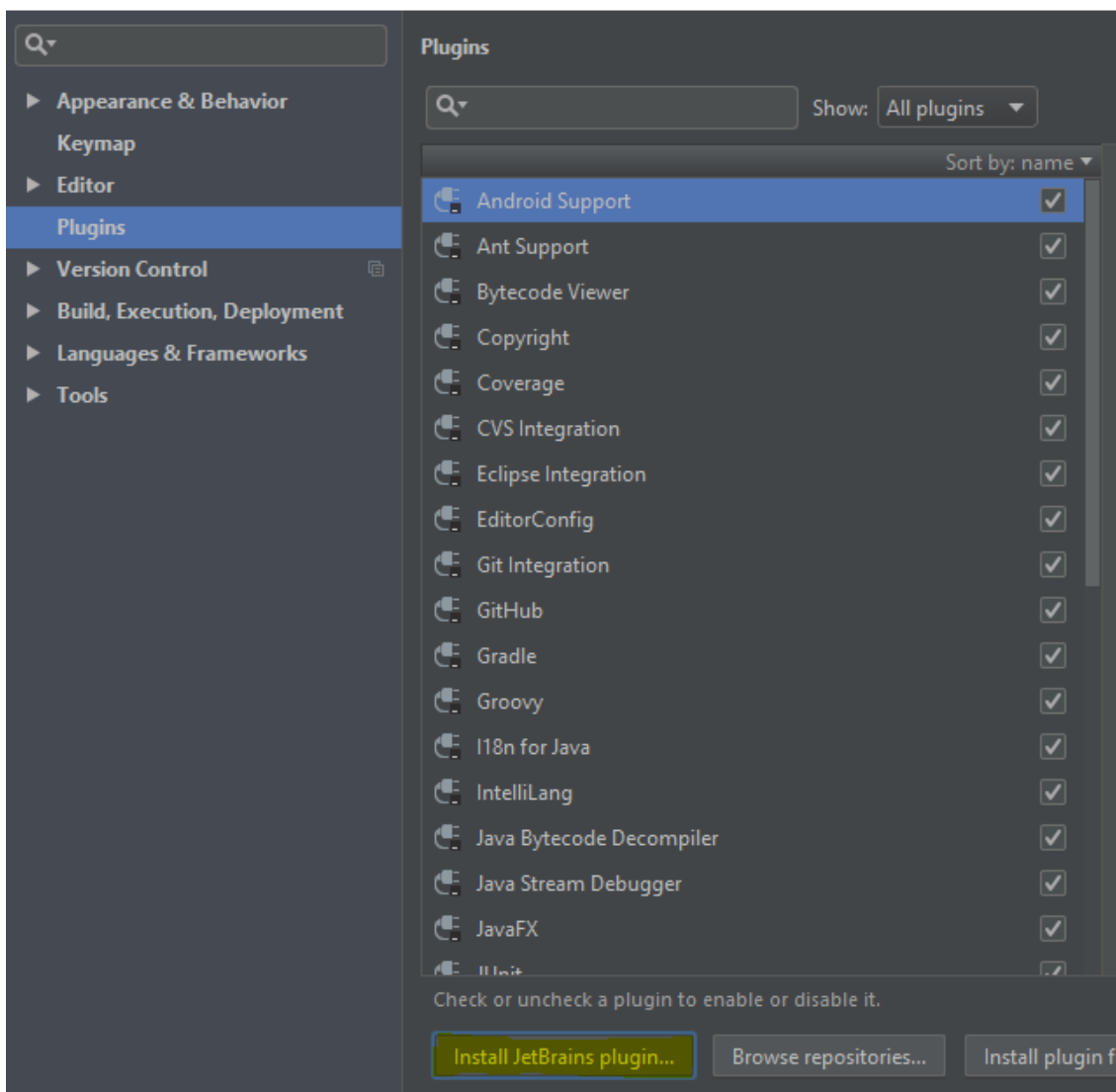
- 3.1. Aparecerá una nueva ventana donde tendremos que clicar en “HTTP Proxy Settings...”. Volvemos a rellenar los campos tal y como hicimos la vez anterior.

Una vez configurado y comprobada la conexión, podremos instalar el Plugin de Scala.



- 3.2. Volvemos a la pagina principal de “Plugins” y clicamos en “Install JetBrains plugin...”

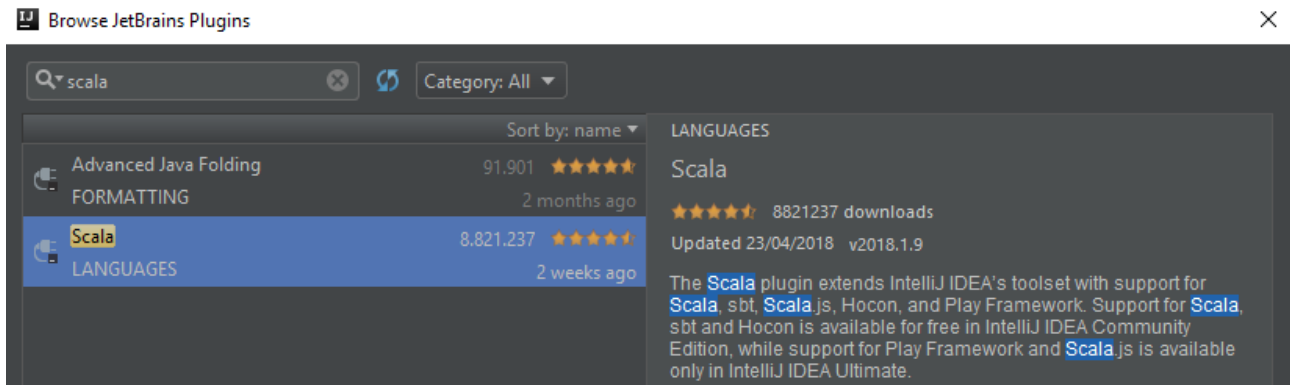
Default Settings





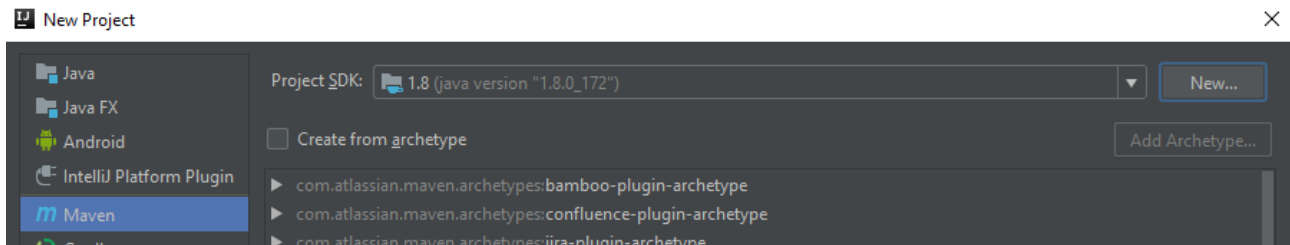
## MANUAL DE INSTALACIÓN DE INTELLIJ – MAVEN

- 3.3. Escribimos en la barra de búsqueda (parte superior, tiene una lupa) “Scala”. Aparecerán dos resultados. Instalamos el plugin “Scala – LANGUAGES”. Una vez que se termine la descarga y la instalación, reiniciaremos el IntelliJ.



## Configuración de un proyecto Maven en IntelliJ

Iniciamos de nuevo IntelliJ IDEA y creamos un nuevo proyecto “Create New Project”

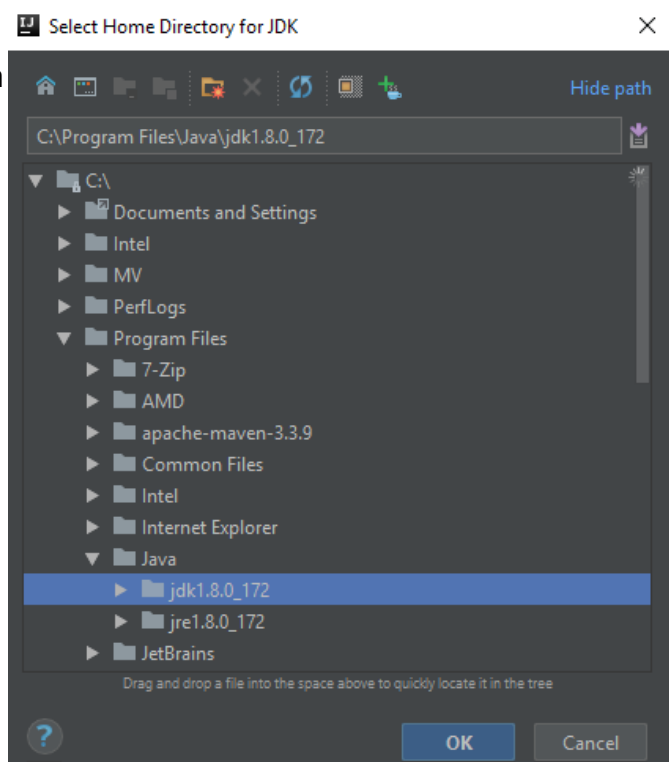


Clicamos en “New” que está a la izquierda de la barra de “Project SDK”. Aparecerá un menú desplegable donde tendremos que seleccionar el directorio donde se encuentre el JDK de java. Una vez lo hemos seleccionado, clicamos en “OK” y volvemos a la pantalla anterior.

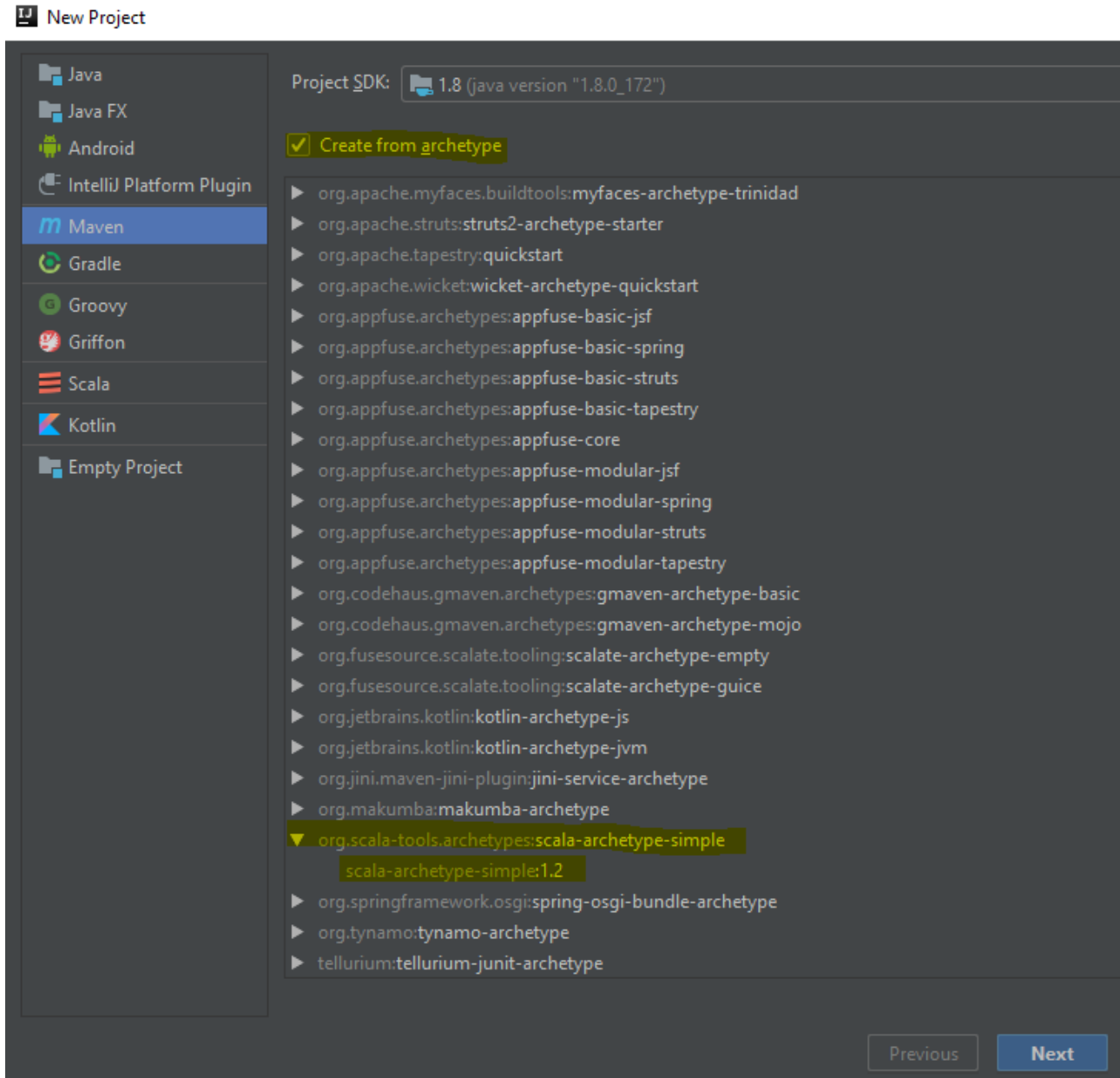
Una vez configurado el JDK, marcamos la casilla “Create from archetype” y descendemos por la lista hasta que localicemos el arquetipo:

“org.scala-tools.archetypes:scala-archetype-simple

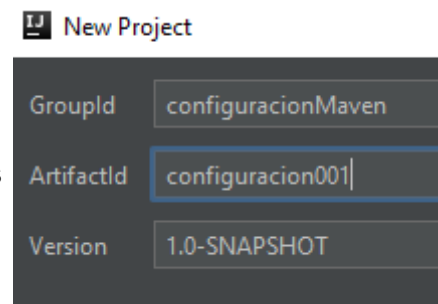
scala-archetype-simple:1.2”



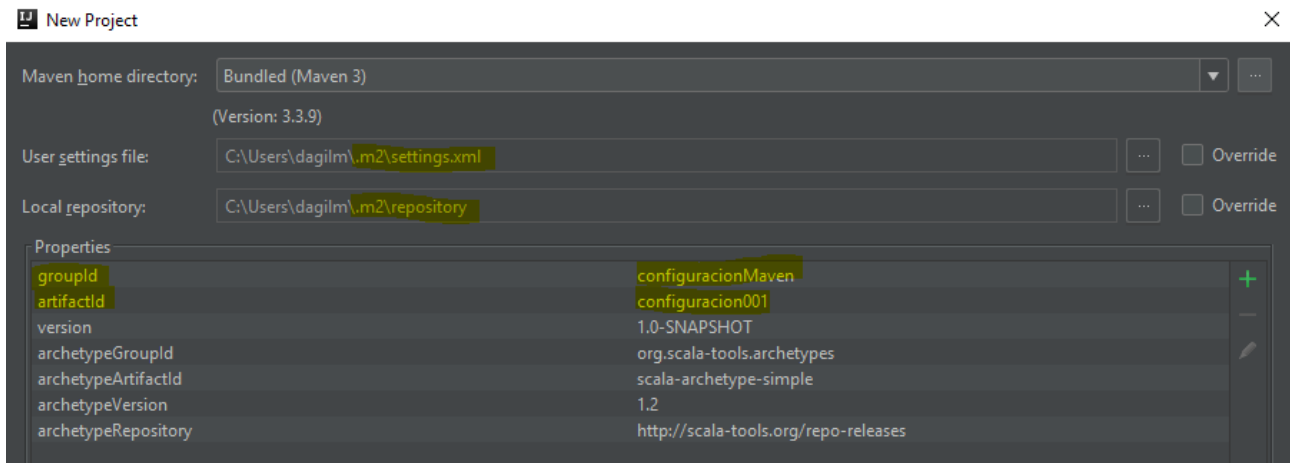
## MANUAL DE INSTALACIÓN DE INTELLIJ – MAVEN



Clicamos en “Next” y rellenamos los campos. En algunos casos, el GroupId será similar a “es.indra.xxx”. ArtifactId también pueden tener una estructura similar, dependiendo del proyecto. Es importante que recordemos estos nombres ya que serán necesarios para actualizar el archivo pom.xml del proyecto. Nos movemos a la siguiente pantalla.

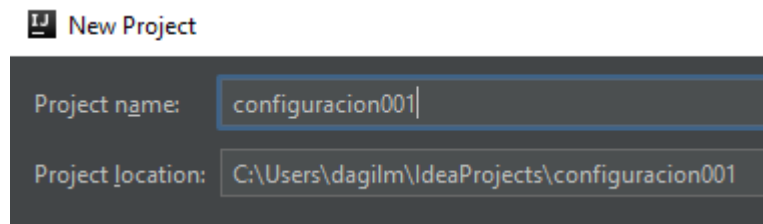


## MANUAL DE INSTALACIÓN DE INTELLIJ – MAVEN

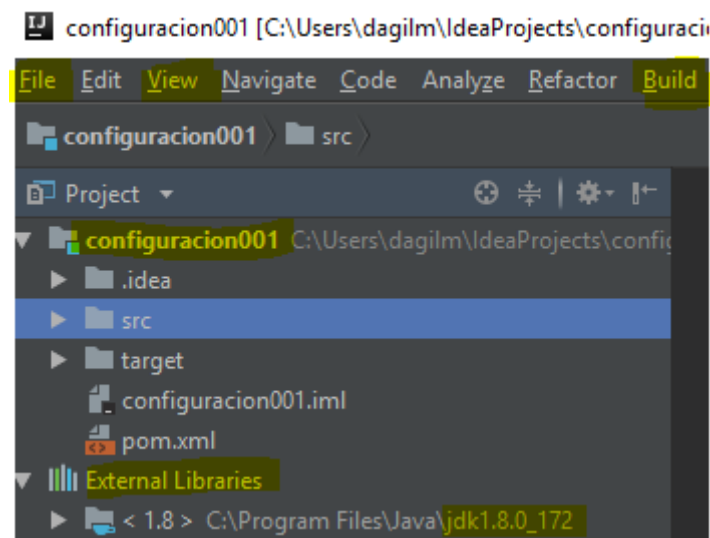


En esta pantalla, comprobamos los datos marcados con amarillo. Nos fijamos especialmente en la localización del archivo settings y el repositorio. Posteriormente tendremos que manipularlos.

Por último, creamos el proyecto al pasar la siguiente pantalla.



Una vez creamos el proyecto, deberíamos ver la siguiente pantalla:



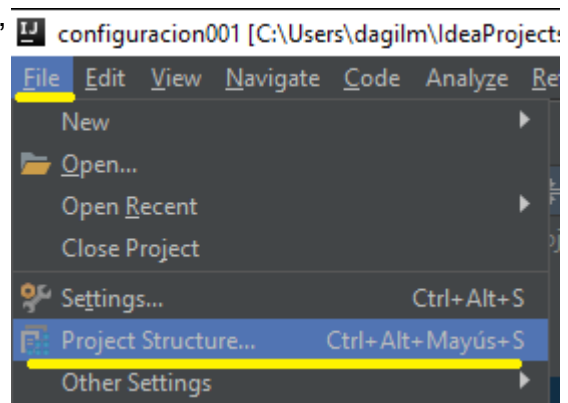
La zona de librerías externas nos muestras las librerías que se encuentran fuera del IntelliJ IDEA y que este utiliza para poder compilar el código correctamente. Dependiendo de cada proyecto necesitaremos más o menos librerías.

## MANUAL DE INSTALACIÓN DE INTELLIJ – MAVEN

En el caso de proyectos escritos en lenguaje Scala, debemos importar otra librería adicional, la SDK.

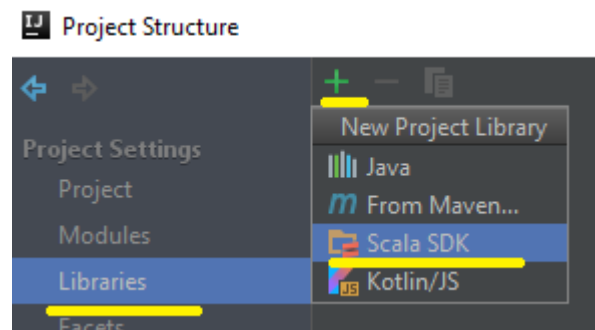
Para ello seguimos los siguientes pasos:

1. Click en “File”
2. Click en “Project Structure”



Aparecerá la ventana de “Project Structure”, la cual cuenta con un (dos) menú(s) lateral(es) de opciones a la izquierda. En ella seleccionamos “Libraries”.

1. En la parte superior de la segunda barra de menús, clicamos en el botón verde de más (+).
2. De las 4 opciones que aparecen en el desplegable, seleccionamos Scala SDK.

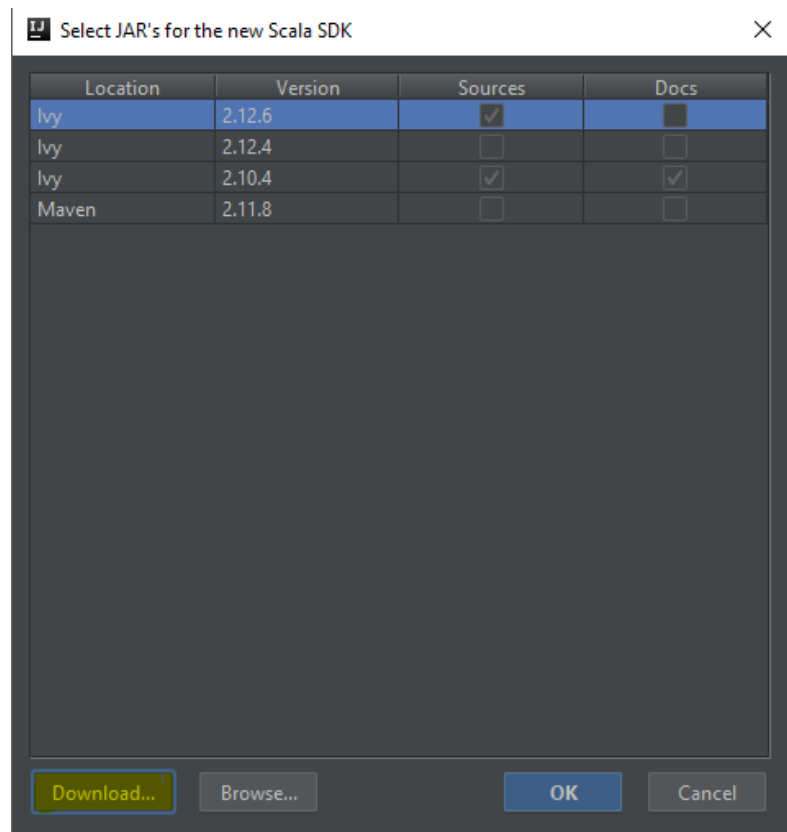


Aparecerá la siguiente pantalla:

Llegados a este punto, existen dos posibilidades. La primera es instalar las librerías a través del IntelliJ IDEA, como se muestra en el manual en los siguientes puntos.

La segunda, es instalar la librería de forma externa (tal y como hemos hecho con Java (JDK) y los binarios de Maven y añadirla al proyecto como una librería externa clicando en el botón “Browse...”

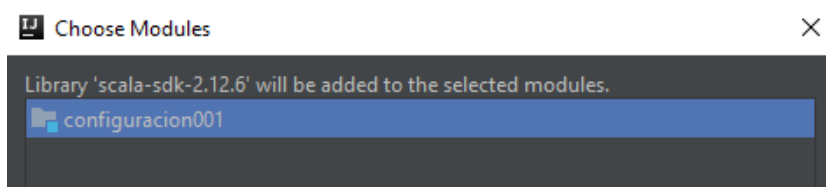
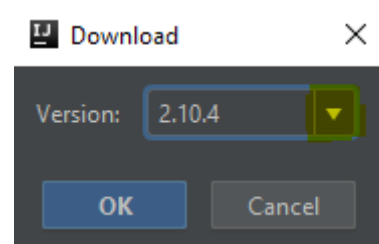
Por tanto, clicamos en el botón “Download...” para instalar internamente las librerías.



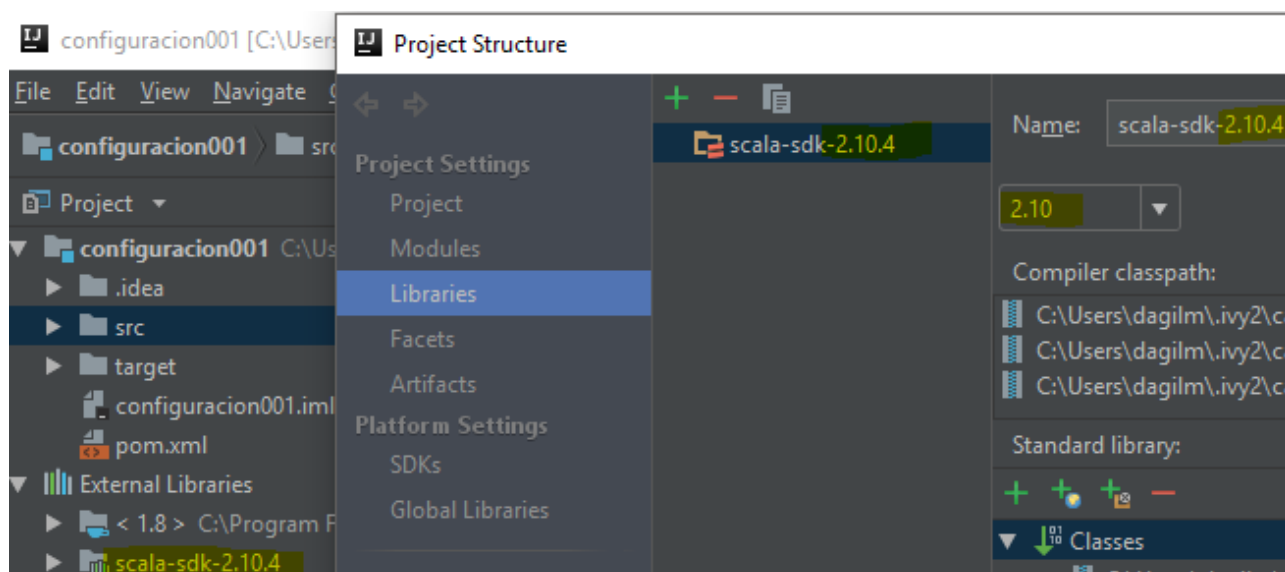
## MANUAL DE INSTALACIÓN DE INTELLIJ – MAVEN

Aparecerá el siguiente cuadro. Si clicamos en la pestaña, aparecerá una lista desplegable desde la cual podremos seleccionar la versión que queramos utilizar de Scala.

Clicamos en “OK” y aparecerá un cuadro que nos mostrará la descarga de Scala. Una vez termine la descarga, aparecerá una nueva ventana y volvemos a clicar en “OK”.

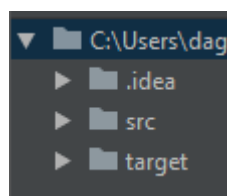


Al volver a la ventana principal de Project Structure, clicamos en “Apply” y deberíamos ver algo así:



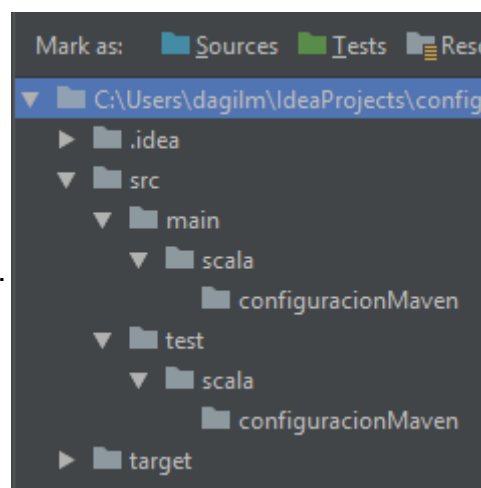
Abrimos de nuevo “File” - “Project Structure” y accedemos el menú de “Modules”.

En la zona de la derecha veremos esta estructura de las carpetas del proyecto.



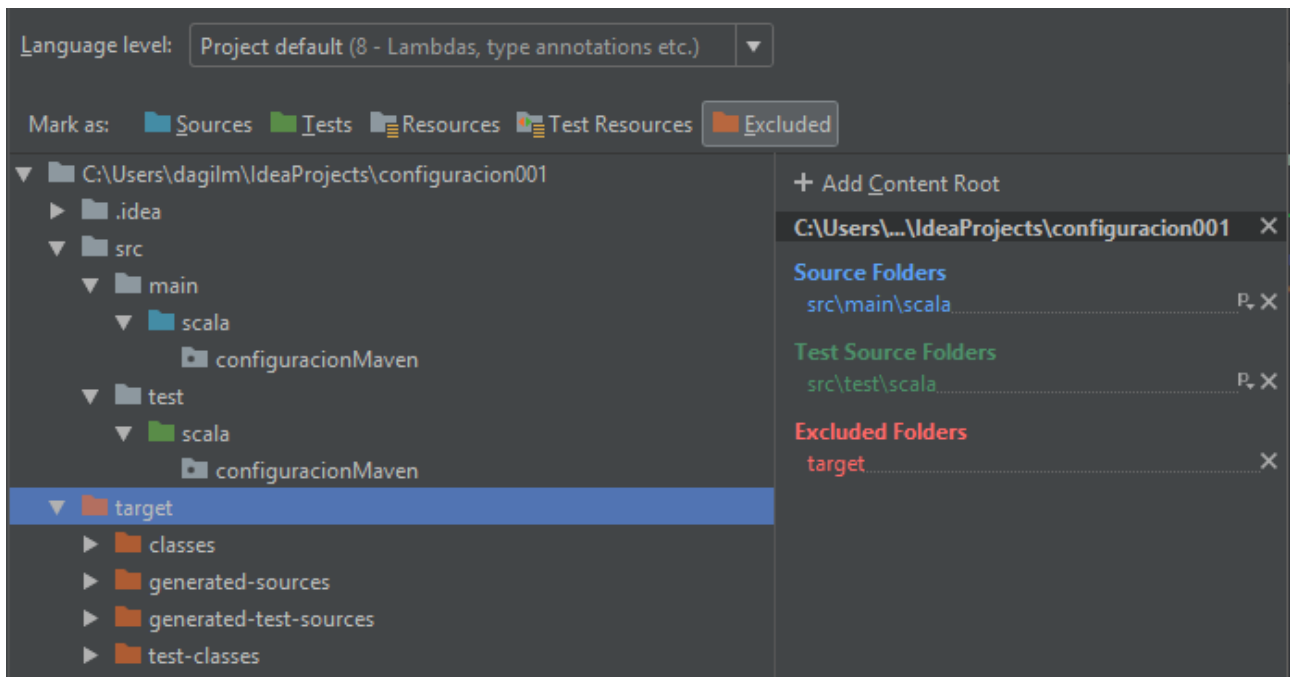
Vamos a asignar a cada una una función.

1. src/main/scala: asignamos Sources (**azul**)
2. src/test/scala: asignamos Test (**verde**)
3. target: asignamos Excluded (**marrón**)



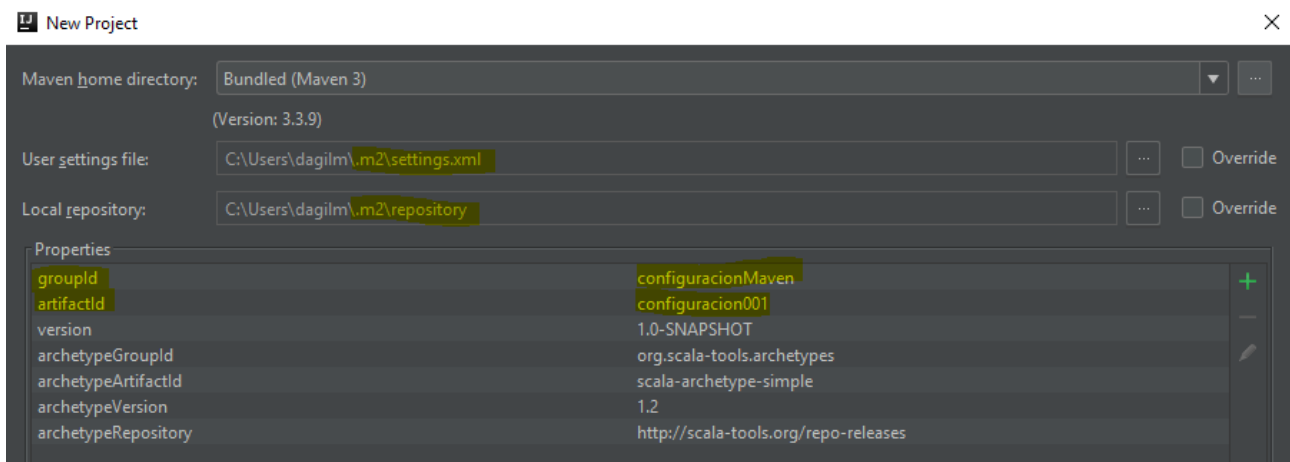
## MANUAL DE INSTALACIÓN DE INTELLIJ – MAVEN

El resultado de la modificación debería mostrar algo similar a esto:



Llegados a este punto, tenemos que copiar el archivo “settings.xml” de la carpeta “C:\Archivos de Programa\apache-maven-3.3.9\conf”. Lo vamos a pegar en dos carpetas:

1. En la carpeta .m2 de tu cuenta de usuario “C:\Users\dagilm\.m2”
2. En la carpeta “repository” que está en la carpeta .m2.



El siguiente paso es sustituir o sobrescribir el archivo “pom.xml” que se encuentra dentro del proyecto. Se puede hacer desde el propio IntelliJ IDEA o desde el sistema de ficheros de Windows con la ayuda de un editor como el Notepad++. El pom.xml de cada proyecto varía. El de este ejemplo es para un proyecto compilado en Scala 2.10.4.

El texto a copiar dentro del archivo “pom.xml” es el siguiente:

## MANUAL DE INSTALACIÓN DE INTELLIJ – MAVEN

```
<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/maven-
v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.fmis.Spread_InterestRate</groupId>
  <artifactId>com.fmis.Spread_InterestRate</artifactId>
  <version>1.0</version>
  <inceptionYear>2008</inceptionYear>
  <properties>
    <scala.version>2.10.4</scala.version>
    <spark.version>1.6.0</spark.version>
    <scala.binary.version>2.10</scala.binary.version>
  </properties>
  <repositories>
    <repository>
      <id>scala-tools.org</id>
      <name>Scala-Tools Maven2 Repository</name>
      <url>http://scala-tools.org/repo-releases</url>
    </repository>
  </repositories>
  <pluginRepositories>
    <pluginRepository>
      <id>scala-tools.org</id>
      <name>Scala-Tools Maven2 Repository</name>
      <url>http://scala-tools.org/repo-releases</url>
    </pluginRepository>
  </pluginRepositories>
  <dependencies>
    <dependency>
      <groupId>org.apache.spark</groupId>
      <artifactId>spark-core_${scala.binary.version}</artifactId>
      <version>${spark.version}</version>
    </dependency>
    <dependency>
```

## MANUAL DE INSTALACIÓN DE INTELLIJ – MAVEN

```
<groupId>org.apache.spark</groupId>
<artifactId>spark-sql_${scala.binary.version}</artifactId>
<version>${spark.version}</version>
</dependency>
<dependency>
  <groupId>org.apache.spark</groupId>
  <artifactId>spark-hive_${scala.binary.version}</artifactId>
  <version>${spark.version}</version>
</dependency>
</dependencies>
<build>
  <sourceDirectory>src/main/scala</sourceDirectory>
  <plugins>
    <plugin>
      <groupId>org.scala-tools</groupId>
      <artifactId>maven-scala-plugin</artifactId>
      <executions>
        <execution>
          <goals>
            <goal>compile</goal>
            <goal>testCompile</goal>
          </goals>
        </execution>
      </executions>
      <configuration>
        <scalaVersion>${scala.version}</scalaVersion>
        <args>
          <arg>-target:jvm-1.7</arg>
        </args>
      </configuration>
    </plugin>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
```



## MANUAL DE INSTALACIÓN DE INTELIJ – MAVEN

```
<artifactId>maven-compiler-plugin</artifactId>

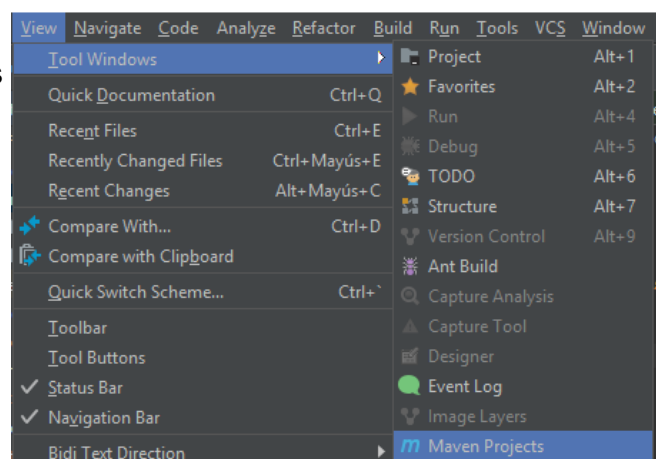
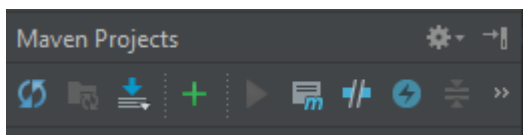
<configuration>
  <source>1.7</source>
  <target>1.7</target>
</configuration>
</plugin>
</plugins>
</build>
<reporting>
<plugins>
<plugin>
  <groupId>org.scala-tools</groupId>
  <artifactId>maven-scala-plugin</artifactId>
  <configuration>
    <scalaVersion>${scala.version}</scalaVersion>
  </configuration>
</plugin>
</plugins>
</reporting>
</project>
```

Una vez hemos reemplazado el “pom.xml” del proyecto, vamos a proceder a la compilación con Maven.

### Compilación del proyecto en Maven

Antes de compilar el proyecto en Maven, debemos abrir la consola de Maven Projects dentro del IntelliJ IDEA. Para ello, clicamos en “View” - “Tool Windows” - “Maven Projects”.

Aparecerá la siguiente barra de menús a la derecha:

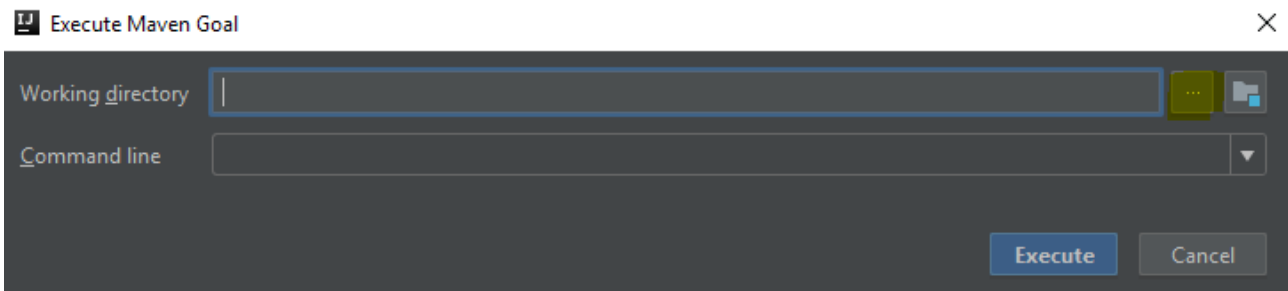


## MANUAL DE INSTALACIÓN DE INTELLIJ – MAVEN

Clicamos en “Execute Maven Goal”



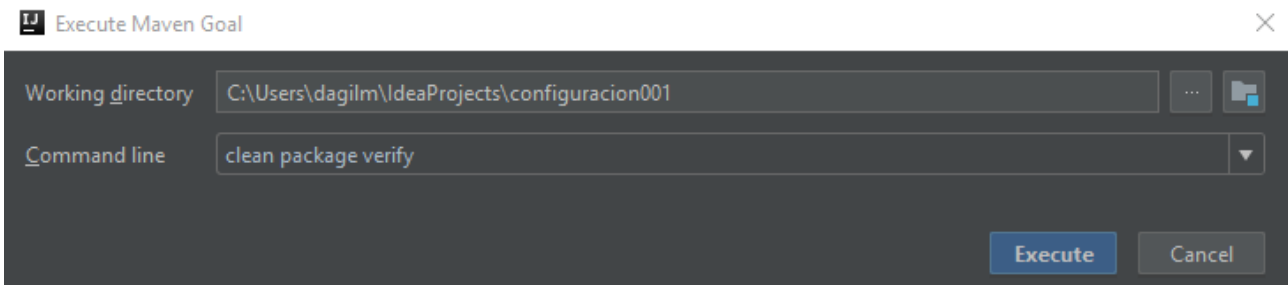
Se desplegará una ventana en la cual se nos pedirá el “Working Directory”.



Seleccionamos el botón “...” para ubicar nuestro proyecto maven. En el caso de este ejemplo, el proyecto se encuentra en “C:\Users\dagilm\IdeaProjects\configuracion001”.

Una vez seleccionado el proyecto, ejecutamos los siguientes comandos:

1. clean
2. package
3. verify



## Posibles errores

### Error de lectura del archivo settings.xml

En caso de que aparezca un error de lectura en el archivo “settings.xml”, debemos copiar el archivo de settings.xml que nos ha facilitado el gestor del proyecto en el block de notas de Windows y desde allí, volver a copiarlo con el Notepad++ en los directorios correspondientes para que se eliminen los caracteres de texto invisibles, como “/n”, “/r” u otros.