



Big Data

(Apache Spark)

Instructor: Trong-Hop Do

April 24th 2021

S³Lab

Smart Software System Laboratory



“Big data is at the foundation of all the megatrends that are happening today, from social to mobile to cloud to gaming.”

– Chris Lynch, Vertica Systems

Apache Spark Installation on Windows



Install Java 8 or Later

- To install Apache Spark on windows, you would need Java 8 or later version hence download the Java version from Oracle and install it on your system.
- <https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html>

Windows x64

166.79 MB



jdk-8u271-windows-x64.exe

Apache Spark Installation on Windows

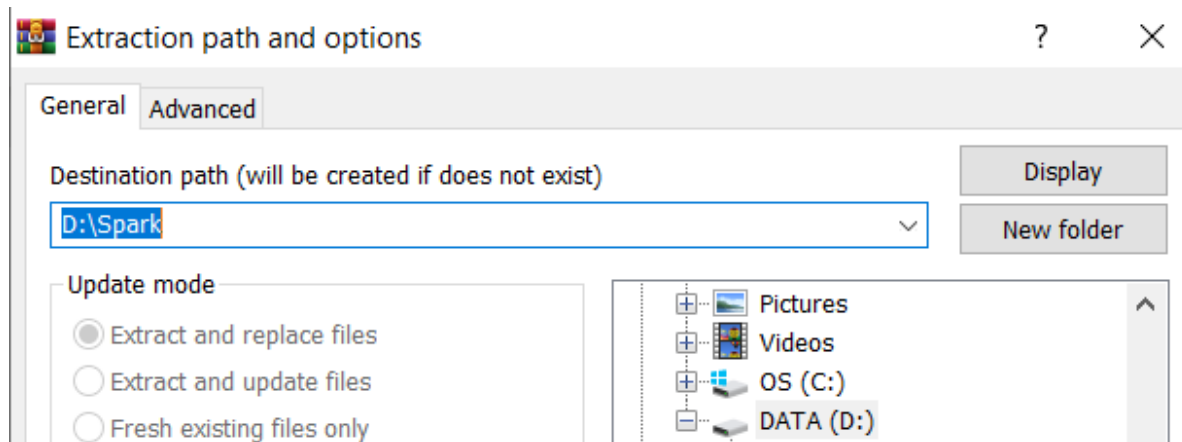
- Download Apache spark
- <https://spark.apache.org/downloads.html>

Download Apache Spark™

1. Choose a Spark release:
2. Choose a package type:
3. Download Spark: [spark-3.0.1-bin-hadoop2.7.tgz](#)
4. Verify this release using the 3.0.1 [signatures](#), [checksums](#) and [project release KEYS](#).

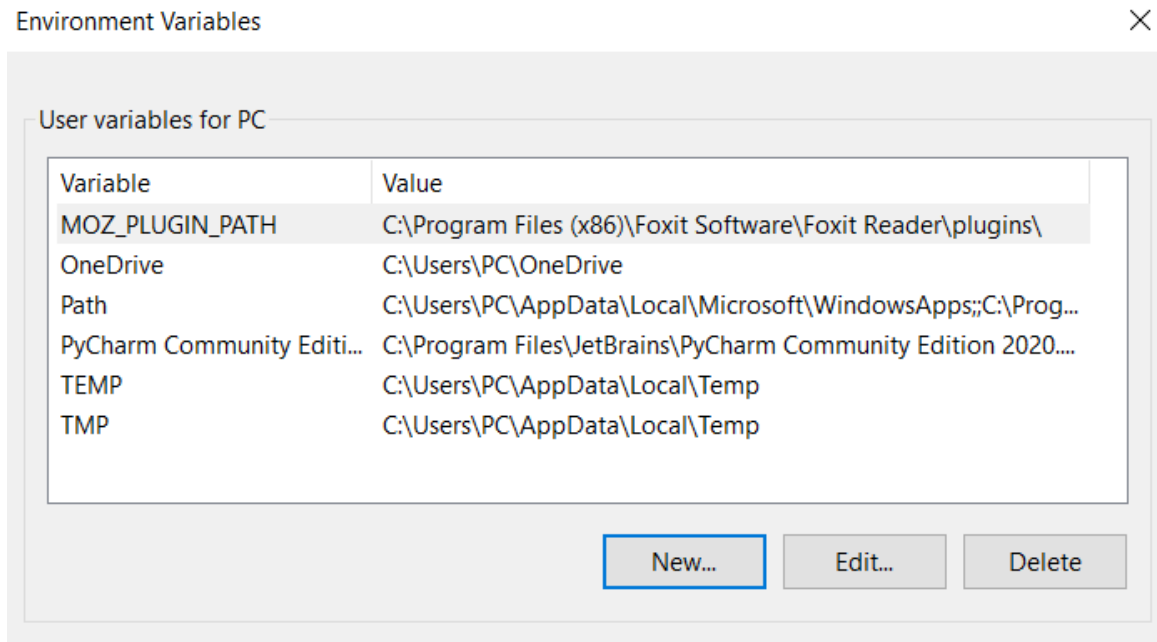
Apache Spark Installation on Windows

- Extract the zip file to any folder

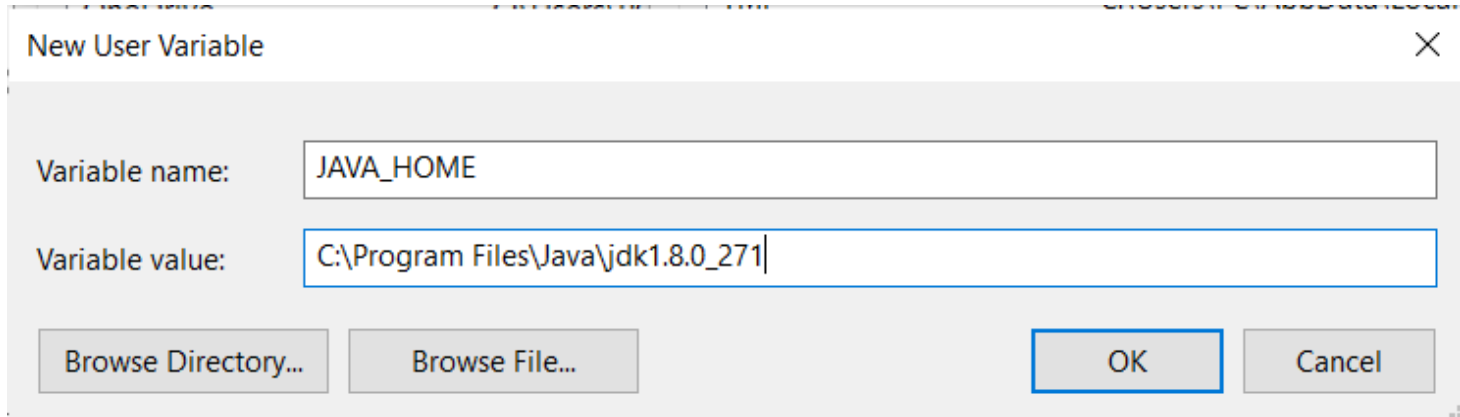


Environment Variables Setting

- Open System Environment Variables window and select Environment Variables.



Environment Variables Setting



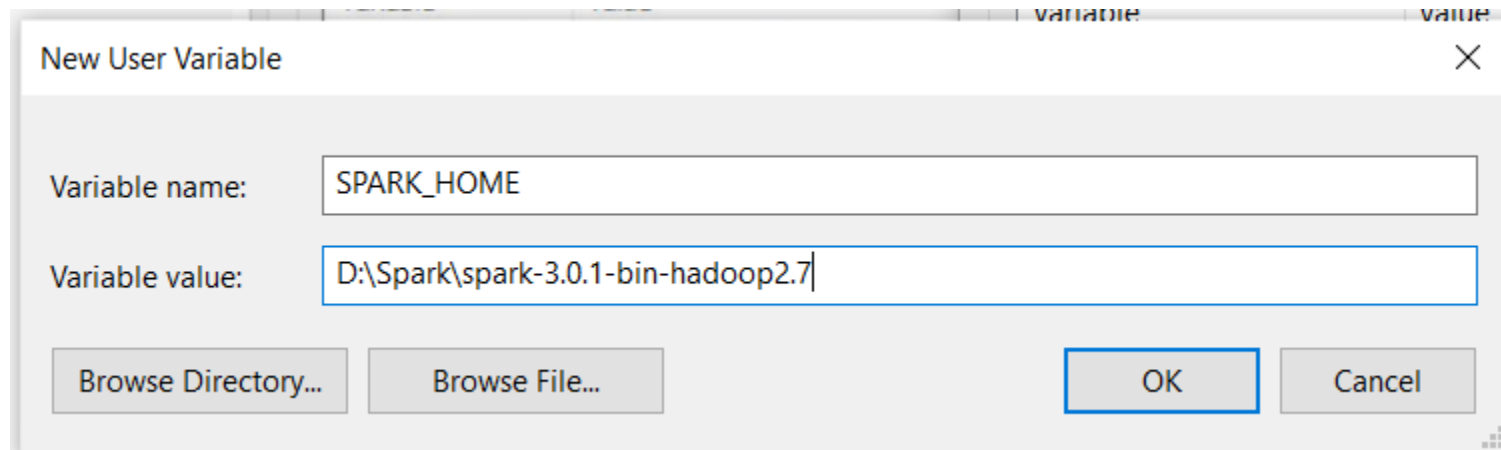
New User Variable

Variable name: JAVA_HOME

Variable value: C:\Program Files\Java\jdk1.8.0_271

Browse Directory... Browse File... OK Cancel

Apache Spark Installation on Windows

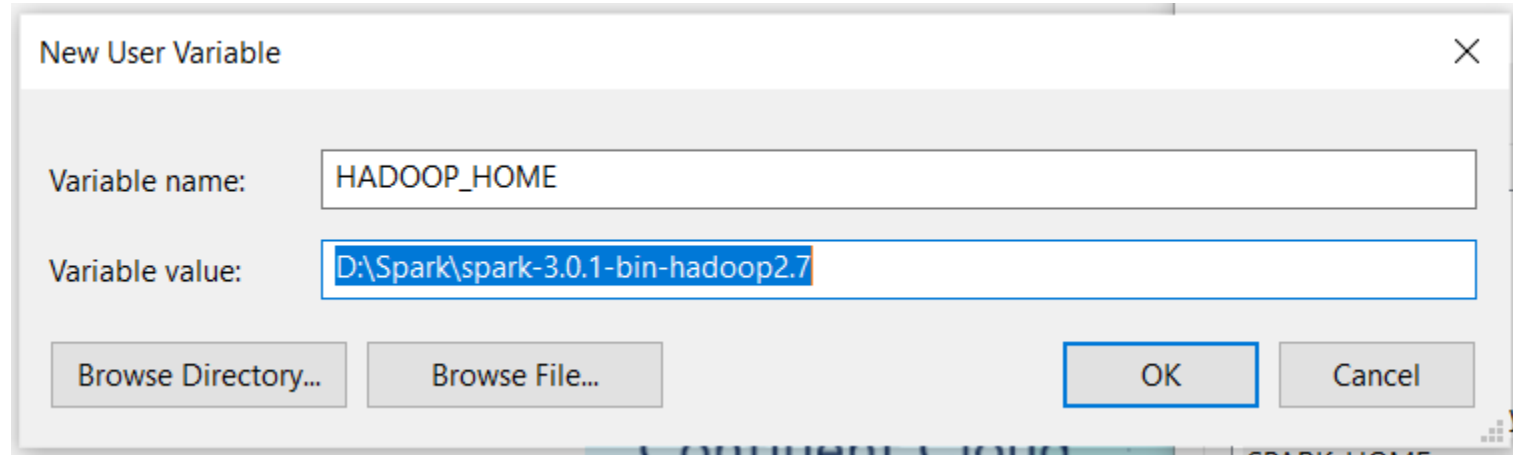


The screenshot shows a Windows 'New User Variable' dialog box. The title bar reads 'New User Variable' with a close button (X) on the right. The dialog contains two text input fields: 'Variable name:' with the text 'SPARK_HOME' and 'Variable value:' with the text 'D:\Spark\spark-3.0.1-bin-hadoop2.7'. Below these fields are four buttons: 'Browse Directory...', 'Browse File...', 'OK', and 'Cancel'. The 'OK' button is highlighted with a blue border. The background of the dialog is light gray.

	variable	value
Variable name:	SPARK_HOME	
Variable value:	D:\Spark\spark-3.0.1-bin-hadoop2.7	

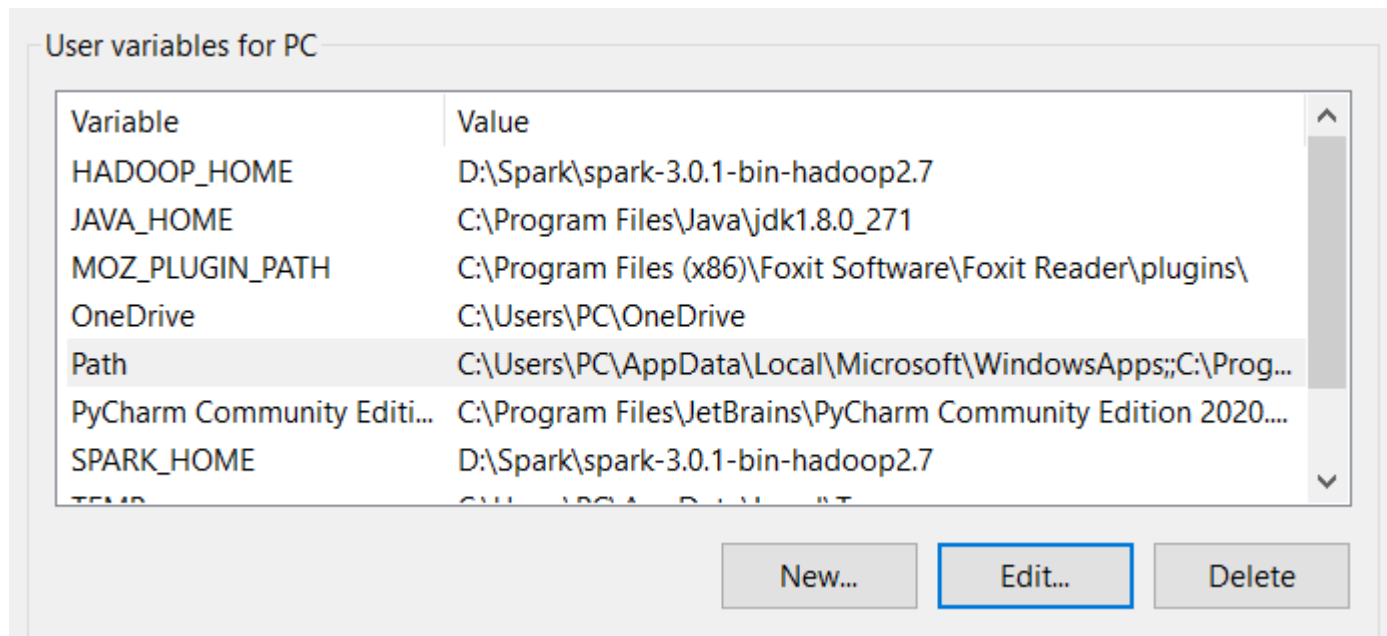
Buttons: Browse Directory..., Browse File..., OK, Cancel

Apache Spark Installation on Windows



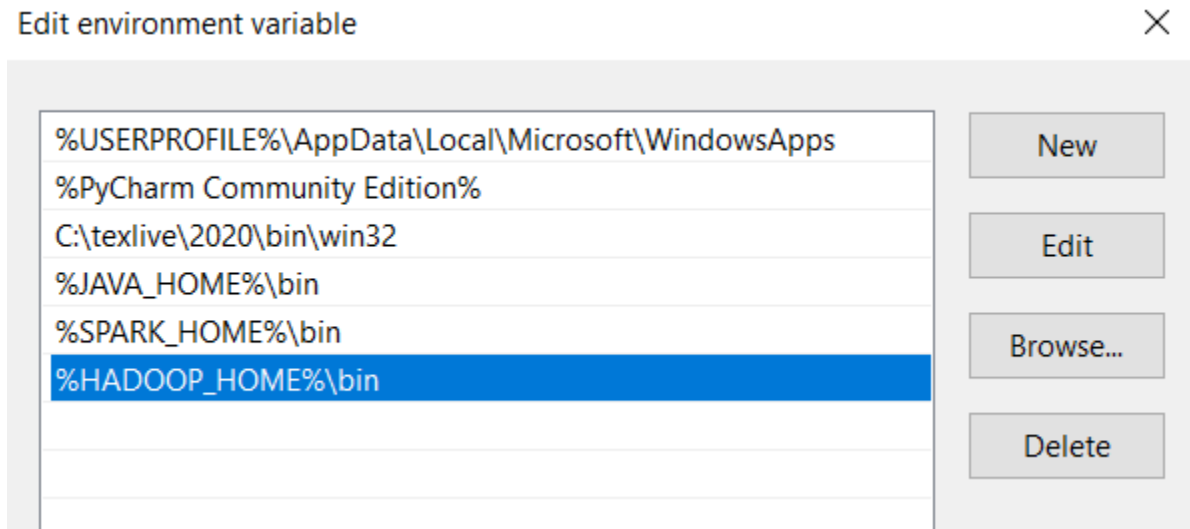
Apache Spark Installation on Windows

- Now Edit the PATH variable



Apache Spark Installation on Windows

- Add Spark, Java, and Hadoop bin location by selecting New option.



```
C:\Users\PC>spark-shell
20/12/07 16:50:25 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
Spark context Web UI available at http://192.168.56.1:4040
Spark context available as 'sc' (master = local[*], app id = local-1607334630777).
Spark session available as 'spark'.
Welcome to

      /_/_/
     / \ V _ V _ / \ / \ ' \
    /__/. _/\_,_/_/_/_/_/_\
     /_/_/
version 3.0.1

Using Scala version 2.12.10 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_271)
Type in expressions to have them evaluated.
Type :help for more information.

scala> 20/12/07 16:50:45 WARN ProcfsMetricsGetter: Exception when trying to compute pagesize, as a result reporting of ProcessTree metrics is stopped

scala>
```

Test apache Spark shell

- Type any command to test Spark shell

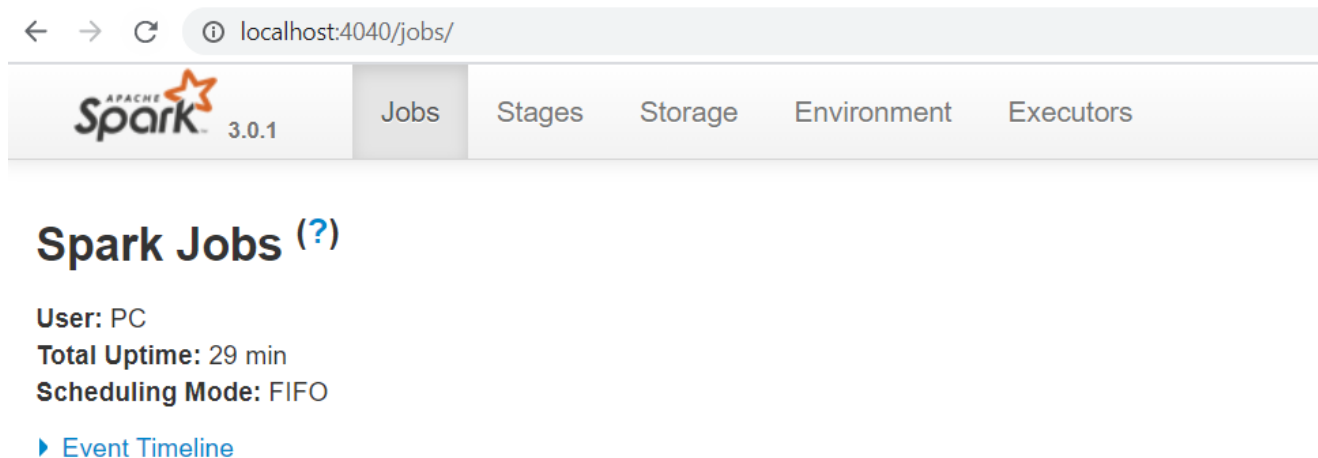
```
scala> spark.version  
res0: String = 3.0.1
```

```
scala> 3+3  
res1: Int = 6
```

```
scala>
```

Web UI on Windows

- Open a web browser and navigate to <http://localhost:4040/>



Spark Setup with Scala and Run in IntelliJ

- download IntelliJ IDEA community edition
- <https://www.jetbrains.com/idea/download/#section=windows>

Download IntelliJ IDEA

Windows

Mac

Linux

Ultimate

For web and enterprise development

Download

.exe ▼

Free 30-day trial

Community

For JVM and Android development

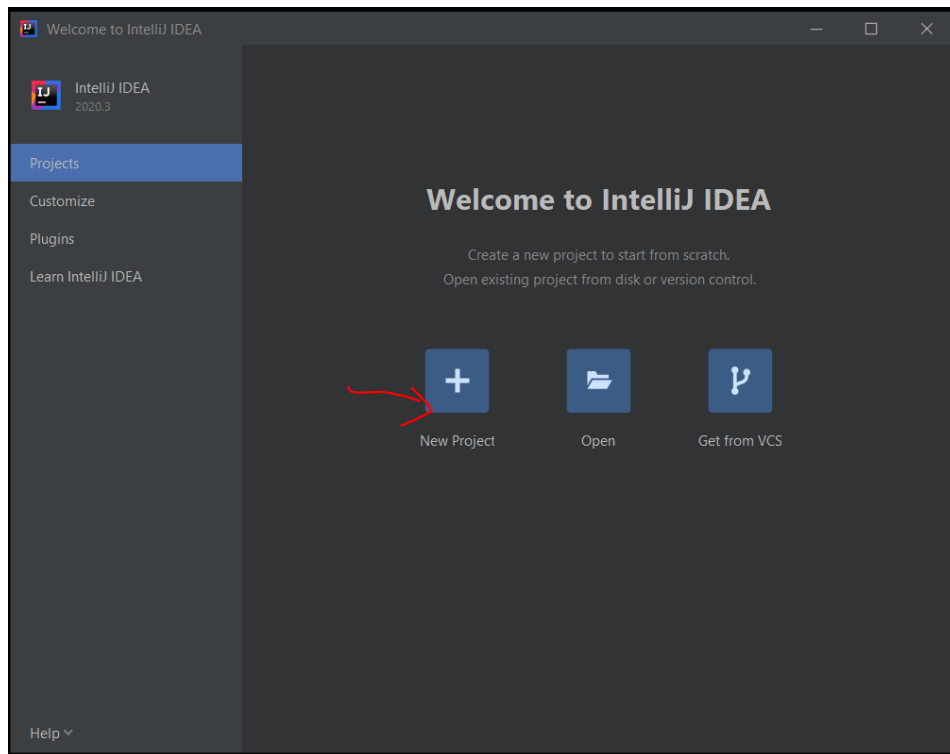
Download

.exe ▼

Free, open-source

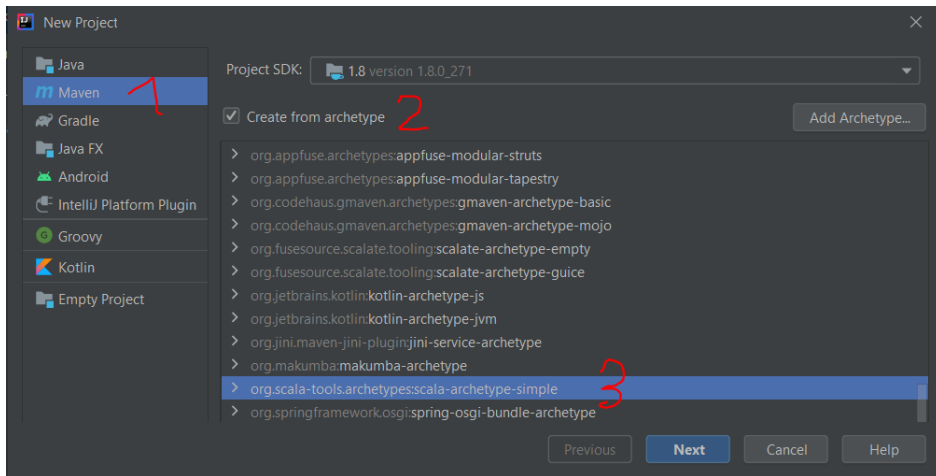


Create a Scala project In IntelliJ

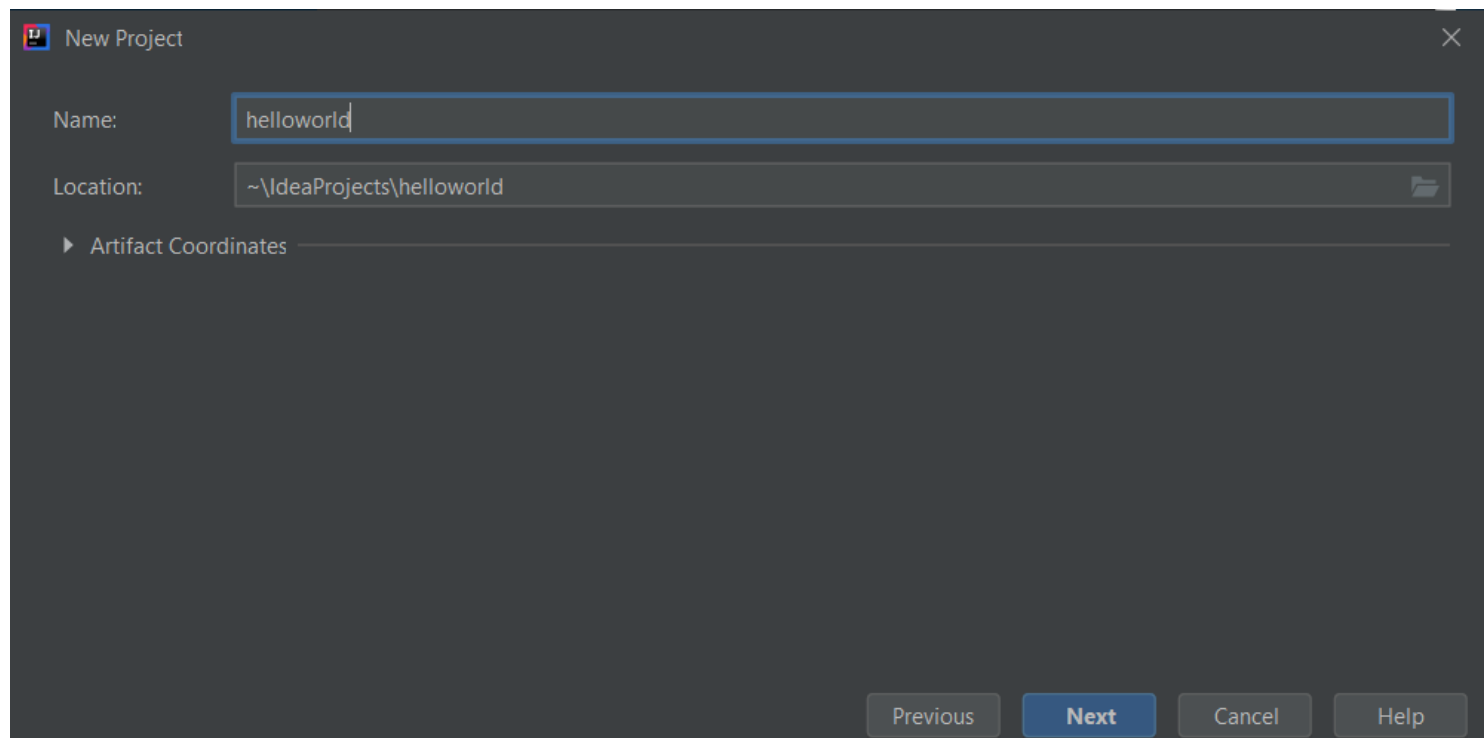


Create a Scala project In IntelliJ

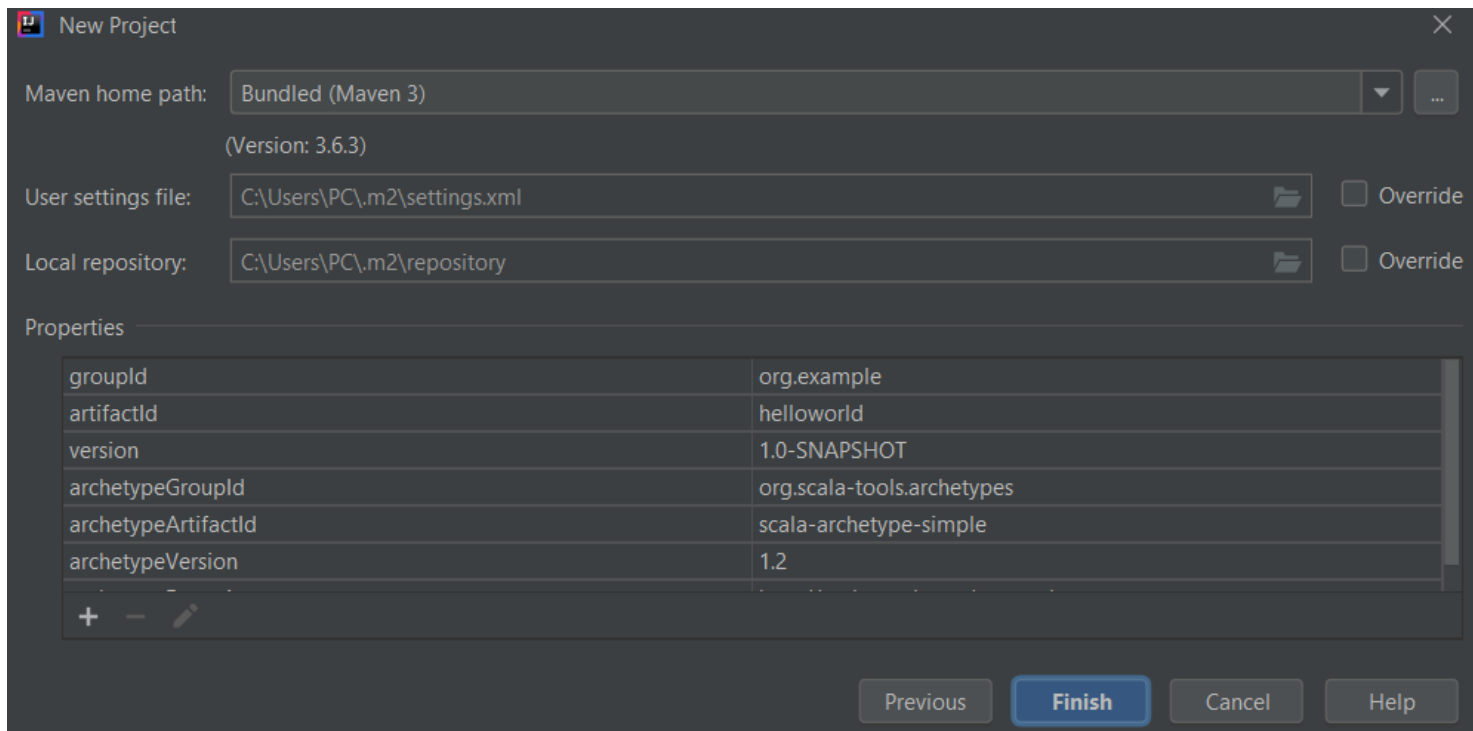
- Select Maven from the left panel
 - Check option Create from archetype
 - Select org.scala-tools.archetypes:scala-archetypes-simple.
- The archetype is a kind of templates that creates the right directory structure and downloads the required default dependencies. Since we have selected Scala archetypes, it downloads all Scala dependencies and enables IntelliJ to write Scala code.



Create a Scala project In IntelliJ



Create a Scala project In IntelliJ



The image shows the 'New Project' dialog box in IntelliJ IDEA. It has a dark theme. At the top, it says 'New Project' with a close button. Below that, there are three main sections: 'Maven home path', 'User settings file', and 'Local repository'. Each has a text field and a folder icon button. The 'Maven home path' is set to 'Bundled (Maven 3)' with a dropdown arrow and a version '(Version: 3.6.3)' below it. The 'User settings file' is set to 'C:\Users\PC\.m2\settings.xml' with an 'Override' checkbox. The 'Local repository' is set to 'C:\Users\PC\.m2\repository' with an 'Override' checkbox. Below these is a 'Properties' section with a table of project properties. At the bottom, there are four buttons: 'Previous', 'Finish' (highlighted in blue), 'Cancel', and 'Help'.

Maven home path: Bundled (Maven 3) (Version: 3.6.3)

User settings file: C:\Users\PC\.m2\settings.xml ☐ Override

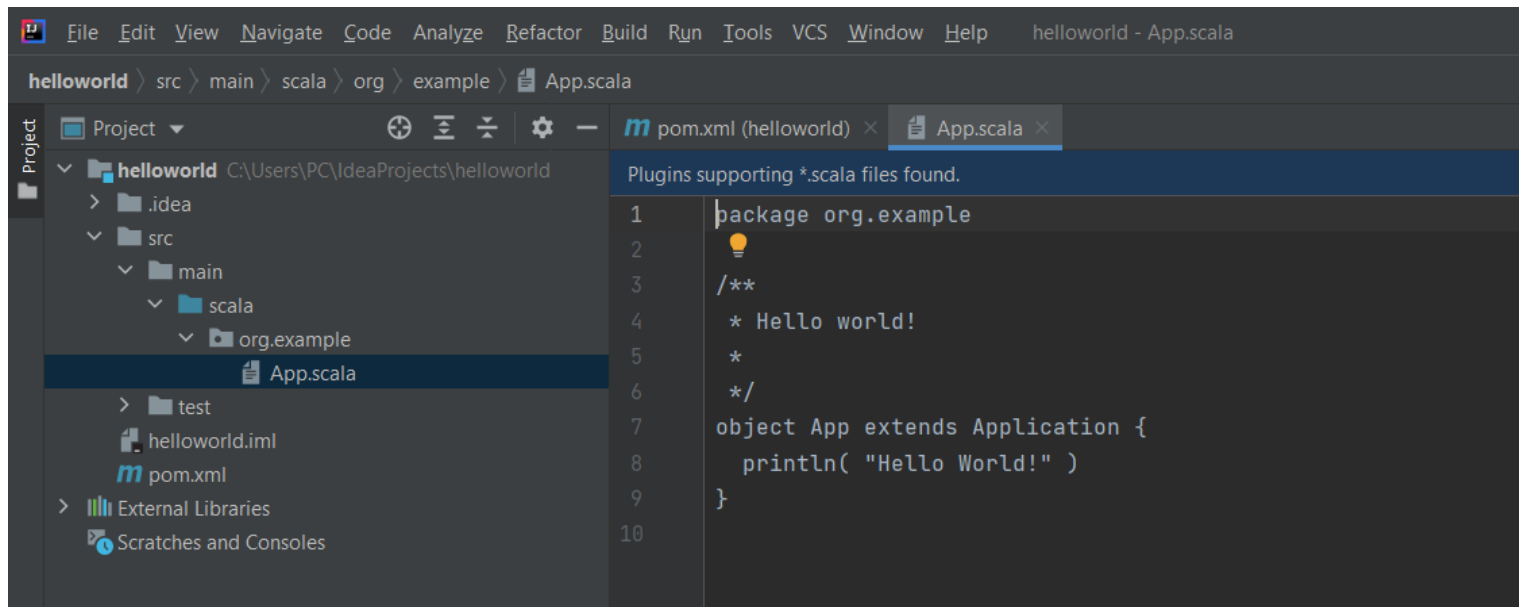
Local repository: C:\Users\PC\.m2\repository ☐ Override

Properties

groupId	org.example
artifactId	helloworld
version	1.0-SNAPSHOT
archetypeGroupId	org.scala-tools.archetypes
archetypeArtifactId	scala-archetype-simple
archetypeVersion	1.2

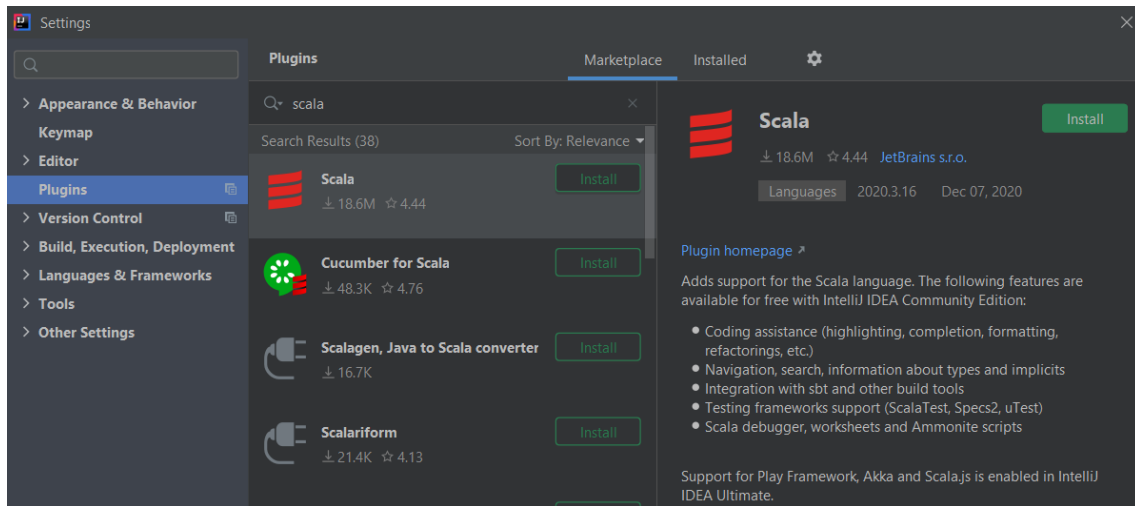
Previous Finish Cancel Help

Create a Scala project In IntelliJ

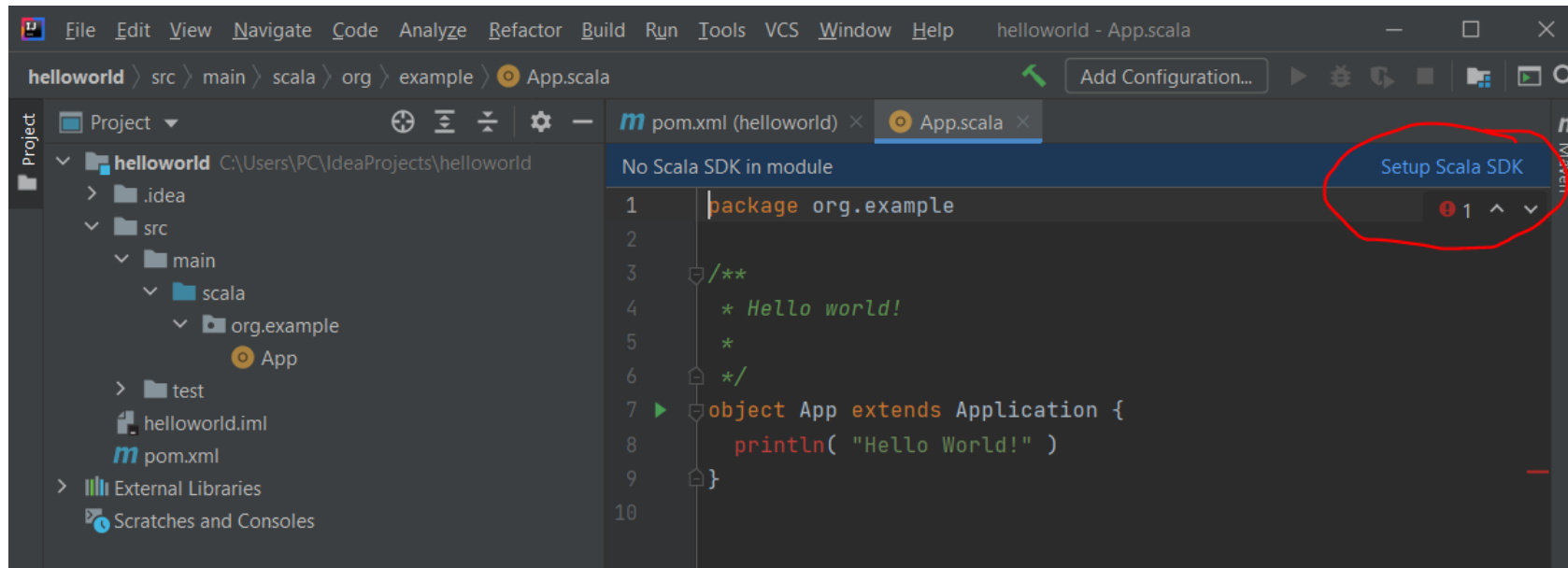


Install Scala Plugin

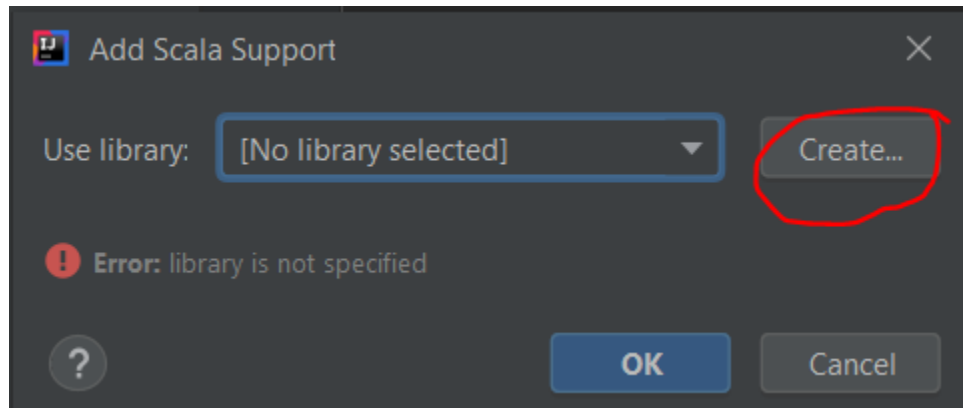
- Open File > Settings (or using shot keys Ctrl + Alt + s)
- Select the Plugins option from the left panel. This brings you Feature panel.
- Click on Install to install the Scala plugin.
- After plugin install, restart the IntelliJ IDE.



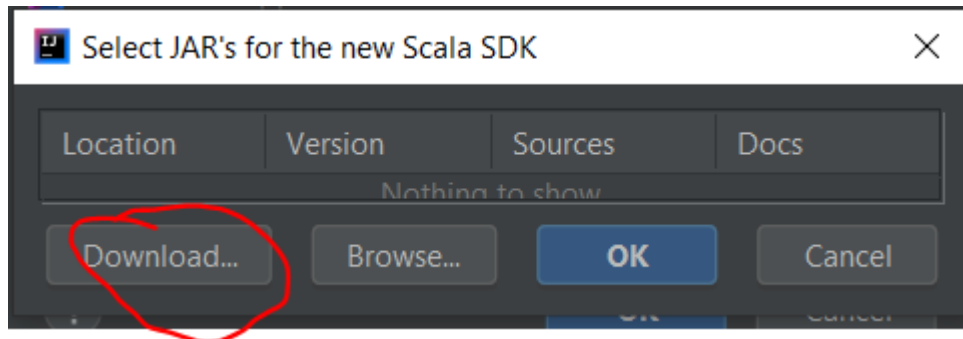
Setup Scala SDK



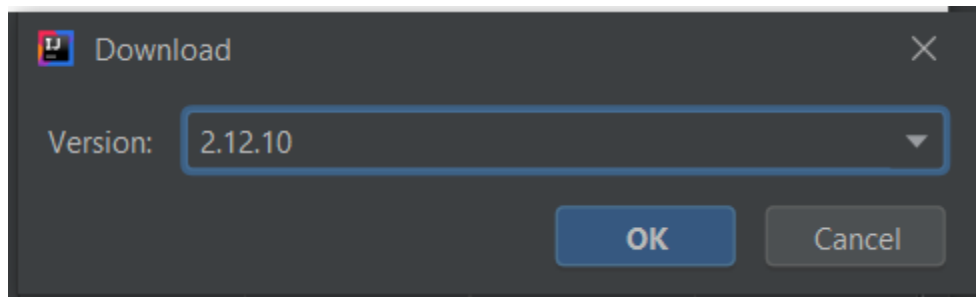
Setup Scala SDK



Setup Scala SDK

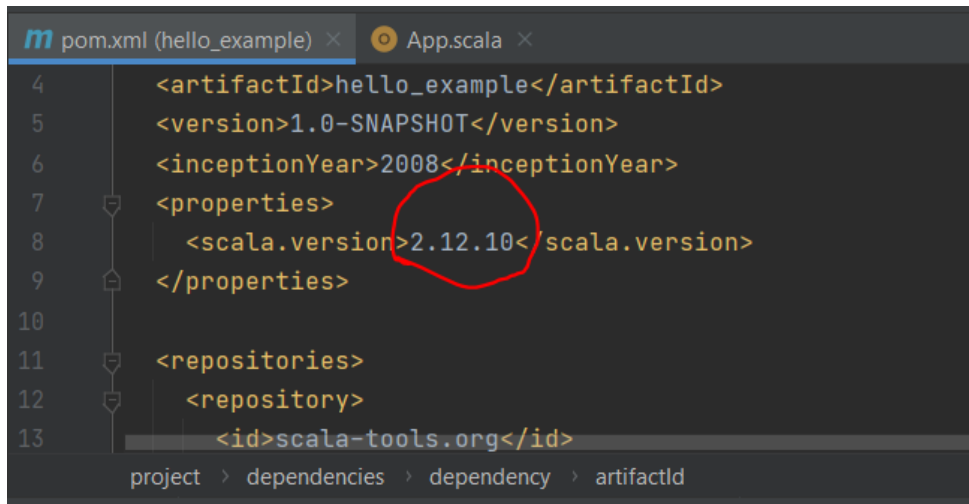


Setup Scala SDK



Make changes to pom.xml file

- First, change the Scala version to the latest version

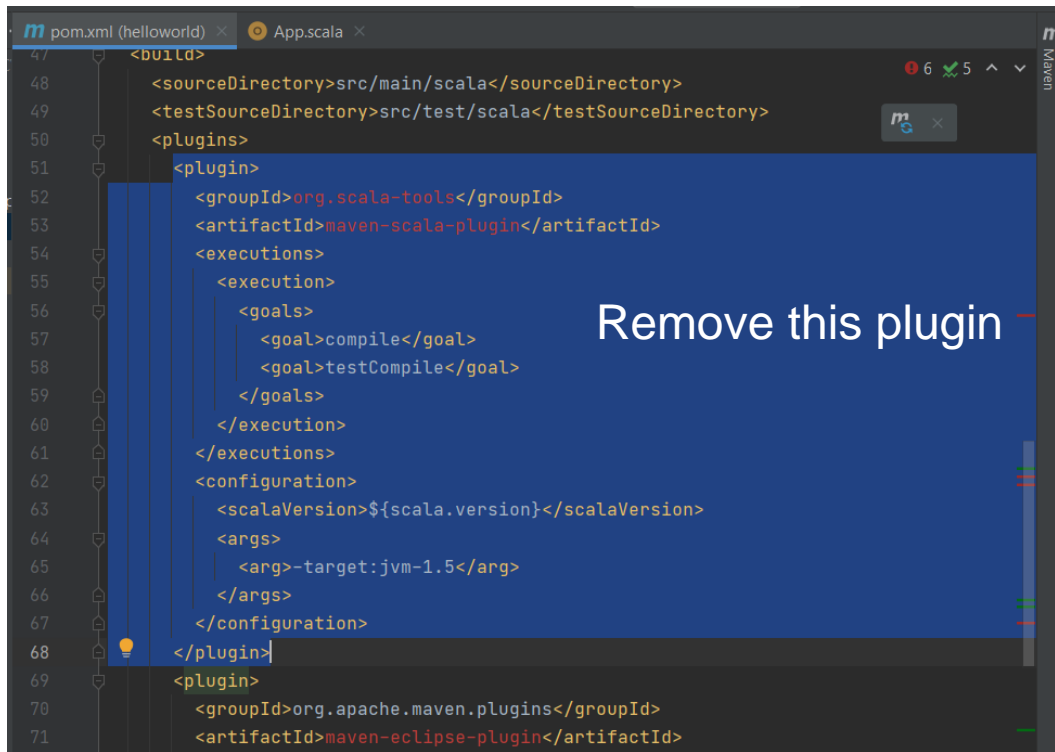


The screenshot shows an IDE with two tabs: 'pom.xml (hello_example)' and 'App.scala'. The 'pom.xml' tab is active, displaying XML code. The code is as follows:

```
4 <artifactId>hello_example</artifactId>
5 <version>1.0-SNAPSHOT</version>
6 <inceptionYear>2008</inceptionYear>
7 <properties>
8   <scala.version>2.12.10</scala.version>
9 </properties>
10
11 <repositories>
12 <repository>
13   <id>scala-tools.org</id>
```

The value '2.12.10' in the `<scala.version>2.12.10</scala.version>` tag on line 8 is circled in red. The bottom of the IDE shows a breadcrumb trail: 'project > dependencies > dependency > artifactId'.

Make changes to pom.xml file

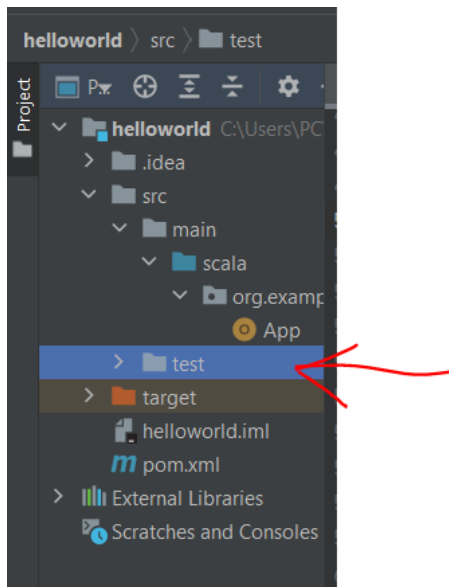


The screenshot shows an IDE window with two tabs: 'pom.xml (helloworld)' and 'App.scala'. The 'pom.xml' tab is active, displaying XML code. A blue rectangular overlay is positioned over the 'maven-scala-plugin' entry in the 'plugins' section. The text 'Remove this plugin' is written in white on the blue background. The XML code is as follows:

```
47 <build>
48   <sourceDirectory>src/main/scala</sourceDirectory>
49   <testSourceDirectory>src/test/scala</testSourceDirectory>
50   <plugins>
51     <plugin>
52       <groupId>org.scala-tools</groupId>
53       <artifactId>maven-scala-plugin</artifactId>
54       <executions>
55         <execution>
56           <goals>
57             <goal>compile</goal>
58             <goal>testCompile</goal>
59           </goals>
60         </execution>
61       </executions>
62       <configuration>
63         <scalaVersion>${scala.version}</scalaVersion>
64         <args>
65           <arg>-target:jvm-1.5</arg>
66         </args>
67       </configuration>
68     </plugin>
69     <plugin>
70       <groupId>org.apache.maven.plugins</groupId>
71       <artifactId>maven-eclipse-plugin</artifactId>
```

Delete Unnecessary Files

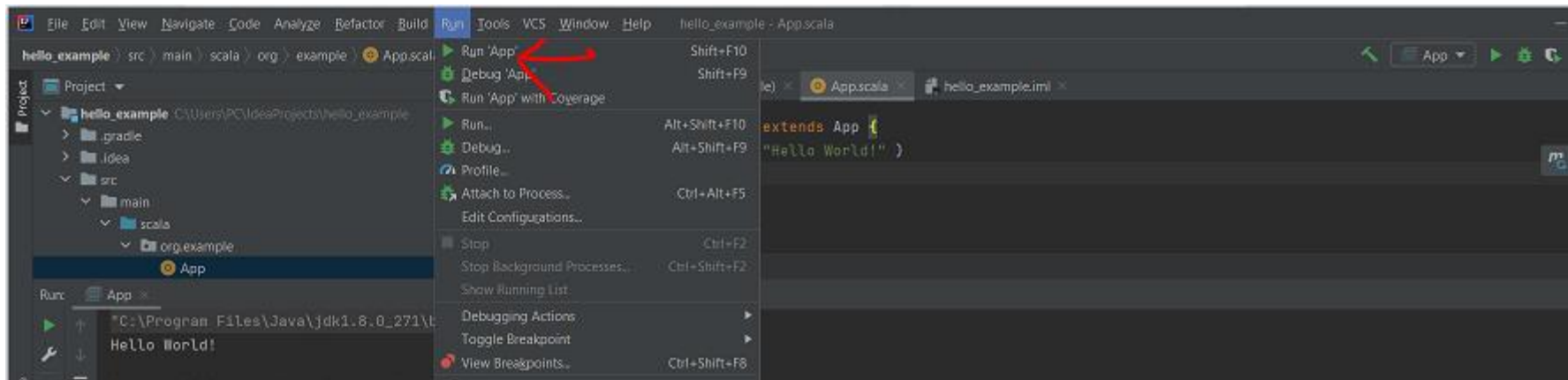
- Delete src/test



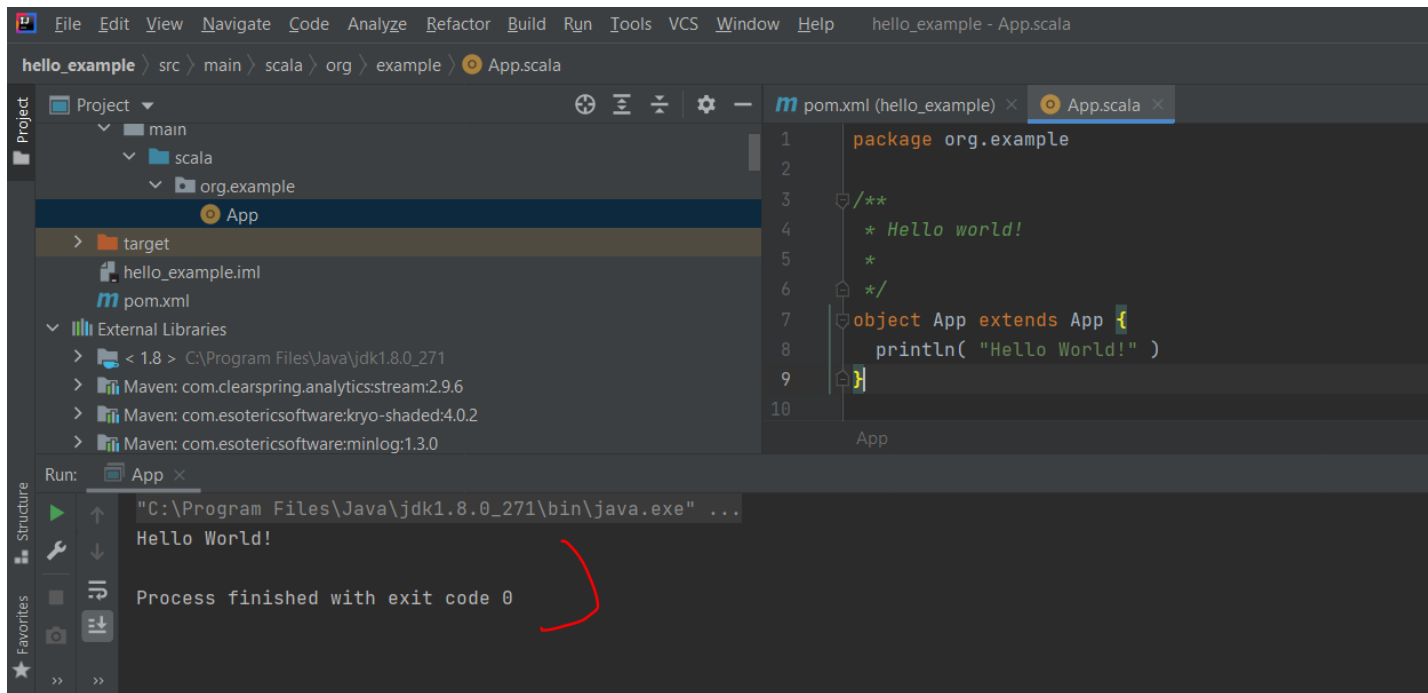
Add Spark Dependencies to Maven pom.xml File

```
<dependency>
  <groupId>org.apache.spark</groupId>
  <artifactId>spark-core_2.12</artifactId>
  <version>3.0.1</version>
  <scope>compile</scope>
</dependency>
<dependency>
  <groupId>org.apache.spark</groupId>
  <artifactId>spark-sql_2.12</artifactId>
  <version>3.0.1</version>
  <scope>compile</scope>
</dependency>
```

Run the application

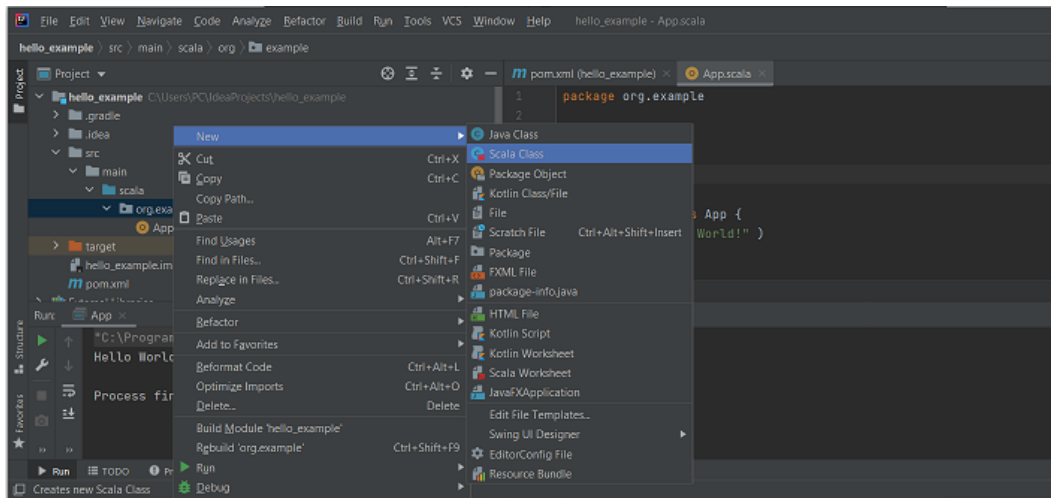


Run the application



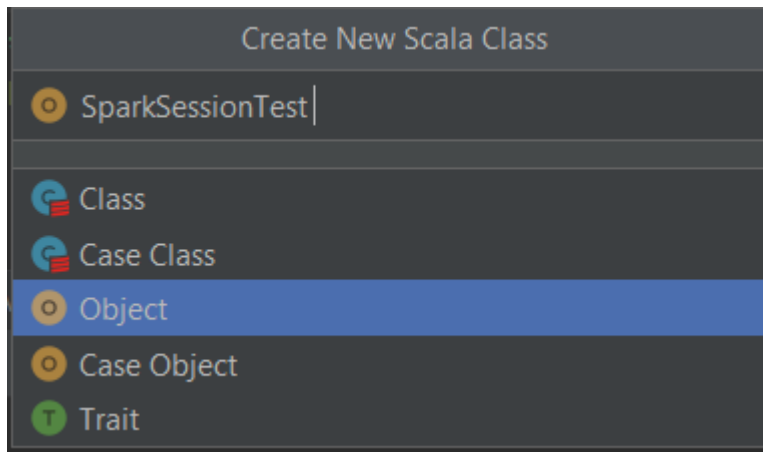
Create new Spark application

- Right click org.example -> New -> Scala Class



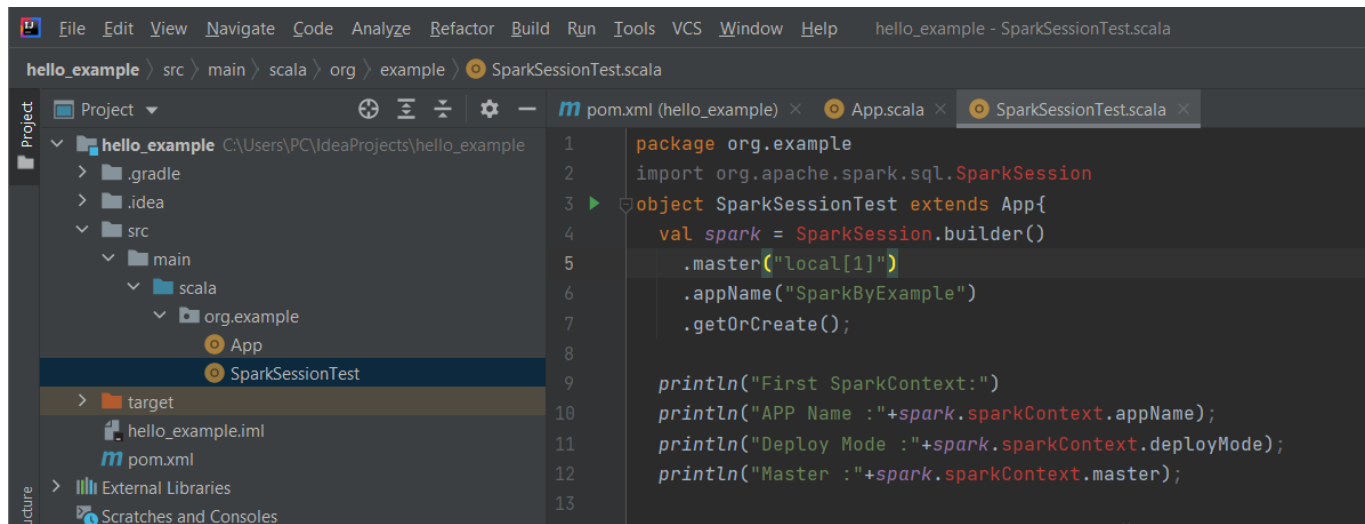
Create new Spark application

- Enter “SparkSessionTest”
- Choose “Object”



Create new Spark application

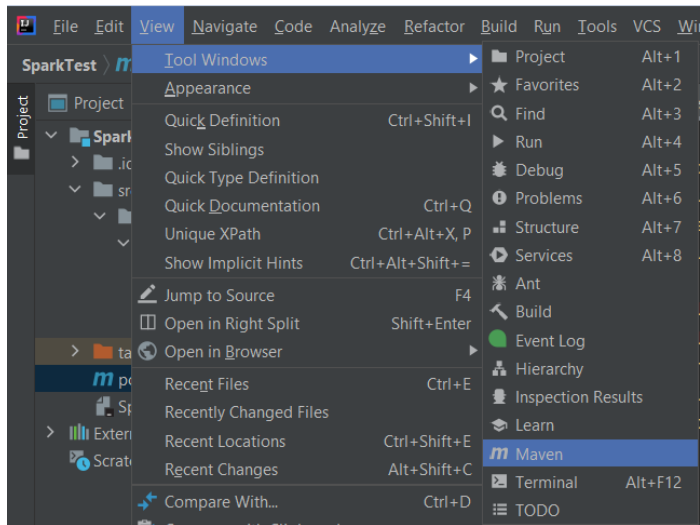
- Copy and past the code



The screenshot shows an IDE window for a project named 'hello_example'. The left sidebar displays the project structure: 'hello_example' (C:\Users\PC\IdeaProjects\hello_example) contains '.gradle', '.idea', 'src', and 'target'. The 'src' directory contains 'main', which contains 'scala', which contains 'org.example'. The 'org.example' directory contains 'App' and 'SparkSessionTest'. The 'target' directory contains 'hello_example.iml' and 'pom.xml'. The 'pom.xml' file is selected, and its content is displayed in the main editor. The code in 'pom.xml' is as follows:

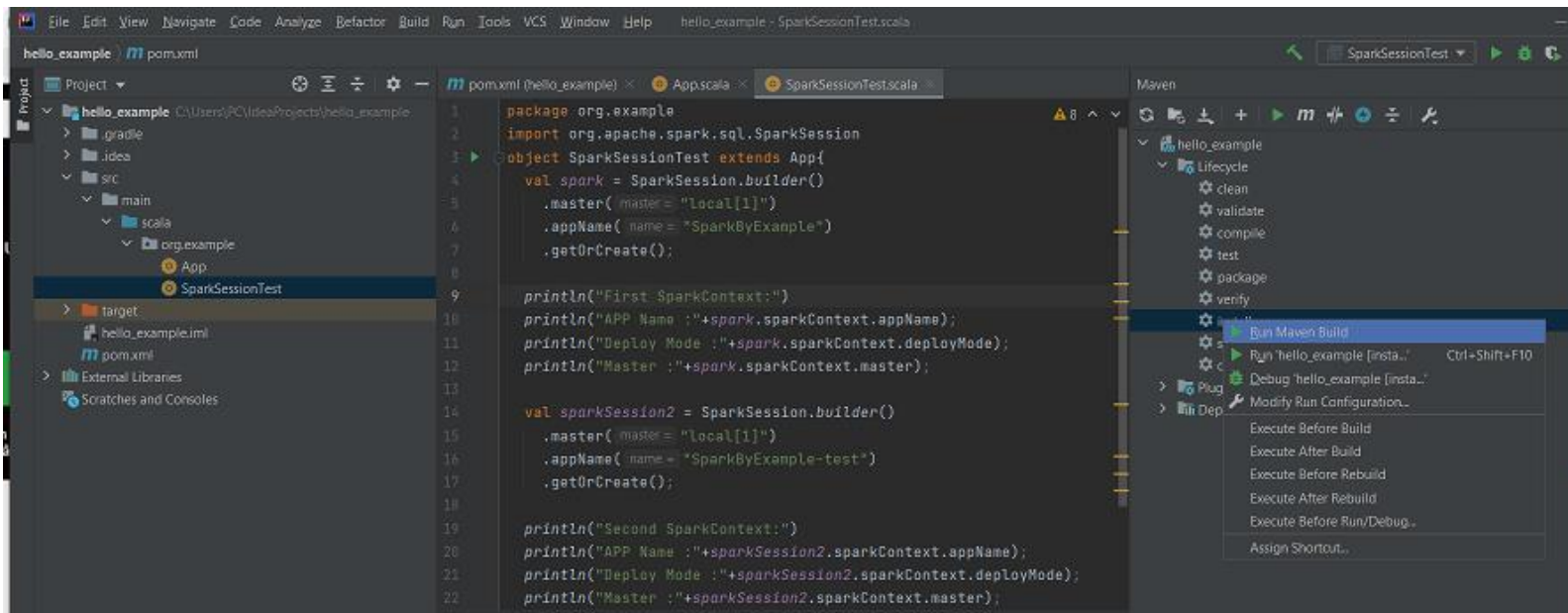
```
1 package org.example
2 import org.apache.spark.sql.SparkSession
3 object SparkSessionTest extends App{
4     val spark = SparkSession.builder()
5         .master("local[1]")
6         .appName("SparkByExample")
7         .getOrCreate();
8
9     println("First SparkContext:")
10    println("APP Name :"+spark.sparkContext.appName);
11    println("Deploy Mode :"+spark.sparkContext.deployMode);
12    println("Master :"+spark.sparkContext.master);
13 }
```

Run Maven Build



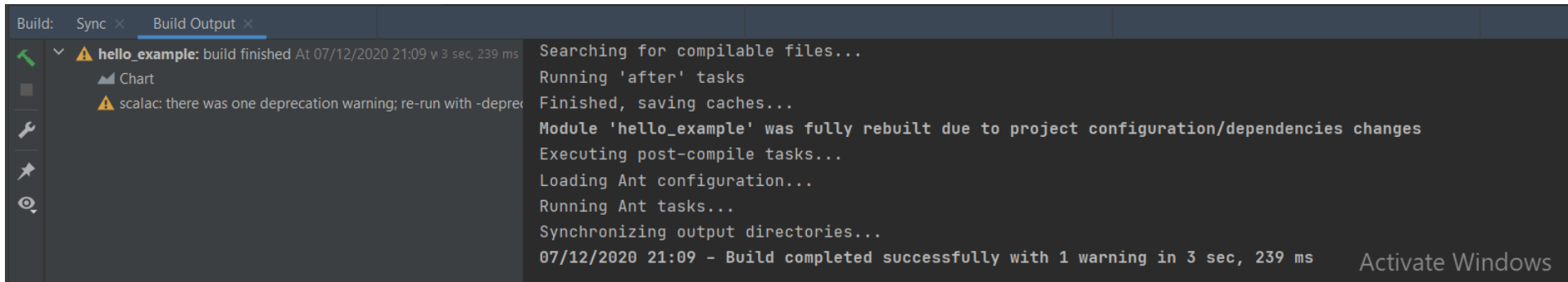
Run Maven Build

- Maven -> Lifecycle > install, right-click, and select Run Maven Build.



Create new Spark application

- Or just Load Maven Changes

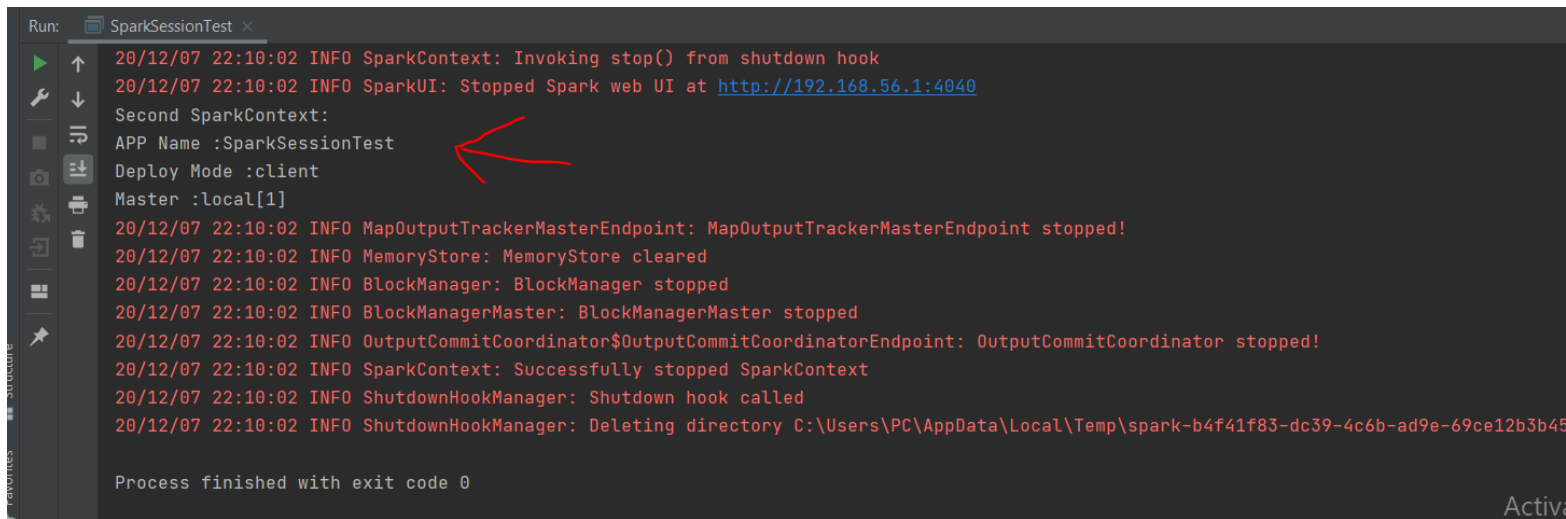


The screenshot shows the 'Build Output' window of an IDE. The title bar indicates 'Build: Sync x Build Output x'. The main content area displays the build log for 'hello_example', which finished at 07/12/2020 21:09 v 3 sec, 239 ms. A warning icon and text indicate a deprecation warning from scalac: 'scalac: there was one deprecation warning; re-run with -depre'. The log details the build process: 'Searching for compilable files...', 'Running \'after\' tasks', 'Finished, saving caches...', 'Module \'hello_example\' was fully rebuilt due to project configuration/dependencies changes', 'Executing post-compile tasks...', 'Loading Ant configuration...', 'Running Ant tasks...', and 'Synchronizing output directories...'. The final line states: '07/12/2020 21:09 - Build completed successfully with 1 warning in 3 sec, 239 ms'. On the left, a sidebar contains icons for a file explorer, a chart, and a settings gear. The bottom right corner of the IDE window has an 'Activate Windows' watermark.

```
Build: Sync x Build Output x
hello_example: build finished At 07/12/2020 21:09 v 3 sec, 239 ms
  Chart
  scalac: there was one deprecation warning; re-run with -depre
Searching for compilable files...
Running 'after' tasks
Finished, saving caches...
Module 'hello_example' was fully rebuilt due to project configuration/dependencies changes
Executing post-compile tasks...
Loading Ant configuration...
Running Ant tasks...
Synchronizing output directories...
07/12/2020 21:09 - Build completed successfully with 1 warning in 3 sec, 239 ms
Activate Windows
```

Create new Spark application

- Run the application



```
Run: SparkSessionTest x
20/12/07 22:10:02 INFO SparkContext: Invoking stop() from shutdown hook
20/12/07 22:10:02 INFO SparkUI: Stopped Spark web UI at http://192.168.56.1:4040
Second SparkContext:
APP Name :SparkSessionTest
Deploy Mode :client
Master :local[1]
20/12/07 22:10:02 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
20/12/07 22:10:02 INFO MemoryStore: MemoryStore cleared
20/12/07 22:10:02 INFO BlockManager: BlockManager stopped
20/12/07 22:10:02 INFO BlockManagerMaster: BlockManagerMaster stopped
20/12/07 22:10:02 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
20/12/07 22:10:02 INFO SparkContext: Successfully stopped SparkContext
20/12/07 22:10:02 INFO ShutdownHookManager: Shutdown hook called
20/12/07 22:10:02 INFO ShutdownHookManager: Deleting directory C:\Users\PC\AppData\Local\Temp\spark-b4f41f83-dc39-4c6b-ad9e-69ce12b3b45

Process finished with exit code 0
```



Cảm ơn đã theo dõi

Chúng tôi hy vọng cùng nhau đi đến thành công.