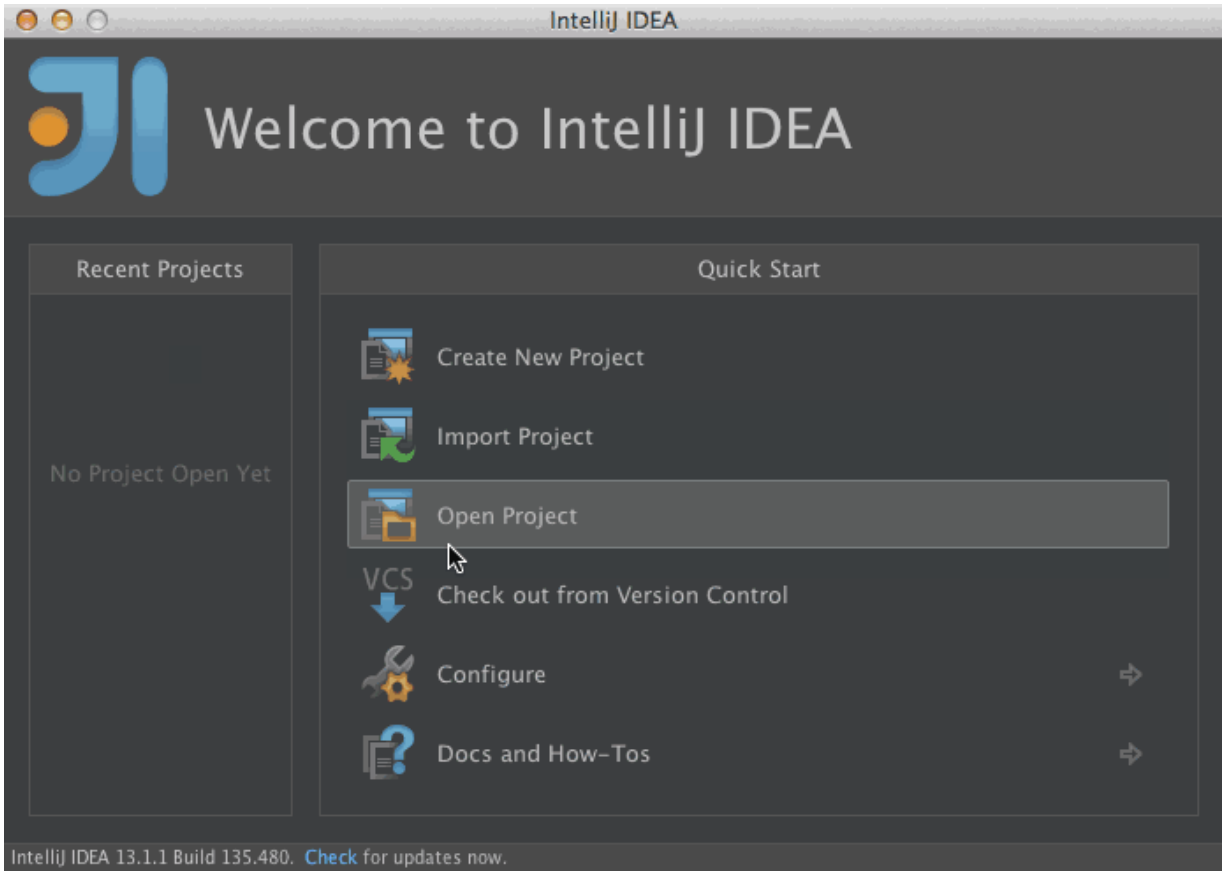
IntelliJ IDEA Tutorial

1. Download IntelliJ IDEA Community Edition

IntelliJ IDEA Community Edition is an open-source version of IntelliJ IDEA, a premier IDE for Java, Scala and other JVM-based programming languages. You can download it from the official website.

2. Install the Scala plugin

When starting Intellij, a Welcome Screen shows up. The first step before creating or opening a Scala project is to install the Scala plugin. For that, go to the bottom right of the Welcome Screen and choose **Configure** → **Plugins** → **Browse JetBrains Plugins**. If such Welcome Screen doesn't appear, go to **Preferences (Settings)** → **Plugins**.



Note that Scala plugin requires restart to complete installation. Thus, **restart Intellij** before following the next steps.

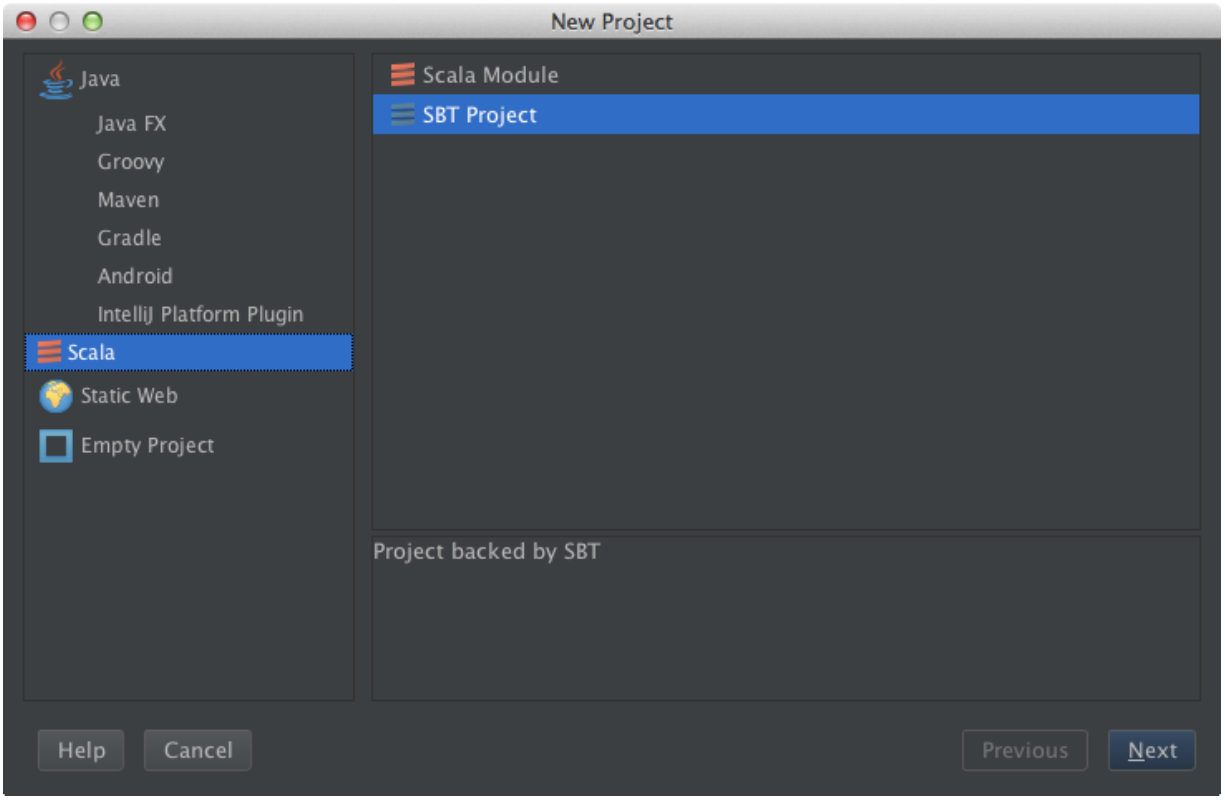
For more information on this step, visit the official documentation.

3. Setup the JDK

From the welcome screen, go to **Configure** → **Project defaults** → **Project structure** and add the JDK. The JDK will be automatically detected, otherwise find the path where it's stored and select it from the file picker. More detailed instructions are here.

4. Creating a project

The easiest way to create a project is to use the Project Wizard. To use it, click **Create New Project** on the Welcome Screen, then select **Scala**, and finally **SBT Project**.

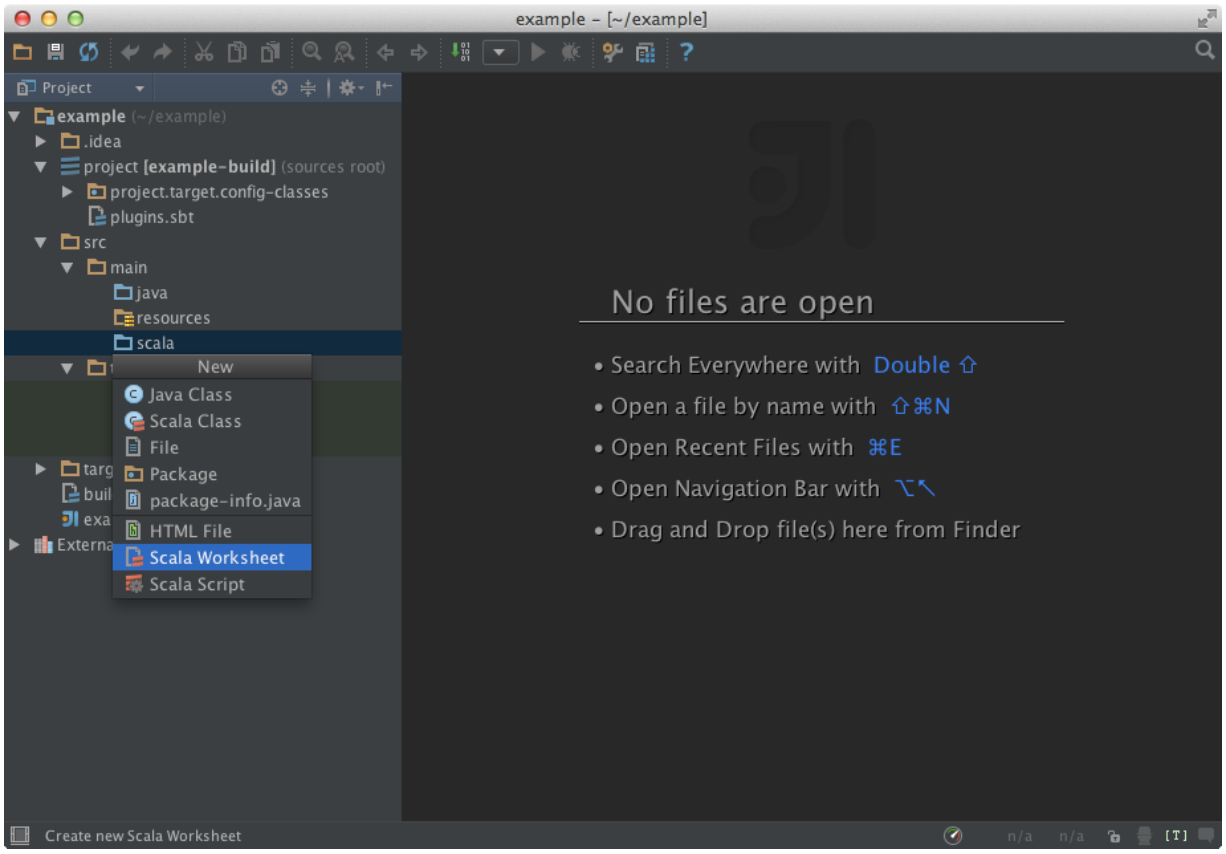


Click Next to specify project name and location. Once you've entered this information, IntelliJ IDEA will create an empty project containing a *build.sbt* file.

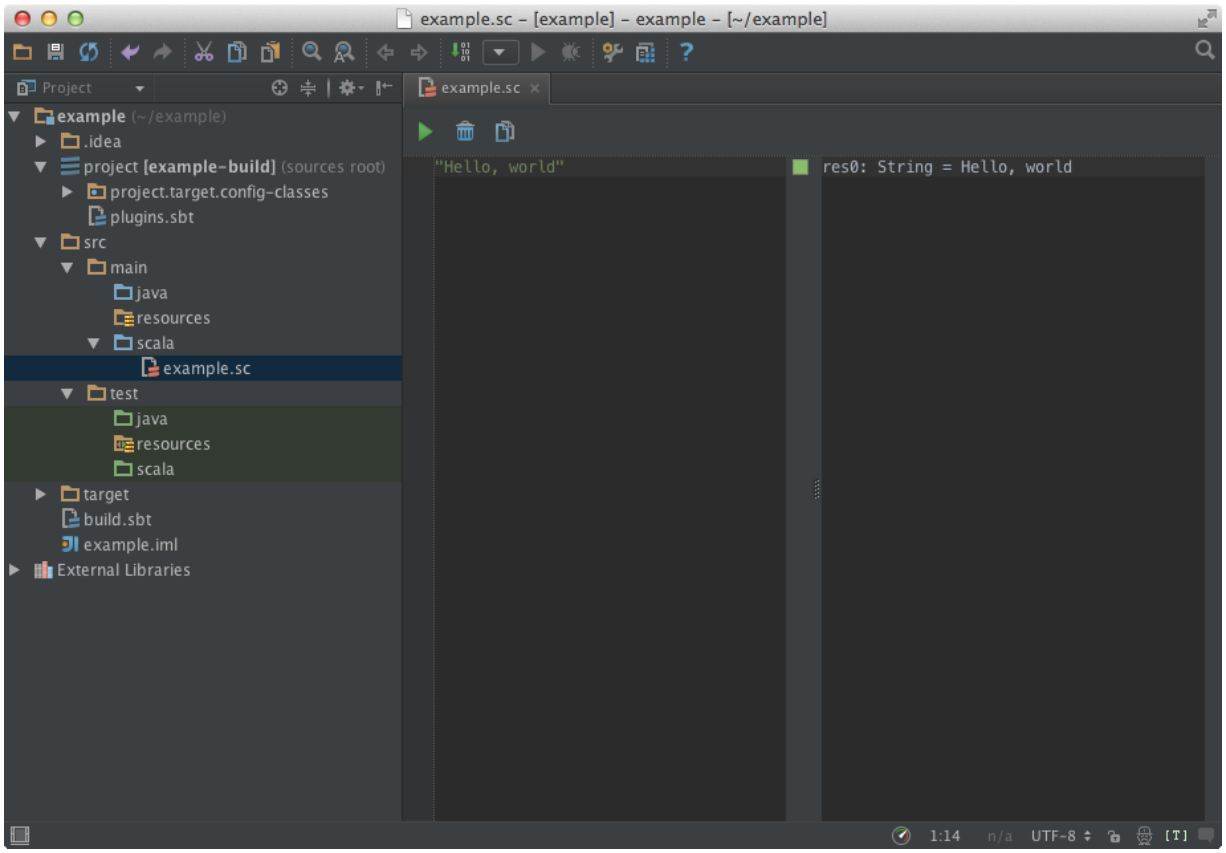
Note: The first time we create or import a SBT project, Intellij may not show the structure of the project right away. This happens because Intellij decides to delay this task until it has successfully executed sbt and downloaded all the required dependencies. Once that's done, the usual `*src*` folder and their subfolders will be created.

5. Creating a Scala worksheet

Simply use the **Create New** action from context menu or from a Scala package (anything under the `*scala*` folder).

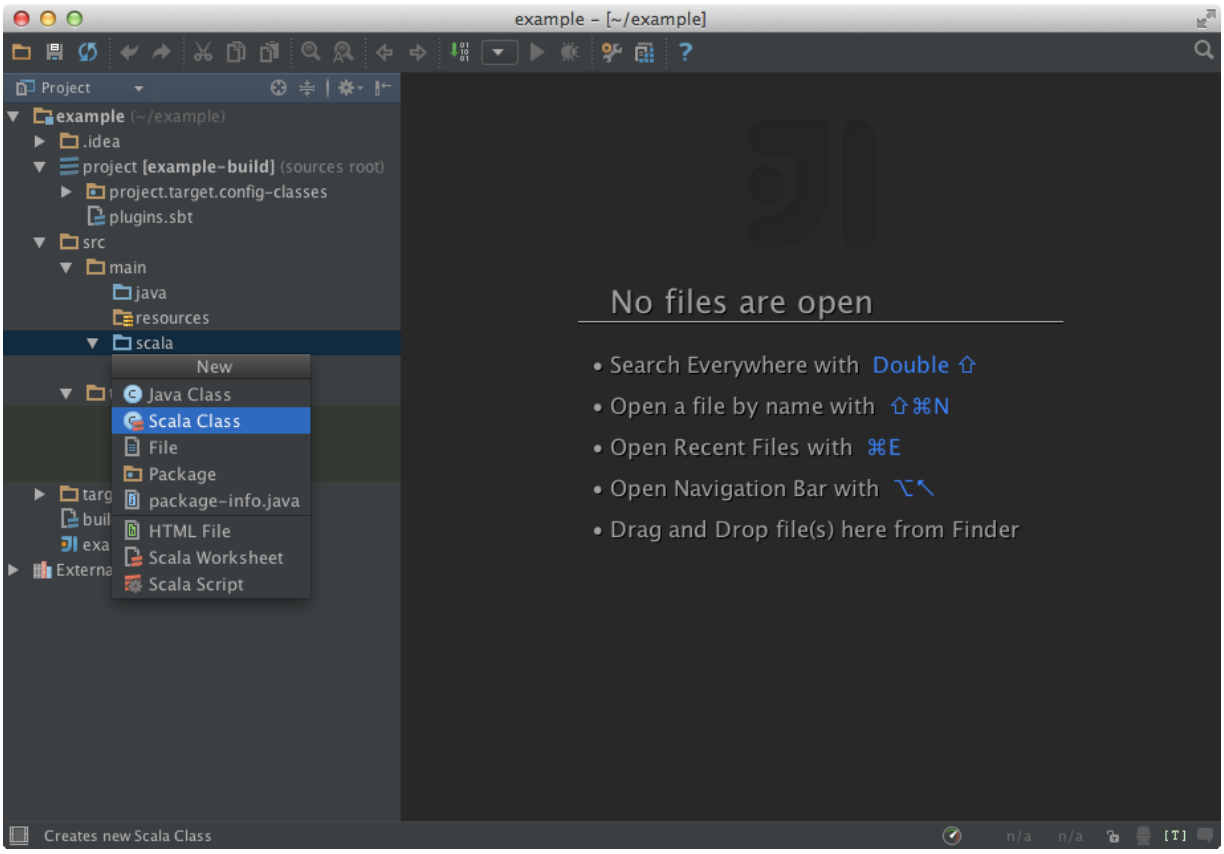


To evaluate worksheets, use the corresponding toolbar icon (the play button), or press **Alt+Ctrl+W** (Alt+Cmd+W on OS X). In case of any problem, you can check the official documentation that thoroughly explains how to run the *Scala Worksheet*.

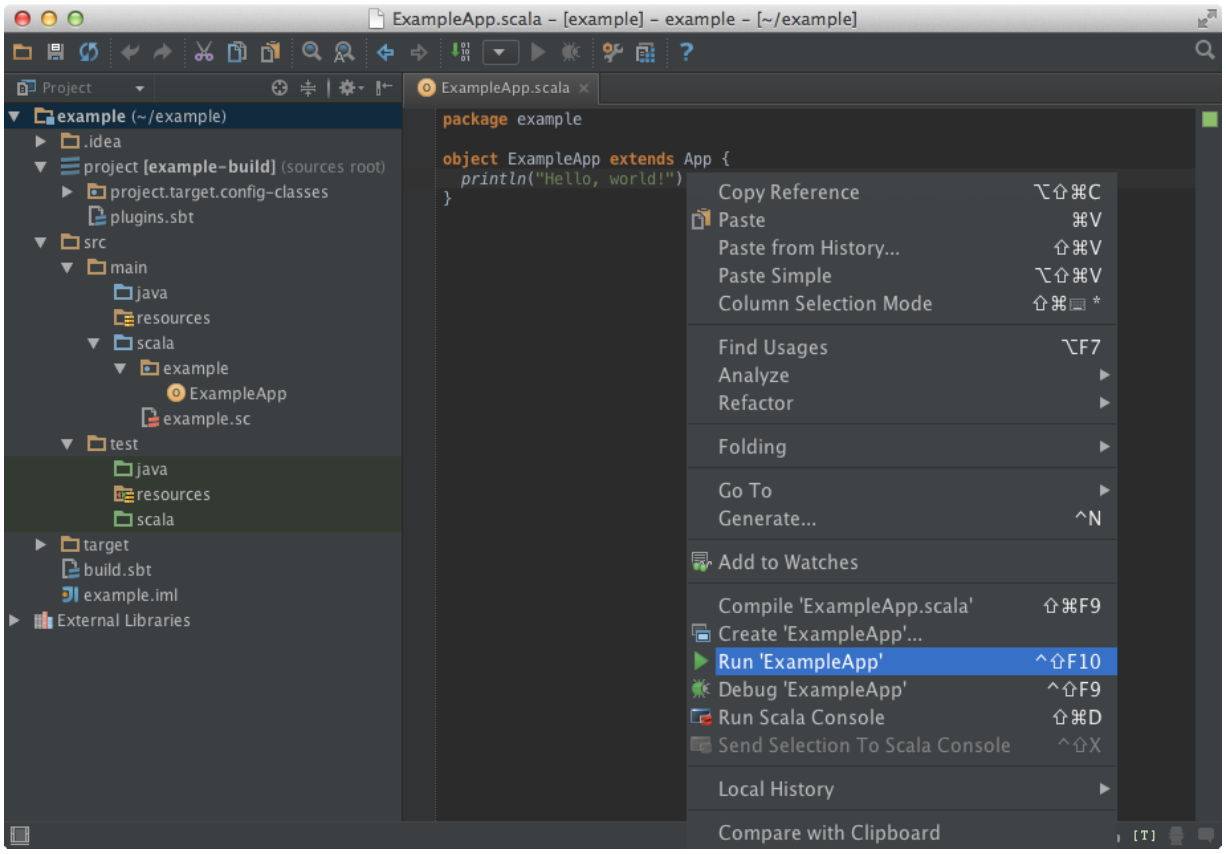


6. Creating a Scala class

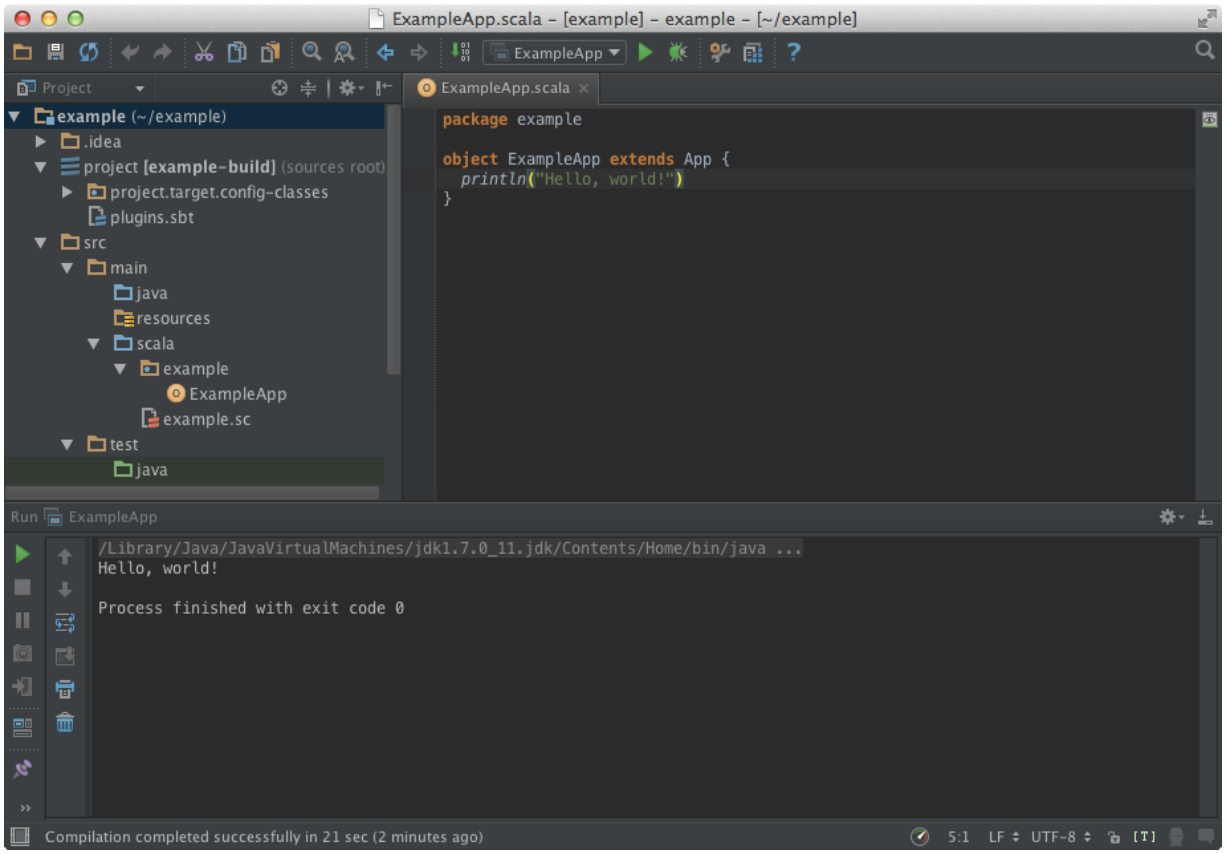
Much akin to worksheets, Scala classes are created via context menu action **Create New**.



Once you are ready, run your application by pressing **Ctrl+Shift+F10**, or using the editor context menu.



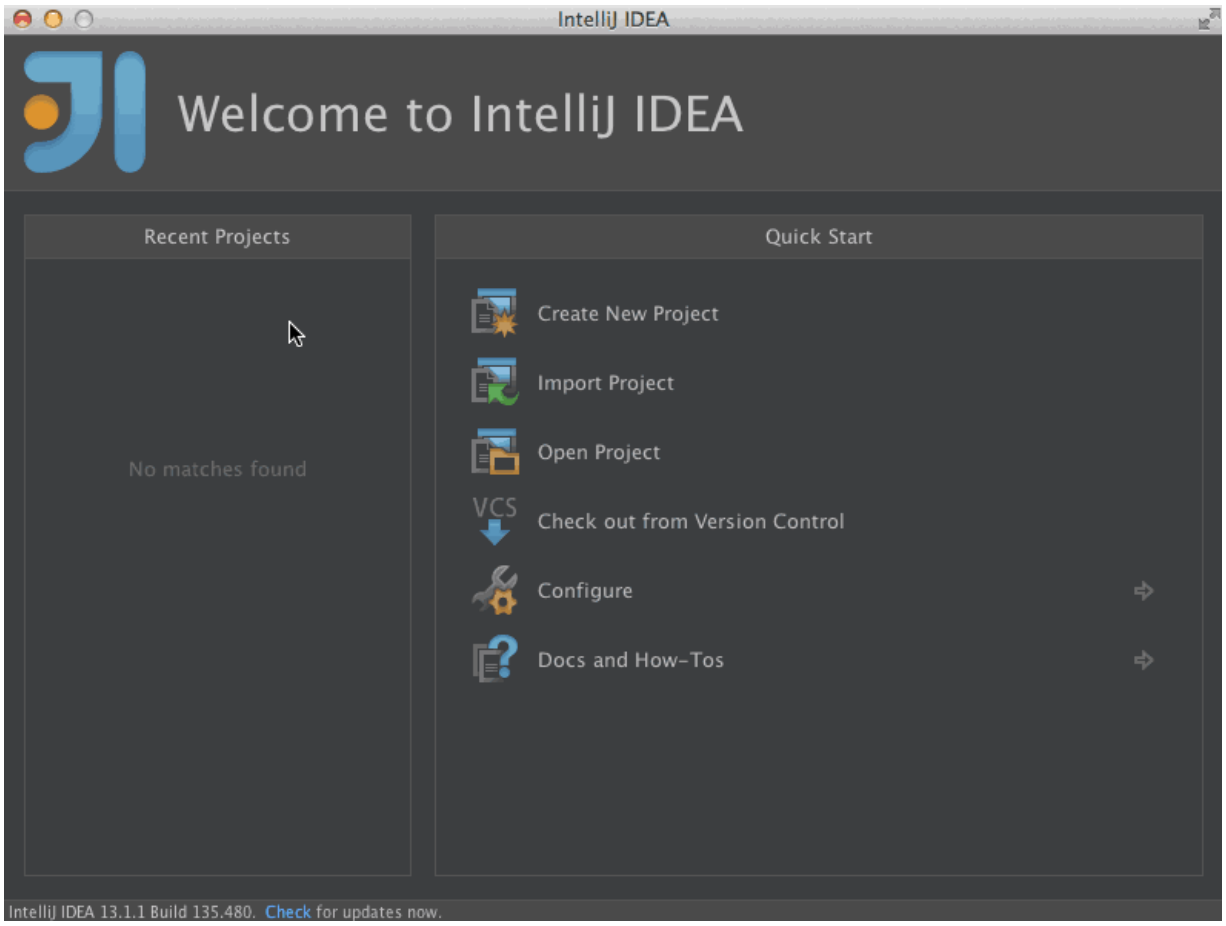
After the application has finished running, you'll see its output in the Run tool window.



7. Opening an SBT project

All the handout files (assignments) are SBT projects. If you want to hack on them using Intellij, you need to import the project first.

To open an SBT project in IntelliJ IDEA, go to the Welcome Screen, click Import Project, and select SBT build file that you wish to open. IntelliJ IDEA will then create a new project and import the selected file to it.



Also, you can open an SBT project without calling the Import Project action. Just click Open Project from the Welcome Screen and select an SBT build file.

8. Synchronizing SBT and IntelliJ IDEA projects

IntelliJ IDEA SBT support synchronizes the project with your build file, so when you change Scala version you're going to use, or add a library, your project is updated accordingly. For the next time, you can avoid this step by checking off the option "Use auto-import" in Step 7.

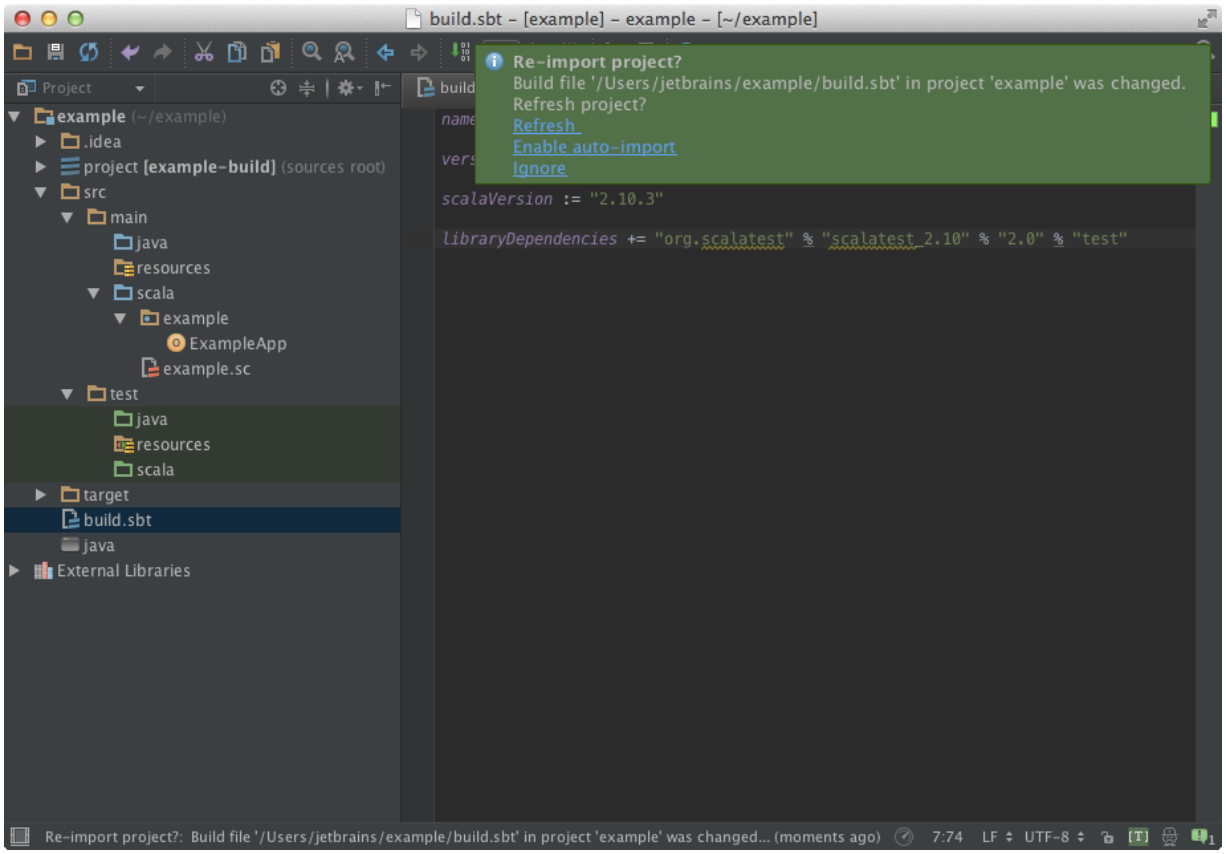
Warning: the scala version shown in this screenshot is 2.10.3 and it's outdated. The scala version that we will use for this course is **2.11.x**. Also, note that the scalatest dependency has the scala version embedded in "scalatest_2.10" and that there's no `2.0` version for 2.11.x. So, replace:

```
1  libraryDependencies += "org.scalatest" % "scalatest_2.10" % "2.0" % "test"
```

by:

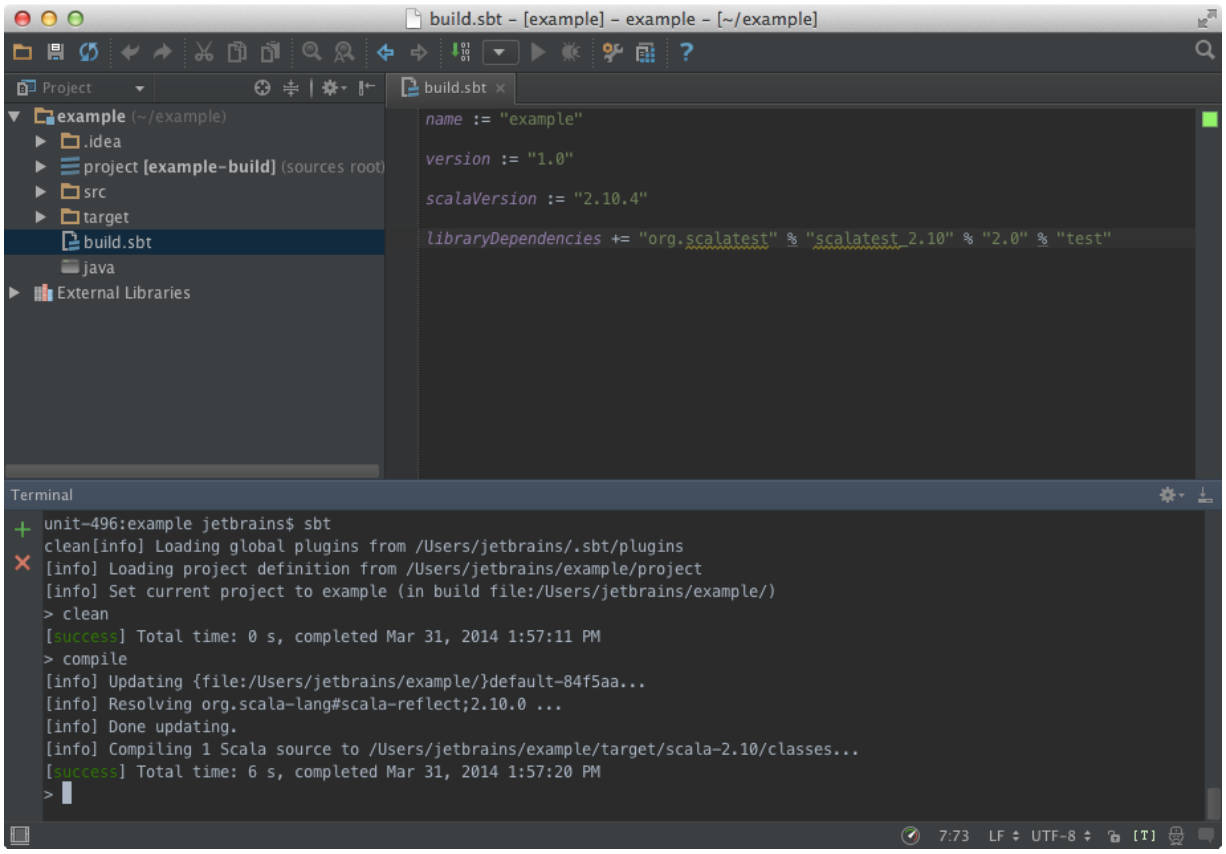
```
1  libraryDependencies += "org.scalatest" %% "scalatest" % "2.2.6" % "test"
```

The double percentage symbol will force sbt to use the current scala version defined in *scalaVersion* (which has to be "2.11.7" or "2.11.8").



9. Using terminal to run SBT commands

The easiest way to run SBT commands from from IntelliJ IDEA is to use the Terminal tool window via **Alt+F12**. It executes the terminal in your project directory. Then, type `sbt` to execute the build tool REPL. If you have any problem (your mappings are different, for instance), check the official documentation.



10. More information and troubleshooting

To learn more about IntelliJ IDEA, read the IntelliJ IDEA official documentation.

Mark as completed