



## Table of Contents

Introduction .....	3
Prerequisites .....	3
Simple Screencast Watch.....	3
Installation .....	3
Switching to the Scala IDE perspective .....	4
Installing SBT .....	4
Creating an SBT-based Eclipse IDE project .....	5
Creating a new SBT project.....	5
Loading the SBTEclipse plugin.....	7
Generating a new Eclipse IDE project with SBTEclipse.....	7
Importing the generated project in Eclipse IDE.....	7

## Introduction

This set of instructions would guide you through the configuration of Scala IDE (Eclipse family) and Scala compiler on Windows Platform.

### Prerequisites

Eclipse with a very recent Scala IDE for Scala 2.11 installed (update site: <http://scala-ide.org/download/nightly.html>). A basic knowledge of the Eclipse user interface is required.

Eclipse IDE does not come with Scala support out of the box. You need to install the Scala IDE plugin to add Scala compatibility to Eclipse IDE. One complication is that the installed Scala IDE dictates which Scala releases would be used in Eclipse IDE. On some occasions, it takes the Scala IDE team some time to add support for the latest Scala version. After the installation, you can switch to Eclipse's new <Scala> perspective.

We will install the Scala IDE version. Please follow the instructor demonstration.

### Simple Screencast Watch

In case you have an existing eclipse, use the below screen shot to install the Scala IDE plugin as shown in the following link:

<https://www.youtube.com/watch?v=dGI04efb6hg>

While Scala IDE is available on the Eclipse Marketplace, its version is often not based on the last available version. Since the installed Scala IDE version dictates which Scala release will be supported, we recommend that you install the most stable, up-to-date version from the repository sites managed by the Scala IDE team itself.

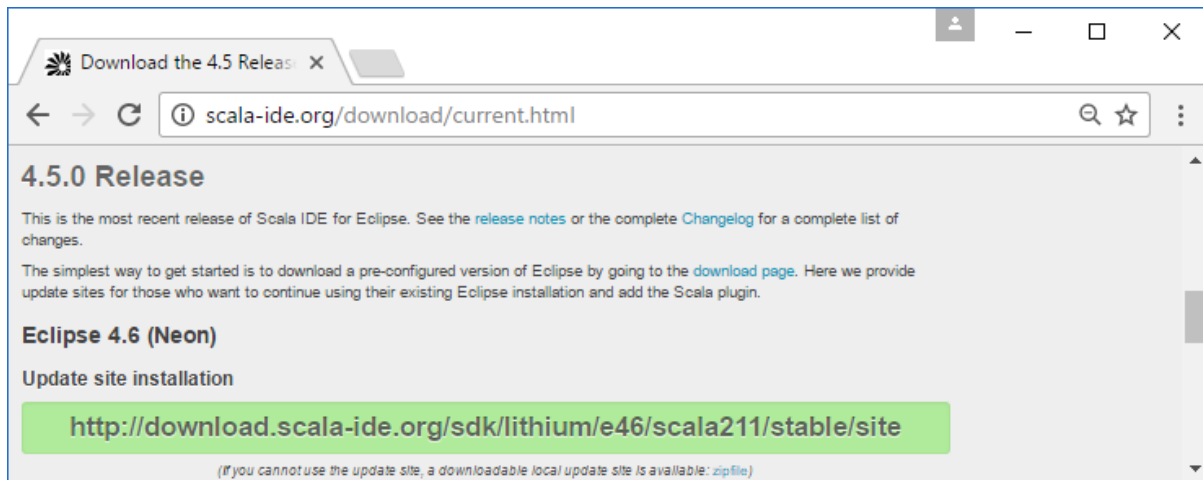
By manually adding Scala IDE's repository, Eclipse can download and install the latest stable version. You'll need to know the Eclipse version for this. You can find this by going to Help | About in Eclipse IDE. Usually, only the major version and minor version numbers are important.

## Installation

Visit the Scala IDE website at <http://scala-ide.org>.

Follow these instructions to install the plugin from Eclipse IDE:

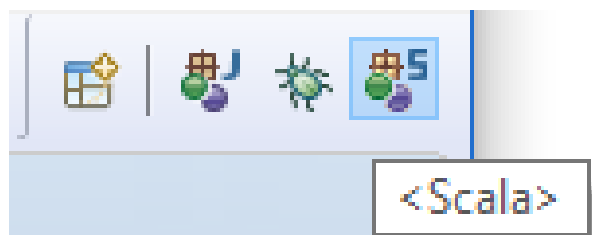
- On the Scala IDE home page, find the update sites link. At the time of writing this, it is located next to the prominent Download IDE button.  
OR
- In the list of supported Eclipse versions, find your installed Eclipse version and copy the link under the Update site installation heading to the clipboard:



## Switching to the Scala IDE perspective

Eclipse supports a lot of different programming languages. After installing Scala IDE, Scala is one of them. Eclipse can optimize its user interface for a specific programming language or environment, by offering different perspectives. Scala IDE adds its own perspective to Eclipse. To switch to the Scala IDE perspective, do the following:

- On the upper right-hand side of the screen, there are a few buttons that represent the last used perspectives. Hover over them; one of them will have the <Scala> tooltip. Click on it:



- If the Scala perspective button is unavailable, click on the button with the Open Perspective tooltip and select the <Scala> perspective from the list. The button will now be added to the toolbar.

The IDE's user interface is now optimized for Scala programming.

*You can switch to a different perspective at any time. Eclipse IDE is well suited for working with projects that use different programming languages.*

## Installing SBT

To install SBT, visit <http://www.scala-sbt.org> and download the latest version for your system.

For Windows, an MSI installer is available that takes care of installing and setting up the path. For other operating systems, an archive file (ZIP or TGZ) must be downloaded, extracted, and added to the path.

Like Gradle, SBT commands can be executed from the command line by specifying the commands as parameters. SBT is unique in that it also has an interactive shell option. When running SBT in interactive mode, SBT commands can run interactively, and as an additional bonus, they also support auto completion; you just need to press the Tab key.

To validate the installation in the command window (Windows) or Terminal (macOS and Linux), type `sbt` and press Enter to start SBT in the interactive shell mode.

## Creating an SBT-based Eclipse IDE project

As mentioned earlier, an important issue with the current version of the Scala IDE plugin for Eclipse IDE is that it has no support for SBT. By installing a plugin in SBT, we can generate Eclipse project files from SBT. The workflow for creating a new SBT-based project that can be opened in Eclipse IDE with the Scala IDE plugin installed is as follows:

- Create a new SBT-based project by applying a project template.
- Add the `sbteclipse` plugin to SBT.
- From SBT, generate a new Eclipse IDE/Scala IDE project using the `sbteclipse` plugin.

## Creating a new SBT project

The easiest way to create a new project is to let SBT generate a `Hello World` project, which includes a blank build file. This command should be run from Eclipse IDE's `workspace` directory. This is the directory that Eclipse IDE uses to store projects. If you are unsure where this directory is located, start Eclipse and try to create a Java project. It will show the path to the `workspace` directory. To create the a new SBT-based project, follow these steps:

- Start Command Prompt (Windows) or Terminal (macOS and Linux) and change the active directory to the `workspace` directory. Type the following command and press *Enter*:

```
sbt new sbt/scala-seed.g8
```

- The first time the preceding command is entered, SBT will download some dependencies. After a while, it will ask for a project name. Type `Akka Quotes` and press *Enter*. A new project will be created in the `akka-quotes` directory.
- Change the current directory to the `akka-quotes` directory and start SBT's interactive shell by simply typing `sbt` and pressing *Enter*.
- Try the generated project by entering the following command and pressing *Enter*:

```
run
```

- Again, SBT will start downloading some dependencies on the first run. When finished, you should see a `hello` message on the console:

```

C:\Users\Vincent\workspace\akka-quotes>sbt
Java HotSpot(TM) 64-Bit Server VM warning: ignoring option MaxPermSize=256m; support was removed in 8.0
[info] Loading project definition from C:\Users\Vincent\workspace\akka-quotes\project
[info] Updating (file://C:/Users/Vincent/workspace/akka-quotes/project/)akka-quotes-build...
[info] Resolving org.scala-sbt.ivy#ivy;2.3.0-sbt-2cf13e211b2cb31f0d3b317289dca70[info] Resolving org.fusesource.jansi#jansi;1.4 ...
[info] Done updating.
[info] Compiling 1 Scala source to C:\Users\Vincent\workspace\akka-quotes\project\target\scala-2.10\sbt-0.13\classes...
[info] Set current project to Hello (in build file://C:/Users/Vincent/workspace/akka-quotes/)
> run
[info] Updating (file://C:/Users/Vincent/workspace/akka-quotes/)root...
[info] Resolving jline#jline;2.14.1 ...
[info] Done updating.
[info] Compiling 1 Scala source to C:\Users\Vincent\workspace\akka-quotes\target\scala-2.12\classes...
[info] Running example.Hello
Hello
[success] Total time: 20 s, completed Feb 8, 2017 10:21:27 PM
>

```

Exit the interactive shell by entering the command `exit` and press *Enter*.

The template used to generate the project is based on the latest stable Scala release. We must check whether this version is supported by your installed Scala IDE version. In Eclipse IDE, from the menu bar, choose Window | Preferences.

In the tree, find and expand the Scala entry. Then, select its Installations entry. Write down the full versions of all the supported built-in Scala releases of your version of Scala IDE. On my installation, the built-in versions were Scala 2.11.8 and Scala 2.10.6. In a text editor, open the generated `build.sbt` build file that was placed in the `akka-quotes` directory. In my case, the file looked like this:

```

import Dependencies._

lazy val root = (project in file(".")).
  settings(
    inThisBuild(List(
      organization := "com.example",
      scalaVersion := "2.12.1",
      version := "0.1.0-SNAPSHOT"
    )),
    name := "Hello",
    libraryDependencies += scalaTest % Test
  )

```

If the specified `scalaVersion` variable was listed in your Scala IDE's Scala installations, then everything is fine. If not, you'll have to change the `scalaVersion` entry to your version. In my case, I had to change it to:

```
scalaVersion := "2.11.11",
```

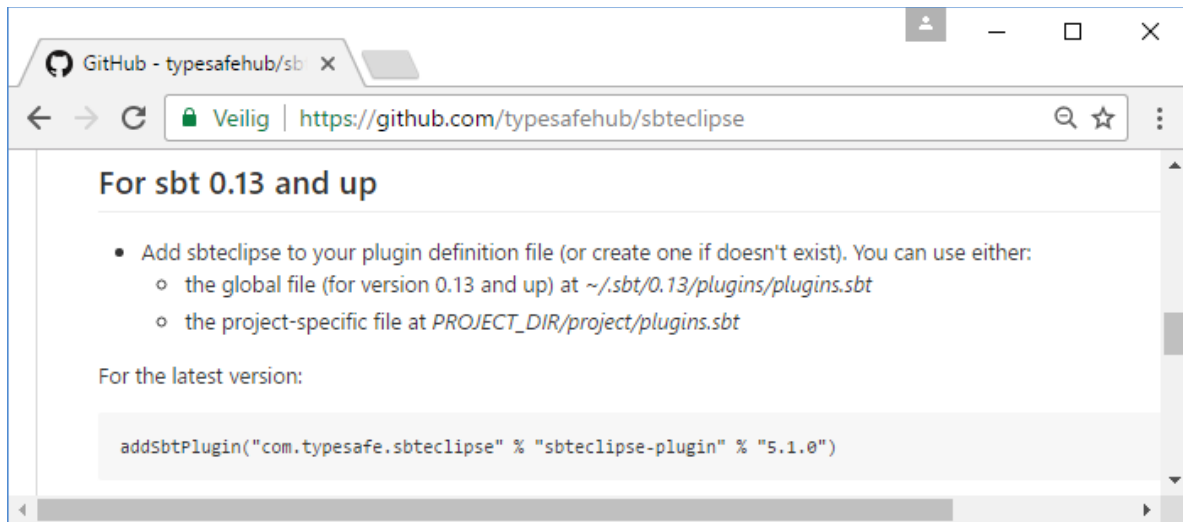
If you had to change the version, run the following commands in the command line to clean and fully recompile the project:

```
sbt clean run
```

SBT will recompile the project using the changed Scala version. Since the template is based on a minimal project, this should usually work without problems.

## Loading the SBTEclipse plugin

To find out which version of the plugin you have to install, visit the project page at <http://github.com/typesafehub/sbteclipse>:



The GitHub project page has instructions to add the plugin to your project. You should look for a line that starts with `addSbtPlugin`. In my case, it mentioned the following:

```
addSbtPlugin("com.typesafe.sbteclipse" % "sbteclipse-plugin"
            % "5.1.0")
```

In the `project` subdirectory of `akka-quotes`, create a new `plugins.sbt` file and copy and paste the line from the website. SBT now knows that this plugin is required for this project.

*Ensure that the file is saved in the `project` subdirectory; otherwise, SBT will not find the file.*

## Generating a new Eclipse IDE project with SBTEclipse

Change the active directory back to the root directory of your `akka-quotes` directory and start the SBT interactive shell again by typing `sbt` and pressing *Enter*.

SBT will download and activate the plugin. From now on, when starting SBT in this project directory, you will be able to create or update an Eclipse IDE/Scala IDE project. Run the following command that is added by the `sbteclipse` plugin:

```
eclipse
```

The `sbteclipse` plugin has now generated project files inside the SBT project's directory that Eclipse IDE with the installed Scala IDE plugin can import.

## Importing the generated project in Eclipse IDE

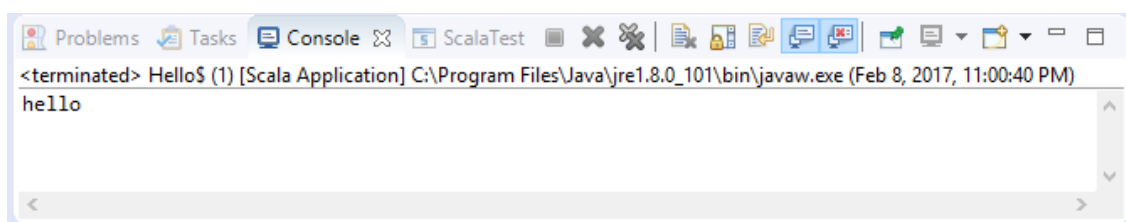
Return to your running Eclipse IDE instance or start Eclipse IDE if it were not running previously. To import the project that SBTEclipse generated for us, follow these steps:

1. Right-click on an empty spot in Package Explorer on the left-hand side of the screen and select **Import**....



2. The Import dialog will appear and ask you to select an import wizard. Choose the Existing Projects into Workspace option and click on the Next button.
3. Click on the Browse... button next to the Select root directory field and navigate to your workspace directory's akka-quotes subdirectory and click on OK.
4. The import wizard dialog's project list will now have an akka-quotes project entry. Click on the Finish button to close the dialog.

The project will now be added to Package Explorer. In Package Explorer, navigate to the `src/main/Scala | example | Hello.scala` file. Press `Ctrl + F11` to run the file; alternatively, click on the Run button in the toolbar or choose Run | Run from the menu bar. If everything goes well, you'll see the `hello` greeting in the Console tab:



*Have fun coding. Cheers to the craft of creating clean code.*



-----END OF DOCUMENT-----

