O'REILLY[®]

Scala Programming Fundamentals Sealed Traits, Collections, and Functions Daniel Hinojosa

About Me... Daniel Hinojosa

Programmer, Developer, Consultant, Instructor, and Speaker

Notable Content:

Testing in Scala (Book)
Beginning Scala Programming (Video)
Java & TDD (Video Training)
Scala Beyond The Basics
Scala Programming Fundamentals Series

Speaker:

OSCON No Fluff Just Stuff Tour DevNexus

<u>dhinojosa@evolutionnext.com</u> @dhinojosa



Other Scala Classes

- Learn the Basics of Scala in 3 hours
- Setting Up Scala Projects
- Scala Beyond the Basics
 - Implicits
 - Pattern matching
- Scala Programming Fundamentals
 - Classes, Methods, Traits
- Scala Programming Fundamentals
 - Sealed, Collections, and Functions

Structure of the Class

- All LiveCoding
- Performed in IntelliJ and Eclipse
- Commits and Pushing performed throughout the class
- Your Choice:
 - Watch
 - Code with Me
- Definitely Ask Questions

Checklist before we proceed...

- We provided pre-setup configuration already.
- Be sure JAVA_HOME is set and that the following works:
 - javac -version
 - java -version
- Be sure SBT is setup and ensure that the following works
 - sbt about
- Be sure that Scala plugins are installed in IntelliJ and Eclipse

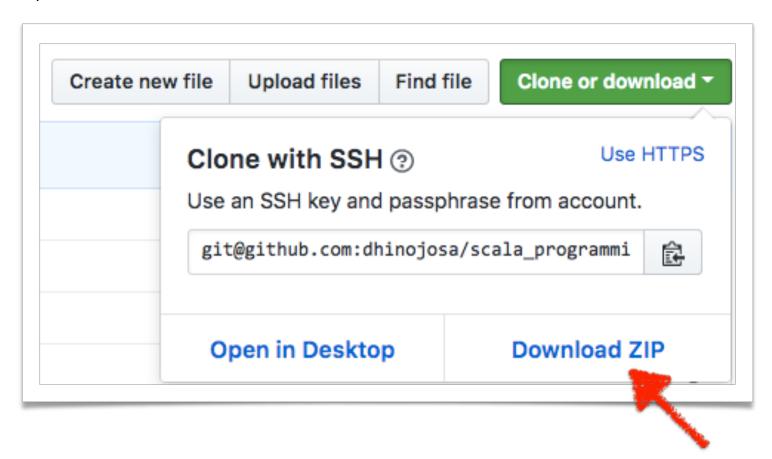
Repository Location, Clone it!

https://github.com/dhinojosa/scala_programming_fundamentals_2

or use the abbreviation...

https://bit.ly/2lgoOyp

But, if don't know Git too well...



Once downloaded...Run SBT

```
~/Development/scala_programming_fundamentals_2(master) » sbt
```

You should see something like

thin

```
[~/Development/scala_programming_fundamentals_2(master) » sbt
[info] Loading global plugins from /Users/danno/.sbt/1.0/plugins
[info] Loading settings from plugins.sbt ...
[info] Loading project definition from /Users/danno/Development/scala/project
[info] Loading settings from build.sbt ...
[info] Set current project to scala_programming_fundamentals_2 (in buopment/scala_programming_fundamentals_2/)
[info] sbt server started at local:///Users/danno/.sbt/1.0/server/30csbt:scala_programming_fundamentals_2>
```

You'll need to update your project...

```
[~/Development/scala_programming_fundamentals_2(master) » sbt
[info] Loading global plugins from /Users/danno/.sbt/1.0/plugins
[info] Loading settings from plugins.sbt ...
[info] Loading project definition from /Users/danno/Development/scala_programmin/project
[info] Loading settings from build.sbt ...
[info] Set current project to scala_programming_fundamentals_2 (in build file:/I opment/scala_programming_fundamentals_2/)
[info] sbt server started at local:///Users/danno/.sbt/1.0/server/30cfdcd777258(sbt:scala_programming_fundamentals_2> update
[info] Updating ...
[info] Done updating.
[success] Total time: 2 s, completed Mar 29, 2018, 12:59:55 PM
sbt:scala_programming_fundamentals_2>
```

Updating Your Project

- This will download Scala and all your dependencies into your ~/.ivy2
- This may take a while depending on your connection speed.
- If you have any problems, there might be issues with network connection, or proxy based issues

Proxy Issues

 If you have problems because of any proxy server, add the following to your environment variables and change username and password accordingly

```
export JAVA_OPTS="$JAVA_OPTS \
-Dhttp.proxyHost=yourserver \
-Dhttp.proxyPort=8080 \
-Dhttp.proxyUser=username \
-Dhttp.proxyPassword=password"
```

Lab: Let's get setup!

Clone or download

```
scala_programming_fundamentals_2
from
https://github.com/dhinojosa/
scala programming fundamentals 2
```

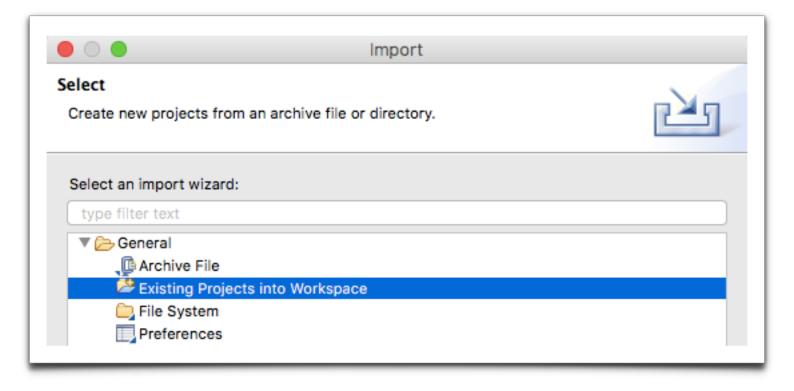
- Run sbt at the command line
- Once in the sbt console, run update
- Wait for the download to complete

Setup for Eclipse

Run the Eclipse Plugin in SBT

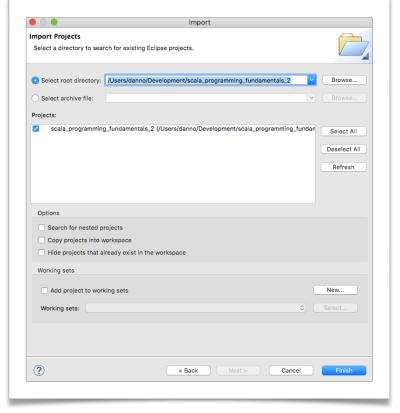
```
sbt:scala_programming_fundamentals_2> eclipse
```

Import Existing Project



File > Import Project >
Select Existing Projects Into Workspace > Next

Select Root Directory



Select the Root Directory of the Project using Browse, Ensure that the Project is Found, hit Finish Project is now ready

```
Quick Access
Packag X Type H Ju JUnit - FunctionsSpec.scala X
                           ▼ 📆 > scala_programming_fundamentals_2 [scal | 2 | 3 | import org.scalatest.{FunSuite, Matchers}
   ▼ # > src/test/scala
                                                50 class FunctionsSpec extends FunSuite with Matchers {
  ▼ ∰ > con-accelaprogrammingfundame

▼ ∰ > ConfectionsSpec.scele

▼ ∰ > CollectionsSpec.scele
                                                    | a function really is an anonymous
| instantiation of a trait.""".stripMargin) {
        ► 🖺 SealedTraitsSpec.scala
    # src/test/iava
                                           # src/test/resources
   ▶ 

Scala Library container [ 2.12.3 ]
                                                  f.apply("Hello") should be(5)
}
   ▶ ➡ Referenced Libraries
   ▶ M JRE System Library [Java SE 8 [1.8.0_12 6,15
   ▶ ( project
                                             19 test("The above can be whittled down to the following:") {
19 val f = (s: String) => s.length
   ▶ (⇒ target
    the build.sbt
                                                       f.apply("Zanzibar") should be (8)
      scala_programming_fundamentals_2.iml
                                                     """If you declare the left hand side you can
                                                     I do some nifty tricks on the right hand side:""".stripMargin) { val f: String \Rightarrow Int = s \Rightarrow s.length
                                                       f.apply("Andorra") should be (7)
                                                      """Also, since the left hand side, has all the type information,

I on the right hand side you can trim the left hand side

I with syntactical tricks like use the placeholder""".stripMargin) {
                                                      val f: String => Int = _.length
                                                    f.apply("Andorra") should be (7)
                                                     The type system has enough information either because

of the left hand side of an assignment, or a parameter in method, or
the way generic types are situated within a class,
you can get rid of some additional code. In the following example, you
can drop the underline and leave $5, but will come with some warnings
                                                       I that you can turn off with """.stripMargin) {
                                                      val f: Int -> Int = 5 +
                                                                                                      Writable Smart Insert 1:1
```

Select the Root Directory of the Project using Browse, Ensure that the Project is Found, hit Finish

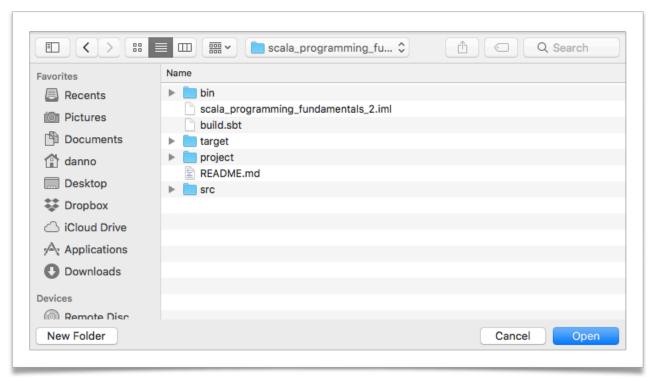
Setup for IntelliJ

From IntelliJ Intro Screen, or from the splash screen, select Import Project



Locate the

scala_programming_fundamentals_2
project and click OK, accept defaults



Project is now ready...

```
● ● ● 🎅 scala_programming_fundamentals_2 [~/Development/scala_programming_fundamentals_2] - .../src/test/scala/com/ora/scalaprogrammingfundamentals/Function...
mentals_2 > 🖿 src > 🖿 test > 🖿 scala > 🖿 com > 🖿 ora > 🖿 scalaprogrammingfundamentals > 💽 FunctionsSpec.scala > 👫
                                                                                                ⊕ # # FunctionsSpec.scala ×
                                                     package com.ora.scalaprogrammingfundamentals
                                                     import org.scalatest.{FunSuite, Matchers}
                                                    class FunctionsSpec extends FunSuite with Matchers {
    project [scala_programming_fundamentals_2 5
    ▼ Imsrc
      ▼ ■ main
                                                        """As a reminder from all those that took the beginner's course,
                                                          a function really is an anonymous
                                                           instantiation of a trait.""".stripMargin) {
             com.ora.scalaprogrammingfundamen 9
                                             10

    ▼ lim test
                                                        val f = new Function[String, Int] {
                                             11
           iava i
                                             12 0
                                                          override def apply(v1: String): Int = v1.length
           resources
                                             13
         scala
           ▼ com.ora.scalaprogrammingfundamen 15
                                                        f.apply("Hello") should be(5)
                CollectionsSpec
                                             16
                 C FunctionsSpec
                                             17
                SealedTraitsSpec
                                             18
                                                      test("The above can be whittled down to the following:") {
    ▶ limitarget
                                                       val f = (s: String) => s.length
         .cache-tests
                                                        f.annly("Zanzihar") should be (8)
       .classpath
      [info] Defining Global / sbtStructureOptions, Global / sbtStructureOutputFile, shellPrompt
      [info] The new values will be used by no settings or tasks.
      [info] Reapplying settings...
      [info] Set current project to scala_programming_fundamentals_2 (in build
       file:/Users/danno/Development/scala_programming_fundamentals_2/)
      [info] Applying State transformations org.jetbrains.sbt.CreateTasks from /Users/danno/Library/Application
        Support/IntelliJIdea2017.3/Scala/launcher/sbt-structure-1.0.jar
       [info] Reapplying settings...
      [info] Set current project to scala_programming_fundamentals_2 (in build
       file:/Users/danno/Development/scala_programming_fundamentals_2/)
      [info] Writing structure to /private/var/folders/_x/2n6ld3jx3ng2w6pr1y29t7540000gn/T/sbt-structure.xml...
       [info] Done.
      [success] Total time: 2 s, completed Mar 29, 2018 8:06:30 PM
   😲 9: Version Control 🏻 sbt shell 🔟 Terminal 🔑 Build 🤏 6: TODO
                                                                                                                       C Event Log
```

Fraits and Abstract Classes, Collections, and F

(Live Coding)

Thank You