

# USING SBT

---

# Directory Structure

```
.
|-- build.sbt
|-- lib
|-- project
|-- src
|   |-- main
|   |   |-- java
|   |   |-- resources
|   |   |-- scala
|   |-- test
|       |-- java
|       |-- resources
|       |-- scala
|-- target
```

- Create main scala class :

```
package foo.bar.baz  
object Main extends App {  
    println("Hello, world")  
}
```

- Open terminal -> cd to project.
- **\$ sbt compile**
- **\$ sbt run**
- **\$ sbt package**

# Managing Dependencies :

- If you have jar, copy them into Lib folder.
- Or if it is managed dependency, write it in build.sbt
- -----
- `libraryDependencies += "net.sourceforge.htmlcleaner" %  
"htmlcleaner" % "2.4"`
- -----
- `libraryDependencies += groupId % artifactID % revision`

Method	Description
<code>+=</code>	Appends to the key's value. The <i>build.sbt</i> file works with settings defined as key/value pairs. In the examples shown, <code>libraryDependencies</code> is a key, and it's shown with several different values.
<code>%</code>	A method used to construct an Ivy Module ID from the strings you supply.
<code>%%</code>	When used after the <code>groupId</code> , it automatically adds your project's Scala version (such as <code>_2.10</code> ) to the end of the artifact name.

As shown in the examples, you can use `%` or `%%` after the `groupId`. This example shows the `%` method:

```
libraryDependencies += "org.scalatest" % "scalatest_2.10" % "1.9.1" % "test"
```

This example shows the `%%` method:

```
libraryDependencies += "org.scalatest" %% "scalatest" % "1.9.1" % "test"
```

Once you add a dependency run **\$ sbt assembly**

# Specifying a Main Class to Run

```
name := "Simple Test Project"
```

```
version := "1.0"
```

```
scalaVersion := "2.10.0"
```

```
// set the main class for packaging the main jar
```

```
mainClass in (Compile, packageBin) := Some("com.alvinalexander.testproject.Foo")
```

```
// set the main class for the main 'sbt run' task
```

```
mainClass in (Compile, run) := Some("com.alvinalexander.testproject.Foo")
```

# Uber Jar/ Fat Jar

- Sbt package jar
  - Class files
  - Source code
  - Src/main/resources
- No libraries
- No dependencies
- Solutions : Use **sbt-assembly**
- <https://github.com/sbt/sbt-assembly>

- In project/plugins.sbt

```
resolvers += Resolver.url("artifactory",  
url("http://scalasbt.artifactoryonline.com/scalasbt/sbt-plugin-releases"))  
(Resolver.ivyStylePatterns)  
  
addSbtPlugin("com.eed3si9n" % "sbt-assembly" % "0.8.4")
```

- In build.sbt

```
import AssemblyKeys._  
  
// sbt-assembly  
assemblySettings
```

- Run **\$ sbt assembly**







