We use sbt for building, testing, running and submitting assignments. This tutorial explains all sbt commands that you will use during our class. The Tools Setup page explains how to install sbt.

Starting up sbt

In order to start sbt, open a terminal ("Command Prompt" in Windows) and navigate to the directory of the assignment you are working on. Typing sbt will open the sbt command prompt.

```
1   shell$ cd /path/to/parprog-project-directory              # This is
      the shell of the operating system
2   shell$ sbt
3   > _                                                        # This is
      the sbt shell
```

Running the Scala Interpreter

You can start the Scala interpreter inside sbt using the console task. The interpreter (also called REPL, for "read-eval-print loop") is useful for trying out snippets of Scala code. Note that the interpreter can only be started if there are no compilation errors in your code.

In order to quit the interpreter and get back to sbt, type ctrl-d.

```
1   > console
2   [info] Starting scala interpreter...
3   Welcome to Scala version 2.11.5 (Java HotSpot(TM) 64-Bit Server VM, Java 1.7
      .0_04-ea).
4   Type in expressions to have them evaluated.
5   Type :help for more information.
6
7   scala> println("Oh, hai!")                                 # This is
      the Scala REPL, type some Scala code
8   Oh, hai!
9
10  scala> val l = List(1, 2, 3)
11  l: List[Int] = List(1, 2, 3)
12
13  scala> val squares = l.map(x => x * x)
14  squares: List[Int] = List(1, 4, 9)
15
16  scala>                                                     # Type [ctrl
      -d] to exit the Scala REPL
17  [success] Total time: 20 s, completed Mar 21, 2013 11:02:31 AM
18  >                                                          # We're back
      to the sbt shell
```

Compiling your Code

The compile task will compile the source code of the assignment which is located in the directory src/main/scala.

```
1   > compile
2   [info] Compiling 4 Scala sources to /Users/aleksandar/example/target/scala-2.11
      /classes...
3   [success] Total time: 1 s, completed Mar 21, 2013 11:04:46 PM
4   >
```

If the source code contains errors, the error messages from the compiler will be displayed.

Testing your Code

The directory src/test/scala contains unit tests for the project. In order to run these tests in sbt, you can use the test command.

```
 1  > test
 2  [info] ListsSuite:
 3  [info] - one plus one is two
 4  [info] - sum of a few numbers *** FAILED ***
 5  [info]   3 did not equal 2 (ListsSuite.scala:23)
 6  [info] - max of a few numbers
 7  [error] Failed: : Total 3, Failed 1, Errors 0, Passed 2, Skipped 0
 8  [error] Failed tests:
 9  [error]   example.ListsSuite
10  [error] {file:/Users/luc/example/}assignment/test:test: Tests unsuccessful
11  [error] Total time: 5 s, completed Aug 10, 2012 10:19:53 PM
12  >
```

Running your Code

If your project has an object with a main method (or an object extending the trait App), then you can run the code in sbt easily by typing run. In case sbt finds multiple main methods, it will ask you which one you'd like to execute.

```
 1  > run
 2  Multiple main classes detected, select one to run:
 3
 4    [1] example.Lists
 5    [2] example.M2
 6
 7  Enter number: 1
 8
 9  [info] Running example.Lists
10  main method!
11  [success] Total time: 33 s, completed Aug 10, 2012 10:25:06 PM
12  >
```

Submitting your Solution to Coursera

The sbt task submit allows you to submit your solution for the assignment. It will pack your source code into a .jar file and upload it to the coursera servers. Note that the code can only be submitted if there are no compilation errors.

The submit tasks takes two arguments: your Coursera e-mail address and the submission password. **NOTE**: the submission password is **not your login password**. Instead, it's a special password generated by coursera. It is available on the Assignments page.

```
 1  > submit e-mail@university.org suBmISsioNPasSwoRd
 2  [info] Packaging /Users/luc/example/target/scala-2.11/parprog-example_2.11-1.0.0
         -sources.jar ...
 3  [info] Done packaging.
 4  [info] Compiling 1 Scala source to /Users/luc/example/target/scala-2.11/classes
         ...
 5  [info] Connecting to coursera. Obtaining challenge...
 6  [info] Computing challenge response...
 7  [info] Submitting solution...
 8  [success] Your code was successfully submitted: Your submission has been
         accepted and will be graded shortly.
 9  [success] Total time: 6 s, completed Aug 10, 2012 10:35:53 PM
10  >
```

Mark as completed