Object Meets Function

Scala

Steven Lolong

steven.lolong(at)uni-tuebingen.de

Programming Language Research Group
Tuebingen University

Oct. 30, 2022

Eberhard Karls
Universität
Tübingen

Summary

Eberhard Karls
Universität
Tübingen

REPL and Compiler

scala open scala terminal for REPL, or run Scala's bit code.

scalac compile Scala's source code to Scala's bit code.

Read - Evaluate - Print - Loop – REPL

```
1 + 1
5 + (2 * 2) - 9 * 1

// value
val a = 5

// variable
var b = 7
a + b

// Method
def aPlusB(a: Int, b: Int): Int = a + b
aPlusB(4, 2)

val max: (Int, Int) => Int = (a: Int, b: Int) => {
  if (a > b) a
  else b
}
max(7, 4)
```

Listing: REPL

REPL

```
scala> :help
```

:help	print this summary
:load	⟨*path*⟩ interpret lines in a file
:quit	exit the interpreter
:type	⟨*expression*⟩ evaluate the type of the given expression
:doc	⟨*expression*⟩ print the documentation for the given expression
:imports	show import history
:reset	[options] reset the repl to its initial state, forgetting all session entries
:settings	⟨*options*⟩ update compiler options, if possible

Compile and Run

Scala compiler will compile Scala's source code into virtual-machine byte code (VMBCode), not a native-machine code (NBCode).
Compiling a file

scalac sourceCode.sc

Execute virtual-machine byte code
scala fileName
fileName only without file extension (.class)

ex.: VMBCode **foo.class** and Scala source code **bar.scala**
>**scala foo**
>**scala bar.scala**

Note: The program file (.class or .sc) to be run requires one method as a starting point.

Entry Point

- Method as an antry point means this method will automatically call.
- It needs an annotation **@main** in front of the function definition to make a function become an entry point.

```
1 @main def hello() =
2   println("Hello world!")
```

Listing: @main annotation