

CS 162 Week 2

Kyle Dewey

Overview

- More on Scala
- Assignment I review
- Secure information flow

More on Scala

Running the REPL

- Just type `scala` at the command line
- Pretty nifty to quickly check to see what an expression does

Unit

- Essentially `void`, but there is an object associated with it
- If you define a method without an equals, (i.e. `def foo {}`), then the return type is `Unit`
- Can say `()` to refer to the `Unit` object

Seq

- **Some Sequence**
- **In many ways very similar to Java's List**
- **List, Vector, and related classes are subclasses**
- **Note: Scala's List refers specifically to a linked list, more like Java's LinkedList**

null

- In general, `null` is an excellent wonderful/
terrible feature
- Often poorly documented whether or not
`null` is possible
- Checking for impossible cases
- Not checking for possible cases

Option

- A solution: encode `null` as part of a type
- For some type, say `Object`, if `null` is possible say we have a `Nullable<Object>`
- Scala has this, known as `Option`
- In general, if `null` is possible, use `Option`

Tuples

- For when you want to return more than one thing
- Can be created by putting datums in parenthesis
- Can pattern match on them

Sequence Processing Functions

AKA: Why `while` is rare and `for` isn't `for`

foreach

- Applies a given function to each element of a Seq

map

- Like `foreach`, in that it applies a given function to each element of a sequence
- However, it also returns a new sequence that holds the return values of each of the function calls

filter

- Takes a predicate, i.e. a function that returns true or false
- Applies the predicate to each item in a list
- A new list is returned that contains all the items for which the predicate was true

foldLeft

- Extremely flexible, but sometimes unwieldy
- Takes a base element
- Takes a function that takes a current result and a current list element
- The function will manipulate result with respect to the current element

flatMap

- Like `map`, but made especially for functions that return `Seqs`
- Will internally “flatten” all of the inner `Seqs` into a single `Seq`
- More on this later in the course

`for` Comprehensions

- Much like Python's list comprehensions
- Internally translated into a series of `foreach`, `flatMap`, `map`, and `filter` operations

Assignment I Review

Secure Information Flow Assignment

Missing File

- As written, there is a missing file:
`util.scala`
- Option 1: Download zip file from the course website (under “Interpreter Code”), copy `util.scala`, and add it to the makefile
- Option 2: Remove all mentions of the pretty printer (from `util.scala`)

Adding a Field for the Label

pc Stack

- Define an object named `pc`
- It internally has a mutable stack
- There are many ways to do this, but `scala.collection.mutable.Stack` is probably the easiest