

Inheritance and Subtyping

9-2-2011

Opening Discussion

- What did we talk about last class?

Traits

- A trait is like an abstract class that can't take arguments.
- You can inherit from multiple traits.
 - `class B extends A with T1 with T2 ... { ... }`
- If methods are duplicated, it searches for the one to use starting at the end of the list and working backward (plus some other details).

final

- Sometimes you have methods that shouldn't be overridden or classes that shouldn't be inherited from. In that case you make them final.
- For example, immutable classes need to be final so that people can't make mutable subtypes.

Inheriting from Function Types

- As you know, functions are used in many places in Scala.
- If you provide an apply method, you can have your class inherit from a function type.
- This would let you pass instances of your function into methods that want functions.

Drawing Project

- Now we will add some code to our project that will utilize inheritance.

Minute Essay

- What questions do you have?
- Show some code next class.