What did we talk about last class?
A trait is like an abstract class that can't take arguments.

You can inherit from multiple traits.

\[
\text{class B extends A with T1 with T2 ... \{ ... \}}
\]

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).
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.
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.
Now we will add some code to our project that will utilize inheritance.
What questions do you have?
Show some code next class.