How do you decide if A should inherit B?

How are method invocations on variables and expressions bound?

What do combinators do?

What is override modifier required for?

What is it optional for?

What do all constructors in Scala ultimately
invoke?

How is this enforced?

Classes must be abstract if they have what?

Prevent type derivation.

Give an alternative to inheritance, and when it should be used.

When should parameterless methods used?

What do parametric field definitions allow?

What does a subclass *not* inherit?

Parameterless methods vs. empty-paren methods

What's a problem with fragile base class?

How does Scala minimize it?

Name the class namespaces in Java and Scala.

What does Scala's class namespace allow?

Give the Scala two factory patterns.

Why do packages share the same
namespace as fields and methods?

How does Scala achieve compatibility with
Java's non-support of uniform access?