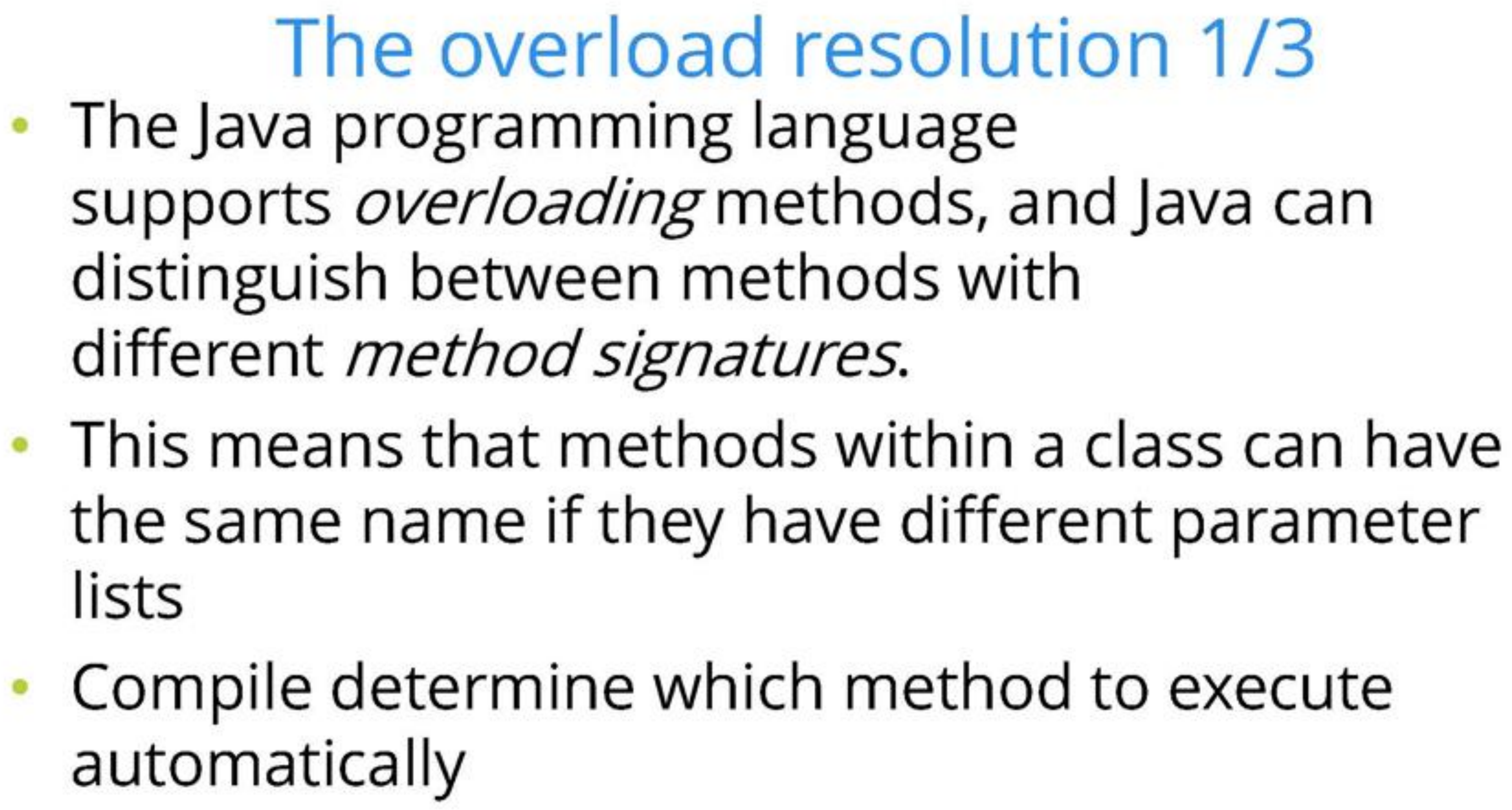
JAVA PROGRAMMING BASICS

Module 2: Java Object-oriented Programming

Training program

Module contents

- Method overload resolution
  - The overload resolution
  - The overload resolution: type or class
  - The overload resolution: type cast or boxing-unboxing
  - The overload resolution and varargs
  - Examples of overload resolution



The overload resolution 1/3

- The Java programming language supports *overloading* methods, and Java can distinguish between methods with different *method signatures*.
- This means that methods within a class can have the same name if they have different parameter lists
- Compile determine which method to execute automatically
- Using overloading makes your code cleaner and easier to read, and also helps to avoid program errors.

See OverloadedTest

The overload resolution: type or class 1/3

- Resolving a method name at compile time is more complicated than resolving a field name because of the possibility of method overloading.
- Invoking a method at run time is also more complicated than accessing a field because of the possibility of instance method overriding.
- The compiler chooses the "most specific" overloaded method with the argument type that is EQUAL or CLOSER to the type of the method parameter in the class hierarchy.
- +WIDENING +AUTBOXING +VARARGS

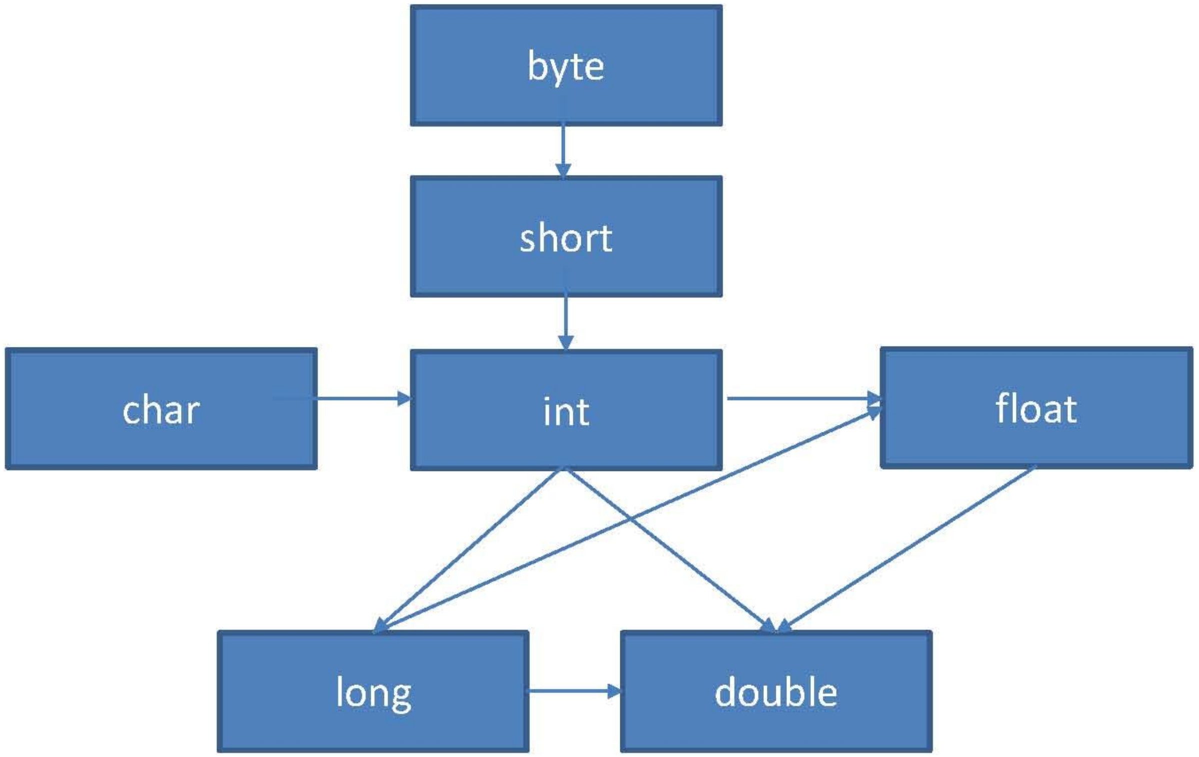
See MostSpecificMethod

The overload resolution: type cast or boxing-unboxing 1/6

- The first phase performs overload resolution without permitting boxing or unboxing conversion

See BoxingForbid1

Examples of overload resolution 7/7



The first phase can use TYPE WIDENING

See Widening1

The overload resolution: type cast or boxing-unboxing 1/6

- The first phase performs overload resolution without permitting boxing or unboxing conversion
- The second phase performs overload resolution while allowing boxing and unboxing, but still precludes the use of variable arity method invocation

See BoxUnbox2

The overload resolution: type cast or boxing-unboxing 1/6

- The third phase allows overloading to be combined with variable arity methods, boxing, and unboxing

See BJPoint3

See VarArgs3