Java Puzzle Ball

Nick Ristuccia

Lesson 3-2
Inheritance
Exercise 3

• Play Inheritance Puzzles 1 through 3.

• Consider the following:
  – What is inheritance?
  – Why are these considered "Inheritance" puzzles?

4 & 5 too, if you want to play more inheritance puzzles
What Is Inheritance?

• Did you notice blue shapes appearing on Green Bumpers?
• **Inheritance** allows one class to be derived from another:
  – A child class inherits the fields and methods of a parent class.
• The parent class is known as the **super class**.
• The child class is known as the **sub class**.
• From playing the game, you've observed and come to understand three key aspects of inheritance and the relationship between super classes and sub classes.
Inheritance Puzzle 1

• In the game:
  – Methods for deflecting the ball which were originally assigned to Blue Bumpers are also found on Green Bumpers.

• In Java:
  – A sub class shares the same methods as the super class.
Inheritance Puzzle 2

• In the game:
  – Green Bumpers contain methods from Blue Bumpers, PLUS methods unique to Green Bumpers.

• In Java:
  – A sub class may have additional methods which aren't found in the super class.
Inheritance Puzzle 3

• In the game:
  – If Green Bumpers inherit unwanted Blue Bumper methods, it's possible to **override** or replace those methods.

• In Java:
  – A sub class may override the methods it inherits.
How Does this Happen?

• Inheritance is enabled in the sub class's declaration.
  – The GreenBumper class before inheritance is enabled:
    ```java
    public class GreenBumper{
        ...
    }
    ```
  – The GreenBumper class with inheritance enabled:
    ```java
    public class GreenBumper extends BlueBumper {
        ...
    }
    ```

• The sub class extends the super class.
  – The BlueBumper class acts as a super class.
  – The GreenBumper class acts as a sub class.
  – The GreenBumper class inherits, or extends, the blueprint of the BlueBumper class.
This is Inheritance!

• You've come to understand the effects of using the `extends` keyword:
  – A sub class shares the same methods as their super class.
  – A sub class may have additional methods which aren't found in the super class.
  – A sub class may override the methods it inherits.

• These are also known as some of the effects of **polymorphism**.

• Keep these three effects in mind as you explore the Java code more deeply in the next part of the lesson.
  – You'll see why inheritance is a useful programming technique.
  – And why we needed to label slots ABCD.