Polymorphism & Method Overloading

Mar. 28, 2017

Overview

- Polymorphism
- Methods
  - Including overriding and overloading

Polymorphism

- Means ‘many forms’
- Difficult to describe, easier to show, so we’ll look at this one in a later lesson
- Allows objects to be treated as if they’re identical

Polymorphism

- In brief though, polymorphism allows two different classes to respond to the same message in different ways
- E.g. both a Plane and a Car could respond to a ‘turnLeft’ message,
  - however the means of responding to that message (turning wheels, or banking wings) is very different for each.

Method Overloading

- A Java class can define several methods with the same name, so long as
  - the parameters are different, the return type is the same

```
public int doSomething(String s, String s2);
public int doSomething(String s);
public int doSomething(int x, int y);
```

  - Java determines the correct method to call by checking the parameters

Method Overloading

- Useful to provide variants of a method that work on different types of object
  - Typically one method does all the work, and the others call it
  - As with constructors, this collects common code into a single method

Method Overriding

- A sub-class can define a method with the same name *and* parameters as as base class
  - Known as overriding

Lab#4-3. Method Overloading

- Create all of possible constructors for the Date class

```
public class Date {
	private int year;
	private String month;
	private int day;

	}
```