CSE 142 Summer 2001

A World of Objects

Introduction

- Java lets us build *simulations* of the world. The *things* in this simulation are called *objects*.
- Objects are just like what we think of as objects:
  - chairs, apples, people, desks, bank accounts, cars, planes, and so on
- Objects have qualities, consist of other objects:
  - The human body, a car, this room
- Objects are *animated*. They can respond to messages that we send them.

Naming Things

- It's often helpful to name things in Java, just like in real life.
- Let's draw some pictures of what name *mean* (what do they refer to?)
- Draw pictures for the following names: Bart, Hal, Lecturer

Simple Things: Numbers

- Numerals vs. Numbers
- Numbers are *quantities*, they are universal concepts.
- Numerals are *names* for numbers.
- We have invented many "numbering" systems over the years:
  - Roman numerals, Decimal numerals, Binary, and so on…
- Note that III (a Roman numeral) and 3 (a decimal numeral) both *name* the same number. Draw a picture.

More on Numbers

- Kinds of numbers: integers and rational numbers
- Imagine that the Java world is already inhabited with all numbers. We can name them using decimal notation.
- So, if we say something like this in Java…

```
3
```

- … we are talking about the number "three."

Naming Numbers

- Sometimes we want to give more useful names to our numbers in Java.
- In Java we name something using this pattern:

  <The kind of thing> <the name> = <the thing we're naming>;
- That looks ugly, so let's do an example:

```
int score = 59;
double temperature = 65.3;
```

- Draw pictures.

Shape Objects

- Many graphics oriented programs need to manipulate *shapes*
- Let's create and name some shapes:

```
Circle theMoon = new Circle(100, 200, 25);
Rectangle theBox = new Rectangle(20, 30, 40, 50);
```

- Note the special word "new". We use the following patterns for creating new objects:

  new <Kind of Object>(<qualities>)

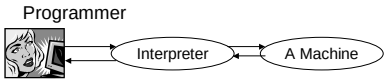
- Why don't we need new for numbers?
- Draw a picture:

```
Rectangle anotherBox = theBox;
```


The Interpreter

- Our first tool is called an *interpreter*
- An interpreter is similar to a language interpreter who will translate your conversation with a speaker of another language
- It does the following (forever):
  - Reads what you type.
  - Translates it and executes or evaluates it
  - Prints the result for you
- Let's try some examples.

Tools In Pictures: Interpreter



Sending Messages

- Let's take a tour (using the interpreter) of some more objects and send them messages.
  - Shapes
  - Pen
  - GWindow
  - BouncingBalls
- We use the following pattern for sending a message:

  <name of the object>.<name of the message>(<message information>)

The Inspector

- We can peek inside of objects by using the inspector.
- The inspector is just a java object that knows how to look inside of other objects.
- Example:

```
Rectangle rect = new Rectangle( );
OBrowser.inspect(rect);
```


Text Objects

- Many programs need to manipulate text, so Java provides us with *Strings* for this purpose.
- Examples:

```
String myName = "Ben";
String momsName = "Ellen";

myName.length( )

myName.charAt(2)
```

Collection Objects

- Many programs need to represent collections of objects.
- Suppose I want to build a course list in Java. Here's one way:

```
ArrayList list = new ArrayList();

list.add("Bob");
list.add("Jill");
```

- Look at it with the inspector…
- Question: How would you find the longest name in our collection?

Simple Things: Characters and Truth Values

- We've seen two other kinds of simples in this lesson:
- Characters:

```
char someChar = 'A';
char anotherChar = 'b';
String name = "Billy";
String anotherName = name.replaceChar('B', 'W');
```

- And truth values:

```
boolean sunny = true;

Rectangle rect = new Rectangle(0, 0, 40, 50, Color.green, false);
```


Summary: Kinds of things we've seen

Kind of thing (Java name)	Used to represent	Example:
int	integers	int x = 34;
double	rational numbers	double y = 34.0;
char	individual characters	char letter = 'x';
boolean	truth values	boolean sunny = false;
String	text	String name = "Bill";
ArrayList	collections of things	ArrayList list = new ArrayList();