Lecture 2: Objects and classes in Java

http://courses.cs.cornell.edu/cs2110

CMS VideoNote.com, PPT slides, DrJava, Book

CMS available. Visit course webpage, click “Links,” then “CMS for 2110.” Not enrolled? Ask Jenna Edwards jls478@cornell.edu to enroll you (needs your netid).

Look at http://cornell.videonote.com/channels/583 to see a previous lecture from fall 2015.

Please download ppt slides the evening before each lecture, have them available in class. Please don’t ask questions on the piazza about that material the day before the lecture!

Please download DrJava (the jar file, not the app). It also requires downloading an old version of Java, so if you don’t want to, don’t.

Book is on reserve in the Uris Library, not the Engineering Library. Engineering Library no longer has (physical) books.

Next week’s recitation!

BEFORE recitation, visit this webpage (part of course webpage)

www.cs.cornell.edu/courses/CS2110/2016fa/online/apiString/01APIString.html

You can get to it from the lecture-notes page of our course webpage, recitation 2.

Read it, watch the short videos —less than 15 minutes!

Come to class with your Cornell id card and your laptop if you have one. Work in groups of 2-3 (your choosing) on questions dealing with the API specs and classes Character and String, with the TA helping, guiding. At the end, the TA will make a note (using your Cornell id card) that you did it.

Java OO (Object Orientation)

Python and Matlab have objects and classes.

Strong-typing nature of Java changes how OO is done and how useful it is. Put aside your previous experience with OO (if any).

This lecture:

First: describe objects, demoing their creation and use.

Second: Show you a class definition and how it contains definitions of functions and procedures that appear in each object of the class.

Third: Talk about keyword null.

Fourth (if there is time). Show you a Java application, a class with a “static” procedure with a certain parameter.

Homework

1. Study material of this lecture.
2. Visit course website, click on Resources and then on Code Style Guidelines. Study
   3. Documentation
   3.1 Kinds of comments
   3.2 Don’t over-comment
   3.4 Method specifications
   3.4.1 Precondition and postcondition
3. Spend a few minutes perusing slides for lecture 3; bring them to lecture 3.

Java OO

References to course text and JavaSummary.pptx

Objects: B.1 slide 10-16
Calling methods: B.2-B.3 slide 18
Class definition: B.5 slide 11

public, private: B.5 slide 11, 12
Indirect reference, aliasing: B.6 slide 17

Method declarations: B.7
Parameter vs argument: B.12-B.14 slide 14

Text mentions fields of an object. We cover these in next lecture

Method calls may have parameters
Method calls may have arguments

Text uses value-producing method for function and void method for procedure.

Get used to terminology: function and procedure
Drawing an object of class javax.swing.JFrame

Object is associated with a window on your computer monitor

Name of object, giving class name and its memory location (hexadecimal).
Java creates name when it creates object

Object contains methods (functions and procedures), which can be called to operate on the object

Function: returns a value; call on it is an expression
Procedure: does not return a value; call on it is a statement

Evaluation of new-expression creates an object

Evaluation of new javax.swing.JFrame() creates an object and gives it the name of the object

If evaluation creates this object, value of expression is JFrame@25c7f37d

2 + 3 + 4

A class variable contains the name of an object

Type JFrame: Names of objects of class JFrame

Consequence: a class variable contains not an object but name of an object, pointer to it. Objects are referenced indirectly.

If variable h contains the name of an object, you can call methods of the object using dot-notation:

Procedure calls: h.show(); h.setTitle("this is a title);
Function calls: h.getX() + h.getWidth()

First class definition

/** An instance (object of the class) has (almost) no methods */
public class C {
    declarations of methods (in any order)
}

Class definition: Describes format of an object (instance) of the class.

This is a comment

Access modifier public means C can be used anywhere

Class definition C goes in its own file named C.java
On your hard drive, have separate directory for each Java project you write; put all class definitions for program in that directory. You’ll see this when we demo.
/** An instance is a subclass of JFrame *\n** An instance is a subclass of JFrame with more methods */

```java
public class C extends javax.swing.JFrame {
    public int area() {
        return getWidth() * getHeight();
    }
    public void setWtoH() {
        setSize(getHeight(), getHeight());
    }
    public void setTitleToDate() {
        setTitle(new java.util.Date().toString());
    }
}
```

An object of class Date contains the date and time at which it was created. It has a function toString(), which yields the data as a String.
null denotes the absence of a name. v2.getName() is a mistake! Program stops with a
NullPointerException You can write assignments like: v1 = null;
and expressions like: v1 == null

Hello World!

/** A simple program that prints Hello, world! */
public class myClass {
  public static void main(String[] args) {
    System.out.println("Hello, world!");
  }
}

args is an array of String elements

We explain static next week. Briefly: there is only one copy of procedure main, and it is
not in any object