

CS100J. Lecture 2, 24 September 2008. Objects & classes

Reading for this lecture: Section 1.3. It's most important that you **study this section over the weekend** and **practice** what is taught using DrJava.

PLive: Activities 3-3.1, 3-3.2, 3-3.4 (not 3-3.3), 3-4.1, 3-4.2.

Summary of lectures: On course page, click on "Handouts" and then "Outline of lectures held so far".

We will hand out assignment 1 next week. No need to look at it yet!

Quote for the day:
Computers in the future may weigh no more than 1.5 tons.
 --Popular Mechanics, forecasting the relentless march of science, 1949

1

About CMS: Course Management System.
 Use to maintain grades, handle submitted assignments, post grades, handle regrades, etc. Developed by the CS Department.

URL: <http://cms3.csuglab.cornell.edu>

Click on "Secure extended login". You will be asked for your cornell netid and password. After entering it, you will either be in the CMS and see the course description or you will see on the right something like this:

AEWs

Sign up for the 1-credit AEW sections for CS100J.

Two hrs per week.
Nothing else.
Not remedial.

CMS Overview
 My Courses
 Com S 100J (student)

If it lists CS100J, click on it. If it doesn't, email Amy Fish at amyfish@cs.cornell.edu and ask her to register you in the CMS for CS100J.

2

Two aspects of a programming language

- Organization – structure
- Procedural – commands to do something

Example: Recipe book

- Organization: Several options; here is one:
 - Appetizers
 - list of recipes
 - Beverages
 - list of recipes
 - Soups
 - list of recipes
 - ...
- Procedural: Recipe: sequence of instructions to carry out

Parts to this course

structural
 objects
 classes
 methods
 inheritance

procedural
 assignment,
 return,
 if-statement
 iteration (loops)
 recursion

miscellaneous
 GUIs
 exception handling
 Testing/debugging

3

A class is a file-drawer. Contents: manila folders, each containing the same kind of information

manila folder: an **object** or **instance** of the class

4

A class is a file-drawer. Contents: manila folders, each containing the same kind of information

name, address, owes: **variables**, called **fields** of the folder

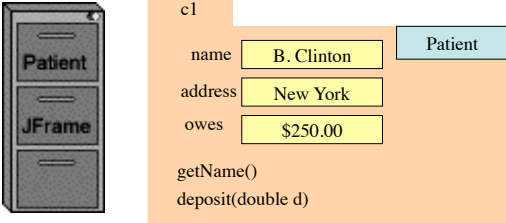
5

A class is a file-drawer. Contents: manila folders, each containing the same kind of information

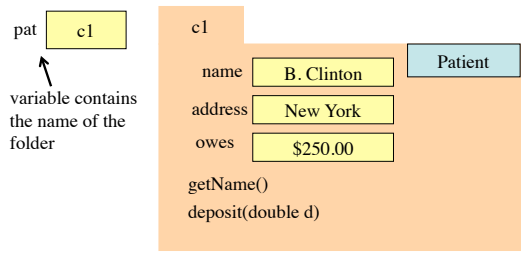
Instructions to be carried out by different people:
 change the name, get the name, bill the patient, receive money from patient, insert teeth xrays into the folder, ...

6

A class is a file-drawer. Contents: manila folders, each containing the same kind of information

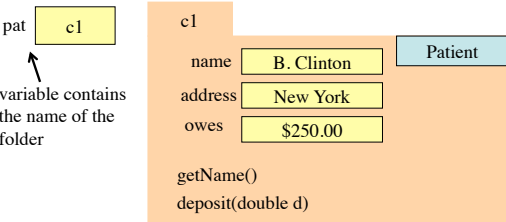


Instructions to be carried out by different people: methods.
 getName is a **function**; it returns a value.
 deposit is a **procedure**; it does some task, doesn't return value



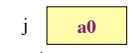
pat.getName() function call. Its value is "B. Clinton"

pat.deposit(250.0); procedure call. Subtract 250.0 from field owes.



new Patient() An expression: create a new folder (put in file-drawer Patient) and give as the value of the expression the name of the folder.

pat= new Patient(); A statement: evaluate newPatient() and store its value (the name of the new folder) in variable pat.



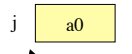
An object (manila folder) of class Javax.swing.JFrame is associated with a window on your computer monitor. It has (among others) these functions:

We will demo the use of each of these methods

getHeight() getWidth() getX() getY() getTitle() isResizable()

and these procedures:
 show() hide()
 setTitle() setSize(int, int)
 setLocation(int, int) setResizable(boolean)

In groups of 2, draw an object (manila folder) of this class, and put the name **a0** on its tab.



`j = new javax.swing.JFrame();
j.show();
...`

Expression **new JFrame()**
Creates new folder and put in file drawer JFrame.
Statement **jf= new JFrame();**
Create new folder, as above, and place its name in variable jf (jf should have first been declared).
Thereafter, use
`jf . method-name (arguments, if any)`
to call methods of folder (object) jf.

- Read section 1.3.
- Practice what we did in class in DrJava.
- Try the self-review exercises on page 40.

package: A collection of classes that are placed in the same directory on your hard drive. Think of it as a room that contains file cabinets with one drawer for each class.

package **java.io** classes having to do with input/output
 package **java.net** classes having to do with the internet
 package **java.awt** classes having to do with making GUIs
 package **javax.swing** newer classes having to do with GUIs

To reference class JFrame in package javax.swing, use:

`javax.swing.JFrame`

Instead: `import javax.swing.*;`

Then use simply JFrame