CS 121 - Intro to Programming: Java - Lecture 6

Announcements

TA office hours today 2-6 or 7; tomorrow 9-5

Programming Assignment 2 due Saturday night(!) - but do it earlier... Hand in through OWL..now up..

Programming Assignment 3 up tonight or tomorrow..

OWL Ch 3 up, due Monday, 11 PM

CNS freshman meetings (MWF in Sept, Oct):
http://cns.umass.edu/students/academic-advising/forms/freshman-survey

LRC - up and running! 10th floor of the library. (starts when? - this Sunday)

http://www.umass.edu/lrc/ is the LRC home page. Click on tutoring, SI schedules...
Beanshell news

Bad: still not working for everyone

Good: If it does work for you - sort of - you can print using

BSHSystem.println(...) & BSHSystem.print(...) (instead of System.out.println(...) etc.)
import java.util.Scanner;
// makes Infant obj interactively, then ups kid's age
public class KidOnFly{
    public static void main(String[] args){

        String name; int age;
        Scanner kidReader = new Scanner(System.in);

    }
}

import java.util.Scanner;
// makes Infant obj interactively, then ups kid's age
public class KidOnFly{
    public static void main(String[] args){
        String name; int age;
        Scanner kidReader = new Scanner(System.in);

        System.out.println("enter name, followed by age");
        name = kidReader.next();
        age = kidReader.nextInt();
        Infant kid = new Infant(name,age);
        kid.anotherMonth();
        System.out.println(kid.getName() +" is " + kid.getAge());
    }
}
Program 2 (AsnTwo on website) due Saturday night!

What do you need to do?

Watch the “Egg Movie” in 3.2

Copy - Paste - Save - Compile Banking.java (on site)

Do this in DrJava. This should place a compiled version of Banking.java (Banking.class) in your directory.
Program 2 (AsnTwo on website) due Saturday night!

Then: create BankingDriver, with main inside.

Do a simple version first:

create nicksAcct - a banking object.

then print out statistics on his acct (bullet 5 from the website write-up)

If this works, you are ready to take on the main event
Program 2 (AsnTwo on website) due Saturday night!

Complete the project: make carolsAcct, and add the statements that capture the script described in the project write-up.
Working from an API

Constructor Summary

```
TreeHouse(int theWidth, int theLength, int theHeight, java.lang.String tree)  
the tree house constructor
```

What does all of this mean?

```
TreeHouse t = new TreeHouse(3,4,5, "maple");
```

Or

```
TreeHouse t = new TreeHouse(3, 4, "maple", 5);
```
Primitive types char and int
char is discrete - $2^{16}$ bit patterns (two bytes)
Every character has a position in the list of chars

'A' is at position 65

(int)'A' is 65
(char)65 -> A

(char)66 -> B

'A'+1 -> 66
(char)('A'+1) -> B
Java arithmetic

Exactly as you might expect, with some caveats:

5/2 is different from 5/2.0

2  2.5

% is the mod operator (remainder upon division)

578 % 10 is 8

7 % 4 is 3

X % 2 is 0 or 1, depending if X is even or odd
The return problem:
Difference between
   str.toLowerCase();
And
   str = str.toLowerCase();
Java and the notion of Encapsulation

• Encapsulation = Information Hiding
• Serves the user: think about nextDouble()
• Serves the implementor: just meet the API
• Underlying principle for code recycling
• Underlying principle for code hygine
Working backwards

[a box of books has a count of books, and a weight of books]
I need a principal class to support this driver... so: what, precisely is needed?

```java
public class BookWork {
    public static void main() {
        BoxOfBooks b = new BoxOfBooks(43, 70.3);
        System.out.println("There are " + b.getCount() + " books.");
        System.out.println("They weigh " + b.getWeight() + " pounds.");
    }
}
```
Working backwards

[a box of books has a count of books, and a weight of books]

```java
public class BookWork {
    public static void main() {
        BoxOfBooks b = new BoxOfBooks(43, 70.3);
        System.out.println("There are " + b.getCount() + " books.");
        System.out.println("They weigh "+ b.getWeight() +" pounds.");
    }
}
```
public class BoxOfBooks{

}
public class BoxOfBooks{

    private int count;
    private double weight;

}

public class BoxOfBooks{

    private int count;
    private double weight;

    public BoxOfBooks(int c, double w){
        count = c;
        weight = w;
    }
}
public class BoxOfBooks{

    private int count;
    private double weight;

    public BoxOfBooks(int c, double w){
        count = c;
        weight = w;
    }

    public int getCount(){return count;}
}