CS 211: Java Syntax Tour

Chris Kauffman

Week 2-2
... I can say that at my workplace I’ve seen more than one freshout who clearly hadn’t made it through college without significant assistance from Stack Overflow and other people’s blogs. None of them lasted very long. Perhaps knowing how to solve problems for yourself isn’t necessary to get a college degree nowadays, but it’s surprising how useful it can be in a career where you solve problems for a living.

▶ bunderbunder, Discussion of cheating using StackOverflow on http://news.ycombinator.com/item?id=4910406
Goals

Today
Project 1 necessities
  ▶ Arrays
  ▶ Static functions

Upcoming
  ▶ Lab 02: Quiz on Basic Java Syntax
  ▶ Project 1 due Sunday
    ▶ Field Questions
    ▶ Submitting to Blackboard

Reading
Building Java Programs Chapters 1-7
Last Time

- Comments
- Statements/Expressions
- Variable Types
  - little types, what about Big types?
- Assignment
- Basic Input/Output
  - Input?
- Conditionals (if-else)
- Iteration (loops)
- Aggregate data (arrays, structs, objects, etc)
- Function Declarations
- Library System
Class exercise available on Practicelt! web site by following this link. Write a program in a class named StarFigure that produces the following output using for loops.

```
////////////////\\\\\\\\
////////////********\\\\\\
////////****************\\\\
////************************\\
********************************
```
Aggregate Data

Define  Now there's a type bleh, it looks like blah
Declare Here is a variable, its type is bleh
Assign  Element foo of variable bar gets value blip
Access  Retrieve element foo of variable bar
Arrays - Multiple of the same kind of thing

See ArrayDemo.java

Define Now there’s a type `bleh`, it looks like `blah`
  ▶ Done for you: part of the java language

Declare Here is a variable, it’s type is `bleh`

```
int ia[] = new int[3];
double doub[] = new double[10];
boolean bool[] = new boolean[4];
```

Assign Element foo of variable bar gets value `blip`

```
ia[0] = 1;
doub[2] = 1.2345;
bool[3] = true;
```

Access Retrieve element foo of variable bar

```
int i = ia[1];
double d = doub[4];
boolean b = bool[0];
```
Length

Arrays carry their length

It's an int (or long?).

```java
int ia[] = new int[3];
System.out.println(ia.length);
int len = ia.length;
for(int i=0; i<ia.length; i++){
    System.out.print(ia[i]+" ");
}
```

Can't change length

```java
// Compile ERROR
ia.length = 20;
```

Why not?

Can cause runtime errors

```java
ia = new int[5];
ia[10] = 12;
Exception in thread "main"
java.lang.ArrayIndexOutOfBoundsException: 10
at ArrayDemo.main(ArrayDemo.java:23)
```
Array Goodies

Declare and Initialize

```
int a[] = {1, 2, 3, 4};
```

Initialize Dynamically

```
int b[];
...
  b = new int[]{7, 6, 5};
myFunc(new int[]{3,1,4,1,5,9});
```
Functions

Are parameterized code

- Referred to as **methods** in java jargon
- Give me some stuff (arguments)
- I’ll give you something back (return value)
- Java: specify types for arguments and return
- User `return` to finish function and give value back
  - Immediately ends function (even inside loop)
  - Useful for project 1
Functions

Live inside classes, see FunctionDemo.java

// Sum up an array
public static int sumIntArray(int a[]){
    int sum = 0;
    for(int i=0; i<a.length; i++){
        sum += a[i];
    }
    return sum;
}

For now, use the magic word static for functions
  ➤ Omitting static changes the meaning of functions significantly
The void

Sometimes a method gives nothing as an answer.
  ▶ Return type is void

public static void downHere()
{
    System.out.println("Calling down here");
}
A Tricky Example

What’s the difference? What gets printed?

**Defined**

```java
public static
void doubler1(int x){
    x = 2*x;
}

public static
void doubler2(int x[]){
    x[0] = 2*x[0];
}
```

**Used**

```java
public static void
main(String args[]){
    int r = 10;
    doubler1(r);
    System.out.println(r);

    int s[] = {20};
    doubler2(s);
    System.out.println(s);
}
```

Code is in Doubler.java
Playing with Functions

It's easy to play with static functions in DrJava's interactive loop. Make sure to use ClassName.functionName(param,parm2).

Welcome to DrJava. Working directory is ...
> Collatz.steps(6)
8
> Collatz.steps(11)
14
> int c6[] = {6, 3, 10, 5, 16, 8, 4, 2, 1};
> Collatz.verify(c6)
true
> int badStart[] = {-4,2};
> Collatz.verify(badStart)
false
Array and Function Practice

Good exercises: functions that manipulate arrays

- BJP3 Self-Check 7.28: arrayMystery5
- BJP3 Exercise 7.6: stdev
- BJP3 Exercise 7.12: priceIsRight
- BJP3 Exercise 7.13: longestSortedSequence