

Computer Science 210: Data Structures

Java review

- Base types
 - boolean, char, byte, short, int, long, float, double
- Class
 - a type; a cookie cutter; blueprint from which individual objects are created
 - A class does not actually exist; it is just a “pattern”
 - A class contains data and methods
- Object
 - an instance of a class; the actual cookie
 - instance variables
 - creating an object
- Methods
 - Declaring methods; parameters, return types
 - Constructor methods; main method
- Expressions
 - operators, the dot operator, casting
- Statements:
 - if, switch, loops, return, break, continue
- Arrays

Base types

- boolean
 - true or false
- char
 - 16 bit character
- byte
 - 8-bit signed integer
- short
 - 16-bit signed integer
- int
 - 32-bit signed integer
- long
 - 64-bit signed integer
- float
 - 32-bit floating point number
- double
 - 64-bit floating point number

Declaring

- variables

```
<type> <variable-name>;
```

- constants

```
static final int MONDAY = 0;
```

- classes

```
[abstract | public | final] class <class-name> extends <super-classname> implements  
<interface_1> <interface_2>..... {  
  
    //instance variables  
  
    //methods  
  
}
```

- abstract class: class has only abstract methods (later)
- final class: can have no subclasses (later)
- public class: class can be instantiated and extended by anything in the same package or by anything that imports the class

Declaring objects

//class definition

```
class Gnome { ...};
```

//declares an object g of type Gnome

```
Gnome g;
```

//object g does not yet exist; to create an object call new

```
g = new Gnome(...);
```

- **Constructor:**
 - a special method that is used to create objects
 - the constructor allocates memory to hold the object and returns a reference to this memory ; this address is then stored in the object variable (g)

Number objects

- We sometimes want to (need to) store integers as objects

```
//an object that represents integer 10
```

```
x = new Integer(10);
```

Instance variables

- Instance vars represent the data associated with the object
- scope
 - **public:**
 - anyone can access public instance variables
 - **private:**
 - only methods of the same class (not subclass) can access private vars
 - **protected**
 - only methods of the same package and subclasses can access protected vars
- **static**
 - a static variable is associated with the class
 - used to store global information about the class
- **final**
 - a constant
 - must have an initial value, which cannot be changed

Methods

- method = code that can be called on a particular object
- declaring methods
 - parameters
 - method modifiers:
 - public , protected , private. abstract , final , static
 - return values and types
- constructor methods
 - a special kind of method that is used to initialize newly created objects
- main method
 - needed in classes that are meant to define stand-alone programs
- java Gnome
 - Java-system invokes the main method in class Gnome
 - main must be public and static

Operators

- assignment
 - `a = b;`
- dot
 - `obj.methodname(...)`
- arithmetic
 - `+, -, *, /, %`
 - `++, --`
- logical operators
 - `<, <=, >, >=, ==, !=`
- operators on booleans:
 - `!, &&, ||`
- bitwise operators
 - `>>, <<, &, |`

Summary

- casting
- if
- break
- continue
- switch
- for loops
 - for (initialization; condition; increment)
 - body
- while loops
- output
 - class System.out
 - System.out.print
- input
 - class System.in and Scanner

- Reading:
 - Java cheat-sheet <http://www.cs.princeton.edu/introcs/11cheatsheet/>
 - (find link on class website)
- Bring any questions to class next time