COE318 Lecture Notes Week 4 (Week of Sept 22, 2014)

Announcements (REPEAT!)

- Quiz (5% of total mark) on Wednesday, September 24, 2014. Covers weeks 1–3 and labs 1–3.
- Midterm (20% of total mark) on Wednesday, October 15. Covers weeks 1–6.
-

Topics

- More about variables
- More about arrays
- Notes on Lab 3
- Notes on Lab 4
- Q&A

More about variables

What is common to ALL variables?

- Any variable must be *declared* before it is used.
- The *declaration* gives the variable a *type* and a *name*. And, since Java is a strongly-typed language, the type cannot be changed (although it may be "cast", a topic we will discuss later on.)
- The name of variable should be sensible. Choosing a good name makes it easier to write, read and maintain code.
- By convention, variable names should be lower case. If the sensible name is more than one word long, the *camel case* convention should be followed. For example, if a variable describes a person's "hair colour", a sensible name would be `hairColour`.
- One exception to this convention is for constants; it is common practice to give the name in all uppercase. For example, in the source code Math.java we read:

```
public static final double PI = 3.141592653589793;
public static final double E = 2.718281828459045;
```

More about arrays

Arrays are fixed in length and hava read-only length "instance variable"

- `int [] arr;  //Declare "arr" to be an array of ints`
- `arr = new int[3]; //Make it an array of 3 ints.`
- Once created, the number of elements in the array cannot be changed.
- The number of elements in an array can always be obtained with the `length` attribute. For example:
- `int i = arr.length; //Would make "i" == 3`
- Note that `length` cannot be modified (i.e. `arr.length++` is illegal; it would not change the number of elements in the array.
- Note also that some objects (like Strings) have a `length()` *method*.
- For example, consider:

```
String [] strings = {"abc", "defg", "xy"};
int i = strings.length; //i would be 3
int j = strings[2].length(); //j would be 2
```

Notes on Lab 3

- Demonstration in class.

Notes on Lab 4

- Demonstration in class.

Questions

1.