Arrays
Ch 3.8

Why science teachers are not asked to monitor recess.
Highlights

- Arrays

```java
double[] x = new double[10];
```
Arrays

Arrays are convenient ways to store similar data types

Much like how we have been thinking of Strings as a list of characters

Arrays are also indexed starting from zero
Arrays vs. classes

Arrays store multiple things of similar type (i.e. 20 ints)
Can easily iterate through array's items

Classes store different types that have a logical connection (i.e. name and age for a person (String and int))

For classes, must specify all variables
Arrays - declare/initialization

When making an array, you need both a type and a length.

The format for making an array is below:

```java
double[] x = new double[10];
```

- **Type, []** means array
- **variable name**
- **length of array**
Arrays - elements

To access an element of an array, use the variable name followed by the index in [ ]

```java
int[] x = new int[5];
x[0] = 2;
x[1] = 8;
```

(See: SimpleArray.java)
Arrays - elements

Note: the use of the [ ] is slightly inconsistent

The first time only, it tells the size:

```java
double[] x = new double[10];
```

After that, if you ever give a [ ], it will try to look up a single index (single item)

```java
x[1] = 2;
```
Arrays - manual initialization

Arrays can be initialized by the following:

```java
int[] x = {1, 2};
```

If you try to access an array element that has not been initialized, your program will crash.

Arrays have a `.length` variable (similar to Strings)

(See: ArrayPrintf.java)
Arrays - length

A char array is not a String

While the important data in String is a char array, classes have methods unlike arrays

Arrays do have one piece of “extra” information: it's own length

```java
char[] x = new char[7];
System.out.println(x.length);
```

(See: ArrayAverage.java)
Arrays - Lazy printing

The most fundamental way to print arrays is to use loops (as we have been doing)

But, as arrays are very common, there exists a lazy way to print basic arrays:
Arrays.toString()

```java
int[] x = {1, 6, 4, 7, 2, 4};
System.out.println( Arrays.toString(x) );
```

(See: Fibonacci.java)