11/14/2007
Let's look at solutions to the interclass problem.
Sitting and listening vs. following along.
We saw in Alice how there are situations where we needed to have collection of different things.

This is a general need in programming. We need to have some simple way of dealing with multiple objects.

Java actually provides many ways for dealing with these collections of objects.

Today we begin with the one type of collection actually built into the language instead of the libraries: arrays.
Just like in Alice, an array in Java is a fixed length collection of things. Unlike Alice, arrays work well in Java.

We can make an array type by placing square brackets after any type in Java.

- `int[] a;`
- `String[] strs;`

Arrays are reference types so we need to instantiate them. During instantiation we tell Java how large the array is.

- `a=new int[10];`
- `strs=new String[6];`
We get the elements out of an array by placing an integer expression in the square brackets after a variable name.

- `a[7]=5;`

The indexes go from zero to the length minus one.

Arrays know their length and you can get it with the length property.

It is common to have loops that run through the contents of arrays.

- `for(int i=0; i<a.length; i++) { ... }`
You can make arrays of arrays (or arrays of those). The syntax is just a logical extension of the 1-D arrays.

- `int[][] a2;`
- `String[][] strs2;`

We can allocate rectangular 2-D arrays with a simple syntax as well.

- `a2=new int[10][20];`
- `strs2=new String[5][4];`
Let's use an array inside of our payroll application.
Write a method that takes an array of ints and returns the sum of all the elements in the array.

Interclass Problem – Do problem 12.2. You can put a number at the top of the file so that it is easy to store all the friends in an array.