Today

• Beginning of primitive arrays
Short Questions?
Questions?
ArrayLists

ArrayList class that we have already seen:

• Ordered list of objects
• List is indexed from 0 .. (Size-1)
• Can access each individual object (through the get() method)
• Can change the object reference (through the set() method)
• Automatically handles reorganization of the list as items are added or removed
The ArrayList class is implemented using Java Arrays (as are many other classes)

Java Arrays:
• Store lists of primitive data types (including references)
• All items in the list are of the same type
• Fixed in size (you declare this ahead of time)
• All items occupy a contiguous region of memory (makes for efficient access)
Arrays

// Allocate an array of size 7:
int[] intList = new int[7];

// Set element values
intList[4] = 9;
intList[5] = 11;

// Accessing the elements
System.out.println(intList[5] + intList[4]);
Arrays

// Allocate an array of size 7:
int[] intList = new int[7];

// Set element values
intList[7] = 11;  // Throws an exception

// Can ask an array how many elements it has
System.out.println(intList.length);
Arrays

// Allocate an array and initialize its contents:
int[] intList = {5, 6, 8, 21, 3, 42};

// How many are in the list?
System.out.println(intList.length);
Array Pitfalls

• Arrays cannot be resized
  • If you want to resize, then you must create a new Array and copy the contents of the old array over to it

• Indexes start at zero

• Using indexes that do not exist: IndexOutOfBoundsException
Wrap Up

Due:
• Project 5: next Tuesday
• HW 7:

Next time:
• Arrays continued