Double Trouble

- so far our data has consisted of string literals and literals and variables of data type `double`
- `double` works well for representing real numbers
- **BUT** `double` variables really store approximations of real numbers

Approximations

- EX: try **Lab 6b Run 1** with a calculator and the user has $7.55 left over
- but your Java program will give you the value `7.549999999999997`
- off by `0.000000000000003`
- will see later how to hide the excess digits

Exact Numbers

- sometimes data values are whole numbers, or `integers`
- values in: ..., -3, -2, -1, 0, 1, 2, 3, ...
- EX: how many children the user has
- EX: a count of the number of times your program has performed a certain act
- computer can represent integers exactly; approximation not needed
Our Second Data Type

- Variables with a data type of int are used to store/represent integers
- Integers
  - Examples: 678, -1987, 2
- FYI: can represent any integer from -2,147,483,648 to 2,147,483,647

Variable Declaration

- Syntax:
  - `dataType variableName;`
- Example:
  - `int numToes;`
  - Assigns the label `numToes` to a memory location
  - Tells Java that integers will be stored at that location

Integer Assignment

- Example:
  - `int a, b, c;`
  - `a = 5;`
  - `b = a + 3 * 4;`
  - `c = 4.5;`
  - Cannot assign a double value to an int variable

Int Gotcha’

- Arithmetic expressions involving int’s have an int result
- Result will be truncated (not rounded)
- Example:
  - `double realNum;`
  - `realNum = 5 / 2;`
  - `realNum` will be 2.0, not 2.5
- If you want a real number result, use real numbers on the right-hand side
Integer Output

- the `Output` class provides a `showValue` method for outputting `int`'s
- syntax:
  
  ```java
  Output.showValue("label", variableName);
  ```
- EX:
  ```java
  int n = 12;
  Output.showValue("n is ", n);
  ```
  - outputs: `n is 12`

Integer Input

- the `Input` class provides the `readInt` method for getting an input integer
- works like `readDouble`, except it returns an `int`
- syntax:
  ```java
  variableName = Input.readInt("prompt");
  ```
  - assigned user-input integer
  - method invocation
  - tells user what to input

readInt

- EX:
  ```java
  int count;
  count = Input.readInt("Enter the count");
  ```
  - prompts user with "Enter the count"
  - after this statement executes `count` will have the user-input integer value

Module 11 Vocabulary

- integer
- int
- integer literal
Questions?

Email Kevin at sahrk@sou.edu