Did you?

- Print & read Course Outline?
- Read Assignment 1?
  - Part 1 is Due before your lab tomorrow!
- Read 1st Chapter of your Textbook??
  - Observe a Reading List is now posted
    - Do the self-check exercises from textbook?

Building Java Programs
Chapter 1
Introduction to Java Programming

Building Java Programs
Chapter 2
Data & Expressions

Check reading list – Now includes Chapter 2!

Comments

- **comment**: A note written in source code by the programmer to describe or clarify the code.
  - Comments are not executed when your program runs.

- Syntax:

```
// comment text, on one line
```

or,

```
/* comment text; may span multiple lines */
```

- Examples:

```
// This is a one-line comment.

/* This is a very long
   multi-line comment. */
```

Using comments

- Where to place comments:
  - at the top of each file (a "comment header")
  - at the start of every method (seen later)
  - to explain complex pieces of code

- Comments are useful for:
  - Understanding larger, more complex programs.
  - Multiple programmers working together, who must understand each other's code.

Comments example

```
/* Suzy Student, CS 101, Fall 2019
   This program prints lyrics about ... something. */

public class BaWitDaBa {
    public static void main(String[] args) {
        // first verse
        System.out.println("Bawitdaba");
        System.out.println("da bang a dang diggy diggy");
        System.out.println();

        // second verse
        System.out.println("diggy said the boogy");
        System.out.println("said up jump the boogy");
    }
}
```

Data and Expressions

Declaring and Assigning

```
public class AssigningValues {
  public static void main(String[] args) {
    int firstValue = 17;
    int secondValue = 2;
    System.out.println("The value is: " + firstValue);
    firstValue = 523;
    System.out.println("The val                alue);
    firstValue = secondValue;
    System.out.println("The val                alue);
    System.out.println("The other value is: " + secon
    firstValue = secondValue *5;
    System.out.println("The value is: " + firstValue);
    firstValue = firstValue +1;
    System.out.println("The value is: " + firstValue);
  }
}
```

Assigns a value

Sets up a memory location to hold the value

Doing Math

Interesting Math !!

Java's Primitive Types

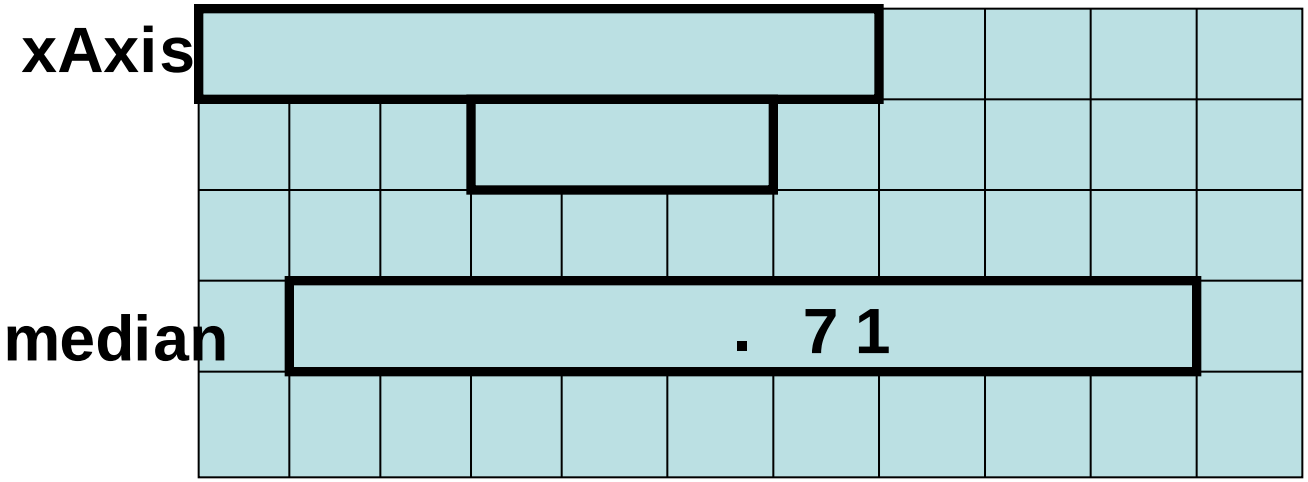
We usually use four kinds (types) of values:

- **int** — A positive or negative integer (32-bit)
- **double** — A "double-precision" floating-point value (64-bit).
  - A real number -- has a fractional part (e.g., 3.141592654).
- **boolean** — A 1-bit value: true or false.
  (actually stored in a 32-bit int)
- **char** — A UNICODE text character (16-bit).

Also

- **byte** — 8-bit integer,
- **short** — 16-bit integer,
- **long** — 64-bit integer,
- **float** — 32-bit floating-point.

Memory and Declaration




```
int xAxis;
double median = 7.1;
```

Your Turn

```
public static void main(String[] args) {
  int currentYear = 2006;
  int lastyear = currentYear - 1;
  int thatYear = 1943;
  int tempYear;
  System.out.println("I think there is a world market for");
  System.out.print(" maybe ");
  System.out.println("five computers.");
  System.out.println("   By Thomas Watson");
  System.out.print("   In the year, thatYear");
  System.out.println("   Or was it" + thatYear);
  tempYear = currentYear - thatYear;
  System.out.print("  In any case it was tempYear or ");
  System.out.println
  tempYear = lastye
  System.out.print                                    "ago");
}
```

```
I think there is a world market for
 maybe five computers.
   By Thomas Watson
   In the year, thatYear   Or was it1943
  In any case it was tempYear or 63ago
  Last year that was only 62ago
```

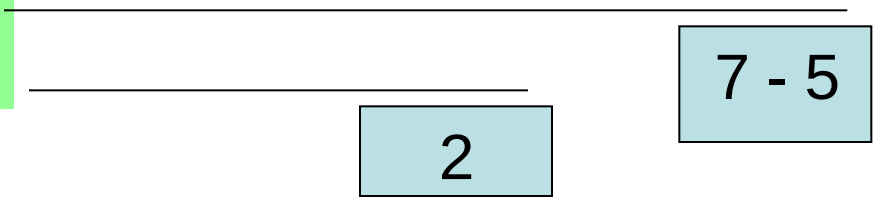
Group Activity:
Find a partner

Evaluating Expressions

Computations occur in a specific order, one at a time

- default Java rule : evaluate **left-to-right**:

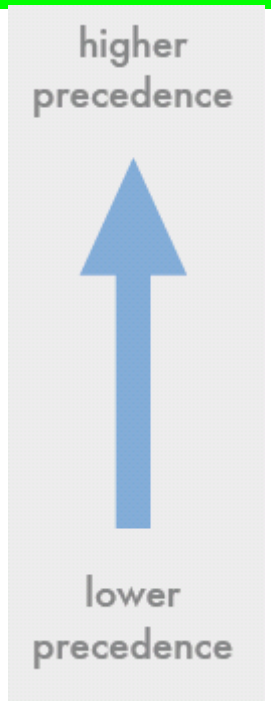
Example: 3 + 4 - 5



- BUT: Every operator has a *precedence* that may override
  - Operators with a “higher” precedence always execute before those with “lower” precedence.

Operator Precedence

Operator Kind	Examples
unary sign modifiers	`+3, -12`
binary multiplicative	`7*3, 2/5, 9%4`
binary additive	`1+2, 3-4`



Example: 29 - 7 * 3

Simple Left to Right:
22 * 3 = 66

Using Precedence:
29 - 21 = 8

Precedence Examples

- Evaluate:

  17 + -5 * 2

  12 - 41 % -18 / 2

Try again:

```
public class VariableTester {
    public static void main ( String[] args ) {
        int perhaps, maybe;
        int niceValue = 4;
        perhaps =  niceValue++;
        maybe = -17 % niceValue;
        System.out.println("perhaps: " + perhaps);
        System.out.println("maybe: " + maybe);
    }
}
```

perhaps: 4
maybe: -2

Assignment 1 Discussion

Computing Concepts

- Computers execute simple instructions known as machine code. Examples:
  - "ADD 1 to value x"
  - "MOVE value y to location z"
  - "IF t = 0, then jump to instruction I"
- Computers only know about numbers—integer values (e.g., 1, -2, etc.), floating-point values (e.g., 3.1415), addresses of memory and instructions.
- Even machine instructions are represented as numbers.

Computer Counting: What is a bit?

- Binary digit
- 8 bits = 1 byte
- 1024 bytes = 1 kilo byte (Kbyte)

$$= 2x2x2x2x2x2x2x2x2x2 \text{ bytes}$$
$$= 2^{10} \text{ bytes}$$

Computer Counting: Powers of 2

- $2^{10} = 1024 = 1K$ (Kilo)

Not to be confused with:

$10^3 = 1000 = 1K$

- $2^{20} = 1,048,576 = 1M$ (Mega)

$10^6 = 1,000000 = 1M$

- $2^{30} = 1,073,741,824 = 1G$ (Giga)

$10^6 = 1,000,000,000 = 1G$

- $2^{40} = 1,099,511,627,776 = 1T$ (Tera)

$10^9 = 1,000,000,000,000 = 1T$