The Learning House

427 S. 4th Street #300

Louisville KY 40202

Java Basics Unit

Lesson 3 - Programs, Statements and Variables

SOFTWARE GUILD

Language

- Does anyone know more than one natural language?
- What makes up a natural language?
- What things do languages have in common?
- Your first computer language:
  - 3 things at once:
    - Industry language
    - How to talk about computer languages
    - The computer language itself

Objectives

- Understand the concept of a ‘program’
- Become familiar with the structure of a simple Java program
- Understand the purpose of the ‘main’ method
- Understand the purpose and function of ‘statements’
- Understand the concept of ‘variables’
- Understand primitive types
- Understand mathematical expressions

Approach

- Bottom up — start with smallest pieces of language
- Build vocabulary
  - “See Spot run!”
  - Scales in music
- Add larger constructs
  - Sentences, paragraphs
  - Melodies, harmonies
- Can’t write the next great American novel if you only know 6 words...

Concepts

- Computers
- Users and programmers
- Software and hardware
- Data vs. information
- Programs and programming
  - Organization and structure
  - Solving problems
- Models and metaphors
- Specifications
- Syntax and semantics
- Which language is best?

Java Program

- Comments
- Identifiers
- Keywords
- Data types
- Literals
- Variables
- Expressions
  - o Operators and operands

Software as a Story

- Should be:
  - o Easy to read
  - o Well formatted — it should look good
  - o Good punctuation and grammar
- “Programs should be written for people to read, and only incidentally for machines to execute.” –Abelson
- “If the program doesn’t run it, it is broken. If people can’t read it, it will be broken. Soon.” –post on stackoverflow

Comments

- Single line : //
- Block: /* */
- Don’t comment the obvious
- Examples

Identifiers

- Used to name a variable, method, class, or package:
  - Cannot span multiple lines
  - Only numbers, letters, underscores, dashes, and dollar signs
  - Must start with a letter, underscore, dash, or dollar sign
  - Cannot contain spaces

Primitive Data Types

Type	Contains	Default	Size	Range
boolean	true or false	false	1 bit	NA
char	Unicode character	\u0000	16 bits	\u0000 to \uFFFF
byte	Signed integer	0	8 bits	-128 to 127
short	Signed integer	0	16 bits	-32768 to 32767
int	Signed integer	0	32 bits	-2147483648 to 2147483647
long	Signed integer	0	64 bits	-9223372036854775808 to 9223372036854775807
float	IEEE 754 floating point	0.0	32 bits	±1.4E-45 to ±3.4028235E+38
double	IEEE 754 floating point	0.0	64 bits	±4.9E-324 to ±1.7976931348623157E+308

Literals

- Represents a data item in source code
- There are six:
  - **Boolean**: ‘true’, ‘false’
  - **Character**: ‘Z’, ‘\n’, ‘u\000’
  - **Floating Point**: 3.14, 3.45E-89, 3.66d, 7,89f
  - **Whole Number**: 189, 0x45AF
  - **Null**: null
  - **String**: “this is a string”, “this is too”

Variables

- Named piece of memory in which you can store a value
- Declaration: `data_type variable_name;`
  - `int score;`
- Declaration and Assignment:
  - `data_type variable_name = some_value;`
  - `int score = 45;`
- Assignment After Declaration:
  - `score = 52;`

Expressions

- Expressions are bits of code that, when run, result in some type of value
- Example:
  - `int sum = 5 + 2;`
- The following can be used to create expressions:
  - Math operators (this lesson)
  - Conditional operators (next lesson)
- Methods are also expressions (more about methods later)

Math Operators

- +, ++, +=
- -, --, -=
- *, *=
- /, /=
- = (assignment)
- Let's play with some expressions...

Some Preliminaries

- We need some functionality right away
- Might not make sense right now — take it on faith
- Simple console input/output example
  - o System.out.println(..)
  - o Scanner

Example: Adder

- Simple example that adds two numbers
- Hard code at first
- Then, add more flexibility

WindowMaster

- New Project
- Type it together
- Analyze line-by-line

GitHub

- Account setup
- Grant permissions
- Look around

Programming by Doing

- www.programmingbydoing.com
- Problem Sets:
  - One NetBeans project per set:
    - Basics and Printing
    - Variables
    - Keyboard Input
- Work in pairs
  - One keyboard
  - Trade off navigating/driving after each problem

WindowMaster Reloaded

- Pair up
- Create a plan first; discuss with instructor
- Implement your plan