AP Computer Science A Study Guide Unit 1

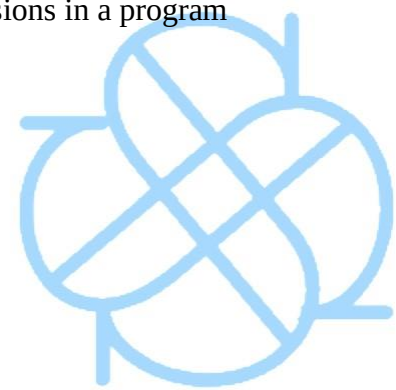
From Simple Studies: https://simplestudies.edublogs.org &

@simplestudiesinc on Instagram

Primitive Types

Learning Objectives

- System class methods print output to the console.
- String literals
- Primitive Data Types
- Declaring variables to different data types
- Use of arithmetic expressions in a program
- Data stored in variables
- Assignment Statements



Why Programming? Why Java?

- Programming Languages: a language used to write instructions that can be executed by a computer
- Instructions in a computer program are known as the code.
    - The instructions written for a computer to execute is a program.
- Machine Code: a set of instructions composed of 1s and 0s the computer can execute without any translation
- High-Level Languages: translate human messages into machine code that the computer can understand
- An interface allows communication between humans and computers.
- Source Codes: program code written in a high-level language before being translated into machine code
- Java is a high-level language and it's easier for programmers to learn and use.
    - User friendly
    - Can use on different kinds of hardware
    - Run slower than the lower-level languages
    - Must be translated to machine code
- Java is one of the most common modern computer languages
    - It is used for web applications and software development
- This programming language was developed by James Gosling and a group of people at Sun Microsystems in California.

Variables and Data Types

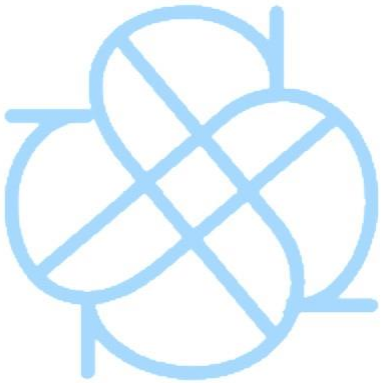
- Variable
    - a group of characters whose value can be changed as needed

    - Stores data in RAM (Random Access Memory)
- Identifier: A name for a parameter, variable, user-defined method, constant or user-define class.
    - A sequence of digits, letters, and the underscore.
    - Can't begin with a digit.
    - Case-sensitive
        - Lowercase when naming identifiers for variables and methods.
        - Uppercase letters are used to separate words.
- Rules you must follow when naming Variables
    - Do
        - Begin variable names with a letter or underscore. (Ex: song, songTitle)
        - After the first letter, the variable name can consist of additional letters or digits (0 to 9).
    - Do Not
        - Variable names should not be a Python keyword.
        - Variable names can't have spaces.
        - Variable names can't have any punctuation.
- Final variable
    - User-defined constant (uses keyword final)
    - Can't change the value of the variable
    - Example:
        - `final double` CLASS_SIZE = 28;
- Make sure you use camelCase in Java (for variables/new instances)
- Non-numeric data: a string which consists of a combination of letters, numbers, and/or symbols. This type of data can't be used in calculations.
- Numeric data: numerical value that can be used in calculations.

Variables and Data Types (continued)

- Built-in Types
    - int: refers to an integer
        - positive, negative whole numbers (including 0)
    - Boolean: a logic which evaluates whether a condition is true or false.

    - Double: decimals (floating-point numbers)
        - Uses 8 bytes
- float: refers to “floating point numbers”. These numbers have a decimal point.
        - positive, negative numbers, including 0.0
        - Uses 4 bytes
- String: Sequence of letters, numbers, spaces, and symbols, or alphanumeric info.
- Boolean: a logic which evaluates whether a condition is true or false.
- Floating-point numbers
    - Stored in two parts (a mantissa and an exponent)
    - Mantissa: Digits of the number

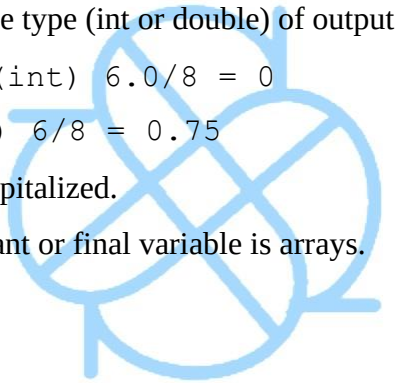


Expressions and Assignment Statements

- Arithmetic expressions
    - Typically consist of parentheses, function calls, and operators
- Arithmetic Operators
    - + (addition)
    - - (subtraction)
    - * (multiplication)

- / (division)
  - Ex: `20 % 8`
    `//returns 2 (NOT 2.5)`
- % (modulus)
  - Gives you the remainder
  - Ex: `11 % 3`
    `//returns 2`

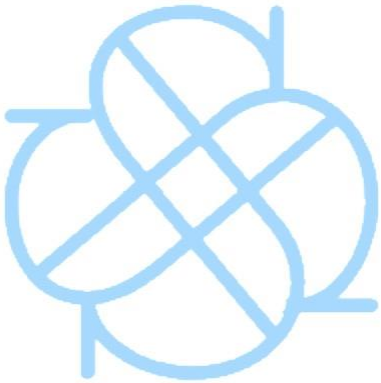
- Assignment Statement: statement which assigns values to variables
- These assignment operators can be applied to the primitive data types `int` and `double`.
  - Even if both of the data types are in the same expression
- Integer division (divisor and dividend are both integers) results in an integer output/quotient.
  - You can control the type (int or double) of output by casting the operands.
    - Example: `(int) 6.0/8 = 0`
      `(double) 6/8 = 0.75`
- Constant identifiers are capitalized.
- A common use of a constant or final variable is arrays.



Expressions and Assignment Statements (continued)

- Relational Operators
  - == (equal to)
  - != (not equal to)
  - > (greater than)
  - < (less than)
  - >= (greater than or equal to)
  - <= (less than or equal to)
- Logical Operators
  - ! (NOT)

- && (AND)
- | | (OR)
- Applied to Boolean expressions (for compound Boolean expressions)
  - To evaluate true or false
- True or false values are assigned based on the result of a truth table for these logical operators.



Compound Assignment Operators

- = (simple assignment)
- Compound Assignment Operators
  - +=
    - x += 5 or x = x + 5
  - – =
    - x -= 7 or x = x - 7
  - *=
    - x *= 9 or x = x * 9
  - /=
    - x /= 10 or x = x / 10
  - %=

        - x %= 4 or x = x % 4
- Increment and Decrement Operators
    - ++
        - i++ or ++i
        - i is incremented by 1
    - –
        - j++ or ++j
        - j is decremented by 1
- Operator Precedence
    - Highest Precedence
        - !. ++. –
        - *, /, %
        - >, <, >=, <=
        - ==, !=
        - &&
        - | |

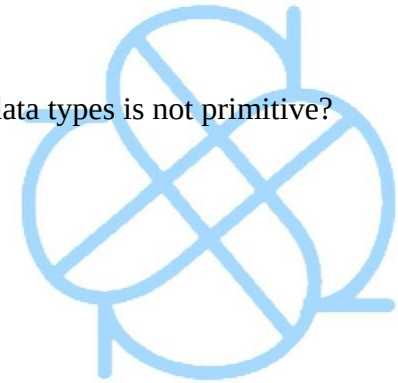
Compound Assignment Operators (continued)

    - Lowest Precedence
        - Simple Assignment
        - Compound Assignment Operators

Input and Output

- Input
    - double x = Call method which reads a floating-point number
    - double x = …;
        - Read user input
    - Scanner Class – simplifies the console and the input
- Output
    - `System.out.print`
    - `System.out.println`
    - `System` class – displays output to the screen.
    - `print` method outputs items without going to a new line while `println` does print the output on the next line.

Example Practice Questions:



1. Which of the following data types is not primitive?
    a. Long
    b. Integer
    c. String
    d. Boolean

    Answer: C
2. A value can't be changed if a variable is declared _____.
    a. final
    b. private
    c. boolean
    d. constant

    Answer: A

    ```
    final  dataType name = value
    ```