Module 1 --1.3

Notebook:	Module 1_note		
Created:	7/11/2020 9:55 AM	**Updated:**	7/13/2020 9:32 PM
Author:	Runan Yang		
Tags:	calculation/connection signs, Log		

Arithmetic Operators and applications

1. addition， +
2. subtract, -
3. multiple *
4. division, /

For division, if the two objects are two integers, by default, the output will be integer (e.g.: 5/2 = 2)

While, if at least one of the two objects are decimal, the output will be decimal.

Normally, one can use *1.0 for divider to get the output as decimal. In this way, two dividers will not be changed in computing the results.

5. balance, %

balance will direct to the residual in division.

Converted the seconds to hours/mins/secs

1. The procedure to solve this problem
   - use scanner to get the values of seconds user input
   - determine the values of hours and saved as a new variable
   - determine the values of mins and saved as a new variable
   - determine the values of secs and saved as a new variable
   - print out these variables in the correct way as answers

2. The arithmetic operators used here
   - to get values of hours, / will be used
   - to get values of minus, / will be used
   - to get values of secs, % will be used

Connecting the two strings

- + is not only the sign for addition, but the sign for connecting two strings.

- At least one string is connected. Before and after "+", the result will be string if at least one string appears.

Relationship operators

- < less than
- > greater than
- == equal
- >= no less than
- <= no more than

++ and -- operators

- Here, VarName++ and ++ VarName could direct to the same output of Variable.
- But, the equation of these two are different. Once ia++, adding 1 to ia would be calculated after copying value of ia to this equation.
- While, ++ia would direct to two steps. First will be copying the value of ia to this equation and the second will be adding 1 to ia.

Logic operators and relationship operators

1. " short circuit " is a characteristic of logical operators. In a equation connected with &&, if the first equation is False, there is no need to judge the second one. The output is False. While, for the equations connected with ||, once the first equation is True, the output of equation will be true without judging the second one.
2. the example of judging three-digit number on java

Three Eye Logical condition

Format is

condition?Statement1:Statement2

could be used to find the max or min

Assign value Operators and application

- Simplest way: assigning values to variables by "=",
- **very important,** different types, usually from types with more space to some lower cases, could not assign automatically.
- While, += can operate this function as well. (VariaName += Value) this is a value. magnitude variable
- Equivalent in programming, usually takes the form "value == VariableName" since it is easy for programmer to distinguish with "VaraibleName = Value". The latter one assigns the value to Variable.

Shift bit operators and application --- for binary, instead of base 10

1. >>. This could be used for shifting bits to the right.
    - VariaName >> Value; the logical operator will shift the bit to the right in magnitude of Value.
    - if this is a positive number of this variable, then on the left side, add 0s
    - if this is a negative number, add 1s
    - by definition of binary shifting bits, right shifting will be the value divided by 2.
2. <<, shift to the left in value bits.
    - add 0s to the right
    - by definition of binary shifting bits, left shifting will be 2^value times.
3. >>>, shift to the right and add one 0 (zero) from the left.
4. &, this operator could be treated as addition in binary;
    - Rule: both values of 1, output is 1; otherwise 0
5. |, used to subtracting
    - rule: at least one 1 appears, outcome is 1.
6. ~, all to the negative (1--0,0--1)
7. ^, could be treated as multiple operator
    - different, outcome is 1; otherwise 0.
    - this one could also be used for multiple to exchange values of two variables
        - Varia1 ^ Varia2 ^ Varial1 = Varial2
        - e.g the binary of 12 (as byte) is 00001100; the binary of 98 is 01100010
        - v1 ^ v2 = 01101110 v1^v2^v1 = 01100010 = v2

( ) always has the priority to operate; and = is the last one to operate.