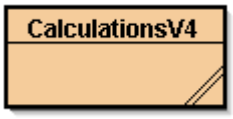
Assessment Instructions

Part 1

Instructions: Modify CalculationsV4 class by replacing all integer values with integer variables.



1. Create a new project called Ints_Assessment in the Unit02 Assessments folder.
2. Create a class called CalculationsV4 in the newly created project folder.
3. Copy and paste the CalculationsV3 class into the CalculationsV4 class. Don't forget to change the version from V3 to V4 in the class name.
4. Assign all integer values to int variables (e.g. iNum1 = 4, iNum3 = 15, etc.). (Note: the program contains integer and non-integer values. Focus only on the integers at this time.)
5. Substitute integer variables for all corresponding integer values throughout the program. (At this time you will only make changes to the integers in the program. Decimal values will be changed later.)
6. Modify the code for the output so that one statement prints the String literal to display the arithmetic problem and one print statement calculates the answer. (Use the practice example as a model.)
7. Model your variable initializations, declarations, and substitutions after the existing code.
8. Add two more arithmetic expressions of your own. Use integer variables and at least three different arithmetic operators (i.e., +, -, *, /, and %) in each expression so that all five are used once you have written the two new ones. (For example, you might use +, -, and * in one expression and -, /, and % in the other.)
9. Make sure your program works flawlessly before moving on to Part 2. (In fact, you probably should not even read Part 2 yet. That would just spoil the experiment!)
10. Compile and run the modified program. Compare the output to CalculationsV2 class.

Expected Output: When your program runs correctly, your output should resemble the following screen shot. Your output will also contain more expressions.

```
Addition
4 plus 8 = 12
32.4425

Subtraction
9 minus 33 = -24
51.596199999999996

Multiplication
15 times 3 times 201 = 9045
78.5

Division
48 divided by 8 = 6
6.625

Modulus
23 modulus 15 = 8
138.3999999999999

2.02 Lab Equations
15 divided by 2.5 times -2
```

Part 2

Are you ready to conduct an experiment to answer an important question "What good are variables?" If the result is the same, there doesn't seem to be any advantage to go to the trouble of assigning values to variables, typing tedious print statements, and writing two print statements instead of one. There must be a good reason for the extra effort.

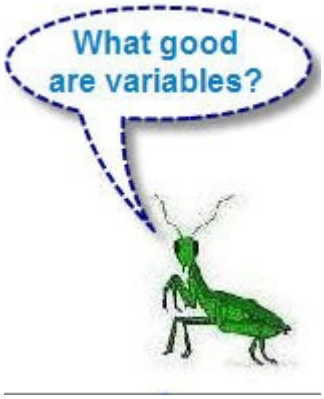
- Choose a variable in the CalculationsV4 class, any variable.
- Change the value of the variable in the statement where it is initialized.
- Can you predict what will happen?

When you run the program the output should be different. How hard was that?

In the 17th century, François Vieta could not have imagined that his decision to substitute symbols for numbers in equations would have such a far reaching impact. Using variables allows programmers to make rapid changes throughout a program instead of having to find every occurrence of a value to update individual statements.

Imagine a program that was 100 lines long and the number 25 was used in 15 different arithmetic expressions. How many lines would you have to change if you wanted to use the number 50 instead of 25?

Now imagine that instead of using the actual number, you assigned 25 to a variable and used the variable in the 15 arithmetic expressions. How many lines would you need to change to replace 25 with 50?



It is good programming practice to use variables instead of values.