# CPSC 1100 – LAB 02

Variable Types, Declarations, Initialization, Assignment Statements, Calculations and Objects

This lab will deal with declaring variables of different types, giving them initial values, assigning them new values, and doing calculations with these variables. We will also create a Rectangle object and interact with it using the methods provided by the Rectangle class. **PLEASE COMMENT YOUR CODE.** You will have points taken off if you do not comment your code. Keep your code neat.

**Some useful links:**

Java tutorial home page: <http://docs.oracle.com/javase/tutorial/>

Start here: <http://docs.oracle.com/javase/tutorial/java/index.html>

Variables: <http://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html>

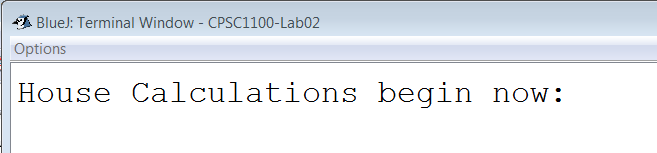
BlueJ Tutorial: <http://www.bluej.org/tutorial/tutorial-201.pdf>

## Tasks: Follow the directions below to complete your lab assignment

For this lab we are going to store some features about a house we are designing. This is a very simple house, it is in the shape of a box. (You should draw this on a piece of paper to start visualizing your dream home – and to help visualize the calculations we will do in this lab). The roof will be a flat rectangle, as will the floor, and each wall of the house. The house will have some doors, each door will be the same size (so doorHeight and doorWidth are the same for all doors in the house). The house may have some windows. Each window will be the same size (i.e. windowHeight and windowWidth are the same for all windows in the house). The roof and the floor will be the same size. Walls *opposite* of each other will have the same dimensions. (Front wall and back wall are same size, side walls are the same size). The area of the roof (and floor) will be widthFrontWall x widthSideWall. (draw a diagram of the house to help visualize, and use variable names to label each measurement for the house – you won't have to turn this in but it will HELP tremendously for you to visualize the problem). Once we have stored and printed out all the information about our house, we will do some calculations involving surface area (length x width = area). A painter may need calculations like this in order to know the total surface area that needs painting.

First you will need to create a new project in BlueJ. You can name it anything you like, including a name with spaces. I named my project **CPSC1100 Lab02**. The name you give here is the name of the folder that BlueJ will store your project in. Next, you will need to create a new class named HouseCalculator. Then, you will need to download HouseCalculator.java.txt from Google Drive (under the Lab02 section). Replace ALL of the code in your HouseCalculator class with the code contained in HouseCalculator.java.txt.

At this point, compile your code, and then run the main method (right click on the HouseCalculator class from the main BlueJ window). (Remember, you can review task 0 from Lab01 for more detailed directions on how to set up a project, and run a program). You should have a terminal window pop up with the following message once you run the main method. (Make sure to select both options in the terminal window, “Clear screen at method call” and “Unlimited buffering”



Now you are ready to complete the following tasks. There is a comment in the starter code that indicates where you will insert your source code. (At the end of this section I have posted a screen-shot of what your final code output should look like).

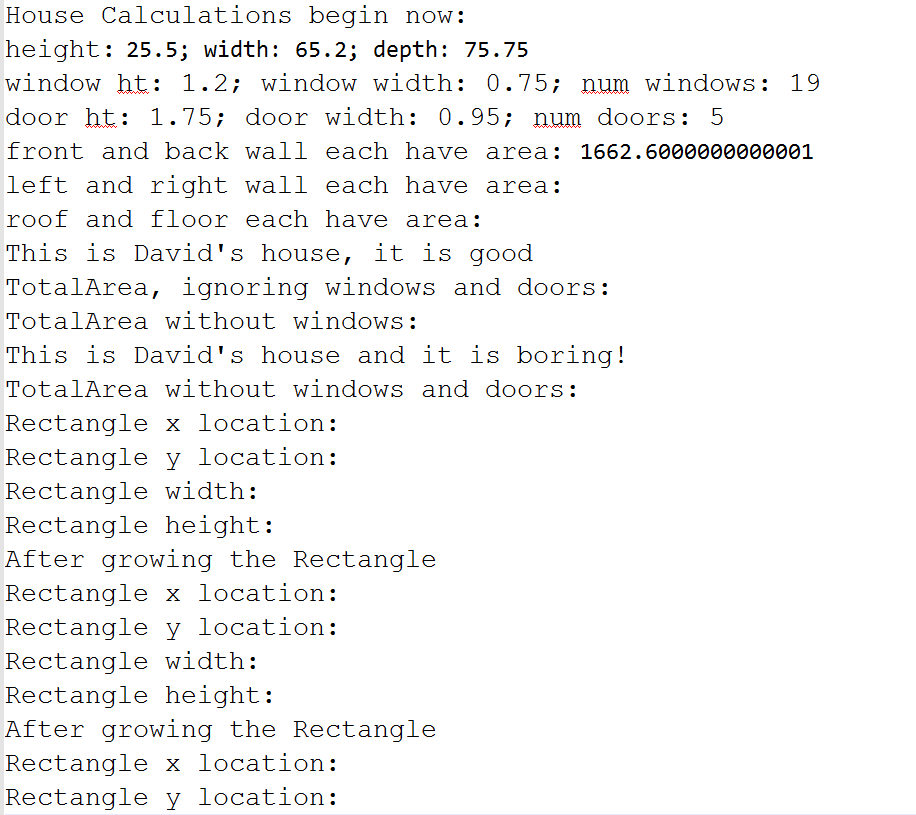
1. *Declare* three variables of type **double** to represent height, width and depth of the house. Let's also *initialize* all these variables such that our house is 25.5 feet tall, front/back walls are 65.2 feet wide, and the side walls are 75.75 feet deep. *Remember that doubles represent numbers with a decimal point, while integers represent whole numbers.*
2. Print these values to the screen using the println() method. Format your text and make sure to label each value you print. (i.e. don't print just the values – give a label to each value in your printout).
3. *Declare* and *initialize* three more variables. All windows have the same size, so we will need a value for height, width, and the number of windows on our house. height = 1.2, width = .75, number of windows = 19. (Pay attention to the data type for these values. Pick an appropriate data type and descriptive names for your variables).
4. *Declare* and *initialize* three more variables. All doors have the same size, so we will need a value for height, width, and the number of doors on our house. height = 1.75, width = .95, number of doors = 5. (Pay attention to the data type for these values).
5. Neatly print out the number of doors / windows, and the size of each.
6. *Declare* six variables of type **double**. One each for the surface area of the roof,floor, side wall left, side wall right, front wall and back wall. Initialize each variable to 0, we will assign values to these variables later as we complete calculations.
7. *Declare* two variables of type **double** for total surface area of the house. The 1st will be the total surface area of our house ignoring windows and doors, the 2nd variable will store the surface area with windows’ and doors’ areas subtracted from the total surface area.
8. *Declare* and *initialize* a **String** variable for each of the following strings (7 strings). “good” “bad” “boring” “exciting” “big” “small” and finally a string with your name. (first name only is ok).
9. For the next several calculations, ignore the number of windows / doors.
10. Create *assignment statements* to calculate and store the surface area of each wall of the house, the roof, and the floor. Print these values out, and make sure to label them in your print statements. (don't just print out the numbers – should be six walls, but only 3 different values). You should use the 6 variables you created in step 6 above to store your results.
11. Create an *assignment statement* that stores the total surface area of your house to the variable declared earlier (step 7) that will hold total surface area. (add values together from previous step).
12. Print a statement that tells whose house this is, uses an adjective to describe the house, and prints the total surface area calculated in the previous step. To print your name and an adjective for the house you should use string concatenation and 2 of the String variables created in step 8. For example, I might print: “This is David's House. It is boring.” I would follow this up with the surface area.
13. Now let’s take into account the windows while calculating the surface area. Note that we can use the surface area calculated in step 11, subtract the total surface area of the windows (# windows X windowHeight X windowLength), and assign this value to our 2nd variable for storing total surface area.
14. Print the value calculated from step 13 as well as a label to indicate this is the total surface area minus windows.
15. Like in step 13, find the (total surface area – window area – door area). Print the value calculated in step 15, as well as a label to indicate this is the total surface area minus windows and doors. Also, using 2 of the String variables created in step 8 and string concatenation, print a message saying whose house this is and an adjective to describe it. I might print “This house belongs to David, and it is very big.”
16. Note that at this point in your code, all the variables still have the values you previously assigned to them.
17. Create a Rectangle object located at (7, 13) with width = 5, height = 10. (Be sure to browse to the java API to verify how the Rectangle constructor works).
18. Print the x location of the Rectangle by using the getX() method from the Rectangle class. The code for this step will look like this:

System.out.println(“Rectangle x location: ” + myRectangle.getX());

Where the word “myRectangle” will be replaced with the variable name you chose in step 17.

1. Print the y location of the Rectangle.
2. Print the width of the Rectangle.
3. Print the height of the Rectangle.
4. Use the grow() method on your Rectangle object to increase the width by 4 and the height by 8. Make sure to use the Java API and read the description of the grow method.
5. Repeat steps 18-21 (hint, copy / paste the code you wrote in those steps). This will print the current location and dimensions of the Rectangle.
6. Use the translate() method to place the Rectangle back at its starting location.
7. Print the Rectangle’s current x and y values to verify that it is in the correct position (7,13)

Your final output should look something like the following. I have left many of the answers blank in this screen-shot so that you can make those calculations yourself.



## To Turn In via Google Drive

Once your program is complete, run the program one final time (right click on HouseCalculator and run the main method). Then, to save a copy of your output from the Terminal window, click on Options → Save to file … give the file a name (something like lab02-output.txt) and choose a location to save your program's output. I would suggest saving this file in your project folder for Lab02. (Note that when you are in BlueJ project folders will display a little BlueJ icon instead of a folder icon, but they are still folders).This is the FIRST of two files you must submit. The SECOND file you must submit is your modified version of HouseCalculator.java.