**IT 1090C Computer Programming I**

**Lab 4 Introduction to Java**

**SPRING 2020**

**12/2 pts (2 extra credit pts)**   
  
Note: Complete the lab individually. You can get help from your group members but it is very important that each student complete the lab themselves using Netbeans. You want to be totally comfortable with creating java projects in Netbeans.

**Learning Goals:**

* how java’s numeric types, (int, and double), interact with the simple arithmetic operators (multiplication, addition, modulo, etc.).
* code several of our previous pseudocode programs in java!

**Part I: Netbeans Project and Java Main File (2 pts)**

1. Create a new Netbeans project called **JavaVariables**
2. Now create a java main class source code file called **Variables.java** within your project

**Take a break and check with your fellow learners. Was everyone able to create the project files correctly? (If you are taking the course online, be sure to watch the support videos before you do this lab!)**

Within your file declare and initialize a series of numeric variables:

int:   
  
***intOperandA, intOperandB, intSum, intProduct, intDifference, intQuotient, intModulo  
  
You can set these to simple integer values 1 5, 13, 24, etc.***

double: (The first two should be decimal values with a fractional part: i.e. 1.5, 3.14 etc. The others will be calculated.)  
  
**doubleOperandA, doubleOperandB, doubleSum, doubleProduct, doubleDifference, doubleQuotient, doubleModulo**

*Note that we do not include the type of variables i.e. int, double, boolean, etc in the name of the variables. I’m having you do that here so you can observe how the typing effects the arithmetic operations.*

Now provide output statements using your variables for each of these arithmetic operations for both the int and the double types:

+, -, \*, /, % (that last one is remainder and is called modulo)

**Example:**   
intSum = intOperandA + intOperandB;

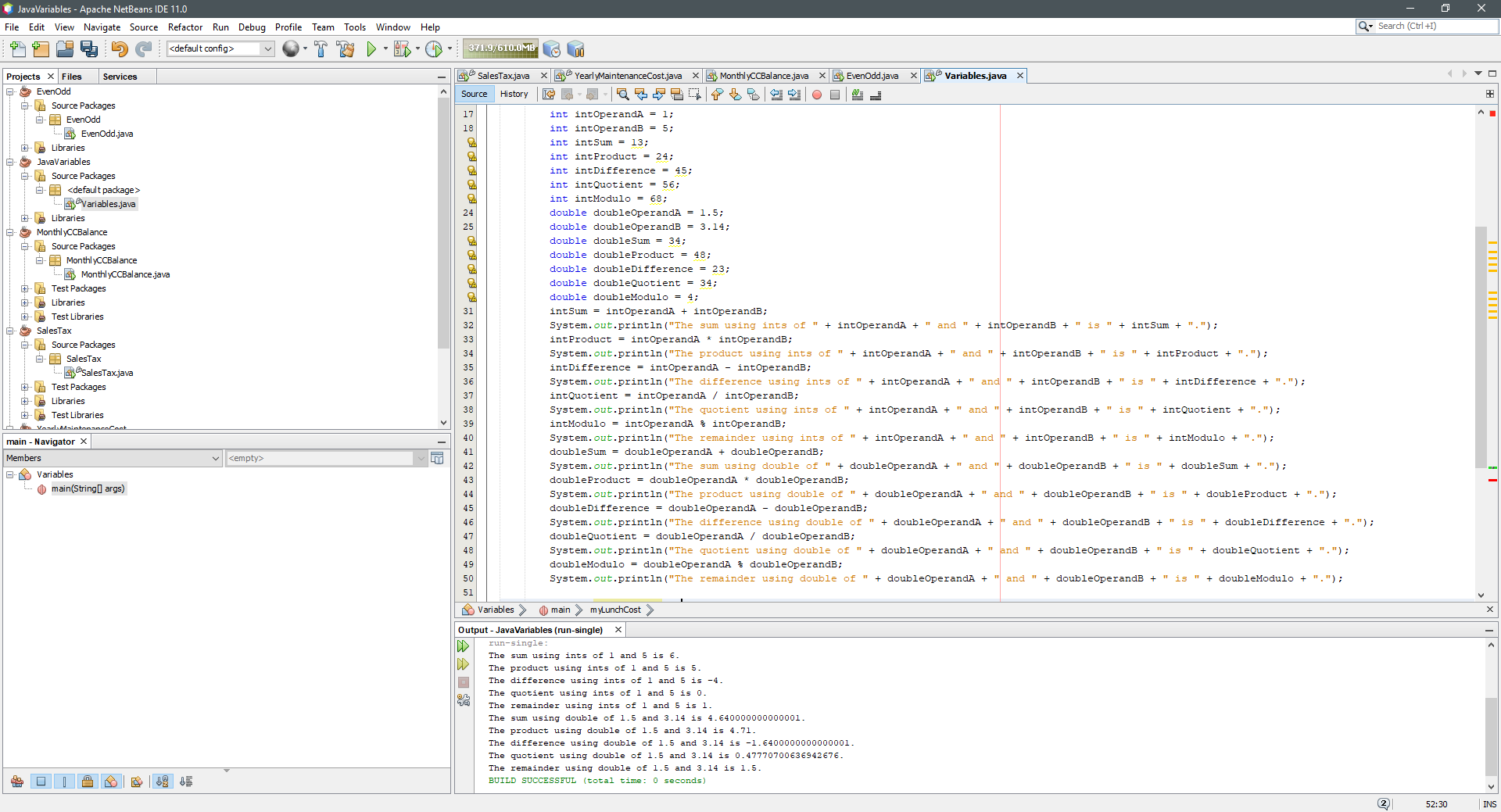
System.out.println(“The sum using ints of “ + intOperandA + “ “ + intOperandB + “ is “ + intSum);

**Do this with the int variables for each of the operations (+ sum, \* product, - difference, / quotient, and % modulo. Notice how we include a space before and after each operator to increase readability. Get in the habit of typing the code this way.**

Now do the same thing with the **double variables**

Now, execute (that is ‘run’) your program. Right click on your **Variables.java** file in the project view and choose **Run File**. Alternatively, you can run the program by clicking on the green arrow on the top menu bar. You can also run the program using the [shift] + F6 shortcut.

Size the output window so you can see the results of the **println** statements for these expressions and grab a screen shot of your desktop that shows this. Alternatively, and what often works better for me, copy the output window as text and paste it into this document. **Be sure when you do these screen shots or output copies that they clearly show me the work I asked you to do.**  
  
[Windows: Alt + printScrn key puts image of desktop in clipboard just use past to insert it here in to this document.   
  
Mac: I think Apple /Command + 2 or 3 does the same thing.]   
 **Insert the screen shot here:**



Did any of these results surprise you? (If not, take another look at the result of int division)

OK complete this part by creating declarations and initializing each of these variables.   
Choose meaningful names of your own for your variables and use camel case:

**Example:  
the cost of my lunch:** double myLunchCost = 12.50;

**the number of kids in your family:**

**the answer to is it raining? :**

**the price of a gallon of gas :**

**your favorite number :**

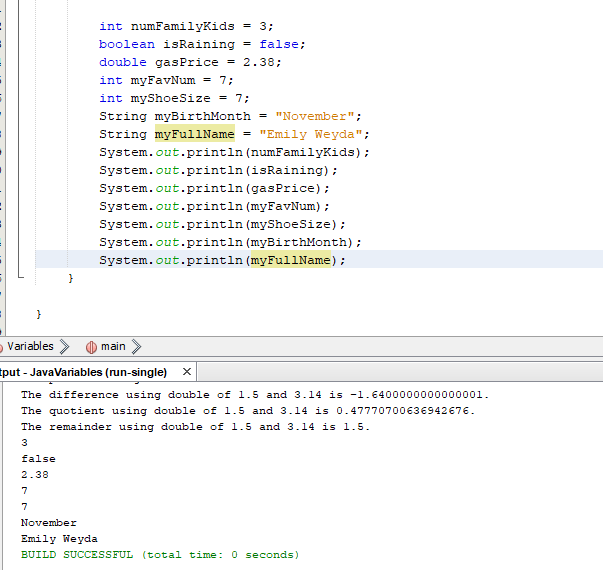
**your shoe size:**

**your birth month:**

**your full name:**

**(If you want, you can print these out but I’ll make that optional so you can finish the lab during the session.)**

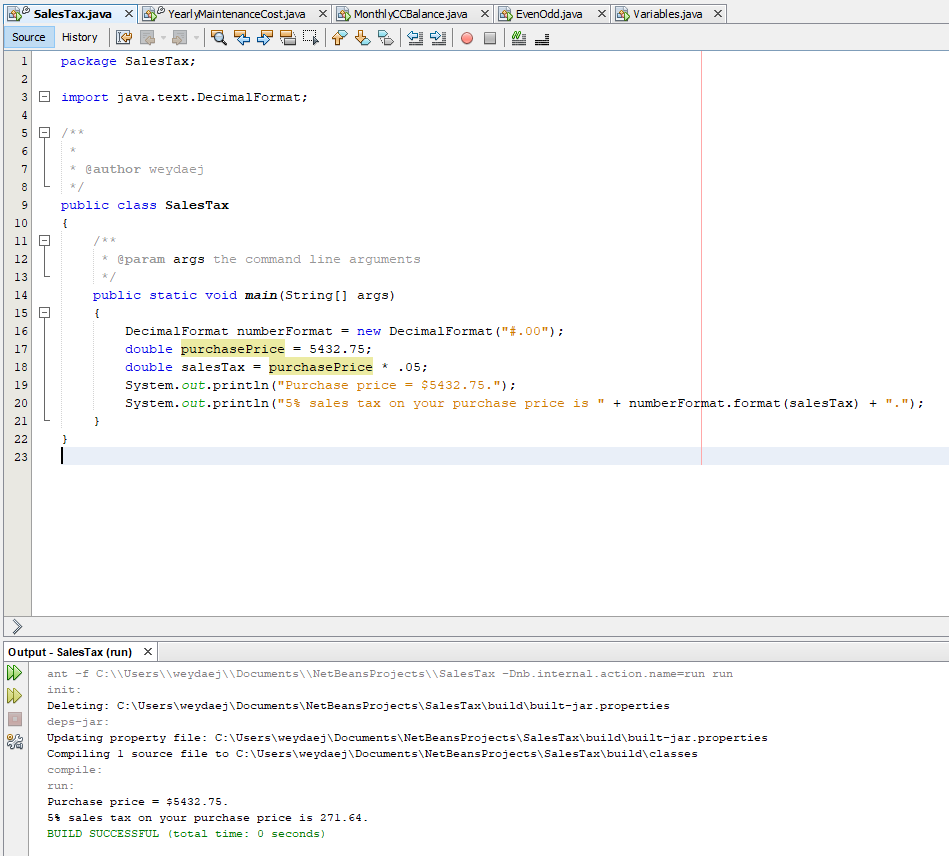
**Get another screen shot or copy of your output window making sure I can read all your variable declarations.**

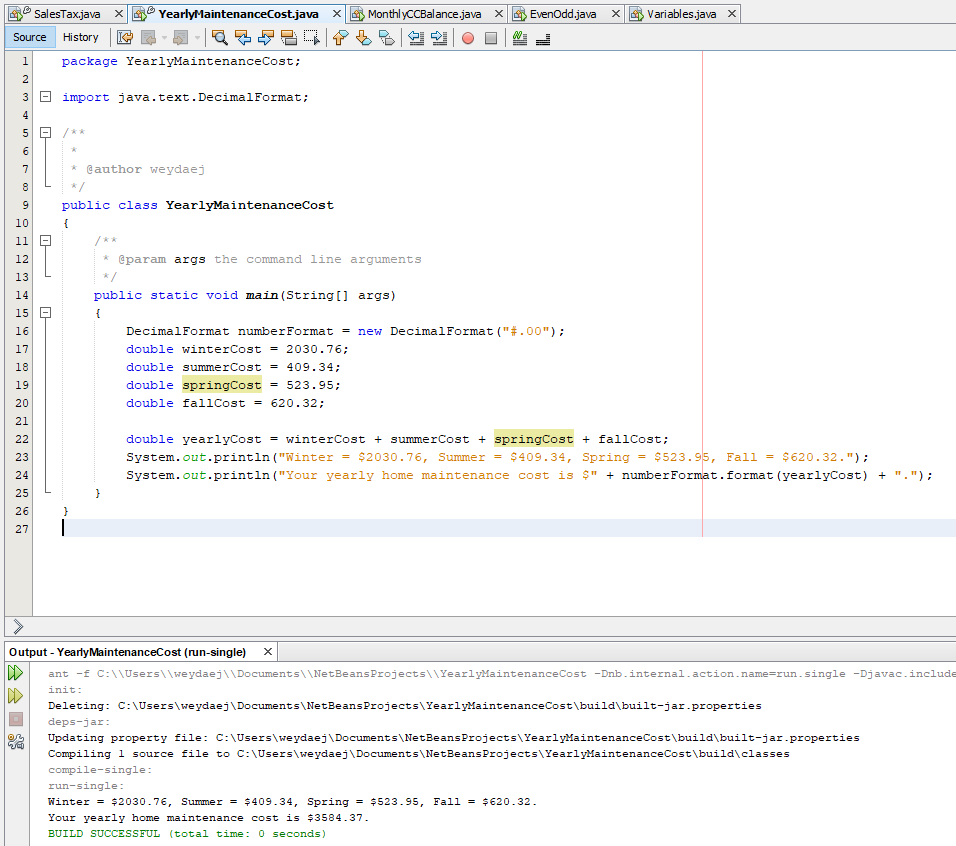
**Paste the screen image or output copy into this document here:  
**

**Part II:** Now code these programs. *Since we can’t do java input yet, just set the input value as a literal value in your code.* I’ll leave it to you to choose the values. Be sure to use clear and meaningful variable names with appropriate types! Try to pick values that make it easy for you to visually confirm the results of your program outputs.  
  
Here is an example:  
  
We did an example where the user enters a number and the program doubles it. Here is the code that I would put within the main() class of my Netbeans project:  
  
**int myFavNum = 23; // Later we would input this but today we simulate that  
int doubleNum = 0;**

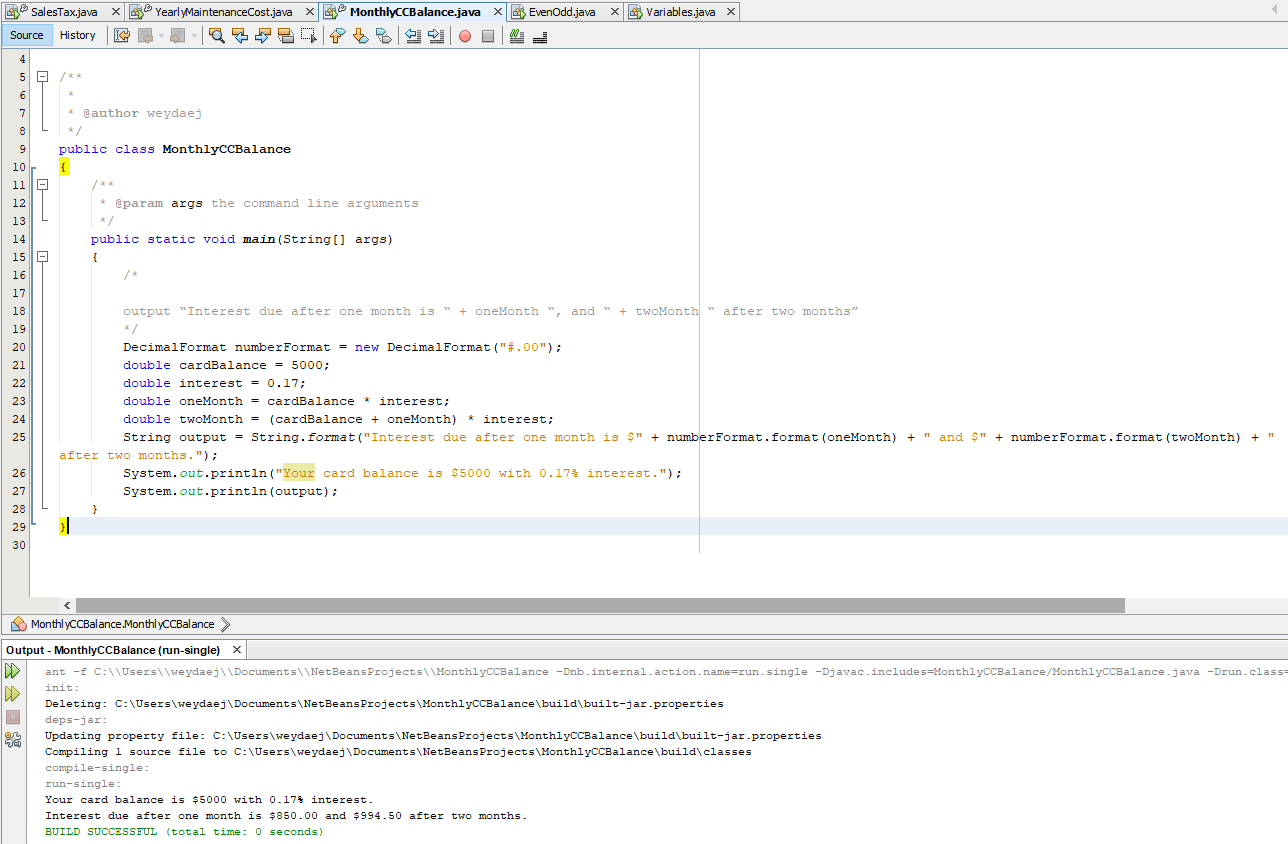
**doubleNum = myFavNum \* 2;  
  
System.out.println(“The number “ + myFavNum + “ doubled is: “ + doubleNum);**

**On with the lab:**  
  
Create a **new** Netbeans Project and java main class for **each** using the specified file names.

**Task 1 (3 pts):   
Project: SalesTax  
File: Year  
  
Given the price of a purchase the program computes and outputs a 5% sales tax. Be sure to use output statements that clearly display the price and the computed sales tax. (No conditions here, the sales tax is always the same rate.)  
  
Screenshots or output copy here from Task 1 (Be sure to include a readable shot of your programs output for each of these tasks):  
  
  
Task 2 (3 pts):   
Project: YearlyMaintenanceCost  
File: YearlyMaintenanceCost.java**

**A program that uses home maintenance costs for each of the four seasons and returns the total yearly maintenance costs. Again, be clear in the output by echoing the values for the seasonal costs when you display the combined total.  
  
Screenshots or output copy here for Task 2:  
  
  
Task 3 (4 pts):   
Project: MonthyCCBalance  
File: MonthlyCCBalance.java  
  
A program that starts with a credit card balance of $5000 and then calculates the interest at a rate of 17%. Assume the user does not make any payments. Display the interest due after one month and again after two months.**

**Screenshots or output copy here for Task 3:**

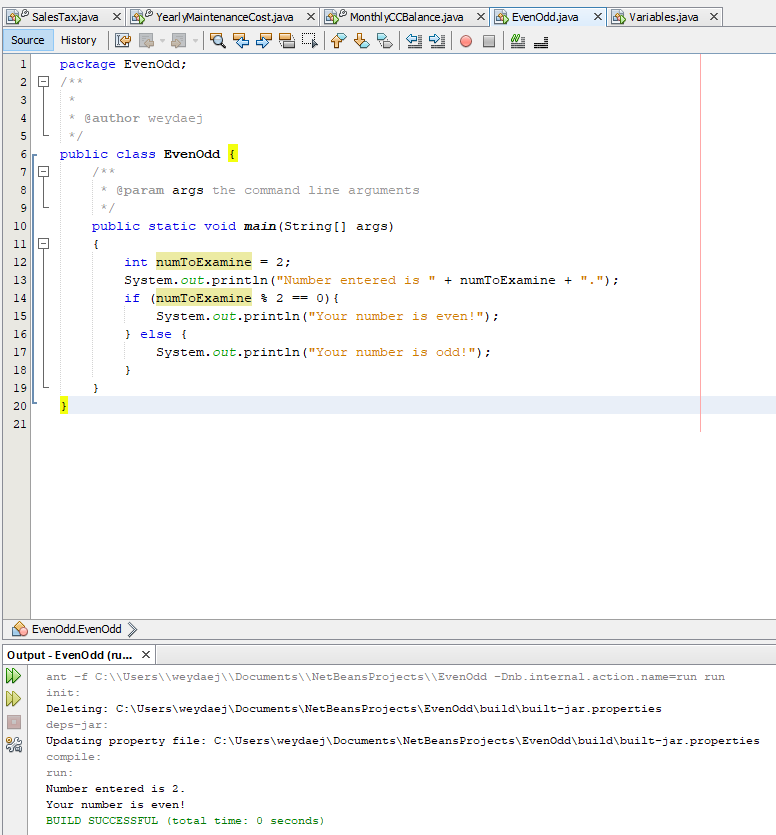


**Task 4 (2 pts): (Extra Credit)   
Project: EvenOdd  
File: EvenOdd.java  
  
A program that takes a integer (java int) and uses the modulo function to indicate if the number is odd or even. Since we can’t do input in java yet, just hard code the value for the input as a variable instantiation.   
  
(Instead of inputting the value which we can’t do yet, set the value in the code.   
  
int numToExamine = 2;**

**// program will then determine that since 2 mod 2 is 0 that numToExamine is Even.**

**)**

**Screenshots or output copy here for Task 4:**



**Submitting your work:   
  
PLEASE READ CAREFULLY. NOW THAT WE ARE USING JAVA YOU WILL HAVE TO SUBMIT ALL THE WORK.**Make sure that you saved your screen shots or output copy in this document and that your file is correctly named: **Lastname\_Firstname\_Lab\_04.docx** using your name. Create a new compressed \*.zip archive folder. (Don’t give me any other type of archive, it will be returned to you ungraded!) Name it: **Lastname\_Firstname\_Lab\_04.zip** using your name. Place a copy of each of your entire Netbeans project folder(s) and a copy of this doc file within your .zip archive. When you submit your lab, upload the complete archive as well as the **separate** copy of this MS Word file which will be used for grading your lab. **This is very important. If you submit the word file, I can insert my feedback to you directly in it.**  
  
Although I won’t run your code this time, I ask that you give me your entire project every time so I can run your code or examine it if I need to for grading. **If you don’t follow the submission directions, your work will not be graded.**  
Use the Canvas Assignment mechanism to submit your two files.  
For the Extra credit resubmit both files using the extra credit submission link.  
  
**Every assignment for the rest of the course will follow these submission guidelines. Please get it right!**