**PROJECT Inventory Valuation - Using Methods in Java 100 points**

**Objective** To type a simple Java program, execute ( run ) the program for some particular values, observe the output and then modify the program.

***PROJECT DESCRIPTION***

Type, compile and run the basic Java program that is shown in **Figure 1** , which follows.

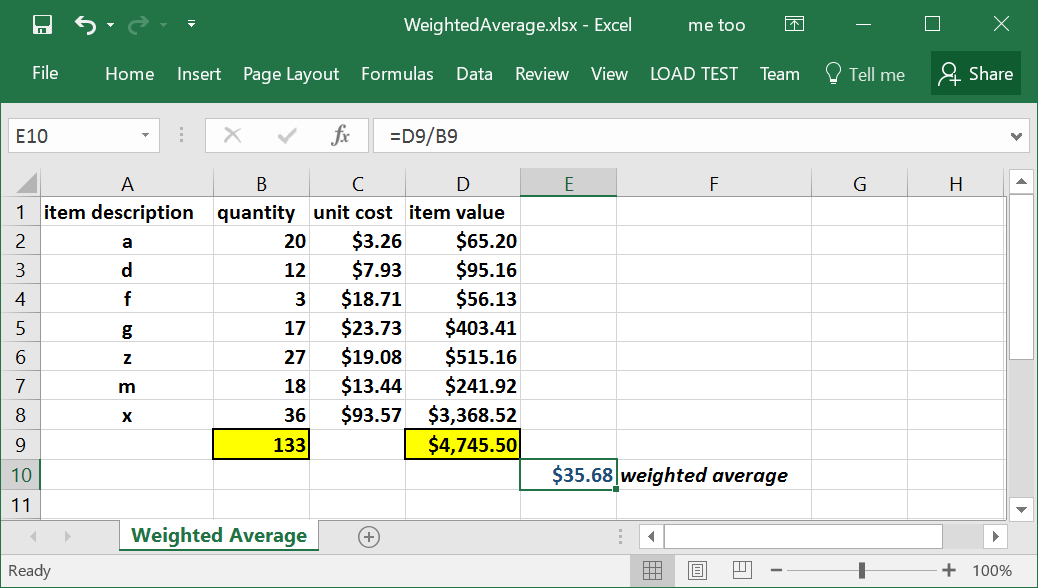
Then compile and run your program, observe the output then modify the program.

***Information About This Project***

For this project, we will create an application that uses the Weighted Average Cost Method to value a product inventory.

The various methods used to value a product inventory include: FIFO, LIFO, Average Cost and Specific Identification.

Here is an example of the use of the Weighted Average Cost Method.



***Steps to Complete This Project***

**STEP 1**  **Open NetBeans**

Open NetBeans and create a Java project with the following details.

For Project Name include **Lab6**

For the Main Class include **lab6.Inventory**

In your **Code** window, shown below, copy in the program code shown in **Figure 1** below, in the appropriate places, except substitute your own name in place of Sammy Student.

**PROJECT Inventory Valuation - Using Methods in Java**

**Figure 1 Source Code for the Inventory Valuation Program**

|  |
| --- |
| **package lab6;**  **import java.util.Scanner;**  **//Sammy Student, Programmer**  **public class Inventory**  **{**  **static Scanner *sc* = new Scanner(System.*in*);**  **public static void main(String args[])**  **{**  **// begin local variable declaration / initialization zone**  **char answer = 'Y';**  **double average = 0.0, cost = 0.0, totValue = 0.0;**  **int number = 0, sumItems = 0, quantity = 0;**  **String item = "";**  **// end local variable declaration / initialization zone**    **// begin program menu**  **System.*out*.println("");**  **System.*out*.println("");**  **System.*out*.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");**  **System.*out*.println("----Inventory Valuation---");**  **System.*out*.println("\*\*\*(Weighted Average Method)\*\*\*");**  **System.*out*.println("");**  **System.*out*.println("");**  **System.*out*.println("--------- M E N U ---------");**  **System.*out*.println("");**  **// end program menu**    **while(answer == 'Y' || answer == 'y')**  **{**  **// begin code block for inventory evaluation**  **System.*out*.println("number of item types in the inventory ->");**  **number = *sc*.nextInt();**    **for(int i = 1; i <= number; i++)**  **{**  **System.*out*.println("enter the item's description");**  **item = *sc*.next();**  **System.*out*.println("item description: " + item);**  **System.*out*.println("enter item quantity");**  **quantity = *sc*.nextInt();**  **sumItems += quantity;**  **System.*out*.println("enter item cost");**  **cost = *sc*.nextDouble();**  **totValue += cost \* quantity;**  **}**  **// outside the for() loop** |

**PROJECT Inventory Valuation - Using Methods in Java**

**Figure 1 Source Code for the Inventory Valuation Program ( continued )**

|  |
| --- |
| **average = totValue / sumItems;**  **// end code block for inventory evaluation**    **// begin code block to display results**  **System.*out*.printf("average cost: $%.2f\n", average);**  **System.*out*.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");**  **// end code block to display results**    **// begin code block to perform additional program run**  **System.*out*.println("run again(Y or N)?");**  **answer = *sc*.next().charAt(0);**  **// end code block to perform additional program run**  **}**  **System.*out*.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");**  **}// end main() method**  **}// end class** |

**STEP 2 Build, Compile and Run the Program**

From the NetBeans Run menu select Run Project (Lab6) to run your app.

**STEP 3 Test the Program**

Once you have successfully compiled your program, review the output **Console** window of NetBeans.

Enter the sample information shown in **Figure 2** that follows as a single run.

You can verify your output with this MS Excel formula:

**=(5\*750+6\*3250+10\*926)/(5+6+10)**

**PROJECT Inventory Valuation - Using Methods in Java**

**Figure 2 Initial Test Run**

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ----Inventory Valuation---  \*\*\*(Weighted Average Method)\*\*\*  --------- M E N U ---------  number of item types in the inventory ->  3  enter the item's description  Chair  item description: chair  enter item quantity  5  enter item cost  750  enter the item's description  Computer  item description: computer  enter item quantity  6  enter item cost  3250  enter the item's description  Desk  item description: desk  enter item quantity  10  enter item cost  926  average cost: $1548.10  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  run again(Y or N)? |

**STEP 4 Construct a Method**

Locate the following code statements, that were given in the initial starter code.

|  |
| --- |
| **System.*out*.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");**  **}// end main() method  }// end class** |

As shown below, define a method that will be used to display the program menu.

**PROJECT Inventory Valuation - Using Methods in Java**

|  |
| --- |
| **System.*out*.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");**  **}// end main() method**  **static void displayMenu()**  **{**  **// place method body statements below**  **}// end method**  **}// end class** |

Locate the following block of code statements.

|  |
| --- |
| **// end local variable declaration / initialization zone**    **// begin program menu** |

Between the above two comment statements, write the statements below that will call the method that you just created.

**// call a method**

***displayMenu*();**

You will now move a block of the original start code into the method.

Locate the following block of code that is given in the original code.

|  |
| --- |
| **// begin program menu**  **System.*out*.println("");**  **System.*out*.println("");**  **System.*out*.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");**  **System.*out*.println("----Inventory Valuation---");**  **System.*out*.println("\*\*\*(Weighted Average Method)\*\*\*");**  **System.*out*.println("");**  **System.*out*.println("");**  **System.*out*.println("--------- M E N U ---------");**  **System.*out*.println("");**  **// end program menu** |

Cut this block of code in your original code and paste the block in the body of the method **displayMenu()** .

**PROJECT Inventory Valuation - Using Methods in Java**

After you move the code block into the method, run your program and test it again. Did the program run the same as it did before?

**STEP 5 Construct Another Method**

Locate the following block of code that appears within the current version of your program.

|  |
| --- |
| **static void displayMenu()**  **{**  **// place method body statements below**  **. . .**  **}// end method**  **}// end class** |

Construct another method named **averageCost()** , as shown below.

|  |
| --- |
| **}// end main() method**  **static void displayMenu()**  **{**  **// place method body statements below**  **. . .**  **}// end method**  **static double averageCost()**  **{**  **// place method body statements below**  **. . .**  **return average;**  **}//end method**  **}// end class** |

Cut the following variable declarations from your current program and place them in your new method, at the top of the method’s body.

**double average = 0.0, cost = 0.0, totValue = 0.0;**

**int number = 0, sumItems = 0, quantity = 0;**

**String item = "";**

Also, within the while() loop of your current program, cut these comment

statements and all of the statements that they enclose. Place this group of statements into the new **averageCost()** method before the return statement.

**PROJECT Inventory Valuation - Using Methods in Java**

**// begin code block for inventory evaluation**

**. . .**

**// end code block for inventory evaluation**

Finally, modify the output statement in the while() loop to call the method, instead of displaying a variable value.

**// begin code block to display results**

**System.*out*.printf("average cost: $%.2f\n", average);**

**System.*out*.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");**

**// end code block to display results**

The output statement will then look as follows.

**// begin code block to display results**

**System.*out*.printf("average cost: $%.2f\n", *averageCost()*);**

**System.*out*.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");**

**// end code block to display results**

After you move the code block into the new method and modify the output, run your program and test it again. Did the program run the same as it did before?

**STEP 6 Construct Even Another Method**

You will now create another method, which will calculate the tax on the weighted average of the inventory. The tax is computed as the product of the number of items in the inventory and the average cost multiplied by the tax rate.

Use these suggestions to perform this task.

• Declare some static (global) integer variable. This integer will hold the

number of items that are in the inventory.

• Declare some static (global) double variable. This double will hold the

avaerage inventory cost.

• Declare a method named computeTax() that will be used to calculate the tax on the inventory. This method is to receive two arguments, the number of items that are in the inventory and the average cost of the inventory. The method will then return the tax on the inventory, based on a tax rate of 6 percent. Your output for this method will include an example statement such as: " at a level of 133 units, the inventory tax is $ 284.73 . "

**STEP 7 Submit Your Project**

Once you have determined that your modified program is correctly displaying the average cost of the inventory and the tax on the average cost, complete the submission process as follows:

Open MS Word and type a heading for a new document that includes your full name, course number, lab number and date.

Within the document paste in a snapshot of your modified code. Label your snapshot of your modified run with a reasonable description.After your snapshot, paste in your finished source code as well copied in from your NetBeans editor.

Submit your MS Word document to Blackboard when complete.