**Assignment #1: Planning and Developing a Real Estate Commission Calculator**

**Application for American Realty, Inc.**

**(Points: 25)**

**(Due: Sunday, January 31 by midnight)**

**PLANNING THE APPLICATION:**

Recall from Note 5 that the first step in writing a Java program to convert a set of business requirements into a business application is to *plan the program*. *Planning* is a critical component of the *program development process*.

As stated in Note 5, we will use a *four-step program planning process* in this class. The four steps are as follows:

1. Understand the *business requirements* and the *program purpose*
2. Design the *Graphical User Interface* (GUI)
3. Plan the program’s *variables*
4. Design the *business logic*

Read the following business requirements carefully and provide the deliverables for each step in the four-step planning process:

The Chief Information Officer (CIO) of *American Realty, Inc*. wants you to next *design a Java application to calculate real estate commission on real estate transactions (buying a house, selling a house, etc.) for both the agency as well as the agent*. Plan and develop the *real estate commission calculator* application using the information provided in the following table:

|  |
| --- |
| **INFORMATION NEEDED FOR REAL ESTATE COMMISSION CALCULATOR** |
| American Realty agency commission = 6% of the property (house) value |
| American Realty agent commission = 40% of the American Realty company commission |

Use the *following placeholders* to provide the *deliverables for each step of the planning process based on the requirements provided above*.

**Step 1: Understand the business requirements and program purpose**

Enter the *program’s purpose*:

|  |
| --- |
| …  To design a Java application to calculate real estate commission on real estate transactions (buying a house, selling a house, etc.) for both the agency as well as the agent. |

Step 1a: Determine the *program’s inputs*

Complete the following table by adding as many *inputs* as necessary:

|  |  |
| --- | --- |
| **INPUT #** | **DESCRIPTION** |
| Input #1: | The property sale price in dollars |
| Input #2: | … |
| Input #3: | … |
| … | … |
| … | … |

Step 1b: Determine the *processing tasks*

Complete the following table by adding as many *processing tasks* as necessary:

|  |  |
| --- | --- |
| **PROCESSING #** | **DESCRIPTION** |
| Processing #1: | Calculate the American Realty agency commission by multiplying the property sale price by 0.06 |
| Processing #2: | Display the calculated the American Realty agency commission back to the user |
| Processing #3: | Calculate the American Realty agent commission by multiplying the American Realty agency commission by 0.4 |
| Processing #4: | Display the calculated the American Realty agent commission back to the user |
| Processing #5: | Exit the application |

Step 1c: Determine the *program’s outputs*

Complete the following table by adding as many *outputs* as necessary:

|  |  |
| --- | --- |
| **OUTPUT #** | **DESCRIPTION** |
| Output #1: | The American Realty agency commission for a given property sale price. |
| Output #2: | The American Realty agent commission for a given property sale price. |
| Output #3: | … |
| … | … |
| … | … |

**Step 2: Design the Graphical User Interface (GUI)**

The next step is to design the *graphical user interface (GUI)*. Design the *GUIs (input dialogs and message dialogs)* based on the *purpose*, *inputs*, *processing tasks*, and *outputs* from *Step 1*.

***Rough Draft of the GUI (input)*:**

|  |  |  |
| --- | --- | --- |
|  | |  |
| Enter the property sale price in dollars: | | |  | | --- | |  | |
| |  | | --- | | OK | | |  | | --- | | Cancel | | |

***Rough Draft of the GUI (output)*:**

|  |
| --- |
|  |
| The American Realty agency commission is: $xx.yy |
| |  | | --- | | OK | |

|  |
| --- |
|  |
| Agent commission is: $xx.yy |
| |  | | --- | | OK | |

**Step 3: Plan the Program’s Variables**

After a sketch of the GUI design (input dialog and message dialogs) is complete, the next step is to determine the variable names for all user input, processing, and output displayed back to the user.

Complete the following *variable names table* based on *Steps 1 and 2*:

|  |  |  |  |
| --- | --- | --- | --- |
| **Variable Name** | **Data Type** | **Input / Output / Processing** | **Example Data** |
| inputPropertySalePriceInDollarsDouble | double | input | 100000 |
| americanRealtyAgencyCommissionDouble | double | processing | 6000 |
| outputAgencyCommissionString | String | output | The American Realty agency commission is: 6000 |
| realtyAgentCommissionDouble | double | processing | 2400 |
| outputRealtyAgentCommissionString | String | output | Agent commission is: 2400 |
| … | … | … | … |

**Step 4: Design the Business Logic**

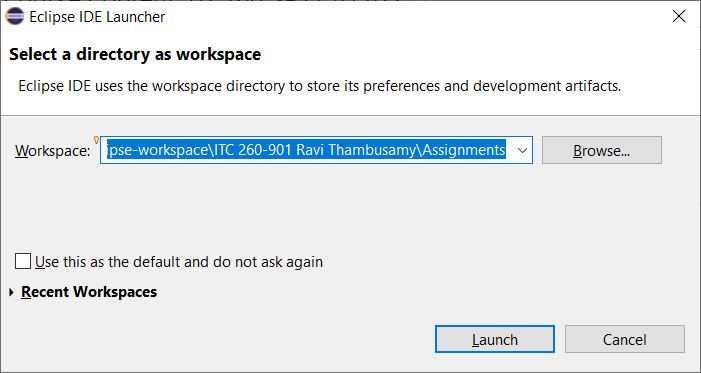
In this step, we are ready to *design the business logic*.   
  
Complete the following *business logic table* based on *Steps 1, 2, and 3*:

|  |  |  |
| --- | --- | --- |
| **Dialog** | **Event** | **Pseudocode** |
| Input Dialog #1 | Clicking the OK Button | Calculate the American Realty agency commission by multiplying the property sale price by 0.06 |
| Input Dialog #1 | Clicking the OK Button | Display the calculated the American Realty agency commission back to the user |
| Input Dialog #2 | Clicking the OK Button | Calculate the American Realty agent commission by multiplying the American Realty agency commission by 0.4 |
| Input Dialog #2 | Clicking the OK Button | Display the calculated the American Realty agent commission back to the user |
| Output Dialog #1 | Clicking the OK Button | Exit the application |

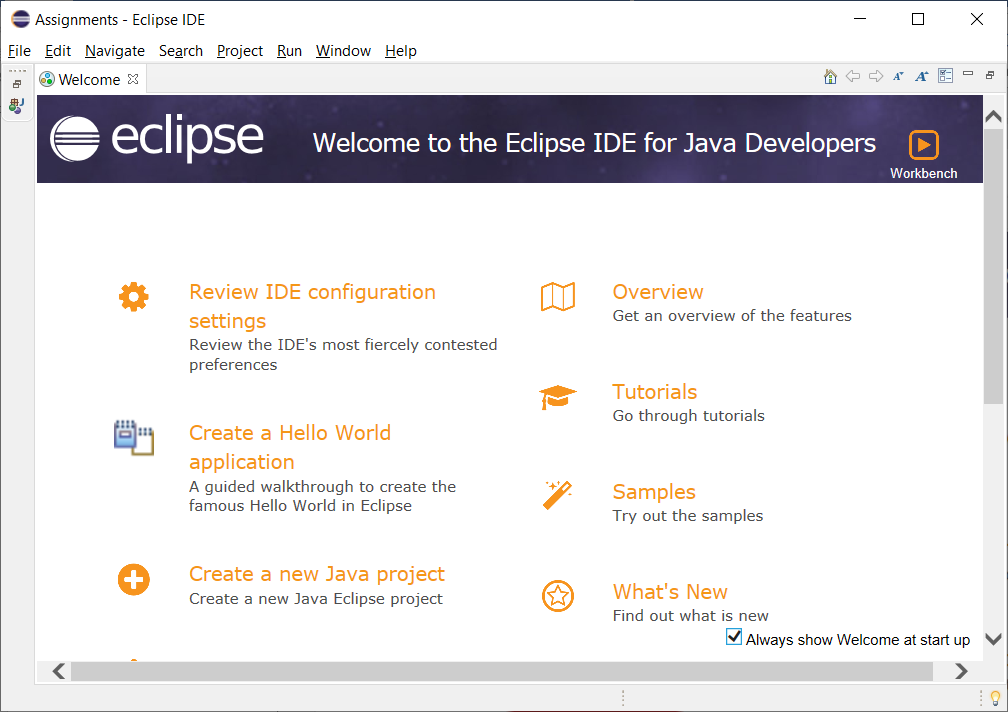
**DEVELOPING THE APPLICATION:**

Please complete the following step-by-step instructions in order to *develop this application*:

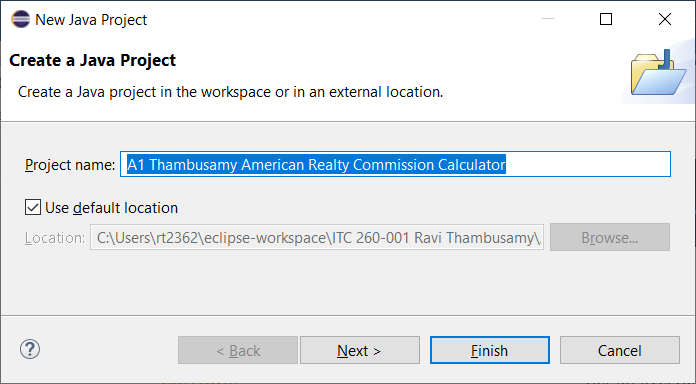
1. Using Windows File Explorer (shortcut is Windows key + e), locate the folder named “*ITC 260-902 your full name*” (example: “*ITC 260-902 Ravi Thambusamy*”). Double click to open that folder and then double click to open the *Assignments* folder.
2. Open the *Eclipse IDE for Java Developers 2020-09* by double-clicking the shortcut on the desktop or by clicking the icon pinned to the task bar.
3. Set the directory as workspace to the *Assignments* folder referred to in *Step 1*.



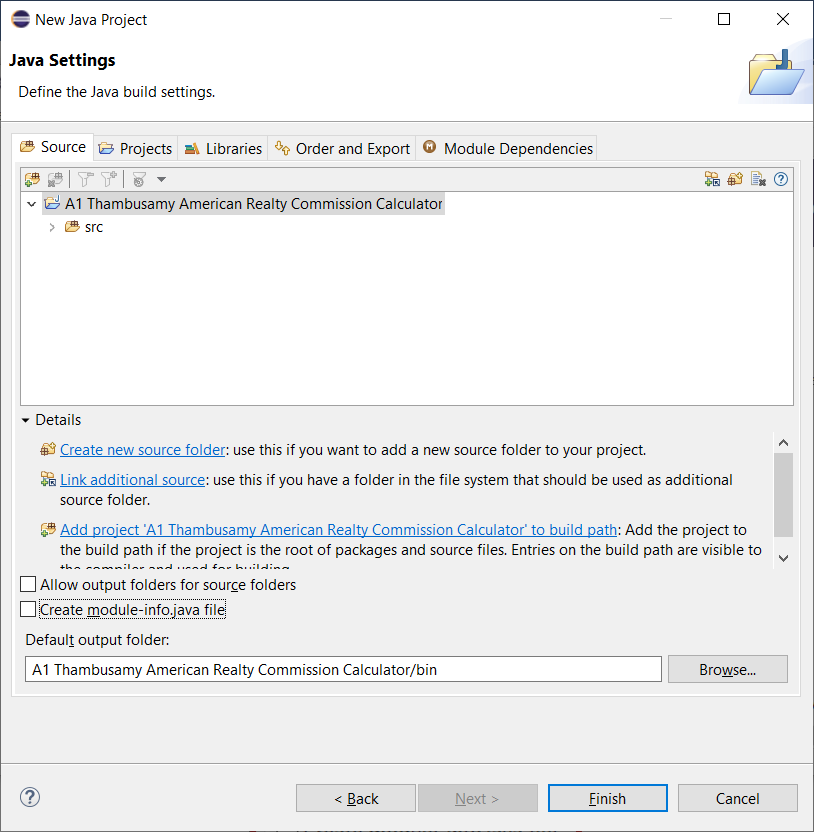
1. Click on the *Launch* button.
2. In the Welcome screen, click on the “*Create a new Java project*”.



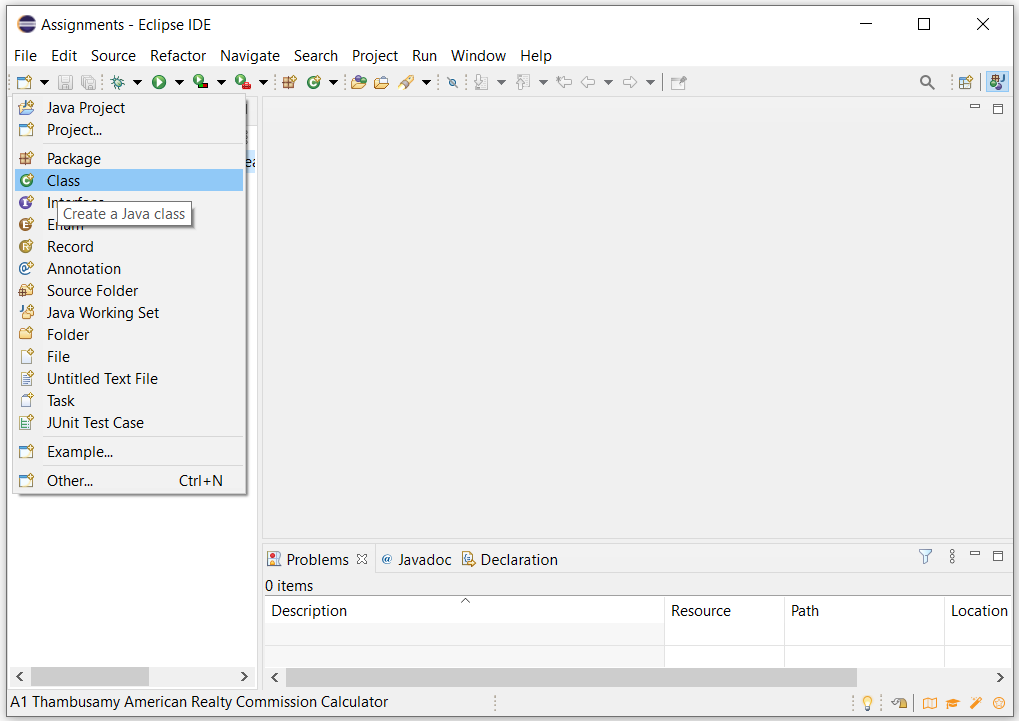
1. In the next screen, enter “*A1 your last name American Realty Commission Calculator*” as the *project name* and then click on the *Next* button.



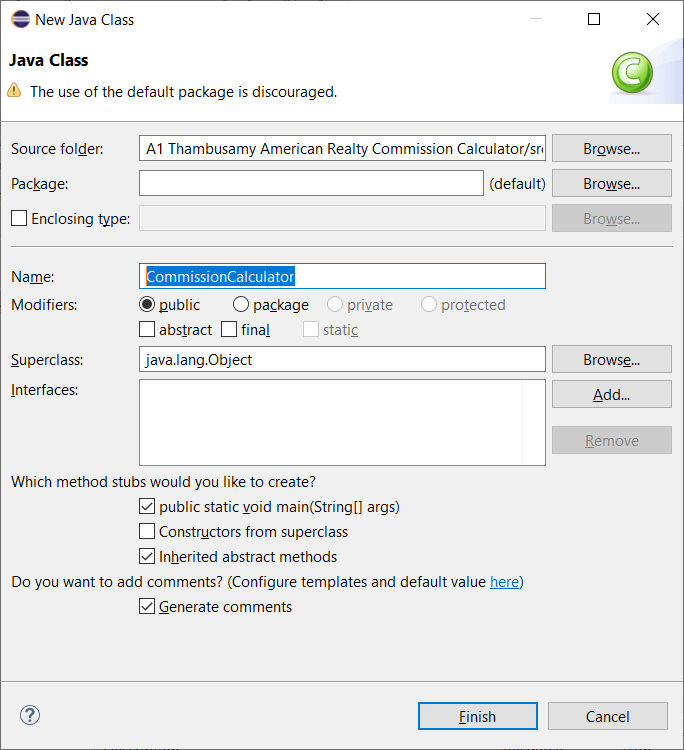
1. In the next screen, *uncheck the check box* next to *Create module-info.java file*.



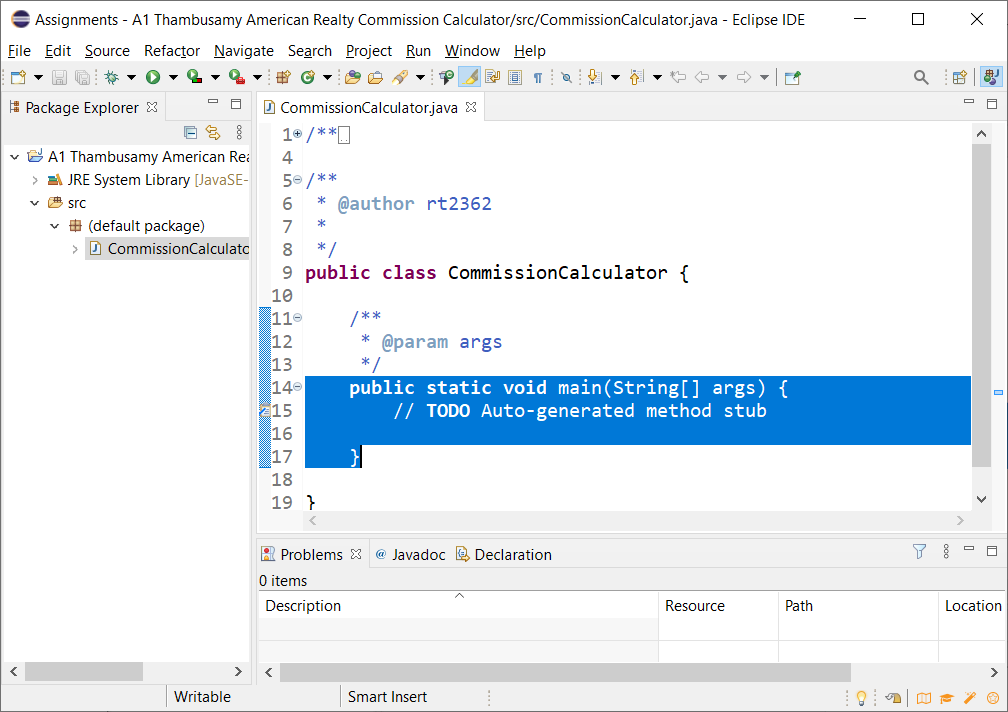
1. Click on the “*Finish*” button.
2. In the Eclipse IDE, make sure that there is only one active project that is open. If any other projects other than this project are open, go ahead and close those first. Close the *Task List* pane and the *Outline* pane. Next, click on the “*New*” drop-down button and then select “*Class*”.



1. This will open a “*New Java Class*” window. Enter “*CommissionCalculator*” in the “*Name*” box. Check the box next to “public static void main(String[] args)”. This will add the empty main() method stub to the class. Check the box next to the “*Generate comments*” box as well. This will add *comment statements* to our code. Finally, click on the “*Finish*” button to create the class.



1. You will see a *main() method stub* and several *comment statements* added to the *CommissionCalculator.java* Java source code file.



1. On *line 1* in the *CommissionCalculator.java* source code, add an import statement that will allow you to work with *input dialog boxes* and *message dialog* *boxes*.
2. Replace *lines 2 through 4* in *CommissionCalculator.java* with the following lines of code:

/\*\* TO DO 13: Provide the following information

\* Class Name:

\* Class Purpose:

\* Programmer Name:

\* Date:

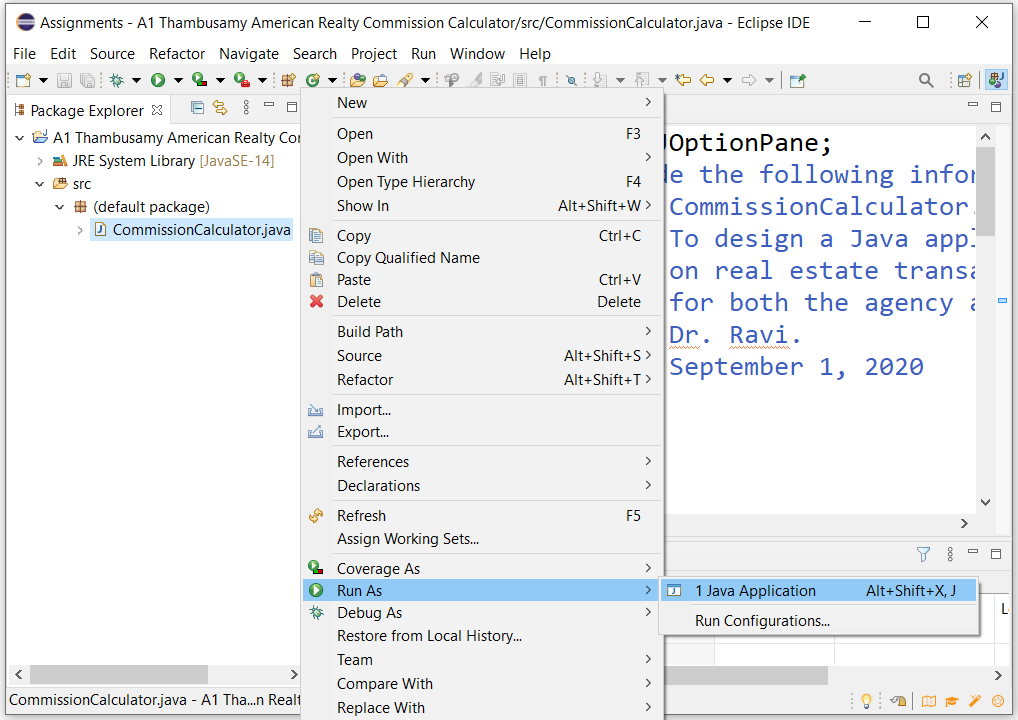
\*/

Add the missing information in the comments section (lines 3 through 6) using the Eclipse IDE.

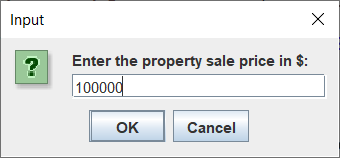
1. Inside the *main()* method, *declare the variables* you identified in the *variable names table on page 4*. Make sure you follow the appropriate *naming conventions* (use the *Camel* naming convention and add the *data type* to the end of the variable names) when you declare the variables.

1. *Implement the* *business logic* from the *business logic table* using the following steps:
   1. Get input *property sale price* value from the user using an *input dialog box*.
   2. Store the value input by the user in the appropriate *variable* (NOTE: You will have to convert the input value into a *double* before you can use it in the calculation).
   3. Calculate the *agency commission* for the given *property sale price* using the *formula* provided in the *business requirements on page 1*.
   4. Store the result in the appropriate variable you declared in *Step 14.* Use string concatenation to concatenate strings in order to display the output message as shown in the program output in *Step 22*.
   5. Display the *output message string* back to the user using a *show message dialog box*.
   6. Calculate the *agent commission* for the given *agency commission* using the *formula* provided in the *business requirements on page 1*.
   7. Store the result in the appropriate variable you declared in *Step 14.* Use string concatenation to concatenate strings in order to display the output message as shown in the program output in *Step 22*.
   8. Display the *output message string* back to the user using a *show message dialog box*.
   9. Refer to the output screenshot in *Step 22* for *sample output*.
2. Make sure that there are no *syntax* *errors*, *logical* *errors*, or *runtime* *errors* in your program.

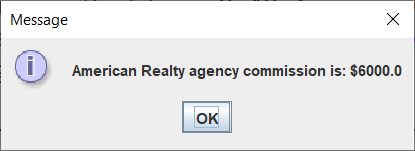
1. Click on the *Save All* button.
2. In the IDE’s *Package Explorer* window, right click on the “*CommissionCalculator.java*” file, select “*Run As*” and then “*Java Application*” as shown below.

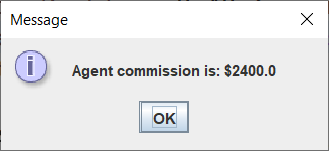


1. If you are prompted to “*Always save resources before launching*”, then check that box in addition to placing a check mark next to *CommissionCalculator.java*. Then click on the “*OK*” button.
2. The program will prompt you to enter the *property sale price in $*. Enter *100000* for *property sale price in $* as shown below:



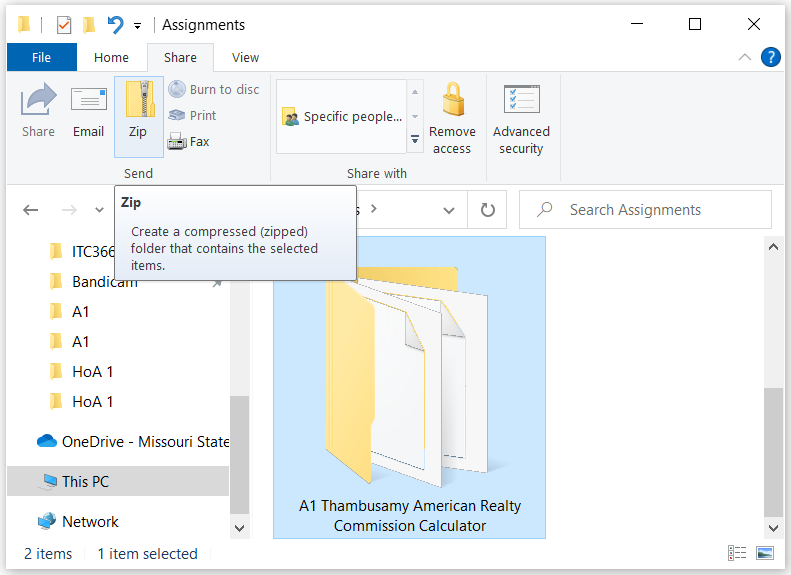
1. You will see the *output* of the *CommissionCalculator.java* program in two *message dialog* *boxes* (one for *agency commission* and the other for *agent commission*) as shown.



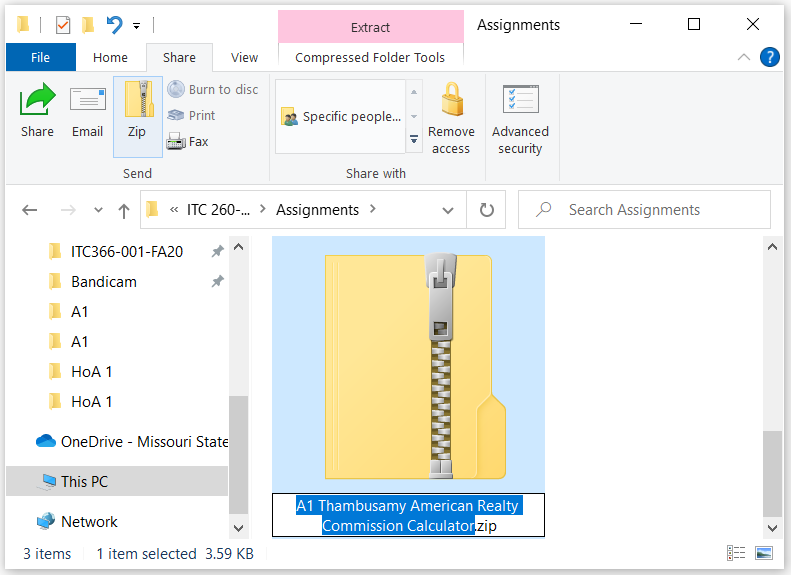


1. *Test the application* by entering different values for *property sale price in $* and make sure that the application works correctly for all possible scenarios of correct and incorrect input and displays the appropriate messages back to the user using *show message dialog* boxes.
2. Close the *Eclipse IDE*.

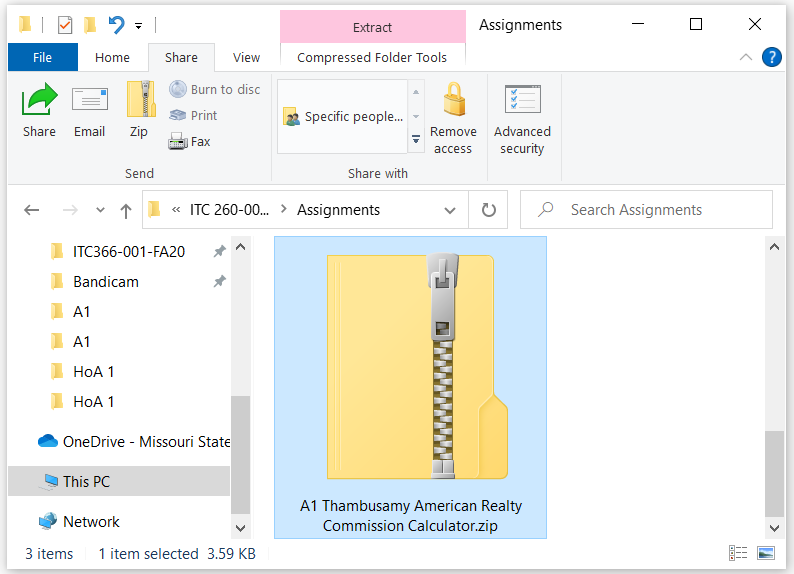
1. Use Windows File Explorer (shortcut is Windows key + e) to go to the “*ITC 260-902\Assignments*” directory you created in *Step 1*.
2. *Save a copy of this Word document* inside the “*A1 your last name American Realty Commission Calculator*” folder [NOTE: If you do not do this, then you will only get credit for developing the Java application, but not for planning the Java application!].
3. In Windows File Explorer, click on the “*A1 your last name American Realty Commission Calculator*” directory to select it. Next, click on the *Share* menu in *Windows File Explorer*.



1. Click on the *Zip* icon.

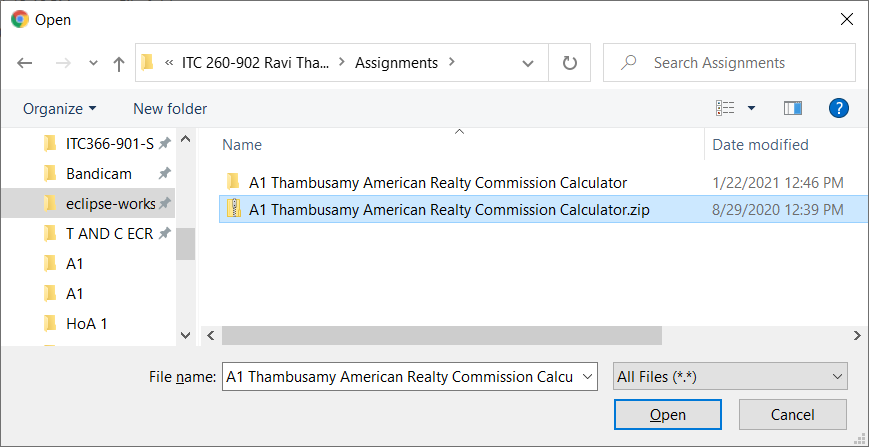


1. Leave the file name as it is and hit the *Enter* button on your keyboard to finish creating the *zip file* for this assignment.



1. Now, log on to Blackboard and click on the “*Assignments*” link under the “*Course Evaluation*” section heading.

1. Next, click on “*Assignment 1 (Due: Sunday January 31 by midnight)*” link.
2. In the “*ASSIGNMENT SUBMISSION*” section, click on the “*Browse My Computer*” button. In the *File Upload* dialog box that opens up, navigate to the *ITC 260-902\Assignments* folder and then select the “*A1 your last name American Realty Commission Calculator.zip*” file and then click on the *Open* button.



1. Finally, click on the *Submit* button on Blackboard to complete the submission. Always verify that your zip file has correctly uploaded on Blackboard. If you accidentally submit the wrong zip file, then you will not be given a second opportunity to resubmit the assignment!