Lab Four Assignment

Due: by 10:00 pm, Thursday, 11/18 (check scheduled tutor time for help)

All labs in this course require to use Eclipse as Java IDE in coding and grading. Please follow instruction in **Steps how to work on your labs and make a zipped file for submission in Eclipse** posted in Modules of Canvas to complete this assignment.

Part I: Apply JavaFX or Swing GUI components and exception handling to design a ***SquareApp*** class using proper GUI components, such as labels, text fields and buttons (Submit, Clear, and Exit) to compute perimeter and area of a square and display the side length and computing results in a proper location in the application window. It will also verify invalid data entries for sideLength (No letters and must be a positive real number) using exception handling. Your code will also have a custom-designed exception class named **NegativeDoubleException** to handle the negative data exception. Your validation info may be displayed in the application window or use output method of **JOptionPane** to prompt user for the correct data entry.

- Run and test your code to meet the requirements.
- Must use meaningful names for fields, variables, methods and classes.
- Must use JavaFX or Swing GUI components (Label, TextField, and 3 Buttons). You may use any layout manager in your code.
- Must document each of your source code (see example in the next slide)

Part II (Optional for extra-credit): The exercises you have completed ***all even-numbered M/C and T/F questions, all even-numbered Predict Output, and all even-numbered Algorithm Workbench*** in chapter 10 to 12 using any text editor or saved in PDF.

Create a folder called answers and copy and paste all of your exercises from the two chapters to this folder, then copy this folder into Eclipse project folder Lab4 (see **Steps**.. In the next sides).

Steps how to work on your labs and make a zipped Eclipse project file for submission

The steps how to work on your labs with Eclipse

Step 1. Open Eclipse you have installed, click on the triangle icon under **File** menu, and you will see a drop-down menu.

Step 2. Click on **Java Project** and you will see a new window to create a Java project, since Eclipse organizes all programs as projects, so does for your labs.

Step 3. Type in your project name, say, Lab4 in the **Project name:** field, and then click on **Finish** button. Eclipse will create a project named Lab4for you.

Step 4. Click on Lab4, highlight the **src** folder, click on the triangle icon under **File** menu, and you will see a drop-down menu.

Step 5. Click on **Class**, a window let you create a new class will be displayed. Type in the class name you are going to code, say, TestScores, and click on **Finish** button. If you are creating a driver class, say, TestScoresApp, after you type in this class name, select the box **public static void main(String[] args)**, and then click on **Finish** button.

Now you can type your code in this editing window and execute your code by clicking Run menu or the play icon shown under menu bar.

Steps to make your project as a zipped Eclipse file for your lab submission

Step 1. Highlight the project you are going to submit, click on **File** menu, and click on Property as shown in Figure 1.

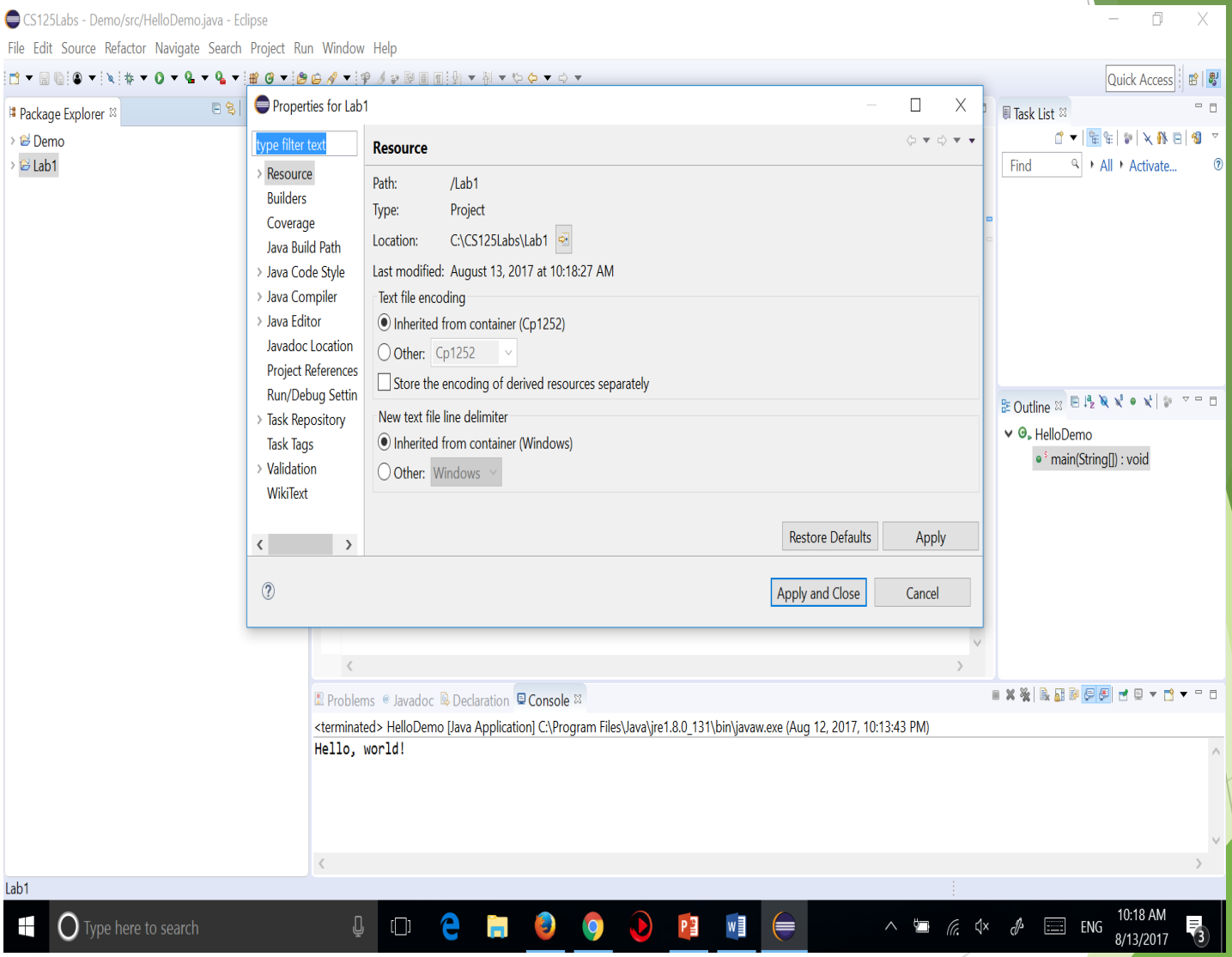

Figure 1 Property window shows the location of your project

Step 2. Click on the icon beside the location path and it will navigate to the directory your project located in your computer.

Step 3. Highlight the folder, copy and paste it to another location in your computer.

Step 4. Navigate to the folder you have done the optional extra-credit exercises, copy and paste this folder into this submission folder. Omit this step if you choose not to do these exercises.

Step 5. Make right mouse button click, select **Send to** and select **Compressed (zipped) folder** as shown in Figure 2.

A zipped file called Lab4.zip will be created and ready for your submission in canvas. Mac users may refer to steps to make a zipped file and requirement is the same.

(See next slide)

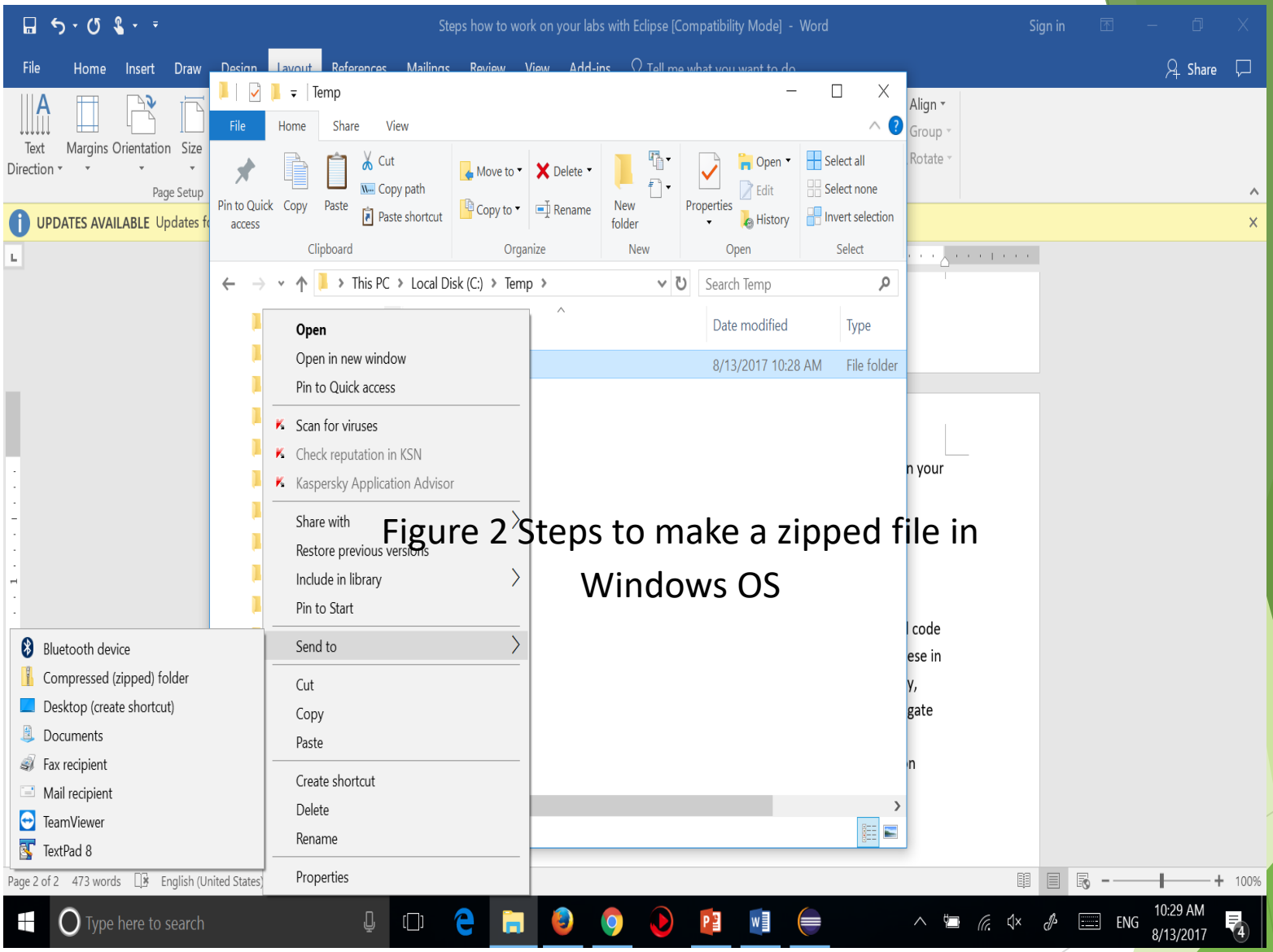

Figure 2 Steps to make a zipped file in Windows OS

Figure 2 Steps to make a zipped file in Windows OS

Step 6. To test if your submission is a correct zipped Eclipse project that can be imported, opened, and run for grading, click on File, Switch Workspace, Other…, Type a new workspace name, say, Test, Eclipse will create a new window as a workspace for you, close the welcome window, then click on File, Import…, click on General, Existing Projects into Workspace, click on Next, choose Select Archive File:, browse to the folder you have saved your submission file, highlight the zipped file, then Open, and click Finish. Your zipped Eclipse project as you lab should be displayed in the Eclipse window, if you did your zipped file correctly.

To submit your lab, log in the course in Canvas, submit it by clicking on the Assignments, clicking on Lab4 and then **Submit Assignment**, **Choose File**, browse and attach the zipped file, click on **open**, then click the **Submit Assignment** button to complete your submission.

Example of documented source code (Required for each of your source code)

```
/*********************************************************************************
Name: Jon Smith
Course: CS125-0X
Lab #: Lab Four
Submission Date: 10:00 pm, Thu (11/8)
Brief Description:  The driver code to run TestScore, create its object, call the methods
to assign and display the letter grade.
*********************************************************************************/
public class TestScoreApp {      //Driver class
      public static void main(String[] args) {      //main method
            TestScore tesScore = new TestScore(); //create object of TestScore
            testScore.inputScore();       //call method to have the score
            testScore.assignGrade();    //call method to assign the grade
            testScore.display();      //call method to display the grade
      }      //end of main()
}  //end of the driver class
```

- You are required to do the documentation like above for each of source code, i.e., for TestScore.java