Lab One Assignment

Due: by 10:00 pm, Thursday, 9/22 (check scheduled tutor time for help if you have any questions)

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** below in this spec (also posted in Modules of Canvas) to complete this assignment.

Part I: Code a ***Rectangle*** class that has the following fields:

- width : double
- Height: double

- The class should have the following methods:
  - Constructor: Accepts the width and height of the rectangle as arguments.
  - setWidth and setHeight: Mutator methods for setting these two fields.
  - getWidth and getHeight: A accessor method for returning the radius field.
  - compputeCircumference Computes the circumference of the rectangle and returns it.
  - computeArea: Computes the area of the rectangle returns it. area = width * height
- Then code a testing class (driver class) **RectangleApp** that tests the Rectangle class by asking the user to enter the width and height of a rectangle (calling methods of Scanner or JOptionPane), creating an object and call its methbods to pass the input data, displaying the values of input data and computed results, by calling correct methods,respectively.
- Coding Requirements:
  - Must use required/meaningful names for fields, variables, methods and classes.
  - Must document each of your source code (see example in the next slide)
  - Must separate operation class ***Rectangle*** from the driver class ***RectangleApp***.

Part II: Design a ***RomanNumerals*** class that takes a number within 1 to 10, convert and display it to the roman numeral version of that number (I, II, III, IV, V … X). If the number entered by user is outside the range of 1-10, the program should display an error message and stop to run. Code a driver class **RomanNumeralsApp** to test the class by asking user to enter a number, creating an object of **RomanNumerals** and calling its methods **getNumber**, **setNumber**, **convertNum** and **displayResult** to perform the described task.

- **Coding Requirements:**
  - Must use required/meaningful names for fields, variables, methods and classes.
  - Must document each of your source code (see example in the next slide)
  - Must separate operation class **RomanNumerals** from the driver class ***RomanNumeralsApp***.

Part III (Optional for extra-credit): The exercises you have completed from Chapter 1 to 4 including ***all even-numbered M/C and T/F questions***, ***all even-numbered Predict Output***, ***and all even-numbered Algorithm Workbench*** in these 4 chapters using any text editor.

Create a folder called answers and copy paste all of your exercises from Chapter 1 to 4 to this folder, and then copy this folder into Eclipse project folder Lab1. (See the **Steps…**)

How to work on your Lab in Eclipse and Lab submission requirement

(See next slides)

Submission requirement:

You must submit all of the parts above as an Eclipse project file named **Lab1** by following instruction in **Steps how to work on your labs and make a zipped file for submission in Eclipse** below (also posted in Modules of Canvas).

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, Lab1 in the **Project name:** field, and then click on **Finish** button. Eclipse will create a project named Lab1 for you.

Step 4. Click on Lab1, 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.

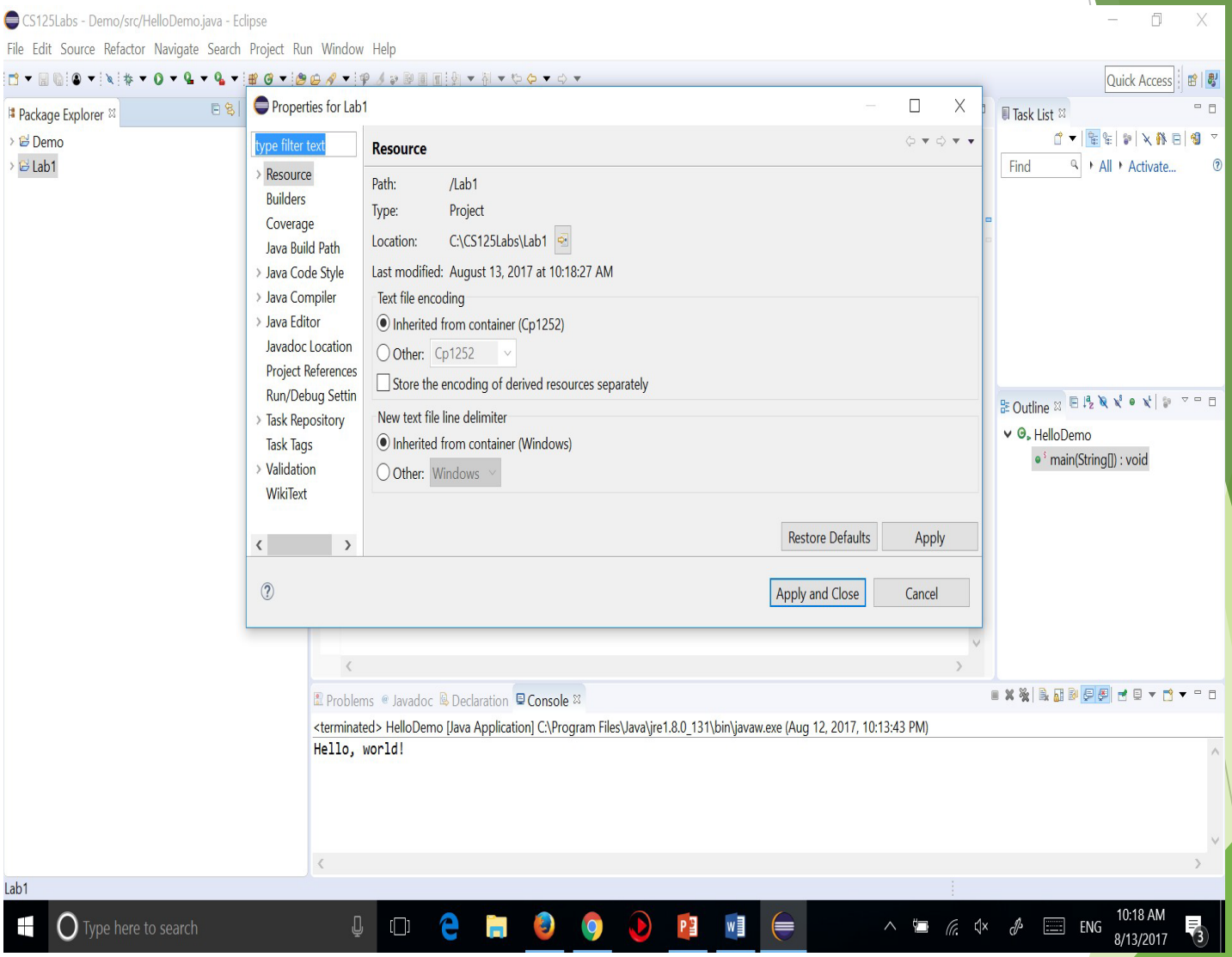

Figure 1 Property window shows the location of your project

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 Lab1.zip will be created and ready for your submission in canvas. Mac users may refer these steps to make a zipped file and requirement is the same.

(See next slide)

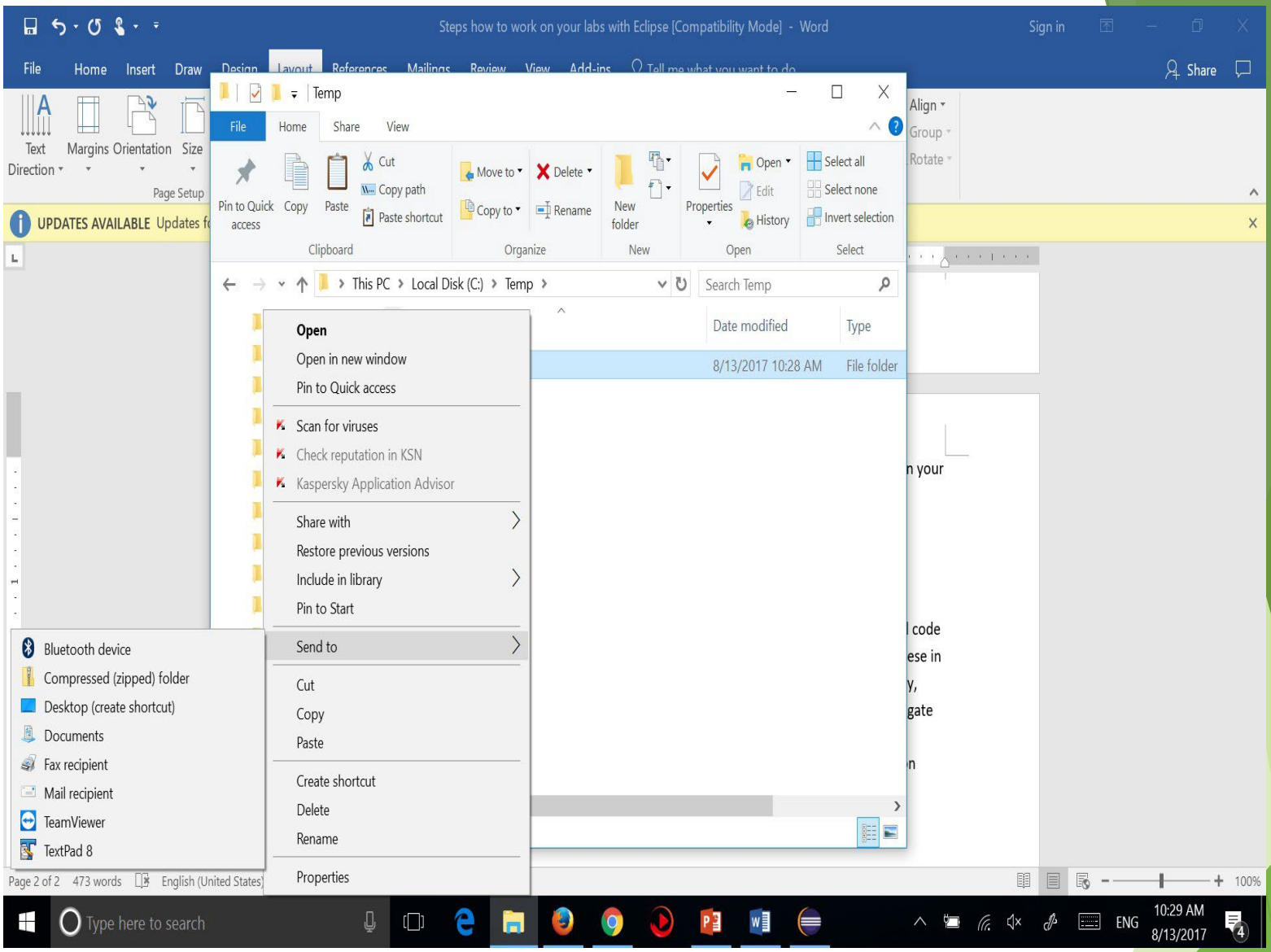

Figure 2 Steps to make a zipped file in Windows OS

Step 6. To test if your submission is a correct zipped Eclipse project that can be imported, opened, and run for instructor's 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 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 Lab1 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 One
Submission Date: 8:00 pm, Wed (9/20)
Description:  The driver code to run TestScore, create its object, call its methods to
accept input, 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. You must have a code title at the top as comments shown above.