

Lab 1

Become familiar with Lab, the Revel Textbook, and the Eclipse IDE.

Pre Lab

- o Make sure that you have read Chapter 1 carefully
- o While reading, do the Self-Check Questions
- o While reading, make sure to watch all of the videos, especially the following ones:
 - o Your First Java Program (Video in 1.7)
 - o Compile and Run a Java Program (Video in 1.8)
 - o Eclipse Brief Tutorial (Video in 1.12)
- o Inside the Revel Textbook, do the '1.7 Exercise Set'
- o **Make sure to review the “Key Terms” and “Chapter Summary” frequently and especially before every Lab Session**

In-Lab Activities

Pair Programming (Swap if necessary)

Watch this [video](#) on Pair Programming and its benefits.



Professional programmers use a technique called **pair programming** – writing computer programs in teams of two. One partner controls the computer, the other helps plan and helps catch errors. *They switch roles often.*

You will do that in this course as well. Even though everyone has their own computer, you must work efficiently with your partner to complete the projects in Lab so that both of you understand the concepts. One person will be coding while the other is “navigating” - you both must discuss the activity while working. As a pair, you only need one copy of each activity, but it is highly recommended that both of you keep a copy of each one for personal reference. Make sure to switch roles after each activity. The icon above will serve as a reminder to do so.

Despite the fact that you both have your own computer, you must only work on the “coder’s” computer. The “navigator” may use their computer to pull up references to be more efficient, but they should not be the one coding (not until roles switch and they are the now the current coder).

Before beginning, make sure you know who your partner is. Every Lab, you will be required to swap partners to work with someone different than the previous week’s lab. In this way, you collaborate with someone new every Lab session, while learning the concepts individually.

Warm-Ups

Go to the Google Forms link on the board and complete your Warm-Up Questions. You have 10 minutes to complete this. If you finish early, feel free to begin the Lab Activities, but make sure your partner is ready as well.



TIP

We many often give you a set of code in which you will have to find and fix errors – expect *many* bugs. These specific problems will be denoted by a debugging icon.

Activity 1: Display five messages

Write a program that displays “Welcome to Java” five times.

Now edit the program to display your own message seven times.



Activity 2: Back to the Introductions

Write a program to display your partner’s introduction from the beginning of Lab.



CHECKPOINT

10 MINUTES



Activity 3: Finding the Error

There is an error within the code shown in the screenshot below. Create a class called Activity3. After typing this code and compiling, you will see a red line at the bottom. What does this line mean? How did you fix the error?



```
public class Activity3 {  
    public static void main(String[] args)  
    {  
        System.out.println("There is a Problem in this code!")  
        System.out.println("If this message is printed that means")  
        System.out.println("The errors are fixed! GOOD JOB")  
    }  
}
```



Activity 4: Finding the Error

There is an error within the code shown in the screenshot below. Create a class called Activity4, and replace all the code which was autocompleted by Eclipse with what is shown below. After typing everything in and compiling, you will see a red line in various places throughout the code. What does this red line mean? How did you fix the errors? What did you learn from this activity?



```
public class DebugActivity2 {  
    public static void main(String[] args);  
    {  
        String course = "ITCS 1212";  
        system.out.println("The Course name is : " + course);  
    }  
}
```



CHECKPOINT

15 MINUTES

Reflection

Go to the Google Forms link given by your TA to work on your Reflection for Lab 1. This counts as a part of your participation grade, and it is important that you put as much detail as possible.

Post Lab (Individual work and submission)

Revel Work

Do the following activities from within the Revel Textbook.

- o Chapter 1 Programming Project 1
- o Chapter 1 Programming Project 2

Eclipse IDE Setup

Go to Canvas and download the Environment Set Up instructions; this will be located under this week's section. Make sure to download the one specific to your machine – either Windows or Mac. Go through all the instructions and take screenshots of your set up process. These screenshots will be turned in with everything else at the end, and counts as a part of your Post Lab grade. Have at least five screenshots minimum and put them in a Word Document.

Start Programming

1. Do Exercise 1.9 (Area and Perimeter of a Rectangle) from the Revel Textbook: Write a program that displays the area and perimeter of a rectangle with the width of 4.5 and height of 7.9.
2. Before starting this next program, it will help to go over parts of Chapter 2. Also, write the pseudocode (steps of the program) in the top comments before coding.
Jean is visiting Britain and must rent a car. However, the system uses kilometers rather than miles. Write a program that allows her to enter a number, and converts from kilometers to miles so that she can make sure she doesn't go over the speeding limit and gets a ticket.
Hint: 1 kilometer = 0.621 mile.

Important Concepts

Answer the following questions to turn in. These can be answered on the same Word Document as your Set-Up screenshots.

1. Explain the differences between different types of programming errors, namely (1) Syntax errors, (2) Runtime errors, and (3) Logic errors.
2. List five major hardware components of a computer.

Submit Work

You may zip all your files from each section (Setup screenshots, Programming projects, and Important Concept answers) into one folder to turn in on Canvas. Name the zipped folder "LastName_PostLab1" before uploading.