

[Support Links](#)[Request Links](#)[WebCampus Links](#)[Moodle Home](#) ► [2014A](#) ► [XLS-CS-501-WS 2014A](#) ► [Assignments](#) ► [Assignment 1 due 6/4](#)

Assignment 0:

Start your programming work by preparing your computer for Java software development by downloading the following from the [Companion Web site](#):

Supplement Part I Modules A Glossary, B Installing and Configuring JDK (Java Development Kit), and C Compiling and Running Java from the Command Window

In Section 2.2 the `System.out.println()` method is used. This should work for executing programs using the Command (CMD) Prompt Window. The Scanner class for input, described in Section 2.3 and Listing 2.2, works in the Command Prompt Window but can be awkward (it may have to be used multiple times to enter more than one input, like a number and a date). For output to the standard console device (the name of the default output device, usually the screen or, formerly, the printer) use one of the following commands:

- `System.out.println("...text...");`
- `System.out.print("...text...");`

The first prints a line and moves the cursor to the next line; the second prints the texts and stops, allowing further output to continue on the same line.

When you submit Java programs they should be in text files with a `.java` extension. A simple text editor will work but you may need to change the `.txt` extension to `.java`. One of the important Java conventions is that the file name, before the extension, has to be exactly the same as the name of the class in the program that contains the `main()` method. See all the Listings (e.g., 1.1 on p 17, 1.3 on p 19, and 1.4 on p 22) and notice how the file name (before the `.java` extension) agrees with the class name. [Java Programming Style Guidelines](#) is a useful site in this regard, there are many others.

For the purposes of this course please name the files using both the Chapter and Exercise number and a short descriptive name. For the first program the name of the file and class should be `C1E1Welcome.java` and the second could be `C2E21FutureInvestmentValue.java`. Also indicate, maybe with first submission, which edition of the text you are using.

Assignment 1:

Read the parts of Chapter 1, Sections 1.1-5, with which you are not familiar. From Section 1.6 until the end of Chapter 1 is basic programming knowledge that you will need it for all assignments. Pay particular attention to Section 1.19 for displaying messages in a dialog box window. After the first few assignments using Scanner you should only use dialog box windows.

Do Exercise 1.1 using the Command Prompt Window.

Read Chapter 2 Sections 2.1-12, paying particular attention to Section 2.3 where the Scanner class, and its principal methods in Table 2.1, is introduced. This class supports user input from the “console”, which usually is the keyboard, but can also be a numeric key pad or any other character/number input device set up on your computer. Until we get to computer graphics the input console will be the keyboard. Do at least one of the exercises in Chapter 2 using scanner for both input and output, but you should try to use pop up windows with JOptionPane classes as soon as you can.

Assignment 1	9th Edition	8th Edition
CS	1.1, 2.4, 2.15, 2.21	1.1, 2.4, 2.14, 2.21
MIS	1.1, 2.4, 2.11, 2.21	1.1, 2.1, 2.13, 2.14

After 1.1 you should begin to work with an Integrated Development Environment (IDE) instead of the CMD. These are programs that have a text editor, provide line numbers for your programs, indicate many syntax errors, color code commands, variables, etc. and support debugging. Most programmers these days use one or more IDEs.

Supplement Part II discusses and provides directions for several of these IDEs; you are welcome to try as many as you want. If you do not care to investigate which one, I suggest you use Eclipse, an open source IDE.

In addition, IDEs usually have debugging tools. If you learn to use the debugging tools you may never again have to use neither Scanner nor `println()` statement at all.

Starting with your work from Chapter 2 on you should use pop-up windows for both input and output instead of Scanner and `println()` classes. Pop up windows can be implemented by paying attention to the

- String Class in Section 2.18,
- To format output in popup windows attention needs to be paid to the so-called escape characters in Section 2.17.2 and Table 2.6. In particular, the `\n` escape character allows multiple lines in such output windows. `JOptionPane.showMessageDialog()` methods in Section 1.10 for an output pop-up window, and
- `JOptionPane.showInputDialog()` methods which create popup windows for user input in Section 2.19. These methods provide for a popup window in which a prompt can be shown and into which user can enter program input.

Take your time with all this; what you learn here will serve you well during the rest of this course and the rest of your career as a programmer.