

Lesson 2.02 — Declaring & Assigning Variables

Overview

Objectives — *Students will be able to...*

- Identify, declare, and assign variables.

Assessments — *Students will...*

- Write a program that converts temperature from Fahrenheit to Celsius.

Homework — *Students will...*

- Read the rest of BJP 2.2
- Complete self-check questions 5, 6, 9, 12-15 (4th edition: 6, 7, 10, 14-17)

Materials & Prep

- Projector and computer (if you are able to/opt to use Eclipse with your students)
- White paper and markers
- Classroom copies of WS 2.2
- Pair or small group student assignments
- Sample online temperature converter (<http://www.onlineconversion.com/temperature.htm>)

Since most of today's lesson follows WS 2.2, you should have read through the worksheet. You may prefer to delete the notes from the worksheet (so it is only a sheet of exercises) if you are working on developing note-taking skills in your classroom. We recommend leaving these sections in for ELL classrooms, so your students can focus on syntax rules instead of translating what they are hearing to vocabulary they need to then write in their notebooks.

Pacing Guide

Section	Total Time
Bell-work and attendance	5min
Introduction and worksheet exercises	25min
Practice exercises	20min
Turn in worksheets, wrap up	5min

Procedure

Since much of this class involves learning syntax, there will be a lot of drilling during the class. Try to spice up the lesson by allowing students to work in pairs, or playing soft music in the background to put students in the right headspace to settle down for work.

Hook your class today by asking which of them are taking or have taken physics or chemistry. Ask students about working with Fahrenheit and Celsius temperatures—do they have to convert temperatures in class? Which measurement are they more familiar with? Which do they use more often? Show students the online calculator and ask if they ever use such online tools, and tell students that they're going to learn how this program is built today.

Bell-work and Attendance [5 minutes]

Introduction and Worksheet Exercises [25 minutes]

Emphasize with students...

Content - Debugging Tools As you continue to program in Eclipse you will notice some of the debugging tools that are available to you. One of these tools is the syntax coloring whereby syntax errors will be highlighted or underlined a specific color. This draws the programmer's attention to the error and allows for quicker fixes.

Even though these tools exist, it is still important for you to get good at identifying syntax and other errors. This will speed up the development process and will result in fewer errors in the final program.

-
1. Using WS 2.2, walk students through the proper way to declare a variable.
 - Be sure to spot-check for understanding by having students give you the definitions of **type**, **syntax**, **declaration**, and **variable** (all bolded in the text).
 - Encourage students to use their notes if needed.
 2. Guide students through the syntax rules for variable declarations by working through the first few examples of Exercise 1 in pairs.
 3. Give students a few minutes to complete Exercise 1 on their own; encourage students to tackle Exercise 2 as well, then check all answers together as a whole group.
 4. Using the figure on Exercise 3 of WS 2.2, walk students through the proper syntax to assign a variable.
 - Spot-check for understanding by asking students to define the italicized words.
 - Ask students for a few sample answers, correct them if needed, then give students a few minutes to complete Exercise 3 in pairs.
 5. As a whole group, walk students through Exercise 4 and 5. Complete 5a together as a group, then let students work on 5b in pairs.

At this point, your class may be raring to get started on the rest of the assignment without your help. If they are, great! Post the practice questions on the board so they can continue to that assignment once they have completed the worksheet. If your class wants you to walk them through string concatenation, go through the examples as above.

Practice Exercises [20 minutes]

1. Have students complete the following practice self-check questions:
 - a. Self-Check 2.8: studentVariables
 - b. Self-Check 2.13: valuesOfABC
2. Have students complete Exercise 2.1: displacement.
3. Students should work on their own, but if the exercise is too challenging, you might opt to have students collaborate on answers. Be sure to remind students that each student should turn in their own set of work.

Students turn in worksheets, wrap up [5 minutes]

At the end of class, collect WS 2.2 and practice problems submissions.

Accommodation and Differentiation

If you have students who are speeding through this lesson, you should encourage them to:

- Complete Self-Check 2.18: timesOperator.
- Challenge them to build their own program that converts Fahrenheit to Celsius (this version won't take user input—yet!)
- Have the student create a classroom poster diagramming the parts of variable declaration & assignment.

If your class is struggling with learning syntax, you can split the lesson into 2 lessons, and/or take off some of the homework questions. If splitting the lesson in two, we recommend stopping today's lesson before string concatenation.

Common Mistakes

Variables common mistakes: <http://interactivepython.org/runestone/static/JavaReview/VariableBasics/commonMistakes.html>

Misconceptions

- Students will draw on their math knowledge when learning variables. This leads to confusion on the differences in a programming language.
- The equal sign = is assignment in Java, not equality. When reading out code, explicitly saying “assignment” will help reinforce the concept that = is not equality: `x = 1` is read as “x is assigned the value 1”.
- A variable is a container for value that can change, it does not denote a fixed value. From WS2.2, consider the following statements:

```
int age = 17;  
age = age + 1;
```

Students may view the second line of code as a math equation and miss the point that age is a variable and contains a value. The `age = age + 1` changes the value of age. When learning Java there is the code which is static, the running of the code, and the state of the variable during runtime which is fundamentally different from an equation in math that can be substituted and manipulated to solve for some unknown.

- Students read `y = x + 2`; and think the *equation* is stored in y, not a *value*.

```
x = 1;  
y = x + 2;  
x = 3;  
System.out.println (y); // what is displayed for y?
```

For students with the misconception that the *equation* is stored, they will incorrectly compute 6 as what is displayed for y.

Java is not a spreadsheet that stores the equation. Explicitly teaching that x and y are independent variables in a programming language and hold values is an important distinction. This differs in math where `y = x + 2` is a relationship. Showing the state of the variables can be achieved using the whiteboard, the debugger by stepping through one line of code at a time, and examining variables, or using a visualization tool like http://cscircles.cemc.uwaterloo.ca/java_visualize/.

Video

- BJP 2–2, *Variables and Assignment* http://media.pearsoncmg.com/aw/aw_reges_bjp_2/videoPlayer.php?id=c2-2
- CSE 142, *Variables* (5:12–12:48) <https://www.youtube.com/watch?v=0eUm1RFGkWw&start=310>
- CS Homework Bytes, *Variable and Assignment, with Elizabeth* https://www.youtube.com/watch?v=fPqGiexXi_Y

Forum discussion

Lesson 2.02 Declaring and Assigning Variables (TEALS Discourse account required)