

## Lesson 2.07 — Scope & Pseudocode

### Overview

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

- **Identify** the scope of a variable.
- **Identify** common scope errors

Assessments — *Students will...*

- **Complete** a worksheet

Homework — *Students will...*

- **Read** BJP 2.4 “Pseudocode”
- **Complete** self-check questions 31-33 (4th edition 34-36)
- **Check class notes** for completion, adding daily summaries if needed
  - Students may use the book to supplement their notes if needed.

### Materials & Prep

- **Projector and computer**
- **Whiteboard and markers**
- **Classroom copies** of WS 2.7

Since most of today’s lesson follows WS 2.7, 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 to scope	10min
Worksheet activity	15min
Introduction to pseudocode	5min
Programming Project 1	20min

### Procedure

Scope is another challenging topic, so we’ve prepared a student handout that diagrams scope. Encourage students to take supplemental notes in their notebook and on the handout.

**Bell-work and Attendance [5 minutes]**

**Introduction to Scope [10 minutes]**

1. Using WS 2.7, walk students through the sample illustration of **scope** and scope-related error.  
Before you explain that brackets define scope, ask students to tell you why they think the error in the example. They may be able to spot the scope from their reading last night.
2. Have students copy the rules for scope off the worksheet and into their notebooks. Students should head the page “The 4 Commandments of Scope,” and illustrate, circle, highlight, star, or otherwise emphasize these rules so they are easy to find in the notebook throughout the year.
  - Transcribing these notes is not busywork—in the ELL classroom, this slows down the pace and allows

students additional time to process the language and study the concepts conveyed. This is a good strategy for all classes (ELL or not)!

At this point, your class may be raring to get started on the rest of the assignment without your help. If they are, great!

If they are still seeming uncertain about how to apply rules of scope, walk through Exercise 1a and ask the class to explain to you how/why you know that scope begins and ends where it does.

### **Worksheet Activity [15 minutes]**

1. Have students work independently or in pairs on WS 2.7.
  - a. Remind students to refer to their 4 Commandments when searching for errors.
  - b. Play music quietly in the background to encourage focus and discourage chatting.
  - c. If students are struggling with the exercise, allow them to pair up. Encourage students to trade papers and look for errors.
2. If more 25% or more of the class is struggling, return to whole group with the stipulation that students who get it may continue working independently.
3. Review answers as a whole group. Make sure to call the variables boxed in Exercise 1 “local variables,” and ask students to explain what makes them localized.
4. Point out to local variables can be declared in the body of constructors and methods. These variables may only be used within the constructor or method and cannot be declared to be public or private.

### **Introduction to pseudocode [5 minutes]**

Using an example that was popular during the last week of class, show students how to plan a program using **pseudocode**.

### **Programming Project 1 [20 minutes]**

As a class, begin planning a program to answer Programming Project 1.

The challenging task of writing the pseudocode and/or structure diagram (you should always require one, both, or a combination of the two) is the most important part of this exercise, so avoid the temptation of giving students too much guidance.

### **Accommodation and Differentiation**

It is likely that students will not finish the programming project during class. You can reduce student stress by setting the expectation that they submit complete pseudocode by the end of class; not a completed program. You may extend the program for advanced students, and ask that they submit an entire program (complete with class header).

### **Video**

- BJP 2–5, *Drawing Complex Figures with for Loops* [http://media.pearsoncmg.com/aw/aw\\_reges\\_bjp\\_2/vid eoPlayer.php?id=c2-5](http://media.pearsoncmg.com/aw/aw_reges_bjp_2/vid eoPlayer.php?id=c2-5)
- CSE 142, *Managing Complexity: Problem Decomposition and Pseudocode* (1:58–16:17) <https://www.youtube.com/watch?v=xQP1OF9KpbM&start=118>
- CSE 142, *Managing Complexity: Nested Loops* (16:17–37:50) <https://www.youtube.com/watch?v=xQP1OF9KpbM&start=977>
- CSE 142, *Managing Complexity: Class Constant* (Optional) (37:51–49:17) <https://www.youtube.com/watch?v=xQP1OF9KpbM&start=2271>

- CSE 142, *Managing Complexity: Review* (Optional) (0:58–21:04) <https://www.youtube.com/watch?v=otR7yEbRZAs&start=58>

## **Forum discussion**

Lesson 2.07 Scope & Pseudocode (TEALS Discourse account required)