

Lesson Plan

Topic: Variables in Python

Resources

Overview Video	Variables in Python - Teacher Overview
Slides	Variables in Python Lesson Slides
Worksheet	Variables in Python Worksheet
Starter File	inclass.py
Finished Files	inclass.py madlib.py

Time Requirement

45 minutes

Instructional Notes

- This lesson can build on the previous lesson of “Printing in Python”. Students can work in the existing inclass.py they created in the last lesson or they can start with a new and fresh Python file.
- The lesson contains three extensions. The first two extensions work in the inclass.py file. The third extension is more comprehensive and will take additional time.
- Depending on your instructional preference, you can live code with the students to walk through the code on the slides.

Do Now (3 min)

Scavenger Hunt - Students will search and count for items in the classroom. They will later use these items and values to create variables.

Lecture, Discussion, & Demo (20 min)

Discuss the different values everyone found for the scavenger hunt. End with pointing out that these items can be represented in variables. (2 minutes)

Lecture begins with comparing variables to empty boxes that should be named and hold an item. An overview of proper naming conventions follows. (7 minutes)

Next, a classroom discussion takes place investigating proper naming conventions and looking at examples. If there is additional classroom time needed, the discussion on legal and descriptive variables can be expanded to look at additional names. (3 minutes)

Have the students turn and talk about the variable names from the scavenger hunt (3 minutes).

Finally, a demonstration shows how to assign values in PyCharm, and has a quick lesson on using the correct data type. (5 minutes)

Activity (15 min)

Students will declare and initialize variables for the classroom items on their worksheet. Next, students move into PyCharm and code these. Finally, students attempt to use these variables in print statements. The trickiest part will be using `str(variable_name)` for integer and float variables. PyCharm will help them solve this problem.

If time allows, students can begin working on a Madlib program. They will need to select a nursery rhyme or sentence, create variables, and substitute those variables in to make a Madlib. Students will complete this activity with user input in future lessons. This will provide them with an opportunity to work with more variables now.

Wrap Up (10 min)

Review the `inclass.py`. Screenshots are included or you can code with them. If anyone started on extension 3, you could have them share the silliest Madlib they coded (if time allows). We will revisit the Madlib in another lesson when we begin working with user input. Take a moment to celebrate their accomplishments while wrapping up the class.

Lesson Goals

- Grasp the Purpose of Variables - Understand why variables are crucial in programming, how they store and manage data, and their role in enhancing code flexibility in Python.
 - Assessed in Wrap-Up Question
- Master Variable Declaration and Initialization - Become proficient in declaring and initializing variables in Python. Learn to choose suitable names, understand data types, and assign values accurately for effective coding.
 - Assessed in Activity Worksheet table
- Independently Create Variables in Python - Create variables in Python confidently. Develop skills in syntax, data types, and best practices to apply variable creation in real-world programming scenarios.
 - Assessed in `inclass.py` and/or `madlib.py`