Model 1 Variables and Assignment

In programming, an ***assignment statement*** saves a value to a ***variable***. The variable "is set to'' the value after the ***= operator***. Choosing good variable names is considered good programming style and will make your programs easier to read.

Do not type anything yet! Read the questions first!

Python code	Shell Output
`data = 12`	
`data`	
`Data`	
`Data = 34`	
`data`	
`Data`	
`my data = 56`	
`my_data = 78`	
`3data = "hello"`	
`data3 = "world"`	
`hot = 273 + 100`	
`273 + 100 = hot`	
`hot`	
`Hot + 100`	
`hot - 100`	

Questions (15 min)

1. Based on the information and Python code in Model 3, give an example representing each of the following:
    a. an assignment statement

    b. the variable being assigned

    c. the assignment operator

    d. the value of the variable after the assignment

2. Type each line of code in a Python Shell and write the corresponding output in the space above. If an error occurs, raise your hand. Place an asterisk (*) next to any output for which you were surprised.

3. Circle each *successful* assignment statement in Model 3. How many are there?

4. What do you see after a successful assignment statement?

5. How can you check the value of this variable?

6. For each assignment statement that executed without an error, write the corresponding variable name.

7. Based on the Model 3 output, indicate whether each statement below is true or false.
    a. Variable names in Python can start with a number.

    b. Variable names in Python must start with a lower-case letter.

    c. Variable names in Python may not include spaces.

    d. Variable names in Python are case-sensitive.

8. Each of the following assignment statements has an error. Write a correct line of Python code that corrects the assignment statement. Double-check your code using a computer.
    a. 3 + 4 = answer

    b. oh well = 3 + 4

    c. 2x = 7

9. Predict the value of the variable ***hot*** after executing all lines of code in Model 3. Then test your prediction on a computer, and explain the result.

10. Write a line of Python code to assign the current value of ***hot*** to the variable ***cold***. Show output that confirms that you have done this correctly, and explain the code.