Learning Activity (2 of 2): Input and Output

Overview

In this lesson, you will build on the basic input/output ideas from your programs in the last lesson, and learn to store more information and display it using better formatting. While not all programs use the console input and output approach that we have taken thus far (where people type the answers and see the results in text), almost all programs make use of many string variables and combine and format them in one form or another.

Preparation Material

Variables

One of the things you learned in the previous lesson was how to store data in a variable. A variable is like a name that we attach to that data, so that later we can refer to it when we need it.

In your programs this week, you are going to start to see many variables used simultaneously in the same program. This is very common, and doesn't cause any problem for the computer. As long as you are consistent and always use the same name for the same data, you won't have any problems.

Because we will have to keep track of more and more variables, it becomes increasingly important to choose good names for them. For example, while you may remember what **`x`** means now, in a few weeks, months, or years, you might forget. On the other hand a variable name like **`color`** or even **`favorite_color`** is much more descriptive and will help you and others better understand your code.

Hint from Instructor:

You can't use spaces in variables, so if you want a long variable name with multiple words, the Python style is to use underscore characters between the words, such as: **`this_is_a_very_long_variable_name`** .

Other style guides, especially for other languages may use different approaches, such as "camel case" where each subsequent word is capitalized, such as **`thisIsAnotherLongVariable`**, but in this class, you should stick with the underscore, or "snake case," approach.

Comments

Comments are a way for you to include notes in your code. They don't affect the program in any way, but they make it easier for someone to understand the code when they look at it later. To add a comment to your code, include the **#** sign before the text that you want to be a comment.

Please watch the following videos for more information about comments:

- Direct link: Comments (3 mins)

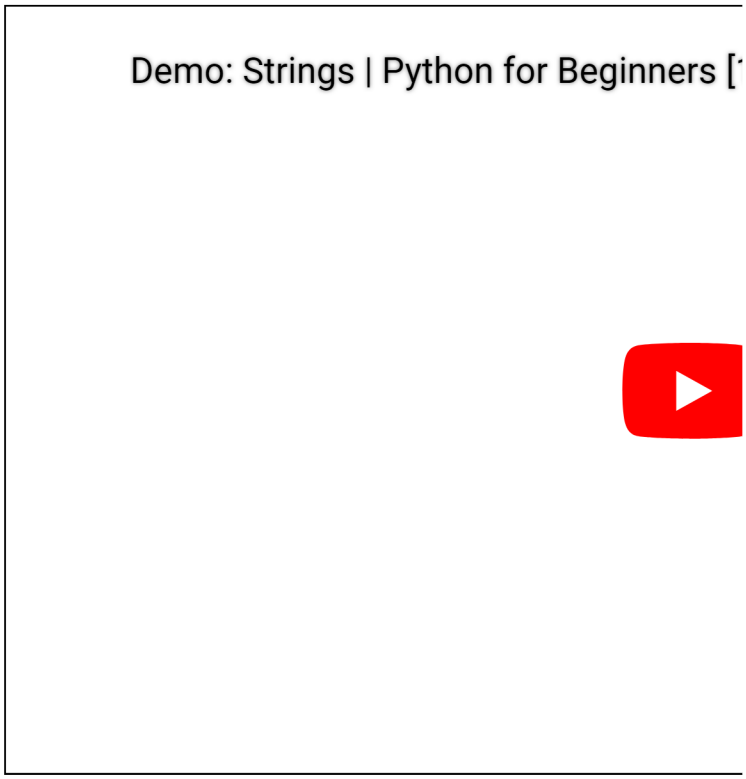
- Direct link: Demo: Comments (3 mins)

Combining and Formatting Strings

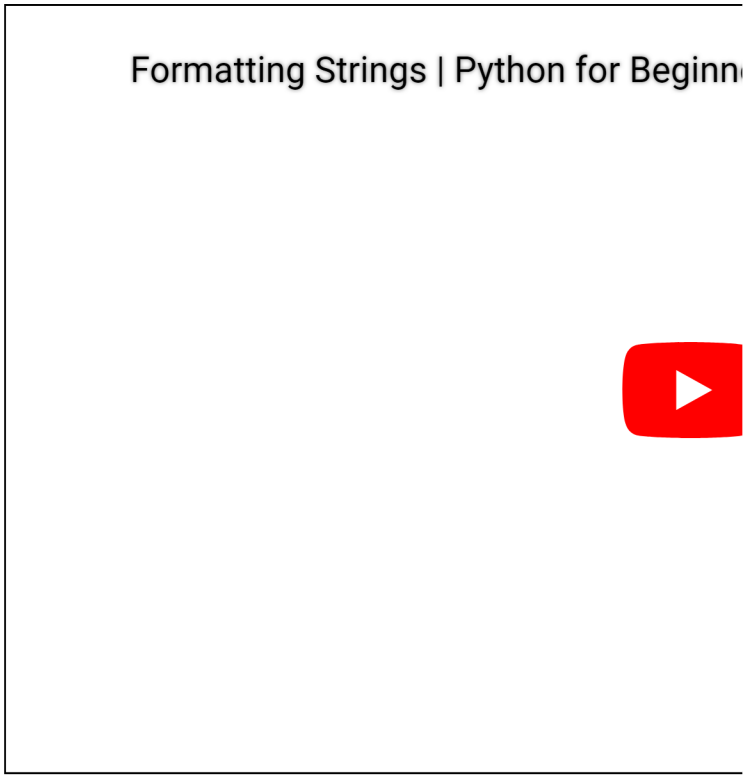
As you learned in the previous lesson, "Strings" are variables that are a sequence of characters (for example, letters, numbers, spaces, symbols, etc.). Please watch the following videos that demonstrate how to combine, format, and display strings in different ways:

- Direct link: String Concepts (6 mins)

- Direct link: Demo: Strings (4 mins)



- Direct link: Formatting Strings (4 mins)



- Direct link: Demo: Formatting Strings (4 mins)

As shown in these videos, some helpful string functions available in Python are:

Code	Result
`words = "the GLORY of GOD is intelligence"`	`the GLORY of GOD is intelligence`
`words.capitalize()`	`The glory of god is intelligence`
`words.title()`	`The Glory Of God Is Intelligence`
`words.upper()`	`THE GLORY OF GOD IS INTELLIGENCE`
`words.lower()`	`the glory of god is intelligence`

Code	Result
`words.count("g")`	1
`words.lower().count("g")`	3

Notice that `words.count("g")` resulted in a 1, because it did not count the two cases of capital "G" in the sentence. On the other hand, `words.lower().count("g")` resulted in a 3, because it first converted everything to lowercase, and then counted them, so when it counted the g's, the capital G's in that sentence were first converted to lowercase g's, and then they were counted.

> **Hint from Instructor**
>
> The examples in this table all say "`words.`" but the name "words" is not special. In this case, it assumes that the string is stored in a variable named `words` but it could have been any variable name such as: `first_name.title()` or `book_title.capitalize()`.

Activity Instructions

Overview

An iconic line from the James Bond movies is that he would introduce himself as "Bond, James Bond." For this assignment you will write a program that asks for your name and repeats it back in this way.

Instructions

Prompt the user for their first name. Then, prompt them for their last name. Display the text back all on one line saying, "Your name is last-name, first-name, last-name" as shown:

```
What is your first name? Scott
What is your last name? Burton

Your name is Burton, Scott Burton.
```

```
What is your first name? Brigham
What is your last name? Young

Your name is Young, Brigham Young.
```

Make sure to be precise! You should have the spacing, comma, and period appear exactly as shown in the examples.

Adjust the Capitalization

Now that the program is displaying the strings back with the correct spacing, improve your program by making it display the words using the **`.title()`** function on each variable so that it capitalizes only the first letter and all the other letters are lowercase.

Test that your program works by trying some words that are capitalized and some that are lower case. The output should be the same. For example:

```
What is your first name? Brigham
What is your last name? Young

Your name is Young, Brigham Young.
```

```
What is your first name? brigham
What is your last name? YOUNG

Your name is Young, Brigham Young.
```

```
What is your first name? brIGham
What is your last name? YOUng

Your name is Young, Brigham Young.
```

Instructor Code

When your program is finished, please view the instructor's version of this program to compare your approach to that one.

You should work to complete this activity first, without looking at the instructor code. However, if you have worked on it for a significant period of time and are still having trouble, you may feel free to use the instructor code to help you finish your program.

- Sample solution

Submission

You have now completed all of the learning activities for the week!

Make sure to:

- Return to Canvas and submit the associated quiz.

Up Next

- Check Your Understanding

Other Links:

- Return to: Week Overview | Course Home