Overview

So far we have printed "Hello, World!" and a few other things to our terminal using *literals*. A literal is any time you write the literal value in your program. We literally wrote "hello world". One of the most useful tools for programmers is using varaibles instead.

Set Up

Create a new python file and call it `stringvariable.py`.

Important Info

Variables in computer science work differently from variables in math. - In math, typically a variable represents an unknown value. - In computer science, a variable is used to store data we want to keep track of.

Variables in computer science are useful because we can: - Change the value of a variable - Use the value of the variable when out code executes

When we create a variable, we give it a name and a value. This process is called **declaring** a variable. A variable cannot be used until it has been declared. At the bottom of this lesson is some best practices for naming variables.

Different variables can store different types of data, which we will learn more about later. For now, we will focus on String variables.

A String variable has a value that contains a string of characters. It can be an empty pair of quotes, a single letter, a word, a phrase, a sentence, or even an entire novel!

Copy

Copy the following code. In this code, we create a variable called `greeting` and set the value to `"Hello, World!"`. When creating a variable, you can choose any name you want. Below we will learn about common naming conventions in python.

```
greeting = "Hello, World!"
print(greeting)
```

What happens when you run the code?

Notice that our `=` sign acts as a way to set the value of our variable. Instead of a mathematical expression that is asserting the equality of 2 values, the `=` sign in code allows you to set the value of a variable.

Change

Change the greeting variable so that it greets you by name rather than `"Hello, World!"`.

Run the program again to verify that it works.

Challenge

Once again, you will print multiple messages. Create 3 different variables to store these 3 messages, and print them to the terminal.

`Hello, World!`

`Hello,` followed by your name

`I love computer science!`

Best Practices

In most code editors, there is some kind of "intellisense" that will help you autocomplete variable names that have already been created in your program.

Because of this, any variable can have any name as long as you need, because you should only really be typing the whole thing once.

These are some best practices that you will be expected to maintain in your code throughout the semester.

1. Use clear and descriptive names for your variables.

   - For example `words` is a worse variable name than `greeting` because `greeting` describes what the text will be.

2. Use `snake_case` for variables in Python.

   - Snake case is the special naming convention of using all lower case letters and separating words with an `_` underscore.
   - All variables in python should use `snake_case`

3. Avoid single letter variable names.

   - `g` is shorter to type than `greeting`, but you only need to type it once the first time you create the variable.

4. Variable names cannot start with a number.

5. If you are creating a value that will never change, you can identify it as a constant value by naming it with all caps.

   - `CONSTANT_VALUE` would be a variable that isn't supposed to change.