

Lesson 2

Strings

We have seen string variables in the previous lesson. Now we will learn a bit more about them and how they can be used in our code. Strings are basically a collection of characters (letters) that are organized one after the other, in the order the user has written them in. In the string “Hello World!”, the first character is “H” and the last character is “!” etc.

Index Access

In Python, we can access a particular letter within a string variable by using square brackets (“[]”) that contain the index (number) of the character you wish to access. Try the below code out yourself!

```
myString = "The Academy of Code"
print(myString[0]) # prints "T"
print(myString[3]) # prints " " (space)
print(myString[7]) # prints "d"
```

Hint: Notice how the first character is accessed by using an index of zero (**not one!**)

Tasks:

1. Print the **ninth** element of the string.
2. Print the **eighteenth** element of the string.

Chatbot Project

Write code that will create a chatbot that will ask the user questions and respond to the answers that are given. Most of the sentences should be hardcoded (written by the coder, not inputted by the user) and the user should input the missing details in each sentence. For example:

```
myName = input("Hello there! What's your name?") # chatbot greets user and asks their name
print("My name is " + myName + ". What's your name?") # tell chatbot your name and ask his/her name
```

The user must first input their own name before you can print the line. Some of the points you and the chatbot should discuss might include (feel free to add more or do your own!):

- Name (give the chatbot a name also!)
- Age
- Where you're from
- Where you go to school
- How many in your family?
- How many pets do you have?
- How long have you been coding?
- What do you want to be when you're older?
- Etc.

Tasks:

1. Write the code that will print the conversation between you and the chatbot.
2. Add in delays between responses to give the chatbot time to respond. See the example below of how to do this.

```
import time # import time library to use it in the project
myName = input("Hello there! What's your name?") # chatbot greets user and asks their name
print("My name is " + myName + ". What's your name?") # tell chatbot your name and ask his/her name
time.sleep(2) # 2 second delay for chatbot to respond
```

Hint: Put a two second delay in after every `print()`.

3. Run the program.
4. Save the program.

Extra Tasks

String Slicing

We can also access a group of characters within a string (substring) using ":" and the index of the character we wish to start/end. Check out the example below and try it out.

```
myString = "I am a String"

print(myString[3:]) # prints "m a String" (from index number 3)

print(myString[:9]) # prints "I am a St" (up to index number 9)

print(myString[4:11]) # prints "a Stri" (between index number 4 and 11)

print(myString[:]) # prints "I am a String" (all of string)
```

Tasks:

1. Print the string **from** index number 5.
2. Print the string **up to** index number 10.
3. Print the string **between** index number 1 and 7.

Index Search

At times we may also need to work backwards (find the index of a certain character/s in a String). Try out the example below!

```
myString = "I love writing Python code"

index = myString.find("o") # stores the index of the first occurrence of "o"

print(index) # prints "3"

index = myString.find("y") # stores the index of the first occurrence of "y"

print(index) # prints "16"

index = myString.find("a") # stores the index of the first occurrence of "a"

print(index) # prints "-1" (character is not found in String)

index = myString.find("Python") # stores the index of the first occurrence of "Python"

print(index) # prints "15"
```

Hint: Notice how the "index" variable can be reused to store another index value.

Tasks:

1. Print the index of "w" in the string.
2. Print the index of "love" in the string.

Concatenation

The final topic we will cover in this lesson is concatenation, which is adding multiple string together to make a longer string. See examples below!

```
myString = "I am " + "a really amazing " + "coder" # add three separate strings and store in variable
print(myString) # prints "I am a really amazing coder"

myString += " , super amazing!" # add additional string to myString variable
print(myString) # prints "I am a really amazing coder, super amazing"
```

Tasks:

1. Add another sentence on to the end of the string.
2. Print out the whole string
3. Print out the string, leaving out the **first and last** two letters.
4. Print out the length of the whole string.

Additional Information

String Length

Python has a simple way of counting the amount of characters in a string (we're too lazy to count them ourselves...). Check out the example below.

```
myString = "How long is this String?"  
  
stringLength = len(myString) # store the length (int) of myString into the stringLength variable  
  
print(stringLength) # prints "24"
```

We can use this method of finding the string length in some interesting ways!

```
myString = "Learn-Code-Grow"  
  
stringLength = len(myString) # store the length (int) of myString into the stringLength variable  
  
print(myString[2:stringLength]) # prints "arn-Code-Grow"  
  
print(myString[2:stringLength-2]) # prints "arn-Code-Gr"  
  
print(myString[stringLength-9:stringLength-3]) # prints "Code-G"
```