Chapter 5 - Lists

Lists

List of Numbers

Learning Objectives - List of Numbers

- **Identify some functions that can be used with lists of numbers**
- **Describe what happens when these functions are used with strings**

Sum

The Sum Function

The next few lessons will be focused on functions that work with lists of numbers. Using a function or a method on a list is similar in that an action performed on the list. The syntax for calling a function is different than calling a method.

Function	Method
sum(my_list) Action - The function is the action performed on the list. List - A list is associated with the function. Parentheses - The function has parentheses. The list is passed as a parameter of the function.	my_list.pop() Action - The method is the action performed on the list. List - A list is associated with the method. Parentheses - The method has parentheses. Sometimes parameters are needed. Other times there are no parameters.

List Functions and Methods

The `sum` function is a built-in function that calculates the sum of a list.

```
my_list = [1, 2, 3, 4, 5]
total = sum(my_list)
print(total)
```

challenge

What happens if you:

- Change the value of `my_list` to `[-45, 21, 6, 0.3]`?
- Change the value of `my_list` to `[]`?

The Sum Function & Strings

You can concatenate strings with the `+` operator, which the same operator used to add numbers. This may seem like it is possible to use the `sum` function with a list of strings. However, this will cause an error message.

```
my_list = ["abc", "def", "ghi"]
print(sum(my_list))
```

challenge

What happens if you:

- Change the value of `my_list` to `[1, 2, 3, "red"]`?

Min

The Min Function

The `min` function is another built-in function that returns the minimum value of a list.

```
my_list = [45, 12, 9, 1]
smallest = min(my_list)
print(smallest)
```

> challenge
>
> ## What happens if you:
>
> - Change the value of `my_list` to `[0, 1, 0.5, -0.2, -1]`?
> - Change the value of `my_list` to `[]`?

The Min Function & Strings

Unlike `sum`, the `min` function can work with strings. Python does this by selecting the element that comes first alphabetically.

```
my_list = ["apple", "boy", "cat", "aaron"]
smallest = min(my_list)
print(smallest)
```

challenge

What happens if you:

- Change the value of `my_list` to `["apple", "boy", "123", "cat"]`?
- Change the value of `my_list` to `["apple", "boy", 123, "cat"]`?

Max

The Max Function

The `max` function returns the largest element in a list.

```
my_list = [50, 11, 0, 72, 102]
largest = max(my_list)
print(largest)
```

challenge

What happens if you:

- Change the value of `my_list` to `[5 / 2, 5 // 2, 5 % 2]`?
- Change the value of `my_list` to `[1, 2, 3, 4, "5"]`?
- Change the value of `my_list` to `[]`?

The Max Function & Strings

If all of the elements of a list are strings, then the `max` function can be used to determine the element that would appear last in alphabetical order.

```
my_list = ["house", "cat", "zebra", "dog", "bike"]
last = max(my_list)
print(last)
```

challenge

What happens if you:

- Change the value of `my_list` to `["zzz", "zz", "zzzz", "z"]`?
- Change the value of `my_list` to `["one", "two", "three", "four"]`?
- Change the value of `my_list` to `["100", "27", "3000", "abc"]`?