Wrap-up; Reference guide: Python concepts from module 2

Great work on making it this far in the Python course!
You've put in a lot of work and effort towards learning more about how you can use Python effectively and efficiently.
Let's quickly recap the concepts you learned throughout the videos.
First, we started by understanding the role of functions in Python.
They can save you a lot of time!
You learned how to incorporate built-in functions and how to develop your own function to meet your needs.
We then shifted our focus towards modules and libraries, which gave us access to a lot more functions than those built into Python.
Lastly, we moved to learning about code readability and best practices to write clean, understandable code.
With these understandings, you're ready to learn how powerful Python can really be for task automation and how it can help you going forward as a security analyst.
Thank you for taking the time to go through this course with me.
I'll meet you in the next videos!

Reference guide: Python concepts from module 2

Google Cybersecurity Certificate

User-defined functions

The following keywords are used when creating user-defined functions.

`def`

Placed before a function name to define a function

`def greet_employee():`
Defines the `greet_employee()` function

`def calculate_fails(total_attempts, failed_attempts):`
Defines the `calculate_fails()` function, which includes the two parameters of `total_attempts` and `failed_attempts`

`return`

Used to return information from a function; when Python encounters this keyword, it exits the function after returning the information

```
def calculate_fails(total_attempts, failed_attempts):
    fail_percentage = failed_attempts / total_attempts
    return fail_percentage
```

Returns the value of the `fail_percentage` variable from the `calculate_fails()` function

Built-in functions

The following built-in functions are commonly used in Python.

`max()`

Returns the largest numeric input passed into it

```
print(max(10, 15, 5))
```

Returns `15` and outputs this value to the screen

`min()`

Returns the smallest numeric input passed into it

```
print(min(10, 15, 5))
```

Returns `5` and outputs this value to the screen

`sorted()`

Sorts the components of a list (or other iterable)

```
print(sorted([10, 15, 5]))
```

Sorts the elements of the list from smallest to largest and outputs the sorted list of `[5, 10, 15]` to the screen

```
print(sorted(["bmoreno", "tshah", "elarson"]))
```

Sorts the elements in the list in alphabetical order and outputs the sorted list of `["bmoreno", "elarson", "tshah"]` to the screen

Importing modules and libraries

The following keyword is used to import a module from the Python Standard Library or to import an external library that has already been installed.

`import`

Searches for a module or library in a system and adds it to the local Python environment

```
import statistics
```

Imports the `statistics` module and all of its functions from the Python Standard Library

```
from statistics import mean
```

Imports the `mean()` function of the `statistics` module from the Python Standard Library

```
from statistics import mean, median
```

Imports the `mean()` and `median()` functions of the `statistics` module from the Python Standard Library

Comments

The following syntax is used to create a comment. (A comment is a note programmers make about the intention behind their code.)

#

Starts a line that contains a Python comment

```
# Print approved usernames
```

Contains a comment that indicates the purpose of the code that follows it is to print approved usernames

""" (documentation strings)

Starts and ends a multi-line string that is often used as a Python comment; multi-line comments are used when you need more than 79 characters in a single comment

```
"""
The estimate_attempts() function takes in a monthly
    login attempt total and a number of months and
    returns their product.
"""
```

Contains a multi-line comment that indicates the purpose of the `estimate_attempts()` function

---

Revision #2
Created 2023-12-23 04:05:41 UTC by naruzkurai
Updated 2023-12-27 11:35:24 UTC by naruzkurai