CAREERFOUNDRY

Exercise 1.3: Functions and Other Operations in Python

Learning Goals

- Implement conditional statements in Python to determine program flow
- Use loops to reduce time and effort in Python programming
- Write functions to organize Python code

Reflection Questions

1. In this Exercise, you learned how to use **if-elif-else** statements to run different tasks based on conditions that you define. Now practice that skill by writing a script for a simple travel app using an **if-elif-else** statement for the following situation:

    - The script should ask the user where they want to travel.
    - The user's input should be checked for 3 different travel destinations that you define.
    - If the user's input is one of those 3 destinations, the following statement should be printed: "Enjoy your stay in _____!"
    - If the user's input is something other than the defined destinations, the following statement should be printed: "Oops, that destination is not currently available."

Write your script here. *(Hint: remember what you learned about indents!)*

```
destination = input('Choose to travel to Iowa, New York City, or Detroit: ')
if destination == 'Iowa' or 'New York City' or 'Detroit':
    print('Enjoy your stay in ' + destination)
else:
    print("I'm sorry, that's not a valid destination")
```

2. Imagine you're at a job interview for a Python developer role. The interviewer says "Explain logical operators in Python". Draft how you would respond.

The three logical operators for Python are 'or', 'not', and 'and'. These are all booleans, meaning they will return either a true or false response. The 'or' operator returns true if at least one of the statements is

true. The ‘and’ operator returns true if all of the statements are true. Otherwise, these operators will return as false. The ‘not’ operator will return the opposite of what the result would normally be.

3. What are functions in Python? When and why are they useful?

Functions in Python are sets of instructions that process or manipulate your code in order to achieve certain things. They are useful as they help to cut down on repetitive code and helps to make your code easier to read.

4. In the section for Exercise 1 in this Learning Journal, you were asked in question 3 to set some goals for yourself while you complete this course. In preparation for your next mentor call, make some notes on how you’ve progressed towards your goals so far.

So far, I have learned that one of the main reasons why Python is widely used in the scientific and research communities is because of its ease of use and simple syntax which makes it easy to adapt for people who do not have an engineering background. It is also more suited for quick prototyping.