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!)*

```python
defined_destinations = [ 'destination_1', 'destination_2', 'destination_3' ]
destination = input("Enter your desired destination: ")

if destination == 'destination_1':
   print("Enjoy your stay in 'destination_1'!)
   elif destination == 'destination_2':
      print("Enjoy your stay in 'destination_2'!)
   elif destination == 'destination_3':
      print("Enjoy your stay in 'destination_3'!)
   else:
      print("Oops, that destination is not currently availabel.")
```

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.

and, or, and not are the 3 logical operators used to connect 2 or more expressiond. And returns True if both or all statements are true. Or returns True if any of the statements are true. Not is used to reverse the result, so it returns False if the statement is true.

3. What are functions in Python? When and why are they useful?
Functions in Python are sets of instructions that process or manipulate code to achieve desired results. They are useful in keeping your code clean by helping you not repeat larger amounts of code, and it makes it easier to implement code that will be repeated frequently.

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.
I wouldl like to have a comfortable grasp of what Python is and how to use it. – I do feel like I am getting there.
I would like to be comfortable reading the python documentation – I have started, I read the tutorial.
I would like to be able to articulate the difference and benefits to using Python over JavaScript – I am getting there as well.