# 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('Where do you want to travel to? ')

if destination == 'Italy':
    print('Enjoy your stay in Italy!')
elif destination == 'Ireland:
    print('Enjoy your stay in Ireland!')
elif destination == 'Iceland:
    print('Enjoy your stay in Iceland!')
else:
    print('Oops, that destination is not currently available.')
```

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 logical operators in Python consist of **and**, **or**, and **not** and are used in conditional statements. When there is more than one condition that is being checked, the **and** / **or** operators are helpful if you need either both conditions met or just one of them. The **not** operator on the other hand is useful for when you need to check if a condition is not met, it does this by taking the True/False result of the expression and flipping it.

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

   Functions are a set of instructions that be called on repeatedly, so it makes reusable code within your Python script. This is useful when you have a task that you'll need to run multiple times, as it saves you from having to rewrite the code every time you need that task run.

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 feel like I've gotten a pretty good understanding on the syntax we've covered so far with Python. Just like with JavaScript, it's straightforward as a syntax and that makes it a lot easier than I expected to get a grasp on it. I feel as though being comfortable with JavaScript before moving onto Python has been incredibly helpful in my understanding. There are obvious differences, but it's made a good jumping off point to learn Python.