

## 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!*)

```
locations = []

def travelplans():
    locations = input("Enter three locations you want to visit: ").split(", ")
    if len(locations) == "london" or "paris" or "rome":
        print("Enjoy your stay in [locations]")
    else:
        print("Oops, that destination is not currently available")

travelplans()
```

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.

- a. Python's three logical operators are "And", "Or" and "Not". They're used to quantify values and check parameters when functions fire to make sure your function is returning the correct response. "And" is looking for a combination of factors, "Or" is looking for one factor or another" and "Not" is looking for the absence of a factor. The easiest way to make sure you're coding "And", "Or" and "Not" is to use a truth board to map out what it is you're attempting to build.
3. What are functions in Python? When and why are they useful?
  - a. Functions are blocks of code that are run when called. Functions return information based on parameters that are passed into the function. They're used to complete both simple and complex processes and are useful because they can verify the information you're receiving is correct and accurate, and actually make for cleaner and easier-to-read code.
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.
  - a. I've felt like I've done a great job of using the python tools to make my code work. I'm finding that this is a very fun language to work with and that's not something I'd really had yet. The biggest one is that I've been able to complete the three lessons so far with minimal outside assistance and that's a major goal for myself. Being more confident in my coding and knowledge. It's a great feeling.