Exercise 1.2: Data Types in Python

Learning Goals

- Explain variables and data types in Python
- Summarize the use of objects in Python
- Create a data structure for your Recipe app

Reflection Questions

1. Imagine you're having a conversation with a future colleague about whether to use the iPython Shell instead of Python's default shell. What reasons would you give to explain the benefits of using the iPython Shell over the default one?

The iPython shell is the choice for a handful of reasons, all stemming from it's practicality. Not only is it easier to read because of syntax highlighting and contrasting fonts/colors but it also lets you test out small chunks of code quickly and easily. Unlike the python shell, each command is executed immediately after typing it in, saving time by not requiring the drafting and executing of separate script files.

2. Python has a host of different data types that allow you to store and organize information. List 4 examples of data types that Python recognizes, briefly define them, and indicate whether they are scalar or non-scalar.

Data type	Definition	Scalar or Non-Scalar?
Integer (int)	Whole numbers with no decimal points	Scalar
Float	Numbers that include decimal points	Scalar
String (str)	String of characters (can be numbers of letters)	Non-Scalar
Boolean (bool)	True/False Statement	Scalar

3. A frequent question at job interviews for Python developers is: what is the difference between lists and tuples in Python? Write down how you would respond.

While they are similar, there are distinct differences between the two, namely the mutable/immutable nature of each. Tuples are linear arrays that allow you to store multiple values of any type, however, they are immutable. Once a tuple has been defined they cannot be altered. Whereas lists in comparison are an ordered sequence in Python that is mutable. This means that, unlike tuples, any of the internal elements of a list can be modified or deleted.

4. In the task for this Exercise, you decided what you thought was the most suitable data structure for storing all the information for a recipe. Now, imagine you're creating a language-learning app that helps users memorize vocabulary through flashcards. Users can input vocabulary words, definitions, and their category (noun, verb, etc.) into the flashcards. They can then quiz themselves by flipping through the flashcards. Think about the necessary data types and what would be the most suitable data structure for this language-learning app. Between tuples, lists, and dictionaries, which would you choose? Think about their respective advantages and limitations, and where flexibility might be useful if you were to continue developing the language-learning app beyond vocabulary memorization.

For this app I would structure the data within a dictionary. The built in key:value structure would make it easy to input all of the necessary data for each flashcard. Additionally, this structure would allow for easy access of the associated data within each flashcard.

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!)*