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?

I would recommend iPython instead of the default because it uses color, indentation, numbered inputs, and syntax highlighting to make it more user-friendly to the end user.

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?
Int	Represents both positive and negative integers	Scalar
Float	Represents both positive and negative decimals	Scalar
Lists	A mutable ordered sequence in Python	Non-Scalar
Strings	An immutable array of characters in Python	Non-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.

A list is mutable so it can be changed, while a tuple is immutable and cannot be changed.

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.

I would use a dictionary for this app for each vocabulary word and they would all follow the same format. The format would be as follows: Vocabulary Word = {‘Definition’, ‘Lexical Class’}. I would use a dictionary for each word because it would allow me to categorize the words into key value pairs. This is the best way to display both its definition and its lexical class. I would then put them into a list so that end users would be able to add and remove words, and also be able to reorder the list alphabetically if they so chose.