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?

IPython is just highly more user-friendly. The color coding alone would be enough of a benefit for me as you are able to distinguish reading the code unlike the python shell where everything looks the same. Other options such as auto indexing and prompts make IPython superior to Python default shell.

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	A whole number (non-fraction)	Scalar
float	A decimal number	Scalar
string	A string of characters such as numbers or letter	Scalar
tuples	Linear arrays that can store multiple values of any type	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.

Lists are able to have modifications made to them unlike tuples. Lists are best whenever changes or additions will be needed.

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 a learning app I would stick with using dictionaries. Since it has key-value pairing it would be best to use for this app since there would be multiple keys for each word and would be easy to index and modify if needed.