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?

Python's default shell is fully functional, however it has a few things that make it not as user-friendly. It is hard to read, it requires the user to indent manually. Ipython is very easy to read with different colors and contrasting fonts. It also indents automatically. I python also allows the user to test out small chunks of code easily.

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	An integer represents both negative and non-negative numbers up to infinity	Scalar
bool	A boolean stores either True or False. These data types are good for storing the output of a condition that is checked.	Scalar
tuple	Tuples are linear arrays that can store multiple values of any type.	Non-Scalar

List	A list is a type of ordered sequence that is mutable. This means you can change or delete anything that is in the list. You can also rearrange them.	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.

Tuples are arrays that can store multiple values of any time. This allows you to save several values into one variable. Lists are similar to tuples, however they are able to be changed, deleted, and rearranged. Even though lists have this perk, it is also advantageous to use tuples because they are able to read and access large amounts of data at once.

1. 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 the dictionary data structure to format this app. This would allow me to use the data as a key:value pair and this would help me organize all of it.
I would then create a list for all of the words for the user to add or remove words.

Word = {'Definition': 'words-definition', 'Categories': 'words category (noun, verb, etc.)'}

vocab_words = {word1, word2, word3}