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

- 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 has syntax highlighting feature that provides clear understanding with usage of colors and fonts.
  - The iPython shell automatically indents lines of codes when needed.
  - iPython allows developers to test small body of code easily.
  - Each command is executed as soon as typed. And expected response is printed.

- 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.

  int: Used for whole numbers (e.g., 42, -10, 0)
  float: Used for decimal numbers (e.g., 3.14, -0.5, 2.0)
  str: Used for text data (e.g., "hello", 'world', "123")
  bool: Used for Boolean values, which can only be True or False

  Eachone of these data types are scalar since they hold one single value at a time.

Data type Definition Scalar or Non-Scalar?

- 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 linear arrays. Series of values can be stored with tuples with their given meaningful names to them. Tuples can contain objects, numbers, strings and more. They are immutable.

A list in Python is an ordered, mutable sequence that allows for changes, additions, and removals of elements. Unlike tuples, which are immutable and cannot be altered after creation.

- 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.

Dictionary would be suitable for such a project because of the key/value structure of this data type. Here is an example:

```
flashcards = {
    'apple': {'definition': 'A fruit that is typically red, green, or yellow', 'category': 'noun'},
    'run': {'definition': 'To move at a speed faster than a walk', 'category': 'verb'},
    'beautiful': {'definition': 'Pleasing the senses or mind aesthetically', 'category': 'adjective'}
}
```