

Learning Goals

Explain variables and data types in Python

- Tuple: arrays that can store values of any type
- List: an array of data types that can be inserted into or modified
- Dictionary: stores a list of values in a key pair with a name and value
- String: string of letters and/or numbers
- Int: whole numbers
- Float: decimal numbers
- Boolean: true or false value

Summarize the use of objects in Python

- There are scalar and non scalar objects
- Scalar objects can only carry singular values and not be modified
- Scalar objects are integers, floats, booleans, and nonetype/null
- Non scalar objects can hold multiple values and can be modified or rearranged
- Non scalar objects are strings, lists, tuples, and dictionaries

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 an interactive realtime environment and autocompletion, as well as more detailed error messages and is a good tool to use for testing.

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? |
|-----------|---------------------|-----------------------|
| Boolean | True or false value | Scalar |

| | | |
|------------|---------------------------|------------|
| Dictionary | Key and value pairs | Non scalar |
| String | String of letters/numbers | Non scalar |
| Int | Whole number | 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.

A list allows for easy modifications like rearranging the order or adding/removing values. Tuples can have a new value added but it must be defined as a single element tuple and then merged into the existing one where the new value is added to the end. Tuples are harder and more strict to modify than lists but have faster read time.

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.

The entire structure of the flashcards would be a list. Each word would be a list containing a dictionary. The key and value of the vocabulary word and its definition would both be strings. The second key would be a string with a name like "Tags", and the value would be a list containing categories (ex. 'verb', 'casual', 'easy') to better describe the word.