**Study Guide: Dictionary Methods**

# Knowledge

Python dictionaries are used to organize elements into collections. Dictionaries include one or more keys, with one or more values associated with each key.

### **Syntax**

my\_dictionary = {keyA:value1,value2, keyB:value3,value4}

## Operations

* **len(dictionary)** - Returns the number of items in a dictionary.
* **for key, in dictionary** - Iterates over each key in a dictionary.
* **for key, value in dictionary.items()** - Iterates over each key,value pair in a dictionary.
* **if key in dictionary** - Checks whether a key is in a dictionary.
* **dictionary[key]** - Accesses a value using the associated key from a dictionary.
* **dictionary[key] = value** - Sets a value associated with a key.
* **del dictionary[key]** - Removes a value using the associated key from a dictionary.

## Methods

* **dictionary.get(key, default)** - Returns the value corresponding to a key, or the default value if the specified key is not present.
* **dictionary.keys()** - Returns a sequence containing the keys in a dictionary.
* **dictionary.values()** - Returns a sequence containing the values in a dictionary.
* **dictionary[key].append(value)** - Appends a new value for an existing key.
* **dictionary.update(other\_dictionary)** - Updates a dictionary with the items from another dictionary. Existing entries are updated; new entries are added.
* **dictionary.clear()** - Deletes all items from a dictionary.
* **dictionary.copy()** - Makes a copy of a dictionary.

# Dictionaries versus Lists

Dictionaries are similar to lists, but there are a few differences:

## Both dictionaries and lists:

* are used to organize elements into collections;
* are used to initialize a new dictionary or list, use empty brackets;
* can iterate through the items or elements in the collection; and
* can use a variety of methods and operations to create and change the collections, like removing and inserting items or elements.

## Dictionaries only:

* are unordered sets;
* have keys that can be a variety of data types, including strings, integers, floats, tuples;.
* can access dictionary values by keys;
* use square brackets inside curly brackets { [ ] };
* use colons between the key and the value(s);
* use commas to separate each key group and each value within a key group;
* make it quicker and easier for a Python interpreter to find specific elements, as compared to a list.

### Dictionary Example:

pet\_dictionary = {"dogs": ["Yorkie", "Collie", "Bulldog"], "cats": ["Persian", "Scottish Fold", "Siberian"], "rabbits": ["Angora", "Holland Lop", "Harlequin"]}

print(pet\_dictionary.get("dogs", 0))

# Should print ['Yorkie', 'Collie', 'Bulldog']

print(pet\_dictionary.get("cats", 0))

print(pet\_dictionary.get("rabbits", 0))

## Lists only:

* are ordered sets;
* access list elements by index positions;
* require that these indices be integers;
* use square brackets [ ];
* use commas to separate each list element.

### **List Example:**

pet\_list  = ["Yorkie", "Collie", "Bulldog", "Persian", "Scottish Fold", "Siberian", "Angora", "Holland Lop", "Harlequin"]

print(pet\_list[0:3])

# Should print ['Yorkie', 'Collie', 'Bulldog']

print(pet\_list[3:6])

print(pet\_list[6:9])

# Resources

For additional information about dictionaries, please visit:

* [Mapping Types — dict](https://docs.python.org/3/library/stdtypes.html#mapping-types-dict) - Official python.org documentation for dictionary methods
* [Python Dictionaries](https://www.w3schools.com/python/python_dictionaries.asp) - Tutorial with interactive code blocks for practicing using dictionary methods and operations;