Data Structure Conversions

This lesson highlights the main ways of converting one data structure to another.

We'll cover the following

We've already seen how we can perform conversions between Python's primitive data types.

Well, the same principle works for data structures as well!

Explicit Conversion

The template for explicitly converting from one data structure to another is as follows:

```
destination_structure_name(source_structure_object)
```

`destination_structure_name` is the name of the data structure that we want to convert to.

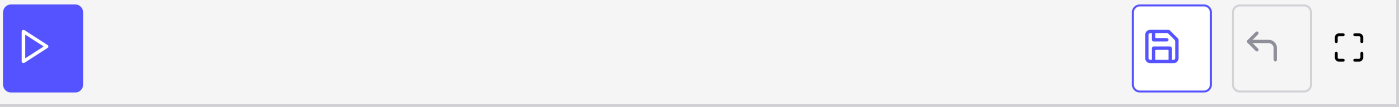
`source_structure_object` is the object which we want to convert.

Converting to a List

We can convert a tuple, set, or dictionary to a list using the `list()` constructor. In the case of a dictionary, only the keys will be converted to a list.

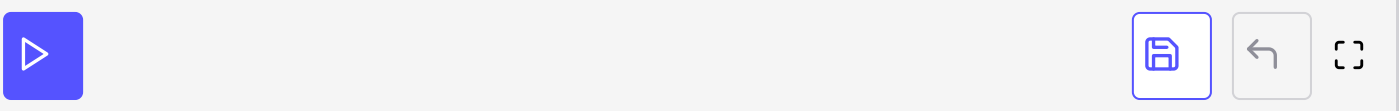
```
star_wars_tup = ("Anakin", "Darth Vader", 1000)
print(star_wars_tup)
```

```
star_wars_set = {"Anakin", "Darth Vader", 1000}
print(star_wars_set)
star_wars_dict = {1: "Anakin", 2: "Darth Vader", 3: 1000}

print(star_wars_dict)

star_wars_list = list(star_wars_tup)  # Converting from tuple
print(star_wars_list)

star_wars_list = list(star_wars_set)  # Converting from set
print(star_wars_list)

star_wars_list = list(star_wars_dict)  # Converting from dictionary
print(star_wars_list)
```



We can also use the `dict.items()` method of a dictionary to convert it into an iterable of `(key, value)` tuples. This can further be cast into a list of tuples using `list()`:

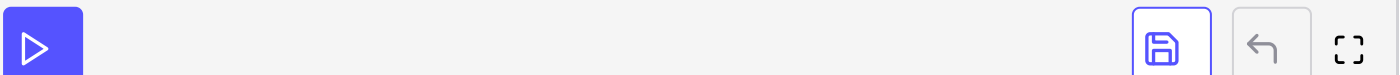
```
star_wars_dict = {1: "Anakin", 2: "Darth Vader", 3: 1000}
print(star_wars_dict)

star_wars_list = list(star_wars_dict.items())
print(star_wars_list)
```



Converting to a Tuple

Any data structure can be converted to a tuple using the `tuple()` constructor. In the case of a dictionary, only the keys will be converted to a tuple:

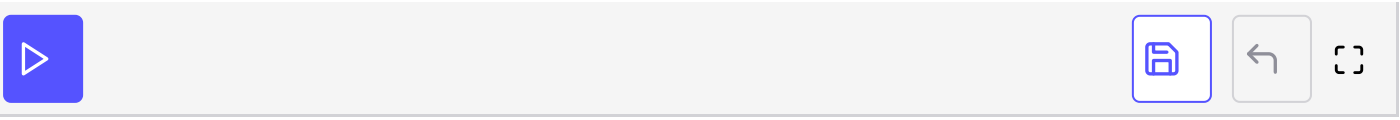
```
star_wars_list = ["Anakin", "Darth Vader", 1000]
print(star_wars_list)
star_wars_set = {"Anakin", "Darth Vader", 1000}
print(star_wars_set)
star_wars_dict = {1: "Anakin", 2: "Darth Vader", 3: 1000}
print(star_wars_dict)

star_wars_tup = tuple(star_wars_list)  # Converting from list
print(star_wars_tup)

star_wars_tup = tuple(star_wars_set)  # Converting from set
print(star_wars_tup)

star_wars_tup = tuple(star_wars_dict)  # Converting from dictionary
print(star_wars_tup)
```



Converting to a Set

The `set()` constructor can be used to create a set out of any other data structure. In the case of a dictionary, only the keys will be converted to a set:

```
star_wars_list = ["Anakin", "Darth Vader", 1000]
print(star_wars_list)
star_wars_tup = ("Anakin", "Darth Vader", 1000)
print(star_wars_tup)
star_wars_dict = {1: "Anakin", 2: "Darth Vader", 3: 1000}
print(star_wars_dict)

star_wars_set = set(star_wars_list)  # Converting from list
print(star_wars_set)

star_wars_set = set(star_wars_tup)  # Converting from tuple
print(star_wars_set)

star_wars_set = set(star_wars_dict)  # Converting from dictionary
print(star_wars_set)
```



Converting to a Dictionary

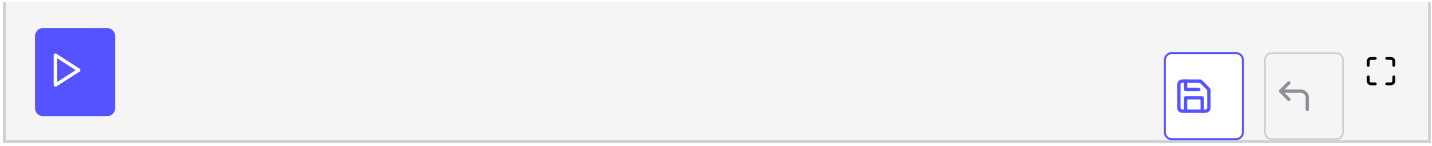
The `dict()` constructor cannot be used in the same way as the others because it requires key-value pairs instead of just values. Hence, the data must be stored in a format where **pairs** exist.

For example, a list of tuples where the length of each tuple is `2` can be converted into a dictionary.

Those pairs will then be converted into key-value pairs:

```
star_wars_list = [[1,"Anakin"], [2,"Darth Vader"], [3, 1000]]
print (star_wars_list)
star_wars_tup = ((1, "Anakin"), (2, "Darth Vader"), (3, 1000))
print (star_wars_tup)
star_wars_set = {(1, "Anakin"), (2, "Darth Vader"), (3, 1000)}
print (star_wars_set)

star_wars_dict = dict(star_wars_list) # Converting from list
print(star_wars_dict)

star_wars_dict = dict(star_wars_tup) # Converting from tuple
print(star_wars_dict)

star_wars_dict = dict(star_wars_set) # Converting from set
```

```
print(star_wars_dict)
```



There are also other ways to convert between data structures that don’t involve explicit conversion.

We urge you to explore such methods on your own.

Before moving on to **Python libraries** in the next section, do check out the data structures quiz and coding exercises in the following lessons!