PYTHON

DICTIONARY

Mind what you have learned, save you it can. - Yoda

WHAT IS A DICTIONARY

A dictionary is an

unordered collection of key / value pairs.

```
#simple Key Values
names_and_force_side = {
    "Luke":"light",
    "Maul":"dark",
    "Han":"grey",#he did shoot first
    "Yoda":"light"
}

#same data... different way of presenting

force_side = {
    "light":["Luke", "Yoda"],
    "Dark":["Maul"],
    "Grey":["Han"]
}
```

Dictionaries are created by coma (,) separating the key value pairs, which are created by seperating the key from the value with a colon (:), inbetween curly brackets ({})

Dictionaries can contain any combination of types as long as the key is a valid key.

```
#simple access item
force_side = {
    "Light":["Luke", "Yoda"],
    "Dark":["Maul"],
    "Grey":["Han"]
}

print(force_side["Dark"])
good_guys = force_side["Light"]
print(good_guys[0])

print(force_side["Light"][1])

falcon_parts = {
    "Navigation":['L3', 'V-5']
```

You get the value of an item by using it's key.

You can access the value of a nested list or dictionary by using the index or key of the nested item.

```
falcon_parts = {
    "Navigation":['L3','V-5'],
    "Deflector_Shield":"Toplex",
    "Hyperdrive":"SSP05",
    "Extras":{
        "Entertainment":['Dejarik Board', "Hinding Spaces"],
        "Luck": 'Aurodium-Plated Gold Dice',
        "Smuggling":"Sensor Proof Smuggler Compartment"
    },
    "Armement":[
        'AG-2G Laser Cannon',
        'AG-2G Laser Cannon'
    ],
    "cockpit_seating":5
}
```

Add an element to a dictionary by assigning a key and then the equal symbol (=) and then the value you want to add.

Replace an element by using the key and giving it a new value.

```
falcon_parts = {
    "Navigation":['L3','V-5'],
    "Deflector_Shield":"Toplex",
    "Hyperdrive":"SSP05",
    "Extras":{
        "Entertainment":['Dejarik Board', "Hinding Spaces"],
        "Luck": 'Aurodium-Plated Gold Dice',
        "Smuggling":"Sensor Proof Smuggler Compartment"

    },
    "Armement":[
        'AG-2G Laser Cannon',
        'AG-2G Laser Cannon'
    ],
    "cockpit_seating":5
}
```

Remove an item from a dictionary much in the same way you remove an item on a list.

```
falcon_parts = {
    "Navigation":['L3','V-5'],
    "Deflector_Shield":"Toplex",
    "Hyperdrive":"SSP05",
    "Extras":{
        "Entertainment":['Dejarik Board', "Hinding Spaces"],
        "Luck": 'Aurodium-Plated Gold Dice',
        "Smuggling":"Sensor Proof Smuggler Compartment"
    },
    "Armement":[
        'AG-2G Laser Cannon',
        'AG-2G Laser Cannon'
    ],
    "cockpit_seating":5
}
```

Sometimes you need to know if a key exists without causing an exception. The get method can help with that.

The 'in' oporator can be used with a dictionary in conjunction with if to add additional logic.

```
falcon_parts = {
    "Navigation":['L3','V-5'],
    "Deflector_Shield":"Toplex",
    "Hyperdrive":"SSP05",
    "Extras":{
        "Entertainment":['Dejarik Board', "Hinding Spaces"],
        "Luck": 'Aurodium-Plated Gold Dice',
        "Smuggling":"Sensor Proof Smuggler Compartment"
    },
    "Armement":[
        'AG-2G Laser Cannon',
        'AG-2G Laser Cannon'
    ],
    "cockpit_seating":5
}
```

Looping through dictionaries are basically the same as looping through lists. With the exception being the key is the itirator instead of the value.

The 'type' is new. The type function returns the type of the argument. The result is compared to the type given. Examples are 'str', 'int', 'list', 'dict', 'float'