



SETS

List Limitations

Dictionary Basics

Modifying
Dictionaries

Dictionary
Methods

Nested
Dictionaries

Sets

A **set** is a collection of unique values

- Sets are **unordered**, which means their values cannot be accessed via index or key
- Sets are mutable (values can be added/removed), but set *values* must be **unique** & **immutable**

```
my_set = {'snowboard', 'snowboard', 'skis', 'snowboard', 'sled'}  
  
my_set  
{'skis', 'sled', 'snowboard'}
```

Sets can be created with curly braces `{}`

The absence of colons differentiates them from dictionaries

```
my_set = set(['snowboard', 'snowboard', 'skis', 'snowboard', 'sled'])  
  
my_set  
{'skis', 'sled', 'snowboard'}
```

Sets can also be created via conversion using **`set()`**

Note that duplicate values are automatically removed when created



WORKING WITH SETS

List Limitations

Dictionary Basics

Modifying
Dictionaries

Dictionary
Methods

Nested
Dictionaries

Sets

You can conduct **membership tests** on sets

```
my_set
```

```
{'skis', 'sled', 'snowboard'}
```

```
'snowboard' in my_set
```

```
True
```

You can **loop through** them

```
for value in my_set:  
    print(value)
```

```
snowboard  
sled  
skis
```

But you **can't index** them (*they are unordered*)

```
my_set[0]
```

```
TypeError: 'set' object is not subscriptable
```