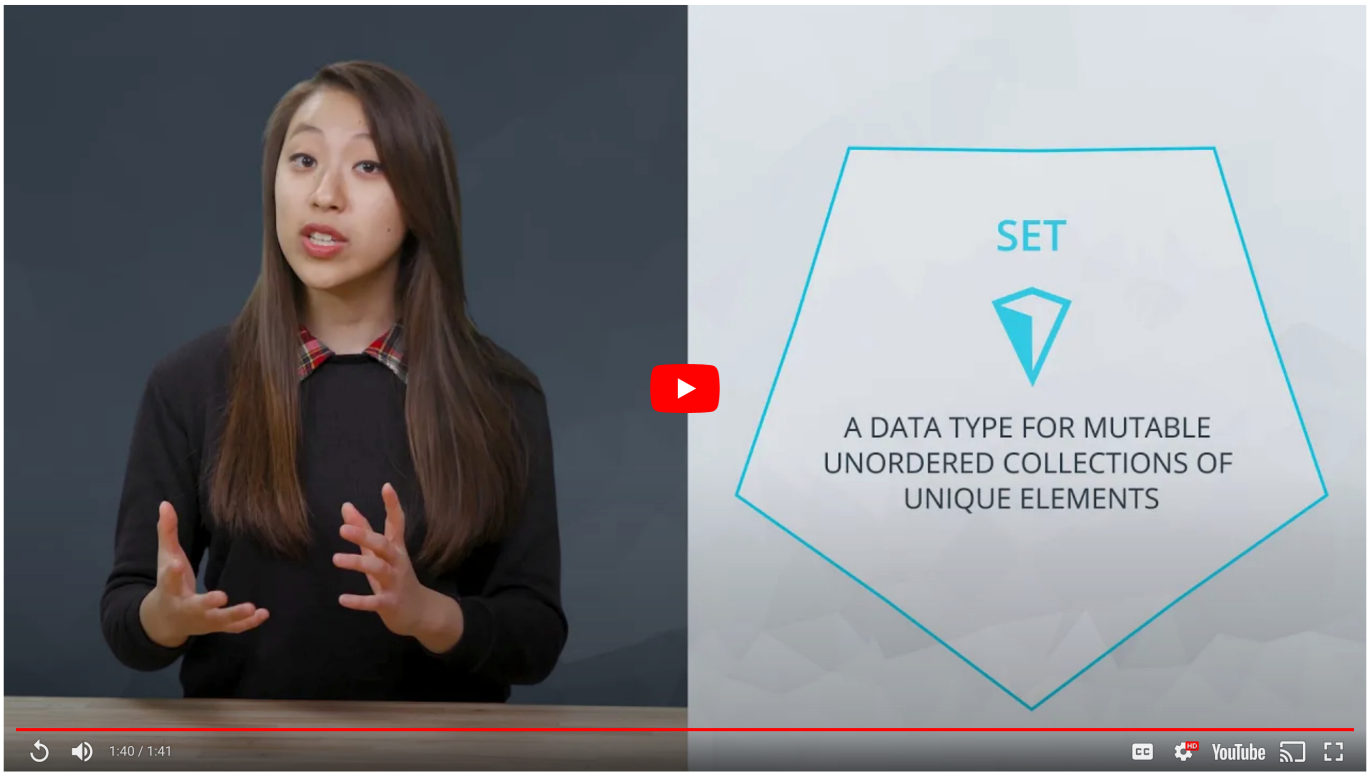
Sets



Sets

A **set** is a data type for mutable unordered collections of unique elements. One application of a set is to quickly remove duplicates from a list.

```
numbers = [1, 2, 6, 3, 1, 1, 6]
unique_nums = set(numbers)
print(unique_nums)
```

This would output:

```
{1, 2, 3, 6}
```

Sets support the `in` operator the same as lists do. You can add elements to sets using the `add` method, and remove elements using the `pop` method, similar to lists. Although, when you pop an element from a set, a random element is removed. Remember that sets, unlike lists, are unordered so there is no "last element".

```
fruit = {"apple", "banana", "orange", "grapefruit"}  # define a set

print("watermelon" in fruit)  # check for element

fruit.add("watermelon")  # add an element
print(fruit)

print(fruit.pop())  # remove a random element
print(fruit)
```

This outputs:

```
False
{'grapefruit', 'orange', 'watermelon', 'banana', 'apple'}
grapefruit
{'orange', 'watermelon', 'banana', 'apple'}
```

Other operations you can perform with sets include those of mathematical sets. Methods

Other operations you can perform with sets include those of mathematical sets. Methods like union, intersection, and difference are easy to perform with sets, and are much faster than such operators with other containers.

Previous Next

Give Page Feedback