14

SETS

WAT LEREN WE?

- Sets
- Methodes **`add()`**, **`update()`**, **`remove()`**, **`discard()`**, **`clear()`**, **`pop()`**, ~~**`copy()`**~~, **`union()`**, **`intersection()`**, **`difference()`**, **`isdisjoint()`**, **`issubset()`**, **`issuperset()`** en **`len()`**
- Frozensets

WAT?

- Ongeordende collectie van elementen
- Unieke elementen
- Geen indices

```
>>> woonplaatsen = {'Waregem', 'Tielt', 'Veurne'}
>>> s[1]
Traceback (most recent call last):
  File "<input>", line 1, in <module>
TypeError: 'set' object does not support indexing
```

IN

- Testen of een element tot een set behoort.

```
>>> woonplaatsen = {'Waregem', 'Tielt', 'Veurne'}

>>> 'Tielt' in woonplaatsen

True

>>> 'Deinze' in woonplaatsen

False
```

LEN

- Aantal elementen in een set.

```
>>> woonplaatsen = {'Waregem', 'Tielt', 'Veurne'}

>>> len(woonplaatsen)

3
```

FOR-LUS

- Elementen van een set overlopen:

```
woonplaatsen = {'Waregem', 'Tielt', 'Veurne'}

for woonplaats in woonplaatsen

    print(woonplaats)
```

```
Tielt

Waregem

Veurne
```

SORTEREN

- Cast naar list

```
>>> woonplaatsen = {'Waregem', 'Tielt', 'Veurne'}
>>> woonplaatsen.sort()
Traceback (most recent call last):
  File "<input>", line 1, in <module>
AttributeError: 'set' object has no attribute 'sort'
>>> l = list(woonplaatsen)
>>> l.sort()
```

SET METHODES

- **add()**: één element toevoegen aan een set.

```
>>> woonplaatsen = {'Waregem', 'Tielt', 'Veurne'}

>>> woonplaatsen.add('Deinze')

>>> print(woonplaatsen)

{'Tielt', 'Veurne', 'Waregem', 'Deinze'}
```

SET METHODES

- `update()`: meerdere elementen toevoegen aan een set.

```
>>> woonplaatsen = {'Waregem', 'Tielt', 'Veurne'}

>>> woonplaatsen.update(['Deinze', 'Ieper'])

>>> print(woonplaatsen)

{'Deinze', 'Ieper', 'Veurne', 'Tielt', 'Waregem'}
```

SET METHODES

- `remove()`: een element verwijderen uit een set.

```
>>> woonplaatsen = {'Waregem', 'Tielt', 'Veurne'}
>>> woonplaatsen.remove('Leuven')
Traceback (most recent call last):
  File "<input>", line 1, in <module>
KeyError: 'Leuven'
>>> woonplaatsen.remove('Veurne')
>>> print(woonplaatsen)
{'Waregem', 'Tielt'}
```

SET METHODES

- **discard()**: een element verwijderen uit een set.

```
>>> woonplaatsen = {'Waregem', 'Tielt', 'Veurne'}

>>> woonplaatsen.discard('Leuven')

>>> woonplaatsen.discard('Tielt')

>>> print(woonplaatsen)

{'Waregem', 'Veurne'}
```

SET METHODES

- `pop()`: geeft een willekeurig element terug.

```
>>> woonplaatsen = {'Waregem', 'Tielt', 'Veurne'}

>>> woonplaatsen.pop()

'Tielt'

>>> print(woonplaatsen)

{'Veurne', 'Waregem'}
```

SET METHODES

- `union()`: de vereniging van twee sets.

```
>>> g1 = {'Jan', 'Piet', 'Joris'}

>>> g2 = {'Mieke', 'Jan'}

>>> g1.union(g2)

{'Mieke', 'Piet', 'Joris', 'Jan'}
```

SET METHODES

- `intersection()`: de vereniging van twee sets.

```
>>> g1 = {'Jan', 'Piet', 'Joris'}

>>> g2 = {'Mieke', 'Jan'}

>>> g3 = {'Korneel'}

>>> g1.intersection(g2)

{'Jan'}

>>> g1.intersection(g3)

set()
```

SET METHODES

- **`difference()`**: de elementen van de eerste set die niet in de tweede set zitten.

```
>>> g1 = {'Jan', 'Piet', 'Joris'}

>>> g2 = {'Mieke', 'Jan'}

>>> g1.difference(g2)

{'Joris', 'Piet'}
```

SET METHODES

- `isdisjoint()`: hebben sets elementen gemeen?

```
>>> g1 = {'Jan', 'Piet', 'Joris'}

>>> g2 = {'Mieke', 'Jan'}

>>> g3 = {'Korneel'}

>>> g1.isdisjoint(g2)

False

>>> g1.isdisjoint(g3)

True
```

SET METHODES

- **`issubset()`**: komen alle elementen van de eerste set ook voor in de tweede set?

```
>>> g1 = {'Jan', 'Piet', 'Joris'}

>>> g2 = {'Jan'}

>>> g1.issubset(g2)

False

>>> g2.issubset(g1)

True
```

SET METHODES

- **`issuperset()`**: komen alle elementen van de tweede set ook voor in de eerste set?

```
>>> g1 = {'Jan', 'Piet', 'Joris'}

>>> g2 = {'Jan'}

>>> g1.issuperset(g2)

True

>>> g2.issuperset(g1)

False
```

FROZENSETS

- Elementen van een `frozenset` kan je niet veranderen.

```
>>> w = ['Waregem', 'Tielt', 'Veurne']

>>> woonplaatsen = frozenset(w)

>>> woonplaatsen.add('Deinze')

Traceback (most recent call last):

  File "<input>", line 1, in <module>

AttributeError: 'frozenset' object has no
attribute 'add'
```