

Ohjelmoinnin peruskurssi Y1

CS-A1111

Listojen metodeita ja operaattoreita

Oppimistavoitteet: tämän videon jälkeen

- ▶ Tunnet lisää erilaisia tapoja käyttää listoja Python-ohjelmissa.

Alilistat

- ▶ Listasta voi ottaa helposti alilistoja (alkuperäisen listan osia):

```
lista = [2, 4, 6, 8, 10, 12, 14, 16]
alilista = lista[2:5]
print(alilista)
```

Tulostus

```
[6, 8, 10]
```

- ▶ Ensimmäinen tai viimeinen indeksi voidaan myös jättää pois:

```
print(lista[:5])
```

Tulostus

```
[2, 4, 6, 8, 10]
```

```
print(lista[5:])
```

Tulostus

```
[12, 14, 16]
```

Alkoita listan lopusta

- ▶ Negatiiviset indeksit tarkoittavat alkioita listan lopusta lähtien:

```
lista = [2, 4, 6, 8, 10, 12, 14, 16]  
print(lista[-1])
```

Tulostus

16

```
print(lista[:-1])
```

Tulostus

[2, 4, 6, 8, 10, 12, 14]

Listan järjestäminen ja kääntäminen

- ▶ Metodi `sort` järjestää listan:

```
lista = [4, 6, 10, 16, 14, 2, -3, -5]
lista.sort()
print(lista)
```

Tulostus

```
[-5, -3, 2, 4, 6, 10, 14, 16]
```

- ▶ Metodi `reverse` kääntää listan järjestyksen päinvastaiseksi.

```
lista.reverse()
print(lista)
```

Tulostus

```
[16, 14, 10, 6, 4, 2, -3, -5]
```

Listasta järjestetty kopio

- ▶ *Funktio* `sorted` tekee listasta kopion ja järjestää tämän kopion. Alkuperäinen lista jää entiselleen:

```
lista1 = [4, 6, 10, 16, 14, 2, -3, -5]
lista2 = sorted(lista1)
print("Uusi lista:", lista2)
print("Alkuperäinen lista:", lista1)
```

Tulostus

```
Uusi lista: [-5, -3, 2, 4, 6, 10, 14, 16]
Alkuperäinen lista: [4, 6, 10, 16, 14, 2, -3, -5]
```

Listojen yhdistäminen

- ▶ Kaksi listaa voidaan yhdistää käyttämällä operaattoria +:

```
lista1 = [1, 2, 3]
lista2 = [4, 5, 6]
yhteislista = lista1 + lista2
print(yhteislista)
```

Tulostus

```
[1, 2, 3, 4, 5, 6]
```


Vielä muutama listojen metodi

- ▶ Listan pituus: `len` eli `len(lista)` antaa listan pituuden.
- ▶ Metodin `insert` avulla voi lisätä listaan annetulle indeksille uuden alkion. Esim. `lista.insert(2,7)`: lisää listan 3. alkioksi luvun 7.
- ▶ Metodi `index` palauttaa parametrina annetun alkion ensimmäisen esiintymän indeksin, esim. `lista.index(6)`
- ▶ Operaattorin `in` avulla voi selvittää, onko alkio listassa, esim. `8 in lista`
- ▶ Metodin `remove` avulla voi poistaa ensimmäisen alkion, jolla on parametrina annettu arvo, esim. `lista.remove(10)`
- ▶ Operaattorilla `del` voi poistaa annetulla indeksillä olevan arvon listasta, esim. `del lista[3]`