

Ohjelmoinnin peruskurssi Y1

CS-A1111

Listat

Oppimistavoitteet: tämän videon jälkeen

- ▶ Osaat käsitellä useita yhteenkuuluvia arvoja yhden muuttujan kautta käyttämällä hyväksi listoja.
- ▶ Osaat luoda listan ja tallentaa listaan arvoja.

Listat

- ▶ Esimerkki: halutaan kirjoittaa ohjelma, joka lukee käyttäjältä 30 lämpötilaa.
- ▶ Kun lämpötilat on luettu, ohjelma tulostaa luetut lämpötilat ja niiden keskiarvon.
- ▶ Ratkaisu: käytetään *listaa*.
- ▶ Lista on rakenne, johon voidaan tallentaa useita eri arvoja.
- ▶ Indeksien avulla voidaan määrätä, mitä listan alkioita käsitellään. Esimerkiksi `lampotilat[4]`.

Listan luominen ja alkioiden lisäys

- ▶ Tyhjän listan luominen:

```
lampotilat = []
```

- ▶ Alkion lisäys listan loppuun:

```
lampotilat.append(arvo)
```

- ▶ Uuden arvon sijoittaminen indeksille *i*:

```
lampotilat[i] = arvo
```

- ▶ Esimerkkejä yksittäisen alkion käyttämisestä:

```
print("5. lampotila", lampotilat[4])
```

```
summa = summa + lampotilat[4]
```

Alkiot listaan listaa luodessa

- ▶ Luotavan listan ei tarvitse olla aluksi tyhjä:

```
numerolista = [2, 4, 6, 8]
```

```
nimilista = ['Matti Virtanen', 'Maija Maki', 'Anu Lahti']
```

- ▶ Voidaan myös luoda lista, jossa on aluksi haluttu määrä nollia.

```
LKM = 30
```

```
lampotilat = [0.0] * LKM
```

ja myöhemmin tietyn alkion arvon voi vaihtaa sijoituskäskyllä:

```
lampo = float(input("Seuraava lampotila: "))
```

```
lampotilat[2] = lampo
```

Merkkijonoja sisältävä lista

- ▶ Listan alkioden ei tarvitse olla lukuja, vaan ne voivat olla esimerkiksi merkkijonoja.
- ▶ Saman listan eri alkiot voivat olla keskenään myös erityyppisiä.
- ▶ Listan alkiona voi olla myös toinen lista.
- ▶ Merkkijonoja sisältävät listat ovat käteviä, kun käyttäjälle halutaan tulostaa erilaisia valintavaihtoehtoja.