

Ohjelmoinnin peruskurssi Y1

CS-A1111

Toistokäsky for

Oppimistavoitteet: tämän videon jälkeen

- ▶ Osaat kirjoittaa for-käskyn avulla ohjelman, joka toistaa haluttua käskyä tai käskyjonoa niin, että käskyjono tarvitsee kirjoittaa vain yhteen kertaan.
- ▶ Osaat valita ongelmasi ratkaisuun sopivan toistokäskyn.

Toistokäsky for

- ▶ For-käskyn avulla voidaan käydä jonkun rakenteen alkiot järjestyksessä.
- ▶ Toistokäsky yleisesti:

```
for alkio in tietorakenne:  
    tee jotain alkiolle
```

Ensin kuitenkin avuksi range-funktio

- ▶ Funktion `range` avulla voidaan generoida lukujono (ei täsmällisesti oikea termi), jonka kaikki alkiot voidaan käydä läpi `for`-käslyn avulla.
- ▶ Esimerkiksi `range(11)` generoi lukujonon, joka sisältää kokonaisluvut 0:sta 10:een asti.
- ▶ Annetaan myös jonon ensimmäisen alkio:
`range(1, 11)`.
- ▶ Määrätään jonon kahden peräkkäisen alkion väli:
`range(1, 11, 2)`.
- ▶ Väli voi olla myös negatiivinen:
`range(10, 1, -2)`.

Toistokäskey for range-funktion avulla

- ▶ `range(11)` generoi lukujonon, joka sisältää kokonaisluvut 0:sta 10:een asti.
- ▶ Voidaan kirjoittaa toistokäskey

```
for i in range(11):  
    tee jotain arvolle i  
jossa i saa vuorotellen arvot 0–10.
```

Esimerkki: kertotaulu for- ja range-käskyjen avulla

```
def main():  
    for i in range(11):  
        tulos = i * 6  
        print(i, "* 6 =", tulos)  
  
main()
```

Hellepäivien lukumäärä for-käskyn avulla

```
def main():
    HELLERAJA = 25.0
    hellepaivien_lkm = 0
    rivi = input("Montako lampotilaa annat?\n")
    lkm = int(rivi)
    for i in range(lkm):
        rivi = input("Anna seuraava lampotila.\n")
        lampotila = float(rivi)
        if lampotila >= HELLERAJA:
            hellepaivien_lkm += 1
    print("Hellepaivia oli", hellepaivien_lkm, "kpl.")
```

```
main()
```


Sisäkkäiset toistokäskyt

- ▶ Toistokäskyn sisällä voi olla toinen toistokäsky.
- ▶ Sisempi toistokäsky suoritetaan kokonaan eli sen kaikki kierrokset suoritetaan jokaisella ulomman toistokäskyn kierroksella.
- ▶ Seuraavien kalvojen esimerkkiohjelmat tulostavat kellonaikoja (tunnit ja minuutit) yhden minuutin välein. Minuutteja muutetaan sisemmässä toistokäskyssä ja tunteja ulommassa toistokäskyssä.

Kello for-käskyn avulla

```
def main():  
    for tunnit in range(24):  
        for minuutit in range(60):  
            print(tunnit, ":", minuutit)  
  
main()
```

Kello while-käskyn avulla

```
def main():
    tunnit = 0
    while tunnit < 24:
        minuutit = 0
        while minuutit < 60:
            print(tunnit, ":", minuutit)
            minuutit += 1
        tunnit += 1

main()
```

Kumpaa käyttää: while vai for?

- ▶ while
 - ▶ Kun toistojen lukumäärään saatetaan vaikuttaa silmukan sisällä.
 - ▶ “Kunnes toisin todistetaan”, eli esim. käyttäjä antaa lopetuskomennon tai jokin ehto toteutuu.
- ▶ for
 - ▶ Kun käydään läpi jonkin tietorakenteen alkioita, eli esim. listassa olevia arvoja tai tiedostossa olevia rivejä.
 - ▶ Kun toistojen lukumäärä on ennakolta tiedossa.
- ▶ Kummallakin käskyllä voi tehdä ihan samat asiat