

Ohjelmoinnin peruskurssi Y1

CS-A1111

Tulostuksen muotoilu

- ▶ Tulostuksen muotoilun avulla voi esim.
 - ▶ määrätä desimaaliluvuista tulostettavien desimaalien määrän
 - ▶ määrätä, kuinka leveä kenttä tulostettavalle asialle varataan
 - ▶ päästä eroon ylimääräisistä välilyönneistä tulostuksessa.

Oppimistavoitteet: tämän luennon jälkeen

- ▶ Tiedät, miten voit muotoilla ohjelman tulostetta, esim. säätää tulostettavissa desimaaliluvuissa desimaalien määrää.
- ▶ Tunnet kaksi eri tapaa:
 - ▶ vanhempi tapa `format`-käskyn avulla
 - ▶ uudempi tapa, joka käyttää Pythonin `f`-stringejä

Miten tulostusta muotoillaan?

- ▶ `print`-käskyssä kirjoitetaan lainausmerkkien sisälle tulostettava vakioteksti.
- ▶ Tekstin keskelle lisätään aaltosulut ja niiden sisään *muotoilumääre* sellaiseen paikkaan, johon tulostuksessa halutaan tulevan joku muotoiltava arvo, esimerkiksi muuttujan arvo.
- ▶ Lainausmerkin jälkeen tulostettavat arvot annetaan `format`-käskyllä.
- ▶ Esimerkki:

```
luku1 = 4.579875
tulo = luku1 * luku1
print("Luku on {:.3f}".format(luku1))
print("{:5.2f} toiseen on {:.6.2f}".format(luku1, tulo))
```

Muotoilumääre

- ▶ Muotoilukoodi alkaa kaksoispisteellä.
- ▶ Kaksoispisteen jälkeen voidaan antaa kokonaislukuna tulostukselle varattavan kentän leveys (voidaan myös jättää pois).
- ▶ Tämän jälkeen desimaaliluvuilla voidaan antaa piste ja sen jälkeen käytettävä tarkkuus (desimaalien määrä) kokonaislukuna.
- ▶ Lopuksi tulee määrittelykirjain, joka kertoo, millaista arvoa ollaan tulostamassa:
 - s merkkijono
 - d kokonaisluku
 - f desimaaliluku ilman eksponenttiosaa
 - e desimaaliluku eksponenttiosan kanssa
 - E desimaaliluku eksponenttiosan kanssa
 - g desimaaliluku

Esimerkkejä

- ▶ Muuttujien arvot tulostetaan kahden desimaalin tarkkuudella:
`print("{:.2f} toiseen on {:.2f}".format(luku1, tulo))`
- ▶ Tulostettaville arvoille varataan vakiolevyiset kentät, desimaaliluvut tulostetaan kolmen desimaalin tarkkuudella:
`print("{:3d} {:7.3f} {:9.3f}".format(vuosi, tuotto, paaoma))`
- ▶ Kaksoispiste tulee suoraan merkkijonon jälkeen ja piste suoraan kokonaisluvun jälkeen ilman välilyöntejä välissä:
`print("Opiskelija {:s}: {:d}.".format(nimi, pisteet))`

Tasaus sarakkeessa

- ▶ Oletuksena merkkijonoja sisältävät sarakkeet tasataan vasempaan reunaan ja lukuja sisältävät oikeaan reunaan.
- ▶ Sarakkeen tasausta voi muokata haluamallaan tavalla seuraavien merkkien avulla:

<, ^ >

- ▶ Esimerkki:

```
print("{:>5s} {:>5s} {:<5d} eur".format(suku1, etu1, palkka1))
Puro   Esa 2800 eur
print("{:>5s} {:>5s} {:<5d} eur".format(suku2, etu2, palkka2))
Pomo  Tiina 11000 eur
```

Uudempi tapa: f-stringit

- ▶ Käytössä Pythonin versiosta 3.6 lähtien
- ▶ Ei käytetä `format`-käskeyä, vaan muuttujien nimet tai muut lausekkeet kirjoitetaan suoraan muotoilumääreen aaltosulkujen sisälle ennen kaksoispistettä.
- ▶ Ennen merkkijonon aloittavaa lainausmerkkiä kirjoitetaan `f`-kirjain.
- ▶ Esimerkki:

```
luku1 = 4.579875
tulo = luku1 * luku1
print(f"Luku on {luku1:7.3f}")
print(f"{luku1:5.2f} toiseen on {tulo:6.2f}")
```