



PYTHON-OHJELMOINTI

Sinikka Leivonen 2021

PYTHON- OHJELMOINTIKIELI

- Helppo ja selkeä ohjelmointikieli
- Editoreja
 - <https://www.matikki.fi/guidebook-python/python/editor.html>
 - replit.com/languages/python3

OHJELMOINNIN KESKEISIÄ KÄSITTEITÄ

- Algoritmi = Yksiselitteinen kuvaus jonkin tehtävän suorittamiseen. Algoritmi muodostuu esim. ohjelman eri komennoista.
- Muuttuja
- Ehtolause
 - Vertailuoperaattorit
- Toistolause = silmukka

ENSIMMÄINEN OHJELMA

- Merkkijono

```
1 #tulostetaan tekstiä
2 print("Hei maailma!")
3 print("Tänään koodataan.")
4
5
6
```

Tulos:

```
Hei maailma!
Tänään koodataan.
```

- Tehtävä 1.
 - Tee ohjelma, joka tulostaa "Me koodataan!".

LASKUTOIMITUKSET

- Print-komento tulostaa laskutoimituksen tuloksen

```
1 print(2+3)
2 print(2-3)
3 print(2*3)
4 print(2/3)
5 print(12/3)
6 print(4-3*3)
7 print((4-3)*3)
```

Tulos:

```
5
-1
6
0.6666666666666666
4.0
-5
3
```

- Tehtävä 2: Tee ohjelma, joka tulostaa:

Laskun $2*3 - 1$ vastaus on:

5

MUUTTUJA

- Muuttuja on ohjelmoinnin keskeinen käsite.
- Muuttujaan "säilötään" tietoa, jota ohjelmassa käytetään.
- Muuttaja voi olla esim.
 - merkkijono
 - kokonais- tai desimaaliluku TAI
 - Totuusarvo (true/false).

```
1 #määritetään muuttajat a, b ,c ja d
2 a=5
3 b=6
4 c=7
5 d=a+b+c
6
7 #tulostetaan muuttujan d arvo
8 print(d)
```

- Ohjelma tulostaa muuttujan d arvon, joka on muuttujien a:n, b:n ja c:n summa

Tulos:

18

MUUTTUJA

- Tehtävä 3.
 - Kirjoita ohjelma, jossa muuttajan a :n arvo on 5 ja b :n arvo on 3. Tulosta lukujen summa, erotus, tulo ja osamäärä.

MUUTTUJA + INPUT-KÄSKY

```
1 #käyttäjän vastaus tallentuu muuttujaan nimi
2 nimi = input ("Anna nimesi:")
3
4 #tulostetaan teksti
5 print (nimi)
```

Viesti osoitteesta www.matikki.fi

Anna nimesi:

Tulos:

Sinikka

- Input-käske pyytää käyttäjältä muuttujaan "nimi" tallentuvan tiedon, jonka print-käske tulostaa.

MUUTTUJA + INPUT-KÄSKY

- Tehtävä 4.
 - Laadi ohjelma, joka kysyy käyttäjältä sanan ja tulostaa seuraavan tekstin: "Kirjoitit sanan xxxx."
 - Vinkki: `print("Kirjoitit sanan, muuttuja1, ".)`
- Tehtävä 5.
 - Kirjoita ohjelma, mikä kysyy nimeäsi ja ikääsi.
Sen jälkeen ohjelma tulostaa tervehdyksen: "Hei, xxxx ! Ikäsi on yyyy"

LUKU SYÖTTEEN ARVONA

- Ohjelma kysyy lukua. Kirjoittamisen jälkeen luku tulostetaan.

```
1 a = input("Anna luku:")
2 |
3 print(a)
4
```

VRT.

Tulos:

5

- Ohjelma kysyy kahta lukua, joiden tuloa on tarkoitus tulostaa. Ohjelma ei toimi.

```
1 a = input("Anna luku:")
2 b = input ("Anna toinen luku:")
3 |
4 print(a*b)
5
6
7
```

Tulos:

[Export code](#)

```
Traceback (most recent call last): module __main__
line 10 exec(t); module line 5 print(a*b) TypeError:
Can't multiply sequence by non-int of type 'str'
```

LUKU SYÖTTEEN ARVONA

- Ratkaisu

```
1 a= int(input("Anna luku:"))
2 b= int(input("Anna toinen luku:"))
3
4 print(a*b)
```

- Input palauttaa annetun arvon merkkijonona. Ohjelmalle täytyy kertoa, että syötettä käsitellään lukuna.
- Int muuttaa syötteen kokonaisluvuksi.

Liukuluku = desimaaliluku

- Alla hyödyllisiä funktioita.
- Funktiot **int**, **float** ja **str** ovat tarpeellisia, koska ohjelmoinnissa lukuja ja merkkijonoja käsitellään eri tavoin. Funktioiden avulla voidaan muuttaa muuttujan tyyppiä.

len	laskee merkkijonon merkkien määrän
int	muuttaa arvon kokonaisluvuksi
float	muuttaa arvon liukuluvuksi
str	muuttaa arvon merkkijonoksi
round	pyöristää luvun halutulle tarkkuudelle
abs	laskee luvun itseisarvon

LUKU SYÖTTEEN ARVONA

- Tehtävä 6:
 - Kirjoita ohjelma, joka laskee tuotteen alennetun hinnan ja tulostaa "Tuotteen alennettu hinta on xx,yy euroa."
 - Vinkki 1: Input-komennon eteen kannattaa laittaa **float** eli **float(input(" "))**. Syy: Hinnat ilmaistaan usein desimaalilukuina (eli liukulukuina).
 - Vinkki 2: **print("Alennettu hinta:", round(hinta, 2))**
 - Vinkki 3: Ohjelmassa desimaaliluvun syöttämisessä käytetään pistettä, ei pilkkua.

EHTOLAUSE: IF-ELSE

```
1 nimi=input("Kerro nimesi.")
2 if nimi=="Sauli":
3     print("Hei pressa!")
4 else:
5     print("OK!")
6
```

- Ehtolause if-else mahdollistaa vastausten vertailun.

ehto	merkitys
$a == b$	a ja b ovat yhtä suuret
$a != b$	a ja b eivät ole yhtä suuret
$a < b$	a on pienempi kuin b
$a <= b$	a on pienempi tai yhtä suuri kuin b
$a > b$	a on suurempi kuin b
$a >= b$	a on suurempi tai yhtä suuri kuin b

- Ehtoja voi myös yhdistää sanoilla AND ja OR.

EHTOLAUSE: IF-ELSE

- Tehtävä 7:
- Kirjoita ohjelma, mikä kysyy kokonaislukua.
 - Jos kokonaisluku on yhtä suuri tai suurempi kuin 100, ohjelma tulostaa "Annoit suuren luvun".
 - Jos kokonaisluku on pienempi kuin 100, ohjelma tulostaa "Annoit pienen luvun".

TOISTOLAUSE ELI SILMUKKA

- Toistolauseita tarvitaan, jos ohjelmassa on samoja komentoja moneen kertaan.
- Toistolauseiden avulla koodista saadaan lyhyempi ja selkeämpi.
- Yleisimmät toistolauseetyypit ovat **for** ja **while**.
 - For-silmukkaa käytetään, kun tiedetään toistojen määrä
 - While-silmukka on silloin, kun silmukan pitää tapahtua niin monta kertaa, kunnes muutos tapahtuu.

FOR-SILMUKKA

- Ohjelma toistaa silmukan kymmenen kertaa aloittaen lukujen tarkastelun luvusta 0.

```
1 ▾ for numero in range(10):  
2     print("Luku " + str(numero))  
3  
4  
5
```

Tulos:

```
Luku: 0  
Luku: 1  
Luku: 2  
Luku: 3  
Luku: 4  
Luku: 5  
Luku: 6  
Luku: 7  
Luku: 8  
Luku: 9
```

- Ohjelma aloittaa lukujen tarkastelun luvusta 0, mutta tulostaa vasta luvusta 1 alkaen.

```
1 ▾ for numero in range(1, 10):  
2     print("Luku " + str(numero))  
3  
4  
5
```

Luku:1
Luku:2
Luku:3
Luku:4
Luku:5
Luku:6
Luku:7
Luku:8
Luku:9

FOR-SILMUKKA

- Tehtävä 8:
- Kirjoita ohjelma, joka tulostaa joka toisen luvun luvusta 2 alkaen lukuun 12 asti.

(Vinkki: `range (x, y, 2)`)

Tämä käskii ohjelman jättämään joka toisen luvun väliin.

Tulos:

Luku on 2

Luku on 4

Luku on 6

Luku on 8

Luku on 10

Luku on 12

WHILE-SILMUKKA

- Ohjelma aloittaa luvusta 1. Lukuun lisätään jokaisen silmukan lopuksi +1 kunnes saavutetaan kokonaisluku, joka on pienempi kuin 11.

```
1 luku = 1
2 while luku < 11:
3     print ("Luku on ", luku)
4     luku = luku + 1
5 print ("Loppu")
6
```

Luku on 1
Luku on 2
Luku on 3
Luku on 4
Luku on 5
Luku on 6
Luku on 7
Luku on 8
Luku on 9
Luku on 10
Loppu

HUOM! Koodi `luku = luku + 1` voidaan kirjoittaa myös `luku += 1`.

WHILE-SILMUKKA

- Ohjelma pyytää antamaan lukuja ja toistaa pyyntöä, kunnes vastataan nolla, jolla ohjelma pysähtyy. Tämän jälkeen ohjelma antaa tulosteen, jossa se ilmoittaa annettujen lukujen määrän. Käytössä on kaksi

muuttujaa

```
1 laskuri=0
2 luku = int(input("Anna ensimmäinen positiivinen luku"))
3 while luku!=0:
4     if luku>0:
5         laskuri= laskuri+1
6         luku = int(input("Anna seuraava luku "))
7 print("Annoit",laskuri,"positiivista lukua")
```

HUOM! Koodi `laskuri = laskuri +1` voidaan kirjoittaa myös `laskuri += 1`.

Viesti osoitteesta www.matikki.fi

Anna ensimmäinen positiivinen luku:

OK

Peru

Annoit 4 positiivista lukua

WHILE-SILMUKKA

- Tehtävä 9.
- Tee ohjelma, joka tulostaa #-merkkiä kuvan osoittamalla tavalla while-silmukkaa käyttäen.
(Vinkki: Hyödynnä print-komennon kanssa kertomerkkiä *)

Tulos:

```
#####
```

```
#####
```

```
####
```

```
###
```

```
#
```


BONUSTEHTÄVÄT:

- Tehtävä 10:
- Tee tehtävä, joka tutkii, onko syötetty kokonaisluku parillinen vain pariton.
- Vinkki: Ohjelmoinnissa jakojäännös ilmoitetaan %-merkillä. Koodi **luku % 2 == 0** tarjoittaa, että kun luku jaetaan 2:lla, jakojäännös on nolla eli luku on parillinen.
- Tehtävä 11:
- Tee ohjelma joka arpoo luvun 1-10 väliltä ja sen jälkeen käyttäjä arvuuttelee lukua.
- Vinkki: Koodin alkuun tuodaan koodikirjastosta satunnaisuuden mahdollistava komento random. Satunnaisuuden mahdollistaa puolestaan random.randint(x,y).

```
import random  
arvottuluku=random.randint(1, 10)
```

TARKISTUKSET:

- Tehtävä 1.

```
1 print("Me koodataan!")|
```

- Tehtävä 2

```
1 print("Laskun 2*3-1 vastaus on: ")  
2 print(2*3-1)
```

TARKISTUKSET:

- Tehtävä 3.

```
1  a=5
2  b=3
3  print(a+b)
4  print(a-b)
5  print(a*b)
6  print(a/b)
7  |
```

- Tehtävä 4.

```
1  sana = input("Anna sana: ")
2  print("Kirjoitit sanan", sana, ".")
3
```

TARKISTUKSET:

- Tehtävä 5.

```
1 nimi = input("Anna nimesi: ")
2 ika = input("Anna ikäsi: ")
3 print("Hei",nimi, "! Ikäsi on", ika)
4
```

- Tehtävä 6.

```
1 a = float(input("Anna tuotteen hinta: "))
2 b = float(input("Anna alennusprosentti: "))
3 hinta = (a-(a*b/100))
4 print("Tuotteen alennettu hinta on", round(hinta,2), "euroa.")
5
6
```

Mahdollistaa desimaalilukujen käsittelyn. Desimaaliluku syötetään käyttämällä pistettä.

Pyöristää luvun kahden desimaalin tarkkuudella.

TARKISTUKSET:

- Tehtävä 7:

```
1 luku = int(input("Anna joku kokonaisluku"))
2 if luku >= 100:
3     print("Annoit suuren luvun.")
4 else:
5     print("Annoit pienen luvun.")
```

- Tehtävä 8:

```
1 for luku in range(2,13,2):
2     print("Luku on ", luku)
```

TARKISTUKSET:

- Tehtävä 9

```
1 risuaita = 5
2 while risuaita > 0:
3     print (risuaita * "#" )
4     risuaita = risuaita-1
```

- Tehtävä 10.

```
1 luku = int(input("Anna luku: "))
2 if luku % 2 == 0:
3     print("Luku on parillinen.")
4 else:
5     print("Luku on pariton.")
6
```


TARKISTUKSET:

- Tehtävä: 11.

```
1 import random
2 arvottuluku=random.randint(1, 10)
3 luku=int(input("Arvaa luku: "))
4 while luku != arvottuluku:
5     if luku > arvottuluku:
6         print("Luku on liian suuri")
7     else:
8         print("Luku on liian pieni")
9     luku =int(input("Arvaa uudestaan. "))
10 print("Arvasit OIKEIN!")
```

```
Arvaa luku: 6
Luku on liian suuri
Arvaa uudestaan. 3
Luku on liian suuri
Arvaa uudestaan. 1
Luku on liian pieni
Arvaa uudestaan. 2
Arvasit OIKEIN!
```

