

# 1 Intro

```
1 # trinket
2 # vi har valgt trinket siden det er nettbasert og
   krever ingen innstallasjon
3
4 # Første program
5 print('Python er gøy!')
6
7 # Python 2 vs Python 3.
8
9
10 # Intro til variabler.
11 # Kanskje hente frem det "keeping score" eksempelet?
12 # Hvis du spiller et spill og skriver ned score på en
   lapp
13 # Evt. Ole petter har en bondegård
14
15 # Vi må ha fnutter for å si at noe er en streng
16 navn = 'Marie'
17
18 # Vi kan printe tekst sammen med variabler
19 print('Hei', navn)
20 print('Hyggelig å hilse på deg!')
21
22 # Hva skjer om vi glemmer fnuttene?
23 navn = Marie
24 print('Hei', navn)
25
26 # Hva skjer om vi har fnutter rundt en variabel?
   Hvorfor?
27 navn = 'Marie'
28 print('Hei', 'navn')
29
30 # Nevne at vi brukte variabler sist med astrokatt
31 # Når vi bygget lego kjente vi kanskje på at det var
   av og til vanskelig å vite
32 # hvilken brikke programmereren mente.
33
34 # til sammen: ca 10 minutter
```

## 2 Regning

```
1 # Vi kan også ha tallvariabler
2 # De skal ikke ha fnutter
3
4 x = 2
5 y = 5
6
7 print("x =", x)
8 print("y =", y)
9
10 z = x + y # - * / **
11
12 print("z =", z)
13
14 # Det er viktig å huske på at formelen ikke blir
    lagret. Kun svaret
15
16
17 print("x =", x)
18 print("y =", y)
19
20 z = x + y # - * / ** % //
21
22 x = 1 # har ingenting å si
23
24 print('z = ', z)
25
26
27 # Oppdatere x
28 x = 3
29 print('x =', x)
30 x = x + 2
31 print('x =', x)
32
33 # 5 minutter
```

### 3 Saldo

```
1
2 saldo = 25450
3 print(f'Startsaldo: {saldo}')
4
5 saldo -= 1000 #det samme som saldo = saldo - 1000
6 print(f'Saldo etter uttak: {saldo}')
7
8 saldo += 10000
9 print(f'Saldo etter innskudd: {saldo}')
10
11 saldo *= 1.0345 # BSU konto 3.45% rente
12 print(f'Saldo etter renter: {saldo:.2f}')
13
14 #4.5 minutter
```

## 4 Volumberegning

```
1 kule_radius = 5
2 pi = 3.14
3
4
5 kule_volum = (4/3)*pi*kule_radius**3
6
7 print(f'Kula har radius: {kule_radius}')
8
9 print(f'Volumet er da: {kule_volum}')
10 # vi kan gjenbruke variabelen
11
12 kule_areal = 4*pi*kule_radius**2
13 print(f'Volumet er da: {kule_areal}')
14
15
16 # Penere utskrift
17 kule_radius = 5
18 pi = 3.14
19
20 kule_volum = (4/3)*pi*kule_radius**3
21
22 print(f'Kula har radius: {kule_radius:.2f}') # endre
    til .1
23
24 print(f'Volumet er da: {kule_volum:.2f}')
25 # vi kan gjenbruke variabelen
26
27 kule_areal = 4*pi*kule_radius**2
28 print(f'Volumet er da: {kule_areal:.2f}')
29
30 # det sparer oss fra feil
31 # Så to fordeler med variabler:
32 # 1. Det sparer oss fra å skrive feil når vi
33 #    gjenbraker
34 # 2. Det gjør kode lettere å lese og mer generell
35 #    (viktig med gode navn)
36
37 # Vi kan også importere pi
```

```
38 from math import pi
39 print(f'Pi er {pi}')
40
41 #Trenger ikke flere desimaler enn 15
42
43 # 13 minutter
```

## 5 Input

```
1
2 # Input tar inn et svar fra brukeren
3 # Minner litt om print
4
5 navn = input('Hva heter du?')
6
7 print(f'Hei, {navn}!')
8
9 # Viktig å lagre svaret i en variabel
10
11 input('Si en farge: ')
12
13
14 # Vi kan lage en adjektivhistorie
15
16
17 # For å printe over flere linjer kan vi bruke
18 # tripple fnutter.
19 """
20 print(f'''En {adjektiv1} høstdag i oktober
21 skulle den {adjektiv2}e klasse 8B på telttur
22 til {sted}. Alle de {adjektiv3}e elevene hadde
23 gledet seg til denne {adjektiv4} turen og
24 hadde med seg både {ting1} og {ting2}.
25 ''')
26 """
27
28 # Vi må definere variablene ellers får vi feil
29
30 adjektiv1 = input('Gi meg et adjektiv: ')
31 adjektiv2 = input('Gi meg enda et adjektiv: ')
32 adjektiv3 = input('Gi meg et til adjektiv: ')
33 adjektiv4 = input('Gi meg et siste adjektiv: ')
34 sted = input('Gi meg et sted')
35 ting1 = input('Gi meg en ting')
36 ting2 = input('Gi meg en til ting')
37
38 print(f'''En {adjektiv1} høstdag i oktober
```

```
39 skulle den {adjektiv2}e klasse 8B på telttur
40 til {sted}. Alle de {adjektiv3}e elevene hadde
41 gledet seg til denne {adjektiv4} turen og
42 hadde med seg både {ting1} og {ting2}.
43 '''
44
45
46 # 10 minutter
```

## 6 Tallinput

```
1 # Problemer med input og tall
2
3 x = input('gi meg et tall')
4
5 print(f'Det dobbelte av {x} er {2*x}')
6
7 # Hvorfor skjer dette?
8
9 # vi må gjøre det om til int (heltall)
10 x = int(input('gi meg et tall'))
11
12 print(f'Det dobbelte av {x} er {2*x}')
13
14 # problemer hvis vi sender inn desimaltall
15
16 # Da må vi gjøre det om til float (flyttall) istedet
17
18 x = float(input('gi meg et tall'))
19
20 print(f'Det dobbelte av {x} er {2*x}')
21
22 # 5 minutt
```



## 7 Regne hypotenus

```
1 # Lag et program som spør om lengden på katetene
2 # i en rettvinklet trekant og skriver ut lengden på
3 # hypotenusen
4
5
6
7 from math import sqrt
8
9 katet1 = float(input('Hvor lang er første katet?'))
10 katet2 = float(input('Hvor lang er andre katet?'))
11
12 # Hva er formelen for hypotenusen?
13
14 # Vi trenger funksjonen sqrt (square root)
15 # fra math
16 from math import sqrt
17
18
19 hypotenus = sqrt(katet1**2 + katet2**2)
20
21 print(f'Hypotenusen er {hypotenus} lang')
22
23 # 5 minutter
```

## 8 Lage muffins

```
1 # muffinseksempel
2
3 # Vi skal lage et program som
4 # Spør en bruker hvor mange muffins de skal lage
5 # Regner ut hvor mye de trenger av hver ingrediens
6 # skriver ut svaret
7
8 # int eller float?
9 antall_muffins = int(input('Hvor mange muffins skal du
    lage?'))
10 print(f'Ditt svar: {antall_muffins}')
11
12 # test at dette fungerer
13 print() # tom linje
14 print('Da trenger du omtrent:')
15 print(f'- {10*antall_muffins} gram smør.')
16 print(f'- {10*antall_muffins} gram sukker.')
17 print(f'- {10*antall_muffins} gram hvetemel.')
18 print(f'- {antall_muffins/5} egg.')
19 print(f'- {antall_muffins/10} ts bakepulver.')
20
21 print('''
22 Smelt smøret og la det kjøle seg litt ned.
23 Pisk eggene sammen med sukker, og rør så forsiktig
24 inn smøret, bakepulver og hvetemel.
25
26 Ta røra i muffinsformer og stek midt i ovnen
27 ved 180 grader i ca 10 minutter.'''')
28
29
30 # Er det rart med et desimaltall med egg?
31
32 print() # tom linje
33 print('Da trenger du omtrent:')
34 print(f'- {10*antall_muffins} gram smør.')
35 print(f'- {10*antall_muffins} gram sukker.')
36 print(f'- {10*antall_muffins} gram hvetemel.')
37 print(f'- {round(antall_muffins/5)} egg.')
```

```

38 print(f'- {antall_muffins/10} ts bakepulver.')
```

```

39
40 print('''Instrukser:
41 Smelt smøret og la det kjøle seg litt ned.
42 Pisk eggene sammen med sukker, og rør så forsiktig
43 inn smøret, bakepulveret og hvetemelet.
44
45 Hell røra i muffinsformer og stek midt i ovnen
46 ved 180 grader i ca 10 minutter.
47
48 Nyt muffins!''')
```

```

49
50
51
52 # Er det noen andre problemer?
53 # Hva om du bare skal lage en muffins og det
54 # blir 0 egg?
55 # Er det noen forslag til forbedringer her?
56 # Hva skjer om noen ber om negative muffins?
57
58 # 13 minutter
```

## 9 Lister

```
1
2
3 # La oss si at vi ønsker å lagre navnene på
4 # alle måneder i et år. Det kan fort bli slitsomt
5 # med så mange variabler
6 måned1 = "januar"
7 måned2 = "februar"
8 måned3 = "mars"
9 måned4 = "april"
10
11 # Det finnes en egen variabeltype for slike samlinger
12 # De kalles lister.
13
14 # Lister lages ved at du "lister" opp alle elementene
15 # med komma mellom og klammeparantes rundt.
16
17 måneder = ["januar", "februar", "mars", "april", "mai"
18            , "juli", "agust", "september", "oktober", "november"]
19
20 # Vi kan printe ut en liste som vanlig
21 print(måneder)
22
23 # For å hente ut av lista kan vi indeksere den med
24 # klammeparanteser. Første element kan hentes slik:
25 print(måneder[0])
26 # Python begynner altså å indeksere på 0
27 # for å hente andre element kan vi skrive:
28 print(måneder[1])
29
30 # For å få lengden til en liste kan vi bruke len
31 # funksjonen
32 print(f'Antall måneder er {len(måneder)}')
33
34 # Men vent! det skal jo være tolv månedr
35
36 # Hvis vi ser nøyerer på lista så ser vi at vi
37 # har glemt desember. Heldigvis kan vi legge til
38 # på slutten av en liste med append
```

```

39 måneder.append("desember")
40 print(måneder)
41
42 # Vi har også glemt en måned til. Hvilken?
43 # Juni? Det er jo bursdagen min, den kan vi ikke
44 # glemme!
45 # Vi kan legge til på hvilken som helst plass med
46 # insert. Hvilken plass skal juni?
47 måneder.insert(5, "juni")
48
49 print(f'Antall måneder er {len(måneder)}')
50 print()
51
52 # Er alt rett da?
53 # august er feil. La oss fikse den. Hvilken indeks
54 # har august?
55 # vi kan overskrive elementer i lister
56
57 måneder[7] = "august"
58 print(måneder)
59
60 # Hva om vi ønsker desember?
61 # print(måneder[12]) # får feil!!!
62 print(måneder[11]) # husk 1 mindre!
63
64 # Vi kan også få siste elementet av lista med
65 # negativ indeksering
66 print(måneder[-1])
67 # nyttig dersom man ikke vet hvor mange elementer
68 # som er i lista.
69
70 # Vi kan kombinere lister med input for å gi brukeren
71 # flere valg
72 print("""Du har valget mellom 3 dører. De er nummerert
    med
73 0, 1 og 2. Bak 2 av dem
74 er det en bare en geit, men en av dørene har en premie
    .
75 Du kan velge dør 0, 1, eller 2. """)
76
77 dører = ["geit", "geit", "flott premie"]

```

```
78 indeks = int(input("Hvilken dør velger du?"))
79 print(f'Du åpner dør {indeks}. Bak den er en {dører[
    indeks]}!')
80
81
82
83 # 13 minutter
```

## 10 Lister

```
1 # Lister er også nyttige i matematikken
2 # Det hender ofte vi trenger å lagre en samling med
3 # tall
4
5 # La oss si at vi ønsker å finne gjennomsnitttiden
6 # Det tok for meg å gå til universitetet forrige uke
7 # Første dagen tok det 41 minutter. Andre dagen tok
8 # det 39 minutter. Tredje dagen tok det 47 minutter.
9 # Siste dagen tok det 43 minutter
10
11 # Vi kan lagre resultatene i en liste
12
13 tid = [41, 39, 47, 43]
14 # på fredag dro jeg ikke til universitetet, men hit.
15 # For å finne gjennomsnittet må vi først finne summen
16 # Det finnes en innebygd funksjon sum som gir summen
17 # av tallene i en liste
18 sum_tid = sum(tid)
19
20 print(f'jeg har tilsammen brukt {sum_tid} minutter')
21
22 # Så må vi dele på antall målinger.
23 # Hvordan kan vi finne det?
24 antall_målinger = len(tid)
25
26 print(f'Over {antall_målinger} dager')
27
28 gjennomsnitt_tid = sum_tid/len(tid)
29 print(f'I gjennomsnitt har jeg brukt {gjennomsnitt_tid
    } minutter')
30
31 # Nå har vi brukt en ferdig funksjon for å finne
    summen
32 # Det er jo kulere å kunne lage sin egen kode for å gj
    øre det
33 # Hvordan ville dere gått frem for å gjøre noe sånt?
34 # Det krever løkker hvilket vi skal prate om etter
    lunsj :)
```

35

36 # 5 minutter



## 11 Microbit

[python.microbit.org](https://python.microbit.org) Trykk på help for guider

```
1 # Hellow world
2 from microbit import *
3
4 display.scroll('Python er kult')
5 display.show(Image.HEART)
6
7 # Vis frem hjerter
8 from microbit import *
9 display.show([Image.HEART, Image.HEART_SMALL])
10
11 # Vis frem eget bilde
12 from microbit import *
13 bilde = Image("59059:"
14               "55055:"
15               "00900:"
16               "90009:"
17               "09990:")
18
19 display.show(bilde)
20
21
22 # Vis frem hjerteanimasjon (VANSKELIG BONUS)
23 from microbit import *
24 display.show([Image.HEART, Image.HEART_SMALL]*10)
25
26
27 # Vis hjemmelagd animasjon
28 from microbit import *
29 bilder = [
30     Image("59059:"
31           "55055:"
32           "00900:"
33           "90009:"
34           "09990:"),
35     Image("95095:"
36           "55055:"
37           "00900:")
```

```
38         "90009:"  
39         "09990:"),  
40     Image("55055:"  
41         "95095:"  
42         "00900:"  
43         "90009:"  
44         "09990:"),  
45     Image("55055:"  
46         "59059:"  
47         "00900:"  
48         "90009:"  
49         "09990:")  
50 ]*5  
51  
52 display.show(bilder)
```