

Oppgaver

I denne seksjonen finner du oppgaver som hører til dag 1 av Kodeskolens kræsjkurs i programmering.

Tema for første dag er *introduksjon til Python, variabler, input, betingelser og løkker*. Dersom du står fast er det bare å spørre. I tillegg anbefaler vi å lese i kompendiet hvis det er noen temaer du synes er spesielt vanskelige. Oppgaver markert som bonusoppgaver er litt mer utfordrende og du velger selv om du har lyst til å prøve deg på dem. God koding!

Variabler og regning

Oppgave 1 *Printing*

- a) Lag et program som skriver ut teksten «Hei, verden!», til skjermen.
- b) Lag et program der du først lagrer navnet ditt i en variabel, og så få programmet ditt til å skrive ut en hilsen direkte til deg.
- c) Hvis du bruker kommandoen `len()` på variabelen din får du ut antall bokstaver i navnet ditt. Endre programmet ditt så det også skriver ut denne informasjonen.
- d) Du kan bruke funksjonen `input()` til å stille brukeren et spørsmål. Bruk dette til å lage et program som spør brukeren om navnet deres, og deretter skriver ut en beskjed som bruker navnet de har oppgitt. Skriv også ut hvor mange bokstaver det er i navnet til vedkommende.

Løsning oppgave 1 *Printing*

a)

```
1 print("Hei, verden!")
```

b)

```
1 navn = "Maria"
2 print(f"Hei, {navn}!")
```

c)

```
1 navn = "Maria"
2 print(f"'{navn}' har {len(navn)} bokstaver i seg.")
   )
```

d)

```
1 navn = input("Hva heter du?\n")
2 print(f"Hei, {navn}! Navnet ditt har {len(navn)}
   bokstaver i seg.")
```

Oppgave 2 *Finn tre feil!*

Her følger det tre programmer som ikke fungerer. Skriv programmet inn på din egen maskin og kjør det, les feilmeldingen, og prøv å tolke den. Når du skjønner hva som er galt, rett opp feilen og kjører programmet.

a)

```
1 Print("Hello, World!")
```

b)

```
1 navn = John
2 print(navn)
```

c)

```
1 navn = input("Hva heter du?")
2 print(name)
```

Løsning oppgave 2 *Finn tre feil!*

a) Vi må bruke liten p i print:

```
1 print("Hello, World!")
```

b) Vi må bruke rundt en string:

```
1 navn = "John"
2 print(navn)
```

c) Vi må passe på at vi bruker riktig variabel:

```
1 navn = input("Hva heter du?")
2 print(navn)
```

Oppgave 3 *Konvertering av temperatur*

I Norge oppgir vi temperaturer i målestokken *Celsius*, men i USA bruker de ofte målestokken *Fahrenheit*. Hvis du finner en kakeoppskrift fra USA kan det for eksempel stå at du skal bake kaken ved 350 grader. Da mener de altså 350°F. Vi vil nå lage et verktøy som kan konvertere denne temperaturen for oss, sånn at vi vet hva vi skal bake kaken ved i Celsius..

For å regne over fra Fahrenheit til Celsius bruker vi formelen:

$$C = \frac{5}{9}(F - 32).$$

Der F er antall grader i Fahrenheit, og C blir antall grader i celsius.

a) Lag et program som spør brukeren om en temperatur i antall grader Fahrenheit, og skriver ut den tilsvarende temperaturen i antall grader Celsius.

Husk å gjøre om svaret til et tall, med enten `int(input())` eller `float(input())`.

b) Bruk programmet ditt til å finne ut hvor mange grader Celsius 350°F svarer til. Virker det rimelig å skulle bake en kake ved denne temperaturen?

Programmet du har lagd tar en temperatur i Fahrenheit, og gjør om til Celsius. Men hva om vi ønsker å gå motsatt vei? Om vi ønsker å lage et nytt program som gjør motsatt, så må vi først ha en formel for F .

c) Klarer du å ta uttrykket

$$C = \frac{5}{9}(F - 32).$$

og løse for F ?

- d) Lag et nytt program som spør brukeren om en temperatur i antall grader celsius, og så skriver ut den tilsvarende temperaturen.
- e) Bruk programmet ditt til å finne frysepunktet og kokepunktet til vann i Fahrenheit målestokken.

Løsning oppgave 3 *Konvertering av temperatur*

a)

```
1 fahrenheit = float(input('Fahrenheit: '))
2 celcius = (5/9)*(fahrenheit-32)
3
4 print(f'{fahrenheit} grader fahrenheit
   tilsvare {celcius:.0f} celcius')
```

b)

```
Fahrenheit: 350
350.0 grader fahrenheit tilsvare 177 celcius
```

177 ° C virker som en rimelig kakebaketemperatur

c)

$$F = \frac{9}{5}C + 32.$$

d)

```

1   celcius = float(input('Celcius: '))
2   fahrenheit = (9/5)*celcius + 32
3
4   print(f'{celcius} grader celcius tilsvare {
      fahrenheit:.0f} fahrenheit')

```

e)

```

Celcius: 0
0.0 grader celcius tilsvare 32 fahrenheit

```

```

Celcius: 100
100.0 grader celcius tilsvare 212 fahrenheit

```

Betingelser

Oppgave 4 *Positive og negative tall*

I denne oppgaven skal vi bruke *betingelser* til å undersøke om et tall er positivt eller negativt.

- Bruk `input` og `float` til å be brukeren om et tall og lagre det i en variabel, `tall`
- Bruk `if` med en *betingelse* til å skrive ut beskjeden “tallet er positivt” dersom tall er større enn 0. Test programmet ditt med å gi 4 som input.
- Bruk `elif` og en betingelse til å skrive ut beskjeden “tallet er null” dersom tall er lik 0. Test programmet ditt ved å gi 0 som input.
- Bruk `else` til å skrive ut beskjeden “tallet er negativt” dersom tallet verken er større enn eller lik 0. Test programmet med -1 som input.

Løsning oppgave 4 *Positive og negative tall*

a)

```
1 tall = float(input('Gi meg et tall: '))
```

b)

```
1 tall = float(input('Gi meg et tall: '))
2 if tall > 0:
3     print('Tallet er positivt')
```

```
Gi meg et tall: 4
Tallet er positivt
```

c)

```
1
2 if tall > 0:
3     print('Tallet er positivt')
4 elif tall==0:
5     print('Tallet er null')
```

```
Gi meg et tall: 0
Tallet er null
```

d)

```
1 tall = float(input('Gi meg et tall: '))
2
3 if tall > 0:
4     print('Tallet er positivt')
5 elif tall==0:
6     print('Tallet er null')
7 else:
8     print('Tallet er negativt')
```

```
Gi meg et tall:  -1
Tallet er negativt
```

Oppgave 5 *Sjekk alder*

En film på kino har aldersgrense 15 år. Vi skal lage et program som interagerer med brukeren, og finner ut om de kan se filmen eller ikke.

- a) Lag et program som printer ut en hilsen til brukeren, og spør hvor gamle de er. Lagre alderen i variabelen `alder`.
- b) Lag en test som sjekker om brukeren er 15 år eller eldre. Skriv ut passende svar avhengig av om de er gamle nok til å se filmen eller ikke.
- c) Utvid programmet ditt til å regne ut hvor mange år det er til brukeren kan se filmen.
- d) La programmet ditt gjøre et unntak for de som er 12 år eller eldre dersom de har med seg en voksen.

Løsning oppgave 5 *Sjekk alder*

a)

```
1 alder = int(input("Hei! Hvor gammel er du? "))
```

b) Vi bruker en `if`-test:

```
1 if alder >= 15:
2     print("Vellommen inn på kino!")
3 else:
4     print("Du er dessverre ikke gammel nok")
```

c) Regner ut antall år til brukeren er 15:

```
1 if alder >= 15:
2     print("Vellommen inn på kino!")
3 else:
```

```

4     år_igjen = 15 - alder
5     print(f"Dú er dessverre ikke gammel nok, kom
        tilbake om {år_igjen} år")

```

d) Legger en ny **if**-test inne i den første testen. Hele programmet blir:

```

1     alder = int(input("Hei! Hvor gammel er du? "))
2
3     if alder >= 15:
4         print("Vellommen inn på kino!")
5
6     elif alder >= 12:
7         med_voksen = input("Har du med deg en voksen (ja
            /nei)? ")
8         if med_voksen == 'ja':
9             print("Velkommen inn på kino!")
10        else:
11            år_igjen = 15 - alder
12            print(f"Dú er dessverre ikke gammel nok. Kom
                tilbake om {år_igjen} år eller ha med deg
                en voksen.")
13
14    else:
15        år_igjen = 15 - alder
16        print(f"Dú er dessverre ikke gammel nok, kom
            tilbake om {år_igjen} år")

```

While-løkker

Oppgave 6 *While løkker for tallrekker*

En **while** løkke har følgende syntaks:

```

1     while [betingelse]
2         [kode som skal gjentas]

```

I denne oppgaven skal vi øve på **while**-løkker til å skrive ut noen tallrekker

- a) Skriv ut alle partall mindre enn 13
- b) Skriv ut hele 7-gangen fra 0 til 70
- c) Skriv ut rekka 1, 3, 9, ..., 243, 729 (**Hint:** Gang med 3 for hver runde i løkka)

Løsning oppgave 6 *While løkker for tallrekker*

a)

```
1 tall = 2
2 while tall < 13:
3     print(tall)
4     tall = tall + 2
```

b)

```
1 tall = 7
2 while tall <= 70:
3     print(tall)
4     tall = tall + 7
```

c)

```
1 tall = 1
2 while tall <= 729:
3     print(tall)
4     tall = tall*3
```

Oppgave 7 *While-løkker for hånd*

Gå igjennom løkkene under for hånd og forutsi hva de kommer til å skrive ut. Etterpå kan du kjøre dem og se om du hadde rett.

a)

```
1 i = 0
2 while i < 10:
3     i += 2
4     print(i)
```

b)

```
1 i = 1
2 while i < 20:
3     i *= 2
4     print(i)
```

c)

```
1 i = 2
2 while i < 10:
3     i = i**2 + 1
4     print(i)
```

d)

```
1 a = 0
2 b = 0
3 c = 0
4
5 while c < 10:
6     a += 1
7     b += a
8     c += a + b
9
10 print(a)
11 print(b)
12 print(c)
```

Løsning oppgave 7 *While-løkker for hånd*

a)

```
2
4
6
8
10
```

b) –

```
2
4
8
16
32
```

c)

```
5
26
```

d)

```
3
6
16
```

Oppgave 8 *While løkker med input*

I denne oppgaven skal vi bruke `while` og `input` til å lage et program som kjører helt til en bruker avslutter det.

- a) Lag en variabel, `runde`, som skal ha verdien `0`.
- b) Lag en variabel, `svar` som skal inneholde en tom streng (`' '`)
- c) Lag en `while`-løkke som fortsetter så lenge svar ikke er lik `'avslutt'` (**Hint:** `while svar != 'avslutt'`). Inne i løkka skal du gjøre følgende:
 - Skrive ut variabelen, `runde`, med `print` for å fortelle brukeren hvor mange ganger programmet har kjørt

- Bruk `input` til å be brukeren å skrive `'avslutt'` dersom de vil avslutte. Lagre svaret i svar variabelen
- Øk runde variabelen med 1 (**Hint:** runde `+=1`)

Løsning oppgave 8 *While løkker med input*

a)

```
1 runde = 0
```

b)

```
1 svar = ""
```

c)

```
1 while svar != "avslutt":  
2     print(f"Denne løkka har kjørt {runde} ganger")  
3     svar = input("Skriv 'avslutt' for å avslutte")  
4     runde += 1
```