

# Programmering i python - avansert

Oppgaveøkt 18.10.22



Planen for i dag (09:00-11:00):

- Litt repetisjon
- Live-koding av et eksempel
- Selvstendig jobbing med oppgaver

```
1  def hilsen(navn):
2      print('~*~*~*~')
3      print(f'Hei, {navn}!')
4
5  hilsen('Ragnhild')
6  hilsen('Oliwia')
7  hilsen('Hanna')
```

```
In [32]: runfile('C:/Users/rawis/OneDrive/Dokumenter/nanna_klasse.py',
wdir='C:/Users/rawis/OneDrive/Dokumenter')
~*~*~*~
Hei, Ragnhild!
~*~*~*~
Hei, Oliwia!
~*~*~*~
Hei, Hanna!
```

## Funksjoner

- Gjør at du slipper å gjenta kode for å gjøre den samme operasjonen flere ganger
- Når funksjonen er skrevet én gang kan du bruke den så mye du vil
- Det finnes mange innebygde funksjoner i Python, f.eks. `print()`

```
11 def funksjon_2(navn, alder):  
12     alder = str(alder)  
13     print(f'{navn} er {alder} år gammel.')  
14  
15 funksjon_2('Ragnhild', 28)
```

```
In [10]: runfile('C:/Users/rawis/OneDrive/Dokumenter/Jobb/Kodeskolen/FU-  
kurs oktober 2022/eksempel.py', wdir='C:/Users/rawis/OneDrive/  
Dokumenter/Jobb/Kodeskolen/FU-kurs oktober 2022')  
Ragnhild er 28 år gammel.
```

- Koden i funksjonen kjøres først når du kaller på funksjonen

```
1 def funksjon(parameter1, parameter2):
2     parameter_sum = parameter1 + parameter2
3     return parameter_sum
4
5 argument1 = 1
6 argument2 = 2
7
8 funksjon_resultat = funksjon(argument1, argument2)
9 print(funksjon_resultat)
```

```
In [7]: runfile('C:/Users/rawis/OneDrive/Dokumenter/Jobb/Kodeskolen/FU-
kurs oktober 2022/eksempel.py', wdir='C:/Users/rawis/OneDrive/
Dokumenter/Jobb/Kodeskolen/FU-kurs oktober 2022')
3
```

- Noen funksjoner returnerer en eller flere verdier, de må da puttes i en eller flere variabler

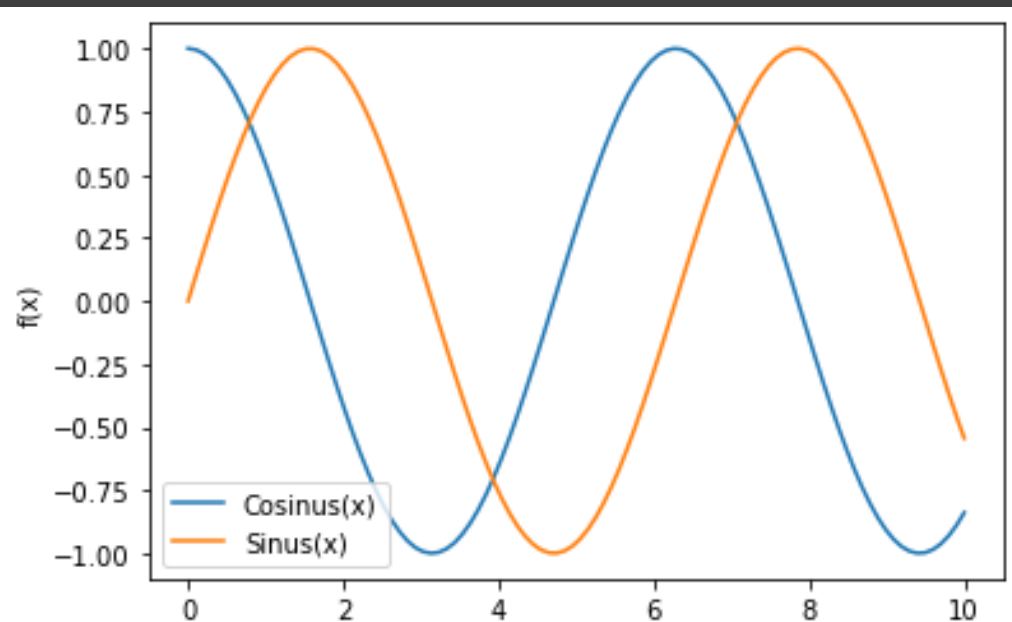
```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 min_array = np.linspace(0, 10, 21)
5 print(min_array)
6 print('-----')
7 min_array_dobbel = min_array*2
8 print(min_array_dobbel)
```

```
In [37]: runfile('C:/Users/rawis/OneDrive/Dokumenter/nanna_klasse.py',
wdir='C:/Users/rawis/OneDrive/Dokumenter')
[ 0.  0.5  1.  1.5  2.  2.5  3.  3.5  4.  4.5  5.  5.5  6.  6.5
 7.  7.5  8.  8.5  9.  9.5 10.]
-----
[ 0.  1.  2.  3.  4.  5.  6.  7.  8.  9. 10. 11. 12. 13. 14. 15. 16. 17.
18. 19. 20.]
```

## Plotting og arrays

- Når vi skal plote data bruker vi ofte arrays
- Arrays er fine å jobbe med, fordi du kan gjøre operasjoner på alle elementene i hele arrayen samtidig
- For å lage arrays må vi importere biblioteket som heter numpy (number python)
- For å plote må vi importere pakken pyplot fra biblioteket matplotlib:

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 x = np.linspace(0, 10, 100)
5 cos_x = np.cos(x)
6 sin_x = np.sin(x)
7
8 plt.plot(x, cos_x, label='Cosinus(x)')
9 plt.plot(x, sin_x, label='Sinus(x)')
10 plt.xlabel('x')
11 plt.ylabel('f(x)')
12 plt.legend()
```



## Plotting og arrays

- Når vi plotter må vi ha to arrays som er like lange, en til x-aksen og en til y-aksen
- Vi kan plotte flere data i samme figur