

Funksjoner

Malin og Bjørn

Hva er en funksjon?

- Avgrenset og navngitt kode som utfører en oppgave
- Kan returnere verdier, eller bare “gjøre ting”
- Kan ta inn verdier (parametre)
- Nyttig for å ikke duplisere kode

```
def <funksjonsnavn>(<parametre>):  
    <innhold>
```

Eksempel:

```
def skriv_noe(noe):  
    print(f"Dü har skrevet {noe}")
```

```
def sirkel_areal(radius):  
    areal = pi * radius**2  
    return areal
```

Bruke funksjoner

- En funksjon gjør ingenting før du ber den om noe
- Vi sier typisk at du “kaller” eller “gjør et kall mot” en funksjon når bruker den

```
def skriv_noe(noe):  
    print(f"Dü har skrevet {noe}")
```

```
skriv_noe("NRK")  
> "Dü har skrevet NRK"
```

```
def sirkel_areal(radius):  
    area = 3.14 * radius * radius  
    return area
```

```
print(sirkel_areal(1))  
> 3.141592653589793
```

Parametre og argumenter

- Du kan definere en funksjon til å ta et vilkårlig antall parametre
- Du må matche hvert parameter med et “argument” når du kaller funksjonen. Et argument er det du kaller funksjonen med.
- Rekkefølgen har betydning

```
def areal_trekant(a, c, theta):  
    return a*c*sin(theta)
```

```
print(areal_trekant(10,100,pi/6)) # Argument er 10, 100 og pi/6, imens parametre er a, c og theta.  
> 499.99999999999994  
print(areal_trekant(pi/6,10,100))  
> -2.6513242969011728
```

Keyword arguments

Du kan også spesifisere hvilket parameter når du kaller funksjonen. Det vil ikke endre utfallet av hva funksjonen gjør. Dette kalles “keyword arguments”.

```
<funksjonsnavn>(<parameter1>=<argument1>, <parameter2>=<argument2>, ... <parameter>=<argument>)
```

Eksempel:

```
print(areal_trekant(theta=pi/6, c = 100, a=10))
```

Default values

Du kan gi et parameter en default-verdi. Dette vil si at dette blir verdien om du ikke spesifiserer den. Du kan velge selv om du vil overskrive verdien

```
def make_tshirt(size, message="Hello World"):  
    print(f"A size {size} tshirt with message \"{message}\")
```

```
make_tshirt('M')
```

```
make_tshirt('M','Goodbye World!')
```

Oppgave 8.5 s. 137

- a) Lag en funksjon som heter `beskriv_by` som tar i mot en by og dens land. Funksjonen skal printe ut en enkel setning, som for eksempel *Reykjavik er i Island*.
- b) Endre til å heller returnere tekststrengen fra funksjonen, og print ut “output” fra funksjonen (fremfor å printe ut i selve funksjonen).

Returverdier

- En funksjon kan returnere en verdi eller et sett med verdier

```
def bygg_tvserie(tittel, antall_episoder, aarstall)
    tv_serie = {'tittel': tittel, 'antall episoder': antall_episoder, 'aarstall': aarstall}
    return tv_serie
```

```
skam_tvserie = bygg_tvserie("skam", 12, 2015)
heimebane_tvserie = bygg_tvserie("heimebane", 10, 2016)
```

```
print(skam_tvserie)
print(heimebane_tvserie)
```


Valgfrie (optional) argumenter

- Du kan definere funksjoner slik at noen parametre har en default-verdi
- Alle valgfrie argumenter må komme til sist i lista over parametre

```
def bygg_tvserie(tittel, antall_episoder, aarstall, land='Norge', orginalserie=''):
    tv_serie = {'tittel': tittel, 'antall episoder': antall_episoder, 'aarstall': aarstall, 'land': land}
    if orginalserie:
        tv_serie['orginalserie'] = orginalserie
    return tv_serie
```

Oppgaver

- a) Lag en liste med minst tre navn. Lag en funksjon som tar inn en string, som printer ut *Hei <navn>!* Kall denne funksjonen på alle elementene i lista.
- b) Lag en ny funksjon som tar inn en liste, og printer ut *Hei <navn>!* For alle elementene i lista.

Lister og funksjoner

- Du kan sende inn lister som argumenter til funksjoner
- Hvis du endrer på en liste inni en funksjon, endrer du på lista også utenfor
- Du kan også sende med en kopi av lista hvis du vil unngå å endre originalen

```
kanaler = ['nrk1', 'nrk2', 'nrk3', 'tv2']
```

```
def fiks_kanaler(kanaler):  
    if "tv2" in kanaler:  
        kanaler.remove('tv2')  
    return kanaler
```

```
fiks_kanaler(kanaler[:])  
print(kanaler)  
fiks_kanaler(kanaler)  
print(kanaler)
```

Kall med vilkårlig antall argumenter

- Du kan be en funksjon ta et vilkårlig antall argumenter
- Dette må defineres sist av parameterne i funksjonen. Det vil si om det er andre ting som må være med til funksjonen, så må disse skrives før.

```
def print_kanaler(*kanaler):  
    for kanal in kanaler:  
        print(kanal)
```

```
print_kanaler('nrk1', 'nrk2', 'nrk3')
```

Oppgaver s. 150

- a) Lag en funksjon for en liste av ingredienser en person vil ha på en sandwich. Det er viktig at denne funksjonen kan ta inn et vilkårlig antall elementer. Kall funksjonen minst tre ganger med ulikt antall elementer.
- b) Lag en funksjon som lagrer informasjon om biler i en dictionary. Funksjonen skal alltid ta i mot en produsent og et modellnavn. Etter det skal den ta i mot et vilkårlig antall keyword-arguments. Kall funksjonen med den nødvendige informasjonen og minst et annet valgfritt parameter (kan være farge). Funksjonskallet kan se slik ut;

```
bil = lag_bil('subaru', 'outback', farge='blå')
```