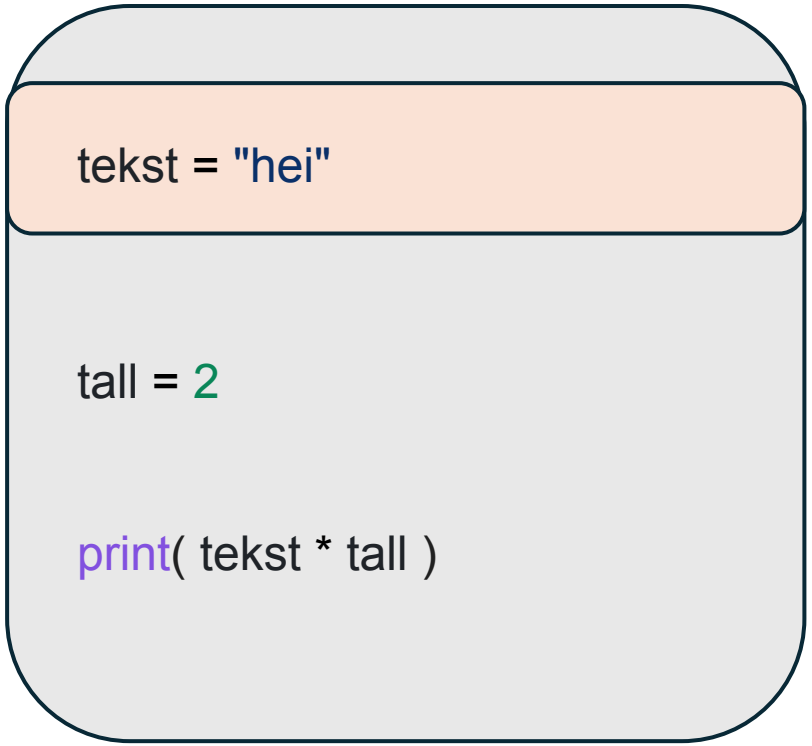
Gruppetime 1 Uke 6

Gruppe 4: Design

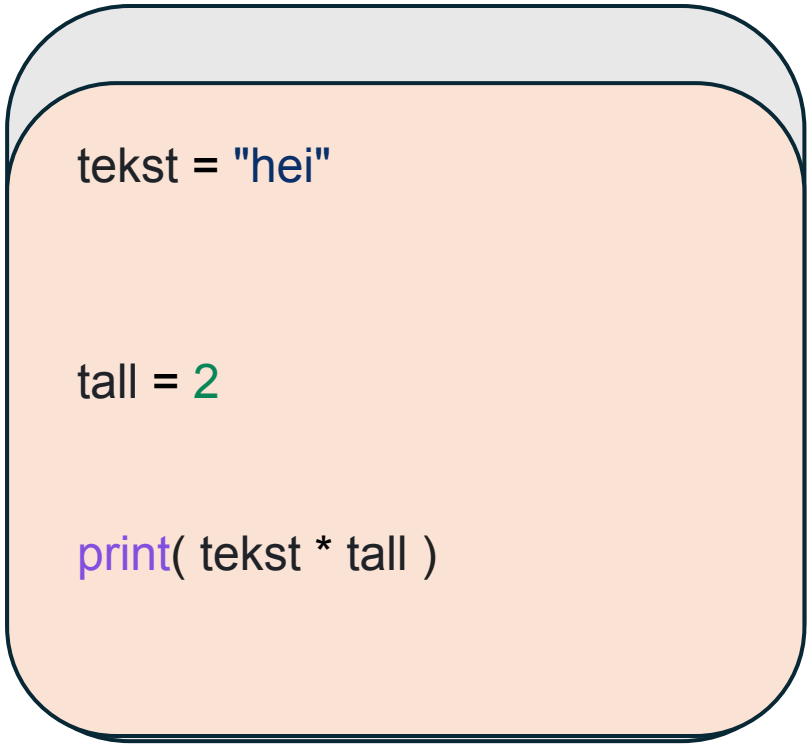
Skop

- Skopet forteller oss hvor variabler er tilgjenglige



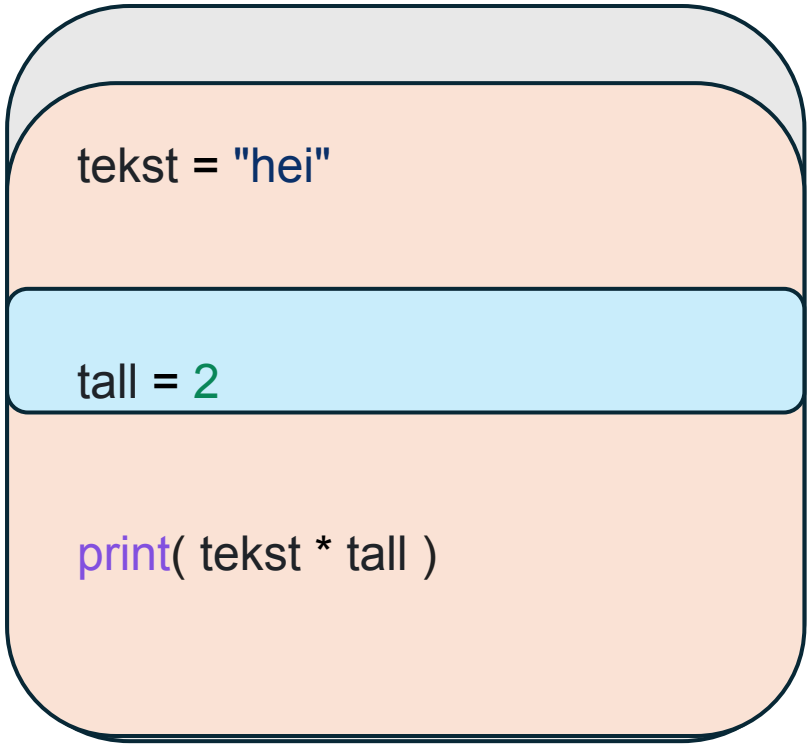
Skop

- Skopet forteller oss hvor variabler er tilgjenglige



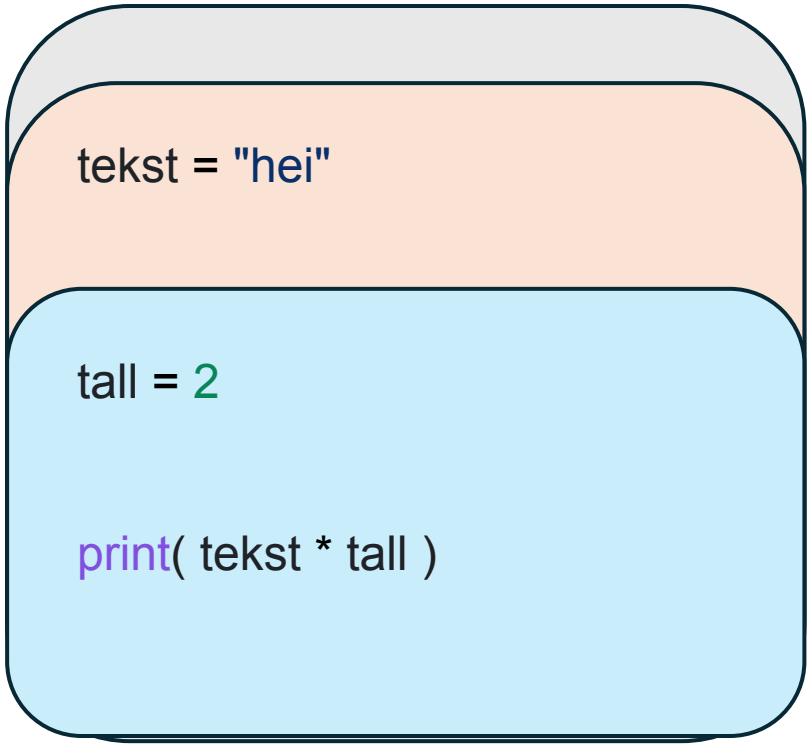
Skop

- Skopet forteller oss hvor variabler er tilgjenglige



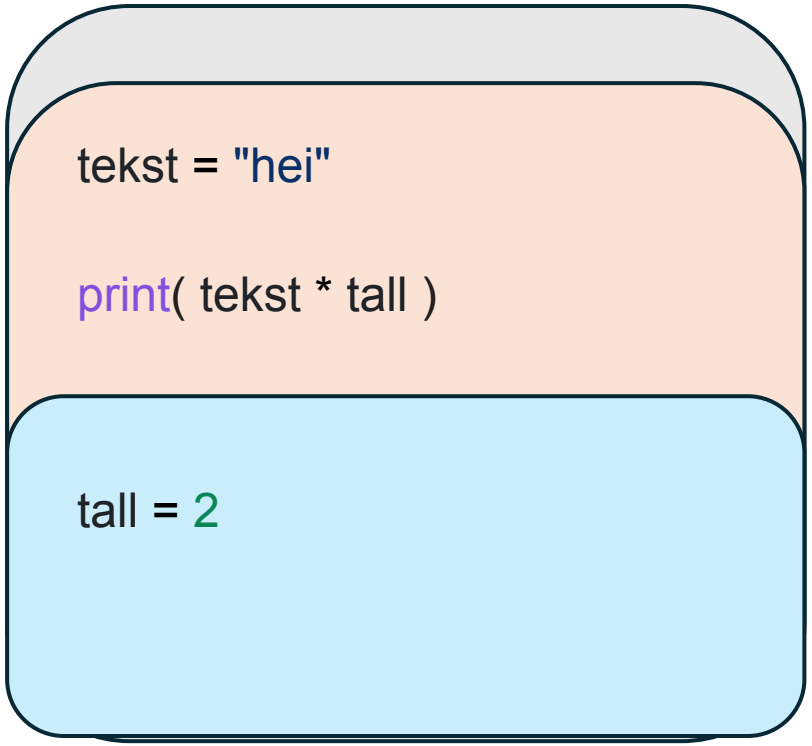
Skop

- Skopet forteller oss hvor variabler er tilgjenglige



Skop

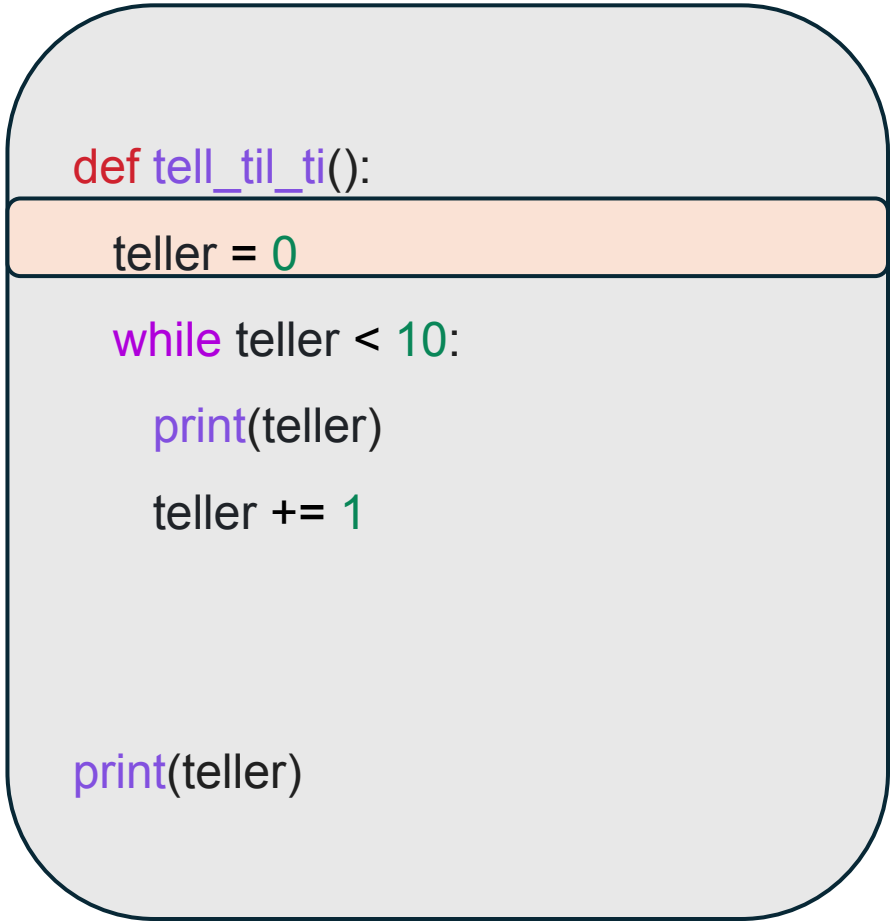
- Skopet forteller oss hvor variabler er tilgjenglige



Skop i subprosesser

Andre skop enn det globale skopet oppstår hovedsakelig gjennom funksjoner/metoder eller klasser

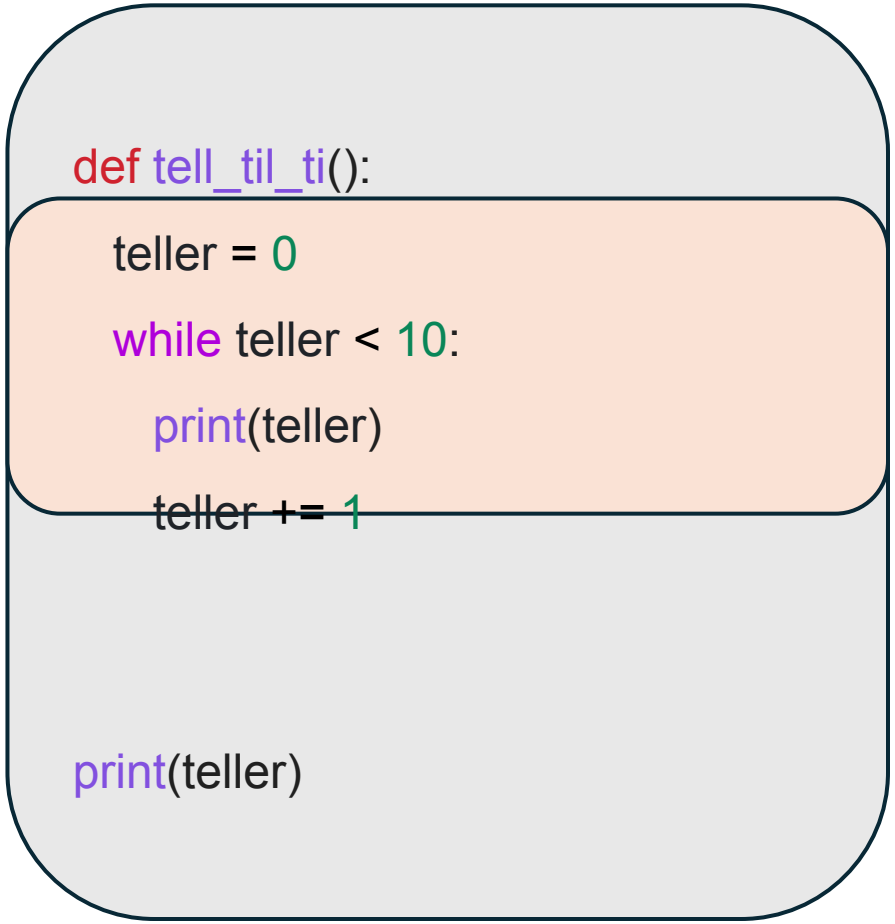
Klasser vil dere ha mer om senere



Skop i subprosesser

Andre skop enn det globale skopet oppstår hovedsakelig gjennom funksjoner/metoder eller klasser

Klasser vil dere ha mer om senere



Skop i subprosesser

Andre skop enn det globale skopet oppstår hovedsakelig gjennom funksjoner/metoder eller klasser

Klasser vil dere ha mer om senere

```
tekst = "Jeg har en bok."

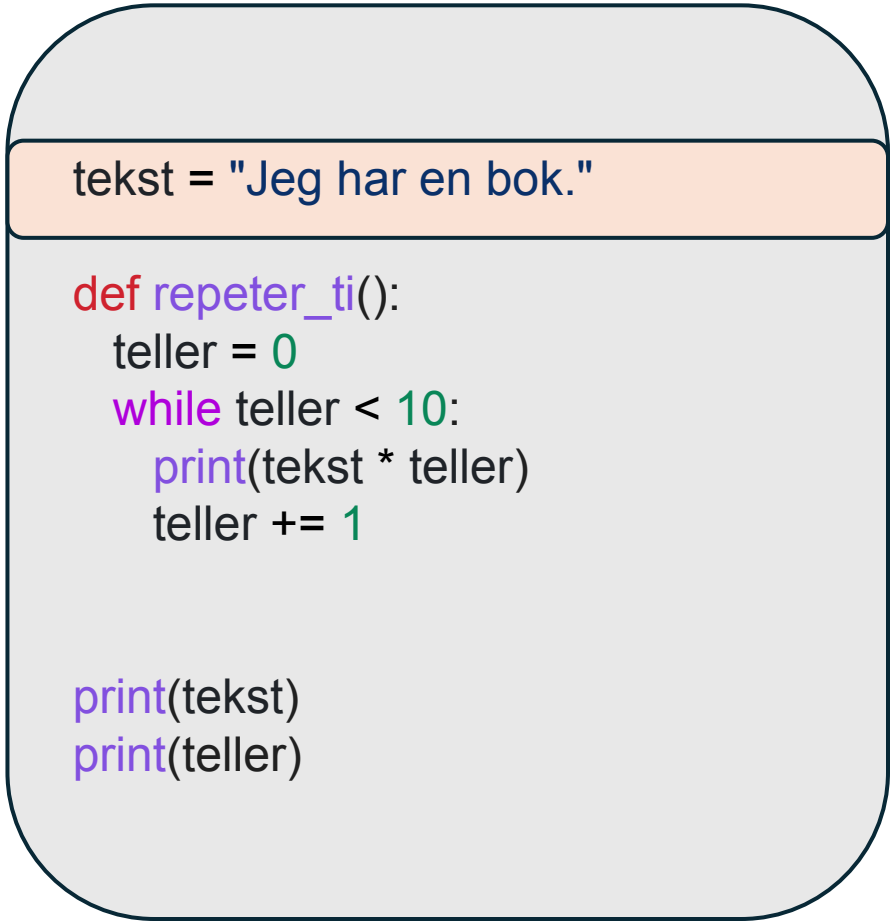
def repeter_ti():
    teller = 0
    while teller < 10:
        print(tekst * teller)
        teller += 1


print(tekst)
print(teller)
```

Skop i subprosesser

Andre skop enn det globale skopet oppstår hovedsakelig gjennom funksjoner/metoder eller klasser

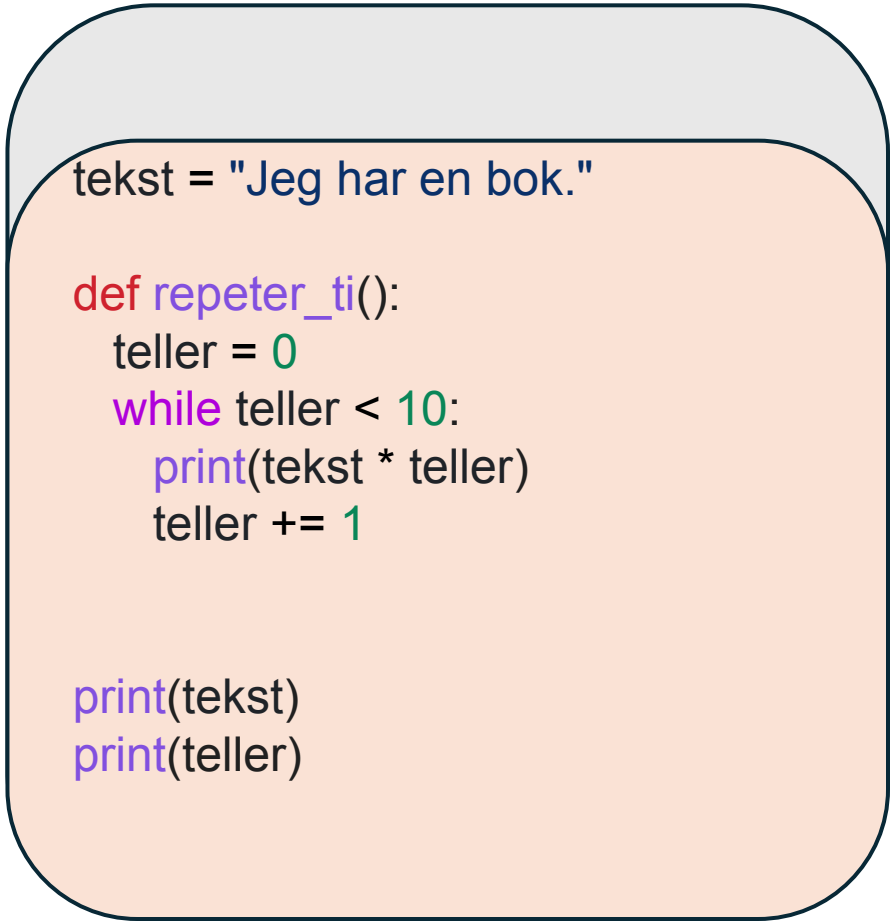
Klasser vil dere ha mer om senere



Skop i subprosesser

Andre skop enn det globale skopet oppstår hovedsakelig gjennom funksjoner/metoder eller klasser

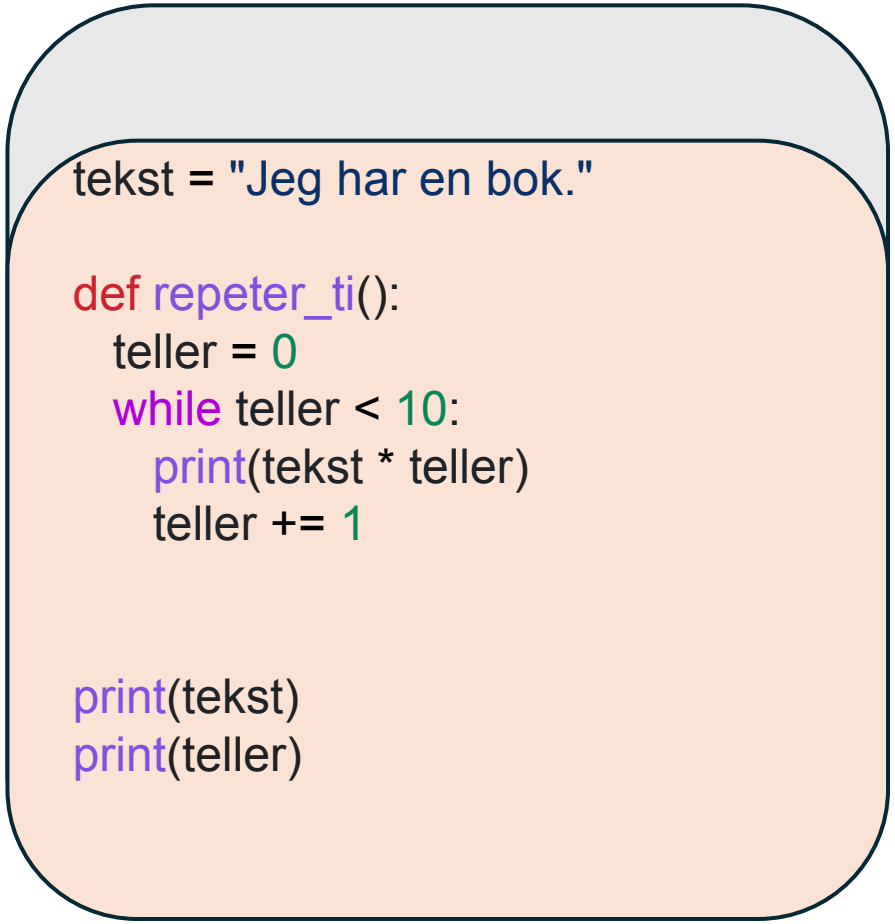
Klasser vil dere ha mer om senere



Skop i subprosesser

Andre skop enn det globale skopet oppstår hovedsakelig gjennom funksjoner/metoder eller klasser

Klasser vil dere ha mer om senere



Skop i subprosesser

- Variabler definert i subprosesser er ikke tilgjengelige utenfor
- Subprosesser *kan* lese av variabler definert utenfor sitt skop
- Men kan ikke skrive til dem!

Hvordan burde man strukturere koden sin?

Prøv å hold globale variabler til en minimum

Som oftest vil man bare trenge noen variabler for veldig spesifikke deler av koden

Lag heller derfor prosedyrer og funksjoner og send variabler gjennom parametere og tilbake med return

Bli vant med å ha et hovedprogram()-prosedyre. Denne prosedyren skal da holde på de “globale” variablene.