

□ Rekne med løkker



PÅ BOKMÅL



LAST NED PDF

Introduksjon

Ein ting menneske ikkje er så flinke til, men som datamaskiner er ekspertar på, er å gjenta noko mange gonger etter kvarandre. I Python kan me gjere det med løkker, og snart skal du sjå at det kan spare oss for mykje tid og skriving.

Hello World!

La oss ta eit døme der me vil ha Python til å seie **Hei!** 100 gonger. Sjølv sagt kan du setje i gang med å skrive:

```
print("Hello World!")
print("Hello World!")
print("Hello World!")
...
```

Som du forstår vil det ta lang tid. Programmerarar vil gjerne løyse oppgåva så enkelt som mogleg, og difor har ein funne opp løkker som kan gjere det for oss! Koden under løyser problemet me skulle løyse på berre to linjer:

```
for i in range(100):
    print(i, "Hello World!")
```

Lurt, ikkje sant?

Range()

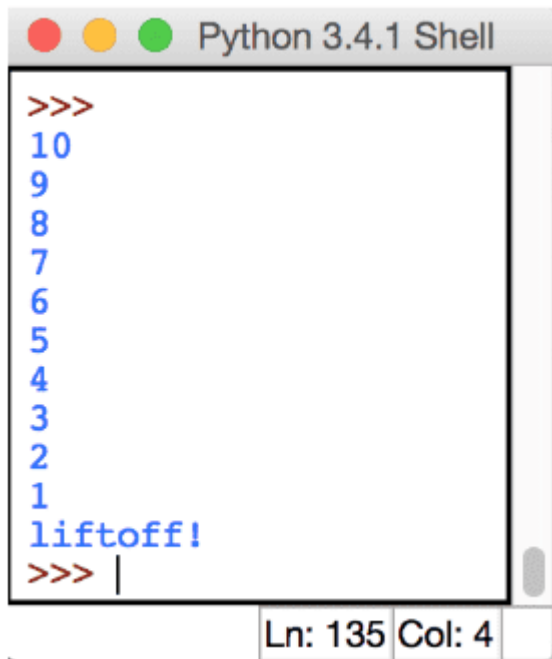
Når me skal jobbe med løkker i Python er `range`-funksjonen svært nyttig. Når du skriv `range(10)` får du ei liste med tala 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. Range kan òg ta fleire parametrar. Skriv du `range(4, 6)`, får du ei liste med tala frå 4 til (men ikkje med) 6, og viss du skriv `range(6, 4, -1)` får du dei same tala i motsett rekkefølge.

Telje til 10

Bruk ei for-løkke til å skrive ut alle tala mellom 0 og 10.

Liftoff

- ☐ Bruk ei for-løkke og `range`-funksjonen til å telje ned frå 10. Når du kjem til 0 skal programmet skrive **liftoff!**



```
>>>
10
9
8
7
6
5
4
3
2
1
liftoff!
>>> |
```

Ln: 135 Col: 4

Summere 100 tal

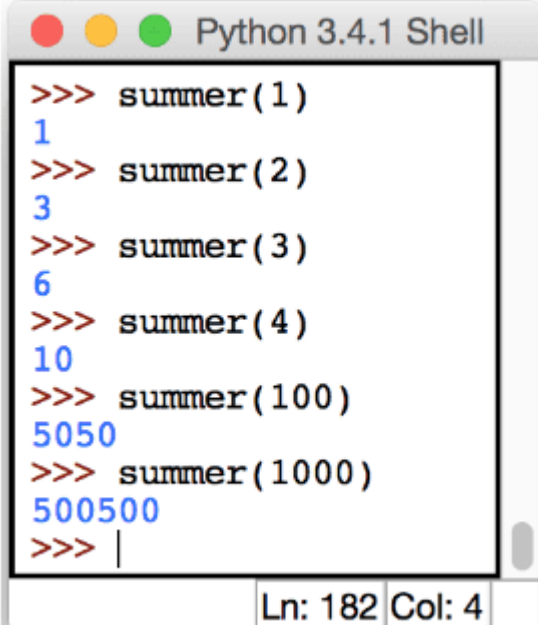
Eit vanleg problem i matematikk er å summere ei følge med tal. Dette er veldig lett når ein kan å programmere! Lag ei for-løkke som går frå 0 til 100, som legg saman alle tala før den skriv ut resultatet. Svaret skal bli 5050.

Summere n tal

- ☐ Lag ein funksjon `summer(n)`, som tek inn ein parameter og returnerer summen av alle tala frå 0 til og med `n`. Definer funksjonen slik:

```
def summer(n):
    sum = 0
    # Din kode
    return sum
```

Når koden din er rett skal den fungere slik som dette:



A screenshot of a Python 3.4.1 Shell window. The window has a title bar with three colored buttons (red, yellow, green) and the text 'Python 3.4.1 Shell'. The main area contains a series of recursive calls to a function named 'summer'. The calls are: 'summer(1)', 'summer(2)', 'summer(3)', 'summer(4)', 'summer(100)', and 'summer(1000)'. Each call is followed by its return value on the next line: 1, 3, 6, 10, 5050, and 500500. The last line shows the prompt '>>>' followed by a vertical bar '|'. At the bottom right of the window, there is a status bar showing 'Ln: 182' and 'Col: 4'.

```
>>> summer(1)
1
>>> summer(2)
3
>>> summer(3)
6
>>> summer(4)
10
>>> summer(100)
5050
>>> summer(1000)
500500
>>> |
```

Ln: 182 Col: 4

Lisens: CC BY-SA 4.0