

# Ordbøker

Skrevet av: Ole Kristian Pedersen, Kodeklubben Trondheim

Kurs: Python

Tema: Tekstbasert

Fag: Programmering, Engelsk

Klassetrinn: 5.-7. klasse, 8.-10. klasse

## Introduksjon

Dette er en kort oppgave som viser hvordan man bruker *ordbøker*.

## Nøkler og verdier

En ordbok (*dictionary* på engelsk) brukes for å lagre *nøkkel/verdi*-par. Tenk deg at du skal ha en norsk-engelsk ordbok. Da vil *nøkkelen* være ordet du slår opp på, for eksempel det norske ordet. *Verdien* vil være det engelske ordet. F.eks. nøkkelen "ost" og verdien "cheese". I Python skrives ordbøker med `{}` slik som dette:

```
>>> d = {'ost': 'cheese', 'brød': 'bread'}
>>> d
{'ost': 'cheese', 'brød': 'bread'}
```

I eksempelet over lagde vi en norsk-engelsk ordbok til variabelen `d`. Nøkkel og verdi har et kolon `:` mellom seg, og `'nøkkel': 'verdi'` -parene skilles med `,`. For å slå opp på en nøkkel, bruker vi `[nøkkel]`, slik som dette:

```
>>> d['ost']
'cheese'
```

Vi kan bruke den samme skrivemåten for å lage nye *nøkkel/verdi*-par eller endre verdien knyttet til en nøkkel:

```
>>> d['farge'] = 'colour'
```

```
# legger til en ny verdi
```

```
>>> d
```

```
{'ost': 'cheese', 'brød': 'bread', 'farge': 'colour'}
```

```
>>> d['farge'] = 'color'
```

```
# endrer verdien
```

```
>>> d
```

```
{'ost': 'cheese', 'brød': 'bread', 'farge': 'color'}
```

En tom ordbok opprettes slik:

```
>>> d = {}
```

```
>>> d
```

```
{}
```

**Merk:** Bare tekst og tall kan brukes som nøkler, men verdiene kan være hva som helst: tekstar, tall, lister, ordbøker, funksjoner, osv.

Vi skal nå skrive et program som lar en bruker lage en ordbok. Programmet skal ta i mot 3 nøkkel/verdi-par, deretter be om en nøkkel å slå opp på og til slutt vise hvilken verdi som tilhører nøkkelen. Det skal fungere slik:

```
Skriv inn en nøkkel: ost
Skriv inn en verdi: cheese
Skriv inn en nøkkel: brød
Skriv inn en verdi: bread
Skriv inn en nøkkel: farge
Skriv inn en verdi: color
Hvilken nøkkel vil du slå opp på? brød
Tilhørende verdi er bread
```

Dette må du gjøre:

☐

Lag ei tom ordbok.

☐

Bruk ei løkke for å hente inn 3 nøkkel/verdi-par.

- ☐ Lagre nøkkel/verdi-parene i ordboka.
- ☐ Spør om en nøkkel.
- ☐ Skriv ut verdien som hører til nøkkelen.

## Gå igjennom ordbøker

Du kan bruke en løkke til å hente ut nøklene til en ordbok:

```
>>> d = {'brød': 3, 'ost': 1}
>>> for key in d:
...     print("Nøkkel:", key)
...     print("Verdi:", d[key])
...
Nøkkel: ost
Verdi: 1
Nøkkel: brød
Verdi: 3
```

Dersom du bare trenger verdiene kan du bruke `d.values()`:

```
>>> for val in d.values():
...     print("Verdi:", val)
...
Verdi: 1
Verdi: 3
```

Dersom du ønsker få tilgang til både nøkkel og verdi kan du bruke `d.items()`:

```
>>> for key, value in d.items():
...     print(key, value)
...
ost 1
brød 3
```

Vi skal nå lage et handleliste-program som lar brukeren velge hva og hvor mye som skal være på handlelista. Programmet skal se slik ut:

```
Skriv en gjenstand: brød
Hvor mange? 2
Skriv en gjenstand: tomat
Hvor mange? 5
Skriv en gjenstand:
Her er handlelista:
2 brød
5 tomat
```

Dette må du gjøre:

- ☐ Ta imot input for gjenstand.
- ☐ Så lenge gjenstanden ikke er en tom tekst "" :
  - ☐ Be om antall.
  - ☐ Lagre til en ordliste.
  - ☐ Bruk gjenstanden som nøkkel og antallet som verdi.
- ☐ Skriv ut handlelista.

**Hint:** Gå gjennom nøklene.

Lisens: CC BY-SA 4.0 (<http://creativecommons.org/licenses/by-sa/4.0/deed>)