

Modul 3 - Del 2 - Hovedpoeng

Hovedpoeng

- En gitt samling av data kan lagres på ulike måter i Python ved hjelp av ulike datastrukturer. Ta deg tid til å velge den beste datatypen for dine spesifikke data og oppgaver.
- Det kan være lurt å konvertere en variabel til en annen type mellom to oppgaver.
- Man bruker en liste (dvs en [sekvens](#) som er [foranderlig](#)):
 - Når man trenger en ordnet samling av elementer.
 - Når man potensielt ønsker å ha duplikater.
 - Når man vil kunne legge til, fjerne eller endre elementer etter opprettelsen.
- Man bruker en [ordbok](#):
 - Når man har behov for å lagre data i form av nøkkel-verdi-par.
 - For å få en rask tilgang til data basert på unike nøkler.
- Man bruker et [sett](#):
 - Når man trenger en samling av unike elementer uten en bestemt rekkefølge.
 - For å eliminere duplikater fra en annen samling.
 - For å få tilgang til matematiske settoperasjoner, som union, snitt og differanse.

Jukseark

Datatyper

Man kan sjekke typen til et objekt med funksjonen `type()`.

Navn (norsk)	Navn (engelsk)	Eksempel	Konvertere til
Ordbok	Dictionary	<code>{"navn": "Odin", "alder": 32}</code>	<code>dict()</code>
Sett	Set	<code>{3, -1, 0}</code>	<code>set()</code>

`None` er et spesielt objekt som representerer fraværet av en verdi eller en null-verdi. `None` er et unikt objekt av typen `NoneType`.

Ordbøker

```
tom_ordbok = {}

hovedsteder = {
    "Norge": "Oslo",          # "Norge" er *nøkkelen* og "Oslo" *verdien*
    "Danmark": "København",
    "Sverige": "Stockholm"
}

# Får tilgang til en verdi ved å bruke dens tilsvarende nøkkel.
print(hovedsteder["Danmark"]) # "København"

# Oppdater en verdi
hovedsteder["Norge"] = "Bergen"
```

```
# Opprett et nytt nøkkel-verdi par  
hovedsteder["Frankrike"] = "Paris"
```

```
# Skriver ut nøkler og verdier  
for (land, hovedstad) in hovedsteder.items():  
    print(f'Nøkkel: {land} | Verdi: {hovedstad}')
```