

Sammenligning av strenger ¶

Læringsmål:

- Betingelser
- Kodeforståelse

Starting Out with Python:

- Kap. 3.2-3.3

Generelt om sammenligning av strenger

Trykk på pilen til venstre for å lese

Når man sammenligner to strenger i Python vil strengene sammenlignes karakter for karakter. F.eks. vil et program som sammenligner strengene "Ola" og "Ole" og sjekker om de er like, først sammenligne 'O' mot 'O', så 'l' mot 'l', og til slutt 'a' mot 'e'. Det er først ved sammenligningen av 'a' mot 'e' at det vil bli returnert `False`.

In []:

```
a = "Ola"
b = "Ole"

print("Sammenligner", a, "og", b)
if a == b:
    print("Navnene er like")
else:
    print("Navnene er IKKE like")
```

Om man derimot sammenligner navnene Markus og Marcus, vil testen av betingelsen returnere `False` når 'k' blir sammenlignet med 'c'. Det betyr at 'u' mot 'u' og 's' mot 's' ikke blir sjekket.

In []:

```
a = "Markus"
b = "Marcus"

print("Sammenligner", a, "og", b)
if a == b:
    print("Navnene er like")
else:
    print("Navnene er IKKE like")
```

Prøv gjerne ut koden over med andre verdier selv. Andre eksempler på kjøring kan være:

```
# a = "Ann", b = "Anne"
Sammenligner Ann og Anne
Navnene er IKKE like

# a = "Anders", b = "Anders"
Sammenligner Anders og Anders
Navnene er like

# a = "Anders", b = "anders"
Sammenligner Anders og anders
Navnene er IKKE like
```

Som du ser i siste kjøring, skiller Python mellom store og små bokstaver, og en stor A er ikke det samme som en liten a. Dette kommer av at tegnene i strengene lagres som tall, og det er disse tallene som sammenlignes. I ASCII er karakterene A til Z representert med tallene 65 til 90, og karakterene a til z er representert med tallene 97 til 122.

a)

Du skal nå lage et program som sammenligner to matvarer, og sjekker om de er like. Det skal ikke spille noen rolle om matvarenavnene som sendes inn har store eller små bokstaver, dvs. at "druer" og "DrUer" skal regnes som like matvarer (se Hint). Det trengs ikke å ta hensyn til entall og flertall, dvs. at "druer" og "drue" regnes som to forskjellige matvarer.

Eksempel på kjøring:

```
Sammenligner druer og DrUer
Det er samme matvare

Sammenligner druer og drue
Dette er to forskjellige matvarer

Sammenligner TomAT og ToMat
Det er samme matvare

Sammenligner tomat og potet
Dette er to forskjellige matvarer
```

Skriv din kode her:

In [3]:

```
a = input('Matvare: ')
b = input('Matvare 2: ')

print('Sammenlikner', a, 'og', b)
if a.lower()==b.lower():
    print('Dette er samme matvare')
else:
    print('Dette er to forskjellige matvarer')
```

Matvare: epler
Matvare 2: pærer
Sammenlikner epler og pærer
Dette er to forskjellige matvarer

Hint

`str.lower()` returnerer en kopi av strengen bestående av små bokstaver.

Dvs., om du har en variabel `a = "DrUer"`, vil `a.lower()` gi strengen `"druer"`. Men husk, `a.lower()` endrer ikke på variabelen `a`, så man må enten lage en ny variabel `b = a.lower()` og bruke variabelen `b` i sjekken, eller så kan man bruke `a.lower()` direkte i sjekken i stedet for `a`.

Eksempel som viser bruk av `str.lower()`:

In []:

```
a = "DrUer"

print(a)          #Output: "DrUer"
a.lower()         #retunerer strengen "druer", men endrer ikke på variabelen a
print(a)          #Output: "DrUer"
print(a.lower())  #Output: "druer"
```

Fremgangsmåte

1. Lag to variabler (`a` og `b`) som representerer hver sin streng.
2. Skriv hvilke variabler du sammenligner ut til skjerm.
3. Bruk en if-else-setning med betingelsen `a.lower() == b.lower()` for å sammenligne strengene, og skriv ut om det er samme vare eller to forskjellige varer til skjerm.

b)

I denne oppgaven skal du lage et program som tar inn to navn og sorterer dem i alfabetisk rekkefølge. Du trenger ikke ta hensyn til små og store bokstaver i denne oppgaven.

Eksempel på kjøring:

```
Første navn: Ole
Andre navn: Ola
Under følger navnene i alfabetisk rekkefølge:
Ola
Ole

Første navn: Bob Bernt
Andre navn: Bob Arne
Under følger navnene i alfabetisk rekkefølge:
Bob Arne
Bob Bernt
```

Skriv din kode her:

In [4]:

```
a = input('Første navn: ')
b = input('Andre navn: ')

print('Under følger navnene i alfabetisk rekkefølge: ')
if a < b:
    print(a)
    print(b)
else:
    print(b)
    print(a)
```

```
Første navn: Ola
Andre navn: Ole
Under følger navnene i alfabetisk rekkefølge:
Ola
Ole
```

Hint (trykk pil til venstre)

Om man ønsker å sortere navnene alfabetisk, kan man bruke `<` eller `>`. For navnene Ola og Ole vil Ole bli sett på som større enn Ola, siden `'e'` er representert av et større tall enn `'a'`.

Bruk av æ, ø og å i oppgave b)

I Python3 brukes Unicode-tabellen til representasjon av bokstaver, og mens verdiene for bokstavene a-z er de samme som i ASCII-tabellen, er det litt annerledes for æ, ø og å. Disse kommer i en litt annen rekkefølge enn vi er vant med og har følgende verdier:

'å' - 229

'Å' - 197

'æ' - 230

'Æ' - 198

'ø' - 248

'Ø' - 216

Dette kan medføre at koden i oppgave b) ikke alltid vil gi korrekt svar, siden 'å' kommer før 'æ' og 'ø' i ASCII-tabellen, men etter 'æ' og 'ø' i det norske alfabetet.

(Frivillig) Om du ønsker kan du endre på koden din slik at dette blir tatt hensyn til. Her må du nok benytte deg av stoff som ikke er gjennomgått enda.

c)

Hva vil kodesnuttene under skrive ut til skjerm?

```
#Kodesnutt 1:
if 'k' < 'b':
    print('k er mindre enn b')
else:
    print('k er større enn b')
```

```
#Kodesnutt 2:
ny = "New York"
la = "Los Angeles"
```

```
if ny < la:
    print(ny)
    print(la)
else:
    print(la)
    print(ny)
```

```
#Kodesnutt 3:
d1 = "DRuEr"
d2 = "drUer"
if d1.lower() < d2.lower():
    print(d1)
    print(d2)
else:
    print(d2.lower())
```

Svar for kodesnutt 1 (Trykk på teksten under for å skrive):

Svar 1: k har et større tall enn b, derfor skriver programmet ut: k er større enn b

Svar for Kodesnutt 2:

Svar 2: N har et større tall enn L, derfor skriver programmet ut: New York Los Angeles

Svar for kodesnutt 3:

Svar 3: De to uttrykkene på sidene av likhetstegnet er like, da stemmer ikke ulikheten, og koden printer "else-uttrykket": druer