

## Løsningsforslag til oppgaver uke 3

### Oppgave 1

Hvor mange elementer er det i listene?

```
liste = [0]
liste1 = [1, 2, 3]
liste2 = [3, 3, 5, 7]
liste3 = ["A", "BC", "D", "E", "F"]
liste4 = ["mange elementer i denne listen"]
liste5 = []
```

Løsningsforslag:

```
liste = [0] #1
liste1 = [1, 2, 3] #3
liste2 = [3, 3, 5, 7] #4
liste3 = ["A", "BC", "D", "E", "F"] #5
liste4 = ["mange elementer i denne listen"] #1
liste5 = [] #0/tom
```

En streng er bare ett element, uansett hvor lang den er. Liste5 er tom, den finnes, så vi kan legge elementer i den, men har ingen elementer i seg ennå.

### Oppgave 2

Ta utgangspunkt i listene over.  
2.1 Hva ligger på indexene?

```
liste3 [4] # =
liste1 [0] # =
liste4 [1] # =
liste3 [5] # =
```

Løsningsforslag:

```
liste3 [4] # = "F"
liste1 [0] # = 1
liste4 [1] # = "mange elementer i denne listen"
liste3 [5] # = ingenting, det er kun 5 elementer, på indeksene 0-4.
```

Et forsøk på å nå en indeks som er utenfor ei liste, gir en *IndexError*. Her er hva som skjer hvis vi prøver å finne liste3[5]:

```
>>> liste3 = ["A","BC","D","E","F"]
>>> liste3[5]
Traceback (most recent call last):
  File "<pyshell#1>", line 1, in <module>
    liste3[5]
IndexError: list index out of range
>>>
```

2.2 På hvilken index ligger:

```
5      (i liste2) # liste2[2]
"BC"   (i liste3) # liste3[1]
1      (i liste1) # liste1[0]
"E"    (i liste3) # liste3[3]
```

2.3 Tegn opp hvordan liste3 ser ut.

index	element
0	"A"
1	"BC"
2	"D"
3	"E"
4	"F"

### Oppgave 3

Skriv en liste "navn" hvor du legger inn 3 navn; "Ola", "Martin" og "Selma".  
Legg deretter til "Kari" nytt navn bakerst i listen.

Tegn opp hvordan listen ser ut nå.

Skriv ut Martin sitt navn ved hjelp av indeksen det ligger på.

Legg til to til navn forrest i listen, navnene velger du selv.

*NB! Husk at når man legger noe inn på en indeks som er i bruk flyttes det som ligger der og alle etterfølgende elementer en indeks opp..*

Tegn opp hvordan listen ser ut nå.

Fjern "Martin" fra listen.

Skriv ut alle navnene på listen.

*Bonus: hva blir utskriften?*

### Løsningsforslag:

```
navn = ["Ola", "Martin", "Selma"]
```

```
navn.append("Kari")
```

0	"Ola"
1	"Martin"
2	"Selma"
3	"Kari"

```
print(navn[1])
```

```
navn.insert(0, "Lise")  
navn.insert(0, "Kai")
```

0	"Kai"
1	"Lise"
2	"Ola"
3	"Martin"
4	"Selma"
5	"Kari"

```
navn.remove("Martin") eller navn.pop(3)
```

```
print("Alle navn: ", navn)
```

### **Utskriften:**

```
> Alle navn: ["Kai", "Lise", "Ola", "Selma", "Kari"]
```

## **Oppgave 4**

Gitt at vi har de tre listene

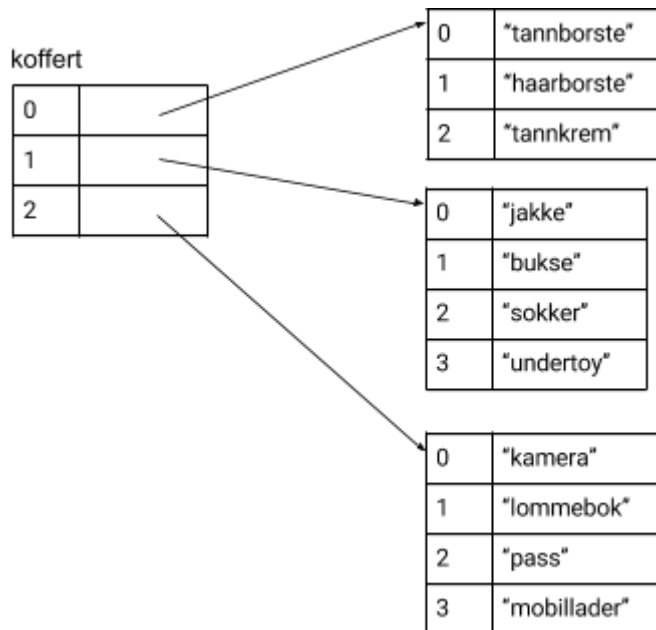
```
toalettsaker = ["tannborste", "haarborste", "tannkrem"]  
klaer = ["jakke", "bukse", "sokker", "undertoy"]  
div = ["kamera", "lommebok", "pass", "mobillader"]
```

Vi legger dem inn i listen "koffert", hvordan vil utskriften av denne se ut? Og hvordan kan vi tegne opp denne (prøv deg fram)?

```
koffert = [toalettsaker, klaer, div]  
print(koffert)
```

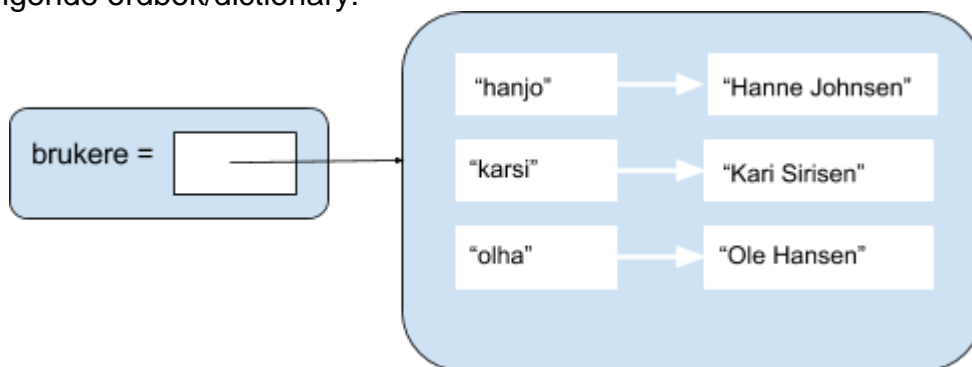
## Løsningsforslag:

[ ['tannborste', 'haarborste', 'tannkrem'], ['jakke', 'bukse', 'sokker', 'undertoy'], ['kamera', 'lommebok', 'pass', 'mobillader'] ]



## Oppgave 5

Gitt følgende ordbok/dictionary:



5.1 Definer en ordbok/dictionary som heter brukere, hvor du legger inn følgende par av nøkler og verdier (brukernavn som nøkler og navn som verdi).

5.2 karsi har registrert seg med feil navn, mellomnavnet hennes mangler, endre navnet hennes til "Kari Marie Sirisen".

5.3 register deg selv som bruker med et brukernavn og navn.

5.4 Skriv ut hanjo sitt navn.

5.4 Fjern Hanne Johansen fra ordboken.

## Løsningsforslag:

### 5.1

```
brukere = {"hanjo" : "Hanne Johansen", "karsi" : "Kari Sirisen", "olha" : "Ole Hansen"}
```

### 5.2

```
brukere["karsi"] = "Kari Marie Sirisen"
```

### 5.3

```
brukere["kristijb"] = "Kristine Berge"
```

### 5.4

```
print (brukere["hanjo"])
```

### 5.5

```
brukere.pop("hanjo")
```

## Oppgave 6

Her er et eksempel på en uleselig/uryddig kode, hva slags endringer ville du gjort med dem?

```
def f1() :  
    i = float(input("input"))  
    print(i*i)  
  
def f2() :  
    i = input()  
    print("velkommen til in1000 ", i)  
  
def f3(    ) :  
    print ( "Hei!" )  
  
f3()  
f2()  
f1()
```

## Løsningsforslag:

Navna på variablene og prosedyrene er lite beskrivende, noe som ikke er bra. Funksjonene burde ha navn som sier noe om hva de gjør. En mulighet er å kalle f1() for *multipliser*, f2() for *velkommen*, og f3() for *hei*, eller noe lignende.

I både f1() og f2() blir det gitt lite informasjon til brukeren. Ved å bare skrive "input" vet ikke brukeren hva slags input det spørres etter. Tall? Streng? Osv. I f1() burde det vært spurt om et tall, i f2() burde det vært spurt om et navn.

I f3() er det mange unødvendige mellomrom som ikke synes i utskriften.