# Velkommen til gruppetime i IN1000 ☺

8. september 2021

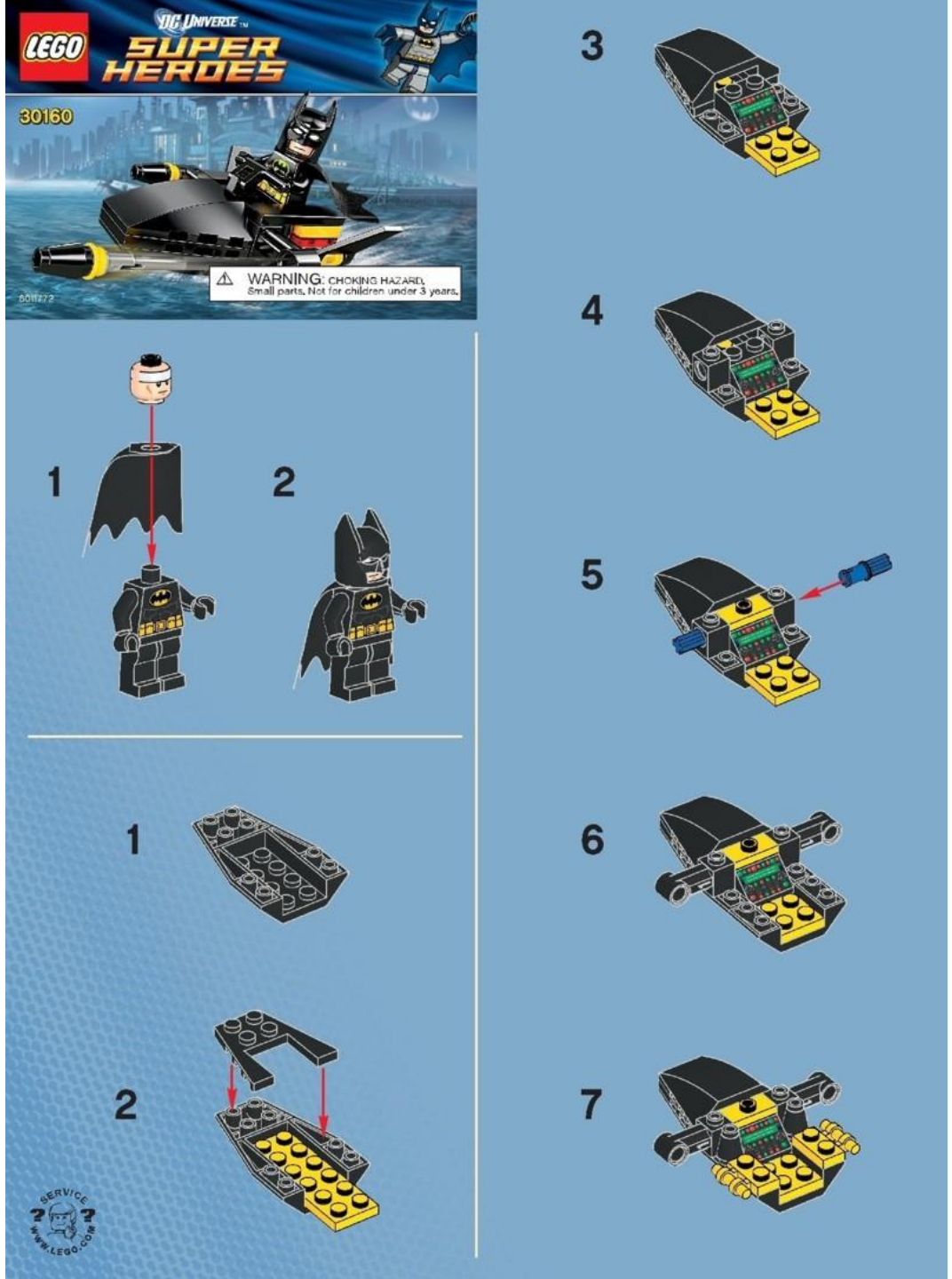
Jessie Yue Guan

# Planen for i dag

- Prosedyrer/Funksjoner
- Parametere/Argumenter
- Returverdier

- Lister
- Mengder
- Ordbøker
- Kahoot! :D

# Prosedyrer/ Funksjoner



# Prosedyrer/Funksjoner

- Tenk på variabler som bokser som ER noe, altså de inneholder en verdi
  - En verdi kan være en tekst, et heltall, et desimaltall, eller en boolsk verdi

---

- Tenk på prosedyrer/funksjoner som bokser som GJØR noe, altså de inneholder en kodeblokk
  - En kodeblokk kan (men må ikke) inneholde if-setninger, tilordning av variabler, osv.

# Prosedyrer/Funksjoner

- Prosedyrer/Funksjoner kan ta imot inndata og/eller gi utdata
  - Inndata = Du forteller boksen hvilke ting det skal utføre operasjoner på
  - Utdata = Du forteller boksen hvilke ting operasjonene skal resultere i

- Må IKKE forveksles med input() og output eller print()
- Lignende konsepter, men ikke det samme

- Inndata for prosedyrer kalles for argumenter/parametere
- Utdata for prosedyrer kalles for returverdier

# Parametere/Argumenter

- En prosedyre kan ha ingen parametere…
- **def hilsen():**
- **    print("Hei Espen Askeladd! :D")**
- … og ingen argumenter…
- **hilsen()**
- Da blir den mindre fleksibel, og kan brukes igjen, men uten variasjoner

- Eller så kan den ha parametere…
- **def personlig_hilsen(navn):**
- **    print("Hei " + navn + "! :D")**
- … og argumenter…
- **personlig_hilsen("Espen Askeladd")**
- Da blir den mer fleksibel, og kan brukes igjen, med variasjoner

# Parametere/Argumenter

- Parametere er det du skriver mellom parentesene når du definerer prosedyren
- **def skriv_hilsen(navn):** ← Parameter: navn
  - **print("Hei, " + navn + "!")**
- Argumenter er det du skriver mellom parentesene når du kaller på prosedyren
- **skriv_hilsen("Kong Harald")** ← Argument: "Kong Harald"
- Det MÅ alltid være like mange parametere som argumenter og de må ha samme rekkefølge!!!

# Returverdier

- En prosedyre kan ha ingen returverdi…
- **def multipliser():**
- **    print(2 * 2)**
- **multipliser()**
- Da blir den mindre fleksibel, og kan ikke brukes av andre prosedyrer/funksjoner

- Eller så kan den ha returverdi…
- **def multipliser():**
- **    return 2 * 2**
- **multipliser(multipliser())**
- Da blir den mer fleksibel, og kan brukes av andre prosedyrer/funksjoner

# Prosedyrer/Funksjoner

- Vi kan enten ha null eller én eller flere parametere/argumenter
- Men vi kan kun ha enten null (prosedyre) eller én (funksjon) returverdi
- Metoder er prosedyrer/funksjoner som er knyttet til en klasse (fremtidig pensum)
- Subrutiner er en fellesbetegnelse for alle prosedyrer, funksjoner, og metoder

# Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! ☺

# Oppgave 1

- Lag en prosedyre som finner ut om brukeren er mer enn 120 cm høy
- Lag en prosedyre som finner ut om brukeren er mer enn 10 år gammel
- Finn ut om brukeren kan ta ThunderCoaster på Tusenfryd ved å bruke prosedyrene ovenfor

# Oppgave 2

- Gjør om prosedyren som finner ut om brukeren er mer enn 120 cm høy til en funksjon
- Gjør om prosedyren som finner ut om brukeren er mer enn 10 år gammel til en funksjon
- Finn ut om brukeren kan ta ThunderCoaster på Tusenfryd ved å bruke funksjonene ovenfor

# Kolleksjoner

# Hvordan lagre flere verdier sammen?

- Hittil har vi for det meste bare jobbet med primitiver (str, int, float, bool)
- Disse kan bare inneholde/lagre en verdi om gangen
- Men nå skal vi begynne å se litt på kolleksjoner (lister, mengder, ordbøker)
- Disse kan inneholde/lagre flere verdier om gangen

# Hvorfor lagre flere verdier sammen?

- **alder1 = 23**
- **alder2 = 19**
- **alder3 = 27**
- **…**

- Dette blir veldig kronglete etter hvert…

- **aldre = [23, 19, 27, …]**

- Dette blir mye mer ryddig og organisert

# Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! ☺

# Lister

# Lister

- En samling av verdier som har en rekkefølge og som kan inneholde duplikater
- For eksempel:
  - **minListe = ["C", "A", "C", "B"]**
- Rekkefølgen i denne lista blir da:
  - "C" på indeks 0
  - "A" på indeks 1
  - "C" på indeks 2
  - "B" på indeks 3
- Duplikater i denne lista blir da:
  - "C" som forekommer 2 ganger

# Hvordan lage en liste

- Vi vet hvor lang lista skal være og vi vet hvilke verdier den skal inneholde:
  - **minListe = [4, 2, 6, 5, 5]**

---

- Vi vet hvor lang lista skal være men ikke hvilke verdier den skal inneholde:
  - **minListe = [0] * 5**

---

- Vi vet ikke hvor lang lista skal være og heller ikke hvilke verdier den skal inneholde:
  - **minListe = []**

# Hvordan legge til en verdi i en liste

- Legge til en verdi i en liste:
  - **minListe = [2, 4, 6]**
  - **minListe.append(8)**
  - **print(minListe)**
- Vil printe [2, 4, 6, 8]

# Hvordan aksessere en verdi i en liste

- Aksessere en verdi i en liste:
  - **spillere = ["Ola", "Ida", "Per", "Hanne"]**
  - **spiller = spillere[1]**
  - **print("Det er nå " + spiller + " sin tur")**
  - **spiller += 1**
- Vil printe "Det er nå Ida sin tur"

# Hvordan endre en verdi i en liste

- Endre en verdi i en liste:
  - **aldre = [25, 20, 32, 21, 29]**
  - **aldre[4] = 30**
- Legg merke til at vi ikke kan skrive følgende:
  - **aldre[5] = 30**
- Vil resultere i en IndexOutOfRangeError

# Hvordan slette en verdi i en liste

- Sletting basert på indeks:
  - **minListe = [1, 3, 5, 7, 9]**
  - **del minListe[2]**
  - **print(minListe)**
- Vil printe [1, 3, 7, 9]

- Sletting basert på verdi:
  - **minListe = [1, 3, 5, 7, 9]**
  - **minListe.remove(7)**
  - **print(minListe)**
- Vil printe [1, 3, 5, 9]

# Hvordan sjekke om en verdi finnes i en liste

- **byer = ["Oslo", "Bergen", "Tromsø"]**

- Sjekke om en verdi finnes eller ikke:
  - **if "Bergen" in byer:**
    - **print("Bergen er registrert!")**
  - **else:**
    - **print("Bergen er ikke registrert!")**

- Legge til en verdi hvis den ikke finnes:
  - **if "Trondheim" not in byer:**
    - **byer.append("Trondheim")**

# Hvordan sjekke hvor ofte en verdi forekommer i en liste

- **kort = ["5", "J", "5", "5", "J"]**

- Telle antall forekomster av en verdi:
  - **print(kort.count("5"))**

- Gjøre noe basert på antall forekomster:
  - **if kort.count("J") == 2:**
    - **print("To par!")**

# Hvordan sortere en liste

- Sortering av tall:
  - **tall = [5, 1, 4, 2, 3]**
  - **tall.sort()**
  - **print(tall)**
- Vil printe [1, 2, 3, 4, 5]

- Sortering av tekst:
  - **bokstaver = ["a", "s", "d", "w"]**
  - **bokstaver.sort()**
  - **print(bokstaver)**
- Vil printe ["a", "d", "s", "w"]

# Hvordan konkatenere lister

- To variabler:
  - **liste1 = [1, 2, 3]**
  - **liste2 = [4, 5, 6]**
  - **sammen = liste1 + liste2**
- To verdier:
  - **sammen = [1, 2, 3] + [4, 5, 6]**
- En variabel og en verdi:
  - **liste = [1, 2, 3]**
  - **sammen = liste + [4, 5, 6]**
- En verdi og en variabel:
  - **liste = [4, 5, 6]**
  - **sammen =[1, 2, 3] + liste**

# Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! ☺

# Mengder


BASIC
Squirtle
Ability
Shell S
As long as this Pokémo
all damage done to th
yours and your opponent
Water
Flip a coin. If heads, this
weakness
×2
resistance
retreat
BASIC
Venusaur
Poison
Your opponent's Active
Jungle
Heal 30 damage from t
Basic Pokémon
Pikachu
Mouse Pokémon. Lengt
Spark If your op
Benched Pokémon
them and this atta
damage to it. (Don
and Resistance for
BASIC
Charizard
Wing Atta
Combusti
This Pokémon can't use
your next turn.
Bulbasaur
50 HP
BASIC
Sleep Poison
Flip a coin. If heads, the Defending Pokémon
is now Asleep and Poisoned.
Razor Leaf
20
retreat cost

# Mengder

- En samling av verdier som ikke har en rekkefølge og som ikke kan inneholde duplikater
- For eksempel:
  - **minMengde = set([1, 3, 3, 2, 1, 1])**

- Ingen rekkefølge betyr at…
  - **print(minMengde[3])**
- … vil resultere i en TypeError

- Ingen duplikater betyr at…
  - **liste = set([1, 1, 1, 5, 5, 9])**
  - **print(len(set))** **#Printer 3**
  - **print(liste)** **#Printer [1, 5, 9]**

# Hvordan lage en mengde

- Lage en tom mengde:
  - **minMengde = set()**
  - **print(minMengde)**
- Vil printe {}

- Lage en ikke-tom mengde:
  - **minMengde = {1, 1, 3, 2, 2, 1}**
  - **print(minMengde)**
- Vil printe {1, 3, 2}

# Hvordan legge til en verdi i en mengde

- Legge til en verdi:
  - **minMengde.add(4)**
  - **print(minMengde)**
- Vil printe {1, 3, 2, 4}

- Legge til flere verdier:
  - **minMengde.update([4, 4, 6, 4, 6, 5])**
  - **print(minMengde)**
- Vil printe {1, 3, 2, 4, 6, 5}

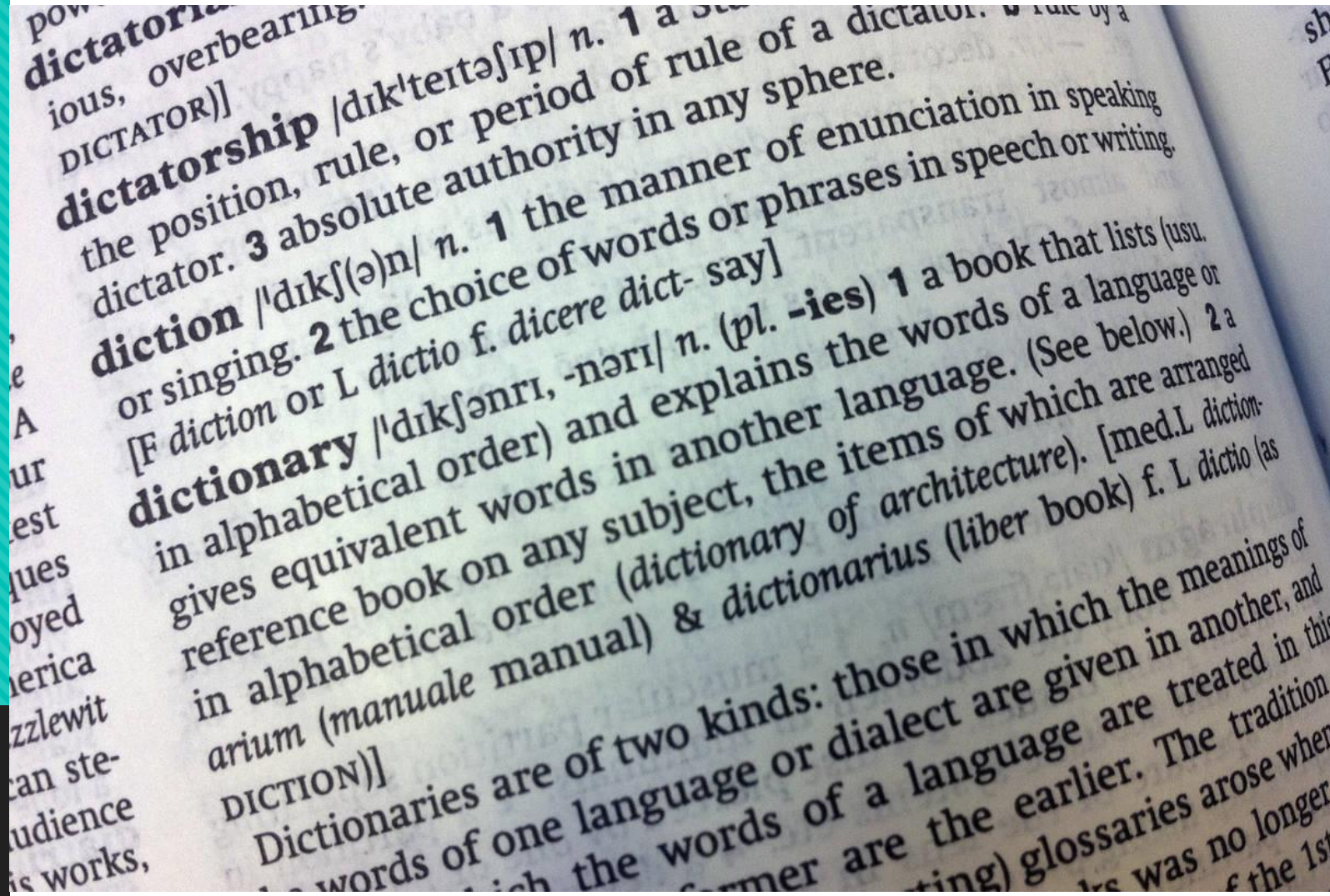
# Hvordan slette en verdi i en mengde

- Slette en verdi:
  - **minMengde = {1, 1, 3, 2, 2, 1}**
  - **print(minMengde)**
- Vil printe {1, 3, 2}
  - **minMende.remove(1)**
  - **print(minMengde)**
- Vil printe {3, 2}

# Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! ☺

# Ordbøker


dictatorship /dɪkˈteɪtəʃɪp/ n. 1 the position, rule, or period of rule of a dictator. 3 absolute authority in any sphere.
DICTATOR)]
diction /ˈdɪkʃ(ə)n/ n. 1 the manner of enunciation in speaking or singing. 2 the choice of words or phrases in speech or writing.
[F diction or L dictio f. dicere dict- say]
dictionary /ˈdɪkʃənrɪ, -nərɪ/ n. (pl. -ies) 1 a book that lists (usu. in alphabetical order) and explains the words of a language or gives equivalent words in another language. (See below.) 2 a reference book on any subject, the items of which are arranged in alphabetical order (dictionary of architecture). [med.L dictionarium (manuale manual) & dictionarius (liber book) f. L dictio (as
DICTION)]
Dictionaries are of two kinds: those in which the meanings of words of one language or dialect are given in another, and
the words of a language are treated in
are the earlier. The tradition
ting) glossaries arose when
was no longer

# Ordbøker og rekkefølge

- Slår opp en innholdsverdi basert på en nøkkelverdi
- En samling av verdier som ikke har en rekkefølge og som kan inneholde duplikate innholdsverdier men ikke duplikate nøkkelverdier
- ordbok = {
  - nøkkelverdi1 : innholdsverdi1,
  - nøkkelverdi2 : innholdsverdi2,
  - nøkkelverdi3 : innholdsverdi3
- }

- **minOrdbok = {**
  - **"Ola Nordmann“ : 12345678,**
  - **"Kari Nordmann“ : 87654321**
- **}**

- Både nøkkelverdier og innholdsverdier kan bestå av en hvilken som helst variabeltype

- Ingen rekkefølge betyr at…
  - **print(minOrdbok[1])**
- … vil resultere i en KeyError

# Ordbøker og duplikater

```
matteordbok = {
    "sum" : "resultatet av addisjon",
    "pi" : 3.14,
    3 : "det andre primtallet",
    3 : "det tredje trekanttallet",
    "totalsum" : "resultatet av addisjon",
    3 : "det andre oddetallet"
}
print(matteordbok)
```

```
{'sum': 'resultatet av addisjon',
'pi': 3.14,
3: 'det andre oddetallet',
'totalsum': 'resultatet av addisjon'}
```

- Legg merke til at…
  - Nøkkelverdien 3 og tilhørende innholdsverdi printes kun én gang
  - Nøkkelverdien 3 og tilhørende innholdsverdi printes når de forekommer først i ordboka
  - Nøkkelverdien 3 inneholder kun den siste innholdsverdien
  - Innholdsverdien "resultatet av addisjon" printes to ganger

# Hvordan lage en ordbok

- Lage en tom ordbok:
  - **minOrdbok = {}**

---

- Lage en ikke-tom ordbok:
  - **minOrdbok = {**
    - **"Norge" : "Oslo",**
    - **"Danmark" : "København",**
    - **"Sverige" : "Stockholm"**
  - **}**

# Hvordan legge til en nøkkelverdi og en innholdsverdi i en ordbok

- Legge til et verdipar:
  - **stoersteLand = {**
    - **"Kina" : 3,**
    - **"Russland" : 1,**
    - **"Canada" : 2**
    - **osv…**
  - **}**
  - **stoersteLand["Norge"] = 68**
  - **print(stoersteLand)**

- Vil printe
  - {"Kina" : 3,
  - "Russland" : 1,
  - "Canada" : 2,
  - osv…
  - "Norge" : 68}

# Hvordan aksessere eller printe en innholdsverdi i en ordbok

- Aksessere en innholdsverdi:
  - **mittLandsStoerrelse = stoersteLand["Norge"]**
- Variabelen får da verdien 68

---

- Printe en innholdsverdi:
  - **print(stoersteLand["Norge"])**
- Vil printe 68

# Hvordan endre eller slette en innholdsverdi i en ordbok

- Endring av innholdsverdi:
  - **stoersteLand["Norge"] = 1**
  - **print(stoersteLand["Norge"])**
- Vil nå printe 1 (i stedet for 68)

---

- Sletting av innholdsverdi:
  - **del stoersteLand["Norge"]**
  - **print(stoersteLand["Norge"])**
- Vil resultere i en KeyError

# Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! ☺