

Mengder

# Læringsmål

# Læringsmål

- Etter modulen:

# Læringsmål

- Etter modulen:
  - Forstår du forskjellen på en mengde og en liste

# Læringsmål

- Etter modulen:
  - Forstår du forskjellen på en mengde og en liste
  - Ser du hvordan du kreativt kan benytte egenskaper ved mengder til å løse et programmeringsproblem

# Forkunnskaper

# Forkunnskaper

- Modulen bygger direkte på:

# Forkunnskaper

- Modulen bygger direkte på:
  - Lister



# Forkunnskaper

- Modulen bygger direkte på:
  - Lister
- Nyttig for å forstå alle aspekter og eksempler:

# Forkunnskaper

- Modulen bygger direkte på:
  - Lister
- Nyttig for å forstå alle aspekter og eksempler:
  -

# Mengder

# Mengder

- Fagene man tar et semester

# Mengder

- Fagene man tar et semester
  - Har ingen spesifikk rekkefølge (hva er fag 1, 2 og 3..)

# Mengder

- Fagene man tar et semester
  - Har ingen spesifikk rekkefølge (hva er fag 1, 2 og 3..)
  - Alle fag må være ulike (kan ikke ta IN1000 og IN1000)

# Mengder

- Fagene man tar et semester
  - Har ingen spesifikk rekkefølge (hva er fag 1, 2 og 3..)
  - Alle fag må være ulike (kan ikke ta IN1000 og IN1000)
- Kan representeres med en liste, men:

# Mengder

- Fagene man tar et semester
  - Har ingen spesifikk rekkefølge (hva er fag 1, 2 og 3..)
  - Alle fag må være ulike (kan ikke ta IN1000 og IN1000)
- Kan representeres med en liste, men:
  - Får uansett en rekkefølge (som dog kan ignoreres):  
["IN1000", "IN1020", "IN1050"]



# Mengder

- Fagene man tar et semester
  - Har ingen spesifikk rekkefølge (hva er fag 1, 2 og 3..)
  - Alle fag må være ulike (kan ikke ta IN1000 og IN1000)
- Kan representeres med en liste, men:
  - Får uansett en rekkefølge (som dog kan ignoreres):  
["IN1000", "IN1020", "IN1050"]
  - Ingenting hindrer å ha samme fag flere ganger:  
["IN1000", "IN1000", "IN1000"]

# Mengder

# Mengder

- Mengde:

# Mengder

- Mengde:
  - En samling av ulike verdier

# Mengder

- Mengde:
  - En samling av ulike verdier
  - Det vil si: a) uten ordning, b) kun ulike verdier

# Mengder

- Mengde:
  - En samling av ulike verdier
  - Det vil si: a) uten ordning, b) kun ulike verdier
- Mengde i Python:

# Mengder

- Mengde:
  - En samling av ulike verdier
  - Det vil si: a) uten ordning, b) kun ulike verdier
- Mengde i Python:
  - `min_mengde = {1,5,1,1}`

# Mengder

- Mengde:
  - En samling av ulike verdier
  - Det vil si: a) uten ordning, b) kun ulike verdier
- Mengde i Python:
  - `min_mengde = {1,5,1,1}`
  - `len(min_mengde)` **2**



# Mengder

- Mengde:
  - En samling av ulike verdier
  - Det vil si: a) uten ordning, b) kun ulike verdier
- Mengde i Python:
  - `min_mengde = {1,5,1,1}`
  - `len(min_mengde)` **2**
  - `print(min_mengde)` **{1, 5}**

# Mengder

- Mengde:
  - En samling av ulike verdier
  - Det vil si: a) uten ordning, b) kun ulike verdier
- Mengde i Python:
  - `min_mengde = {1,5,1,1}`
  - `len(min_mengde)` **2**
  - `print(min_mengde)` **{1, 5}**
  - `5 in min_mengde` **True**

# Mengder

- Mengde:
  - En samling av ulike verdier
  - Det vil si: a) uten ordning, b) kun ulike verdier
- Mengde i Python:
  - `min_mengde = {1,5,1,1}`
  - `len(min_mengde)` **2**
  - `print(min_mengde)` **{1, 5}**
  - `5 in min_mengde` **True**
  - ~~`min_mengde[1]`~~

# Mengder

- Mengde:
  - En samling av ulike verdier
  - Det vil si: a) uten ordning, b) kun ulike verdier
- Mengde i Python:
  - `min_mengde = {1,5,1,1}`
  - `len(min_mengde)` **2**
  - `print(min_mengde)` **{1, 5}**
  - `5 in min_mengde` **True**
  - ~~`min_mengde[1]`~~
  - `min_mengde = set(min_liste)` #dermed også `set([1,5,1,1])`

# En liten oppgave

- Oppgave:
  - Gitt en liste med fem terningkast (fem tall fra 1 til 6), fortell brukeren om hun har fått yatsy (holder å skrive *True/False*)  
  
*(skriv kun kode for å finne ut om yatsy  
- anta at du allerede har en variabel med liste av terningkast, f.eks:  
terninger = [3,5,3,3,3]*
  - Prøv selv med blyant og papir! (3 minutt)  
*(NB! Bruk tiden på å tenke, ikke skrive  
- løsningen kan være veldig kort)*

# Løsning

# Løsning

- {yatzy.py}

En liten nøtt



# En liten nøtt

- Oppgave:
  - Gitt en liste med fem terningkast (fem tall fra 1 til 6), fortell brukeren om hun har fått hus (tre av ett terningkast, og to av et annet)
  - Løs det gjerne ved å kombinere teknikkene fra firere og yatzy (count og mengder)

# En liten nøtt

- Oppgave:
  - Gitt en liste med fem terningkast (fem tall fra 1 til 6), fortell brukeren om hun har fått hus (tre av ett terningkast, og to av et annet)
  - Løs det gjerne ved å kombinere teknikkene fra firere og yatzy (count og mengder)
- Løs gjerne først følgende oppgave:
  - Brukeren mangler kun fire like og hus. Fortell brukeren om hun har fått noe hun kan bruke (*uten å skille mellom hvilken av de to*)
  - *Hint: Hvordan kan det ha seg at denne oppgaven er enklere!?*

# En liten nøtt

- Oppgave:
  - Gitt en liste med fem terningkast (fem tall fra 1 til 6), fortell brukeren om hun har fått hus (tre av ett terningkast, og to av et annet)
  - Løs det gjerne ved å kombinere teknikkene fra firere og yatzy (count og mengder)
- Løs gjerne først følgende oppgave:
  - Brukeren mangler kun fire like og hus. Fortell brukeren om hun har fått noe hun kan bruke (*uten å skille mellom hvilken av de to*)
  - *Hint: Hvordan kan det ha seg at denne oppgaven er enklere!?*
- *Prøv selv med blyant og papir!*

# En liten nøtt

- Oppgave:
  - Gitt en liste med fem terningkast (fem tall fra 1 til 6), fortell brukeren om hun har fått hus (tre av ett terningkast, og to av et annet)
  - Løs det gjerne ved å kombinere teknikkene fra firere og yatzy (count og mengder)
- Løs gjerne først følgende oppgave:
  - Brukeren mangler kun fire like og hus. Fortell brukeren om hun har fått noe hun kan bruke (*uten å skille mellom hvilken av de to*)
  - *Hint: Hvordan kan det ha seg at denne oppgaven er enklere!?*
- *Prøv selv med blyant og papir!*
- *Etterpå diskuter med nabo!*

# Løsning på hus eller fire like

- {hus\_eller\_4like.py}

# Løsning på hus

- {hus.py}