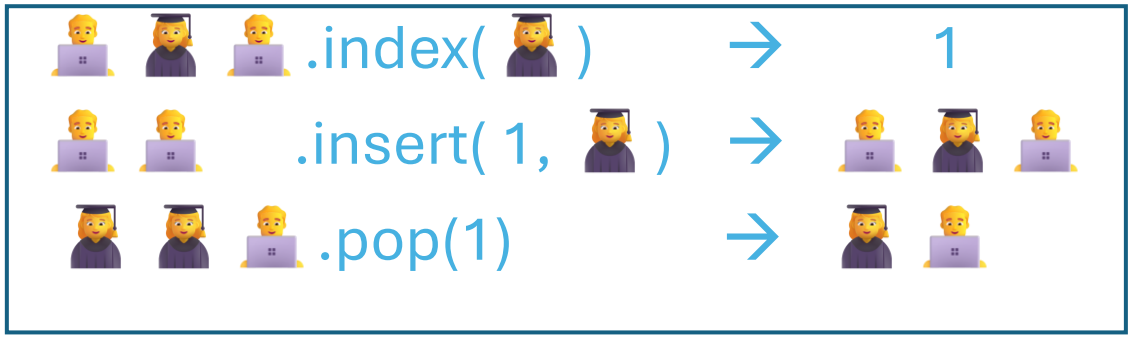
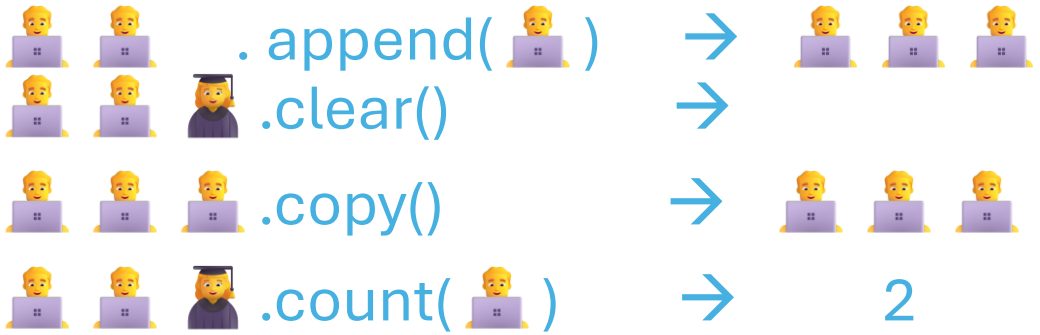
```
def funksjon(parameter1, parameter2):
    nested_list = [["N“, "Ø“, "S“, "T“, "E“, “T“], ["LISTE"]]
    return “REPETISJON”
```

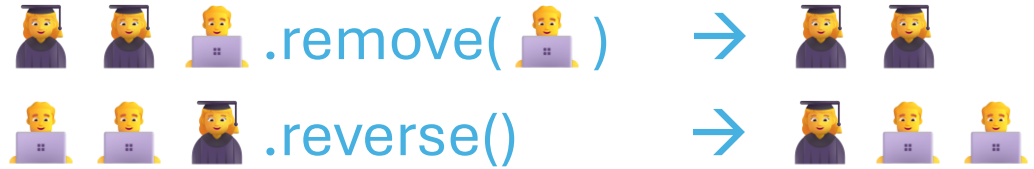


["IN1000", "Gruppetime"]

#REPETISJON

```
1  print("["IN1000", "Gruppetime"]")
2
3
4
5  funksjon(argument1, argument2)
6
```





Obligtips

- Les oppgaveteksten nøye – gjør presist hva den ber om, steg for steg
- Kommentarer - Viktig! Men ikke kommenter alt for mye
    - Kommentarer er for at andre skal forstå hva som blir gjort i koden.
    - Også nyttig for deg selv

Feilmeldinger

- Ikke la deg stoppe av feilmeldinger!
- Det er en naturlig del av det å kode
- I feilmeldingen står det på hvilken linje feilen er, og hva som er feil.

```
Traceback (most recent call last):
  File "c:\Users\Bruker\Desktop\IN1000\uke3.py", line 30, in <module>
    print(liste[5])
          ~~~~~^^^
IndexError: list index out of range
```

IndexError: list index

Når koden ikke fungerer og det ikke er feilmelding:

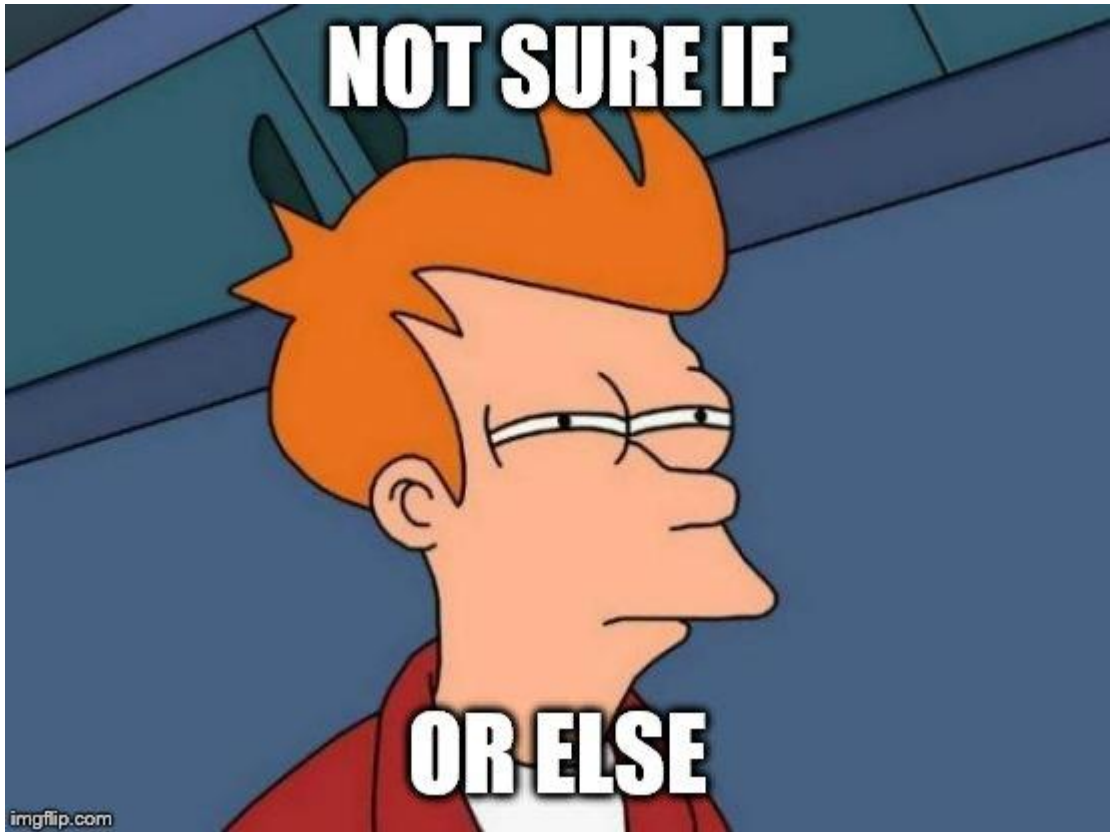
- - Forklar til deg selv eller andre hva koden din gjør steg for steg
- - Python Tutor
- - Forsøk å printe underveis for å se hva som skjer og fange opp feil
- - Tegn
- - Spør om hjelp på Lab / gruppetime

If – elif – else

if → Hvis....

elif → Eller hvis...

else → eller så ....



if - Hvis....

```
x = 2
if x < 5:
    print("x er mindre enn 5")
```

Elif – eller hvis...

```
x = 10
if x < 5:
    print("x er mindre enn 5")
elif x > 5:
    print("x er større enn 5")
```

```
x = 12
if x > 10:
    print("A")
elif x == 12:
    print("B")
```

Fasit:
A

```
x = 12
if x > 10:
    print("A")
if x == 12:
    print("B")
```

Fasit:
A
B



Else ..

```
x = 10
if x < 5:
        x = x + 2
else:
        x = x + 1
print(x)
```

```
x = 10
if x < 5:
        x += 2          ←
else:
        x += 1          ←
print(x)
```

x = x+1
er identisk med:
x += 1

Oppg 1a Eksamen 2022

```
a = 3
if a > 5:
    a = a*2
print(a)
```

#Svar: 3

```
a = 5
b = a > 2
if a == 3 or b:
    a += 10
print(a)
```

#Svar: 15

Kommandolinjen

Pwd → path

cd + <mappe du vil besøke> → tar deg til mappen du vil besøke, denne må være i mappen du allerede er

cd .. → hopper tilbake til forrige mappe

ls → se hva som er i en mappe

```
liste = ["L", "I", "S", "T", "E"]
#Indeks:    0    1    2    3    4
```

HVA PRINTES HER?

```
print(liste[0])

print(liste[4])

print(len(liste))

print(liste[5])
```

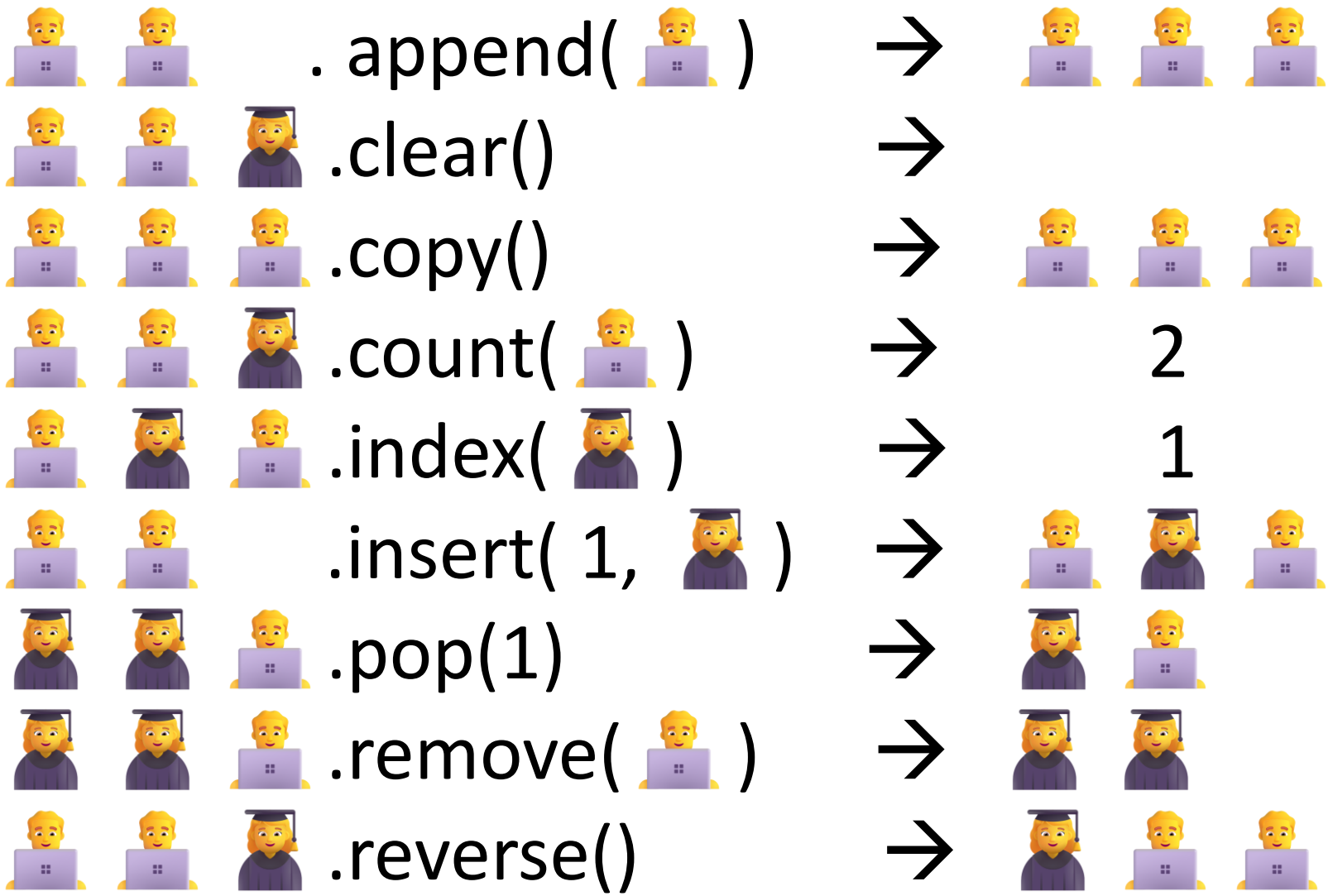
```
IndexError: list index out of range
```

INDEKSEN ER UTENFOR REKKEVIDDEN: Vi har indekser frem til 4, dermed er indeks 5 utenfor rekkevidden

```
Traceback (most recent call last):
  File "c:\Users\Bruker\Desktop\IN1000\uke3.py", line 30, in <module>
    print(liste[5])
          ~~~~~^^^
IndexError: list index out of range
```

```
IndexError: list index out of range
```

Liste – noen funksjoner



Liste

```
liste = []                  #tom liste
# liste.append(3)    # legger til 3 i liste

# Kan også gjøre slik: (konkatenere liste):

liste += [3]

print(liste)

# da printes: Fasit: [3, 3]
```

Konkatenere lister

```
liste1 = [5]

liste2 = ["hei", 2]

# Konkatenere liste og liste2:

liste3 = liste1 + liste2

print(liste3)

# da printes: Fasit:[5, 'hei', 2]
```

Nøstede lister – liste med lister i

```
[["I", "N"],[1000]]
```

```
Indeks:                 0                       1
Indeks:             0       1
Indeks:                                         0
```

Hva printes her?

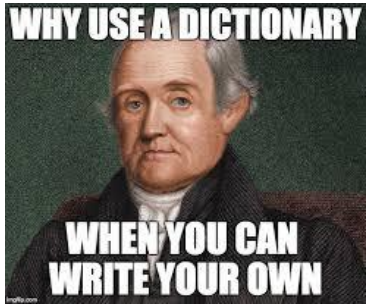
```
Gruppetime = [["I", "N"],[1000]]
print(Gruppetime[0][0])
```

Mengde

- En samling av ulike verdier – ingen duplikater!

```
mengde = {3,3}
print(mengde)
# da printes: Fasit:{3}
```

Ordbok



- En ordbok (dict) er en samling mappinger (transformasjoner) fra én verdi til én annen

```
ordbok = {"Tomat": 2, "Eple": 4}
```

nøkkelverdi(key)

innholdsverdi(value)

Hente innholdsverdien i en ordbok:

```
ordbok = {"Tomat": 2, "Eple": 4}

print(ordbok["Eple"])
```

Endre innholdsverdien i en ordbok

```
ordbok = {"Tomat": 2, "Eple": 4}

ordbok["Eple"] = 3

print(ordbok)
```

Fasit: {"Tomat": 2, "Eple": 3}

Legge til et element i en ordbok

```
ordbok = {"Tomat": 2, "Eple": 4}

ordbok["Appelsin"] = 5

print(ordbok)
```

```
Fasit: {'Tomat': 2, 'Eple': 4, 'Appelsin': 5}
```

Hva skjer her?

```
ordbok = {"Tomat": 2, 3:"Hei", "Tekststreng":"Tekst"}

print(ordbok)
```

Kan man ha ulik type verdier i en ordbok?

Fasit: JA!

```
{'Tomat': 2, 3: 'Hei', 'Tekststreng': 'Tekst'}
```

If-test + liste -- Hvor mange ja’er printes?

```
if 1 in [1, 2, 3]:
    print("ja")


if 1 in ["1", 2, 3]:
    print("ja")
```

For-løkker

```
for i in range(0, 5):
    print(i)
```

→ NB! Printer ikke ut 5

```
liste = [0, 1, 2, 3, 4]
for element in liste:
    print(element)
```

```
ordbok = {“tomat”: 1, “brokkoli”: 2}

for nokkel in ordbok:
  print(nokkel)

for nokkel in ordbok:
  print(ordbok[nokkel])

for x in ordbok.values():
  print(x)
```

While-løkker

```
while <condition>:
        #gjør noe

x = 0
while x < 5:
    print(x)
    x += 1
```

Evigkjørende while-løkker

```
x = 0

while x < 5:

    print(x)
```



Åpne fil

```
fil = open("filnavn.txt")
```



Deretter iterere gjennom hver linje i filen

```
fil = open("filnavn.txt")
for linje in fil:
        print("Her fant jeg: " + linje)
```

Også mulig å legge de til i liste:

```
nyEngelskListe = []
innfil = open("1000engOrd.txt")
for linje in innfil:
    nyEngelskListe.append(linje)

print(nyEngelskListe)
print("Antall ord i lista var", len(nyEngelskListe))
print("Takk for nå")
```

Eksamen høst 2022

```
a = 5
b = a > 2
if a == 3 or b:
    a += 10
print(a)
#Svar: 15 (tall)
```

Skrive til en tekstfil

- Først åpne en fil for skriving:

```
min_fil = open("mittUtFilNavn.txt", "w")
```

- Deretter skrive tekst (en streng-verdi) til slutten av filen:

```
min_fil = open("mittUtFilNavn.txt", "w")
min_fil.write("Dette havner i filen")
```

- Man vil typisk ha linjeskift i det man skriver ut:

```
min_fil.write("Dette havner i filen\n")
min_fil.write("Dette havner på\n ulike\n linjer\n")
```

- Når man er ferdig bør man lukke filen:

```
min_fil.close()
```

Strip()

- *s*.**strip**()
- *s*.**strip**(*chars*)

This method returns a new version of string *s* in which white space (blanks, tabs, and newlines) is removed from both ends (front and back) of *s*. If an argument is provided, characters in the string *chars* are removed instead of white space.

Parameter: *chars* A string specifying the characters to remove

Returns: A new version of the string

Split()

- `s.split()`
- `s.split(sep)`
- `s.split(sep, maxsplit)`

This method returns a list of words from string *s*. If the substring *sep* is provided, it is used as the delimiter; otherwise, any white space character is used. If *maxsplit* is provided, then only that number of splits will be made, resulting in at most *maxsplit* + 1 words.

Parameters: *sep* A substring specifying the separator used for the split
maxsplit The maximum number of splits

Returns: A list of words that results from splitting the string

Parameter vs. Argument

- Definerer funksjonen → da heter det parameter

```
def funksjon(parameter1, parameter2):
```

- Kaller på funksjonen → da heter det argument

```
funksjon(argument1, argument2)
```

Litt om uttrykk

a = 2

Variabel

Uttrykk

Verdi

Int

2 er tilordnet a

"Hei! "

Uttrykk

Tekststreng

2+2

Uttrykk

Scope

- Globale og lokale variabler
- Kodeeksempel

Siste oppgave:
- hva printes?

```
liste = []
x = 10

if x < 15:
    liste.append(1)
elif x > 3:
    liste.append(2)
else:
    liste.append(3)

while x < 11:
    liste.append(1)
    x += 1

for element in liste:
    print(element)
```