

MENGDER, CSV og JSON

INF100

HØST 2024

Torstein Strømme

MENGDER

superraskt



```
s = {'kake', 'fluor', 'tran'}
```

Spør om et element er i mengden

Se gjennom alle elementene i mengden

(Destruktivt) legg til et element til mengden

(Destruktivt) fjern et element fra mengden

```
'kake' in s
```

```
for element in s: ...
```

```
s.add('vitaminbjørner')
```

```
s.discard('kake')
```

MENGDER

```
s = {'kake', 'fluor', 'tran'}
```

Elementer har ikke posisjoner

Samme element eksisterer kun én gang

Ikke mulig å ha muterbare elementer

```
s[1]
```

```
s.add('tran')
```

```
s.add(['brus', 'saft'])
```

*uten effekt, fordi
'tran' allerede er i s*

JSON

En streng som oppretter oppslagsverk og lister med data

<https://dogapi.dog/api/v2/facts?limit=3>

```
{"data":[{"id":"8fa94033-0160-4f27-a6b0-009e77bef24c","type":"fact","attributes":{"body":"Dogs have lived with humans for over 14,000 years. Cats have lived with people for only 7,000 years."}},{id":"c898f9c5-0172-4a18-b67c-227b27ab4045","type":"fact","attributes":{"body":"More than one in three U.S. families owns a dog."}},{id":"c6fe1512-611d-44b1-972a-e44d15967a7e","type":"fact","attributes":{"body":"The names of 77 ancient Egyptian dogs have been recorded. The names refer to color and character, such as Blackie, Ebony, Good Herdsman, Reliable, and Brave One."}}]}
```

```
{
  "data": [
    {
      "id": "8fa94033-0160-4f27-a6b0-009e77bef24c",
      "type": "fact",
      "attributes": {
        "body": "Dogs have lived with humans for over 14,000 years."
      }
    },
    {
      "id": "c898f9c5-0172-4a18-b67c-227b27ab4045",
      "type": "fact",
      "attributes": {
        "body": "More than one in three U.S. families owns a dog."
      }
    },
    {
      "id": "c6fe1512-611d-44b1-972a-e44d15967a7e",
      "type": "fact",
      "attributes": {
        "body": "The names of 77 ancient Egyptian dogs have been recorded."
      }
    }
  ]
}
```

JSON

- Strukturerert data lagret som en streng/tekstfil
- Brukes ofte for kommunikasjon
- Støtter de vanligste datatypene skrevet (nesten) akkurat som i Python:
 - dict *Men støtter ikke «trailing comma»*
 - list
 - str *Men du må bruke " ", ikke ' '*
 - int
 - float *Noen få unntak (NaN og infinity)*
 - bool *Men bruker lowercase*
 - None *Men kaller den for «null»*

JSON

Fra streng til data

```
import json

json_string = '{"foo": 42, "bar": 95}'
data = json.loads(json_string)

print(data['foo']) # printer 42
```

JSON

Fra data til streng

```
import json
```

```
data = {'foo': 42, 'bar': 95}  
json_string = json.dumps(data)
```

```
print(repr(json_string)) # '{"foo": 42, "bar": 95}'
```


- En CSV-fil: ren tekst som representerer tabell-data

```
personnummer,medisin,startdato  
01010111111,kake,2022-08-22  
22020222222,fluor,2022-01-11  
01010111111,tran,2022-06-01
```



	A	B	C
1	personnummer	medisin	startdato
2	10101111111	vitaminbjørner	22/08/2022
3	22020222222	fluortabletter	11/01/2022
4	10101111111	tran	01/06/2022

- En CSV-fil: ren tekst som representerer tabell-data

```
personnummer,medisin,startdato  
01010111111,kake,2022-08-22  
22020222222,fluor,2022-01-11  
01010111111,tran,2022-06-01
```

- I Python: tabell-data representert som streng

```
table_string = '''  
personnummer,medisin,startdato  
01010111111,kake,2022-08-22  
22020222222,fluor,2022-01-11  
01010111111,tran,2022-06-01  
'''
```

- En CSV-fil: ren tekst som representerer tabell-data

```
personnummer,medisin,startdato
01010111111,kake,2022-08-22
22020222222,fluor,2022-01-11
01010111111,tran,2022-06-01
```

- I Python: tabell-data representert som liste av strenger

```
table_string_list = [
    'personnummer,medisin,startdato',
    '01010111111,kake,2022-08-22',
    '22020222222,fluor,2022-01-11',
    '01010111111,tran,2022-06-01',
]
```

- En CSV-fil: ren tekst som representerer tabell-data

```
personnummer,medisin,startdato  
01010111111,kake,2022-08-22  
22020222222,fluor,2022-01-11  
01010111111,tran,2022-06-01
```

- I Python: tabell-data representert som 2D-liste

```
table_2d_list = [  
    ['personnummer', 'medisin', 'startdato'],  
    ['01010111111', 'kake', '2022-08-22'],  
    ['22020222222', 'fluor', '2022-01-11'],  
    ['01010111111', 'tran', '2022-06-01'],  
]
```

- En CSV-fil: ren tekst som representerer tabell-data

```
personnummer,medisin,startdato
01010111111,kake,2022-08-22
22020222222,fluor,2022-01-11
01010111111,tran,2022-06-01
```

- I Python: tabell-data representert som liste av oppslagsverk

```
table_dict_list = [
    {'personnummer': '01010111111', 'medisin': 'kake', 'startdato': '2022-08-22'},
    {'personnummer': '22020222222', 'medisin': 'fluor', 'startdato': '2022-01-11'},
    {'personnummer': '01010111111', 'medisin': 'tran', 'startdato': '2022-06-01'}
]
```

- Liste av lister: hvilke medisiner er registrert på person 01010111111?

```
table_2d_list = [  
    ['personnummer', 'medisin', 'startdato'],  
    ['01010111111', 'kake', '2022-08-22'],  
    ['22020222222', 'fluor', '2022-01-11'],  
    ['01010111111', 'tran', '2022-06-01'],  
]
```

```
for row in table_2d_list:  
    if row[0] == '01010111111':  
        print(row[1])
```

magiske tall (fy!)



- Liste av oppslagsverk: hvilke medisiner er registrert på person 01010111111?

```
table_dict_list = [  
    {'personnummer': '01010111111', 'medisin': 'kake', 'startdato': '2022-08-22'},  
    {'personnummer': '22020222222', 'medisin': 'fluor', 'startdato': '2022-01-11'},  
    {'personnummer': '01010111111', 'medisin': 'tran', 'startdato': '2022-06-01'}  
]
```

```
for row in table_dict_list:  
    if row['personnummer'] == '01010111111':  
        print(row['medisin'])
```



FRA CSV TIL LISTE AV OPPSLAGSVERK

- Fra CSV til streng

```
from pathlib import Path
table_string = Path('org.csv').read_text()
```

```
table_string = '''
personnummer,medisin,startdato
01010111111,kake,2022-08-22
22020222222,fluor,2022-01-11
01010111111,tran,2022-06-01
'''
```

- Fra CSV til streng
- **Fra streng til liste av linjer**
 - Øverste linje er header
 - Øvrige linjer er data

```
from pathlib import Path
table_string = Path('org.csv').read_text()

lines = table_string.splitlines()
header_line = lines[0]
data_lines = lines[1:]
```

```
lines = [
    'personnummer,medisin,startdato',
    '01010111111,kake,2022-08-22',
    '22020222222,fluor,2022-01-11',
    '01010111111,tran,2022-06-01',
]
```

```
data_lines = [
    '01010111111,kake,2022-08-22',
    '22020222222,fluor,2022-01-11',
    '01010111111,tran,2022-06-01',
]
```

```
header_line = 'personnummer,medisin,startdato'
```

- Fra CSV til streng
- **Fra streng til liste av linjer**
 - Øverste linje er header
 - Øvrige linjer er data

```
from pathlib import Path
table_string = Path('org.csv').read_text()

lines = table_string.splitlines()
header_line = lines[0]
data_lines = lines[1:]
headers = header_line.split(',')
```

```
lines = [
    'personnummer,medisin,startdato',
    '01010111111,kake,2022-08-22',
    '22020222222,fluor,2022-01-11',
    '01010111111,tran,2022-06-01',
]
```

```
data_lines = [
    '01010111111,kake,2022-08-22',
    '22020222222,fluor,2022-01-11',
    '01010111111,tran,2022-06-01',
]
```

```
header_line = 'personnummer,medisin,startdato'
```

```
headers = ['personnummer', 'medisin', 'startdato']
```

- Fra CSV til streng
- Fra streng til liste av linjer
 - Øverste linje er header
 - Øvrige linjer er data
- **Fyll tabell med rader**

```
table_dict = [  
    {},  
    {},  
    {},  
]
```

```
from pathlib import Path  
table_string = Path('org.csv').read_text()  
  
lines = table_string.splitlines()  
header_line = lines[0]  
data_lines = lines[1:]  
headers = header_line.split(',')  
  
table_dict = []  
for line in data_lines:  
    row_dict = {}  
    # begynn fyll opp row_dict her ...  
  
    # ... ferdig fylt opp row_dict her  
    table_dict.append(row_dict)
```

- Fra CSV til streng
- Fra streng til liste av linjer
 - Øverste linje er header
 - Øvrige linjer er data
- Fyll tabell med rader
- **Fra linje til oppslagslagsverk**

```
line = '01010111111,kake,2022-08-22'
```

```
cells = ['01010111111', 'kake', '2022-08-22']
```

```
table_dict = []
for line in data_lines:
    row_dict = {}
    # begynn fyll opp row_dict her ...

    cells = line.split(',')
    for i in range(number_of_cols):
        header = headers[i]
        value = cells[i]
        row_dict[header] = value

    # ... ferdig fylt opp row_dict her
    table_dict.append(row_dict)
```

- Fra CSV til streng
- Fra streng til liste av linjer
 - Øverste linje er header
 - Øvrige linjer er data
- Fyll tabell med rader
- **Fra linje til oppslagslagsverk**

```
number_of_cols = 3
```

```
i = 0
```

```
header = 'personnummer'
```

```
value = '01010111111'
```

```
number_of_cols = len(headers)
```

```
table_dict = []
```

```
for line in data_lines:
```

```
    row_dict = {}
```

```
    # begynn fyll opp row_dict her ...
```

```
    cells = line.split(',')
```

```
    for i in range(number_of_cols):
```

```
        header = headers[i]
```

```
        value = cells[i]
```

```
        row_dict[header] = value
```

```
    # ... ferdig fylt opp row_dict her
```

```
    table_dict.append(row_dict)
```

```
row_dict['personnummer'] = '01010111111'
```

- Fra CSV til streng
- Fra streng til liste av linjer
 - Øverste linje er header
 - Øvrige linjer er data
- Fyll tabell med rader
- **Fra linje til oppslagslagsverk**

```
number_of_cols = 3
```

```
i = 1
```

```
header = 'medisin'
```

```
value = 'kake'
```

```
number_of_cols = len(headers)
```

```
table_dict = []
```

```
for line in data_lines:
```

```
    row_dict = {}
```

```
    # begynn fyll opp row_dict her ...
```

```
    cells = line.split(',')
```

```
    for i in range(number_of_cols):
```

```
        header = headers[i]
```

```
        value = cells[i]
```

```
        row_dict[header] = value
```

```
    # ... ferdig fylt opp row_dict her
```

```
    table_dict.append(row_dict)
```

```
row_dict['medisin'] = 'kake'
```

- Fra CSV til streng
- Fra streng til liste av linjer
 - Øverste linje er header
 - Øvrige linjer er data
- Fyll tabell med rader
- **Fra linje til oppslagslagsverk**

```
number_of_cols = 3
```

```
i = 2
```

```
header = 'startdato'
```

```
value = '2022-08-22'
```

```
number_of_cols = len(headers)
```

```
table_dict = []
```

```
for line in data_lines:
```

```
    row_dict = {}
```

```
    # begynn fyll opp row_dict her ...
```

```
    cells = line.split(',')
```

```
    for i in range(number_of_cols):
```

```
        header = headers[i]
```

```
        value = cells[i]
```

```
        row_dict[header] = value
```

```
    # ... ferdig fylt opp row_dict her
```

```
    table_dict.append(row_dict)
```

```
row_dict['startdato '] = '2022-08-22 '
```


- Fra CSV til streng
- Fra streng til liste av linjer
 - Øverste linje er header
 - Øvrige linjer er data
- Fyll tabell med rader
- **Fra linje til oppslagslagsverk**

```
number_of_cols = len(headers)

table_dict = []
for line in data_lines:
    row_dict = {}
    # begynn fyll opp row_dict her ...

    cells = line.split(',')
    for i in range(number_of_cols):
        header = headers[i]
        value = cells[i]
        row_dict[header] = value

    # ... ferdig fylt opp row_dict her
    table_dict.append(row_dict)
```

```
row_dict = {'personnummer': '01010111111', 'medisin ': 'kake', 'startdato': '2022-08-22'}
```

- Fra CSV til streng
- Fra streng til liste av linjer
 - Øverste linje er header
 - Øvrige linjer er data
- Fyll tabell med rader
- **Fra linje til oppslagslagsverk**

```

number_of_cols = len(headers)

table_dict = []
for line in data_lines:
    row_dict = {}
    # begynn fyll opp row_dict her ...

    cells = line.split(',')
    for i in range(number_of_cols):
        header = headers[i]
        value = cells[i]
        row_dict[header] = value

    # ... ferdig fylt opp row_dict her
    table_dict.append(row_dict)

```

```

table_dict_list = [
    {'personnummer ': '01010111111', 'medisin': 'kake', 'startdato': '2022-08-22'},
    {'personnummer ': '22020222222', 'medisin': 'fluor', 'startdato': '2022-01-11'},
    {'personnummer ': '01010111111', 'medisin': 'tran', 'startdato': '2022-06-01'},
]

```

```
from pathlib import Path
```

```
table_as_csv = Path('filename.csv').read_text(encoding='utf-8')
```

```
lines = table_as_csv.splitlines()
```

```
headers = lines[0].split(',')
```

```
data_lines = lines[1:]
```

```
number_of_cols = len(headers)
```

```
table_dict = []
```

```
for line in data_lines:
```

```
    row_dict = {}
```

```
    cells = line.split(',')
```

```
    for i in range(len(headers)):
```

```
        header = headers[i]
```

```
        value = cells[i]
```

```
        row_dict[header] = value
```

```
    table_dict.append(row_dict)
```

NOEN MÅ HA GJORT DETTE FØR

Alternatív A

```
from pathlib import Path
from csv import DictReader

with Path('filename.csv').open('rt', encoding='utf-8', newline='') as file:
    reader = DictReader(file, delimiter=',', quotechar='')

    headers = reader.fieldnames
    data = list(reader)
```

Alternativ B

```
from pathlib import Path
from csv import DictReader
from io import StringIO

file_string = Path('filename.csv').read_text(encoding='utf-8')
reader = DictReader(StringIO(file_string), delimiter=',', quotechar='')

headers = reader.fieldnames
data = list(reader)
```

Fungerer også hvis du har en CSV-streng som ikke kommer fra en fil
(men er litt langsommere med store filer)

Skríve til streng

```
from pathlib import Path
from csv import DictWriter
from io import StringIO

headers = ...
data = ...

output = StringIO()
writer = DictWriter(output, fieldnames=headers, delimiter=',', quotechar='"')
writer.writeheader()
writer.writerows(data)

output_string = output.getvalue()
output.close()
```

Skrive til fil

```
from pathlib import Path
from csv import DictWriter
```

```
headers = ...
data = ...
```

```
with Path('newfile.csv').open('wt', encoding='utf-8', newline='') as file:
    writer = DictWriter(file, fieldnames=headers, delimiter=',', quotechar='"')
    writer.writeheader()
    writer.writerows(data)
```


ÅPEN LAB

ÅPEN LAB

- Gjør (nesten) akkurat hva du vil
- Bestått tilsvarende bestått på lab 7 og lab 8
- Krav:
 - Må benytte *uib_inf100_graphics.event_app* som rammeverk
 - Må være interaktivt (reagerer på mus eller tastatur)
 - Må ha en viss kompleksitet. Spør hvis du er usikker.