Programmering i python - avansert

Oppgaveøkt 24.10.22

simulalearning

X

Folkeuniversitetet

Planen for i dag (09:00-11:00):

-Litt repetisjon: tekstbehandling, ordlister (dictionaries), pandas
-En kort demonstrasjon av pandas
-Selvstendig jobbing med oppgaver

Tekstbehandling

```
1   with open('tekstfil.txt') as fil_tekst:
2       innlest_tekst = fil_tekst.read()
```

- Vi kan lese inn tekst fra fil med syntaksen vist til venstre
- Vi kan lete etter en bokstav eller et ord i teksten. (OBS: Dette er følsomt for store og små bokstaver.)

```
 8   mitt_ord = 'adresse'
 9   if mitt_ord in innlest_tekst:
10       print('Ordet var i teksten.')
11   else:
12       print('Ordet var ikke i teksten.')
```

```
29  tekst = 'I dag er det mandag 24. oktober.'
30  print(tekst.split())
```

```
['I', 'dag', 'er', 'det', 'mandag', '24.', 'oktober.']
```

- Vi kan dele opp en tekststreng med funksjonen split(). Vi kan splitte på mellomrom, ny linje, eller tegn.

```python
with open('sang.txt') as f:
    sang_tekst = f.read()

import random
sang_linjer = sang_tekst.split('\n')
random.shuffle(sang_linjer)
print('\n'.join(sang_linjer))
```

```
Then you can start to make it better.
Take a sad song and make it better.
Remember to let her into your heart,
Hey Jude, don't make it bad.
```

Tekstbehandling

- Funksjonen random.shuffle() bytter om på rekkefølgen på elementene i en liste.
- Når vi bruker split() blir alt mellom hver splittelse av strengen et element i en liste.

Tekstbehandling

```
15    tekststreng = 'hallo'
16    for bokstav in tekststreng:
17        print(bokstav)
```

```
In [82]: runfile('C:/Users/rawis/OneDrive/Dokumenter/Jobb/
Kodeskolen/FU-kurs oktober 2022/tekstbehandling_eksempel.py',
wdir='C:/Users/rawis/OneDrive/Dokumenter/Jobb/Kodeskolen/FU-kurs
oktober 2022')
h
a
l
l
o
```

- Vi kan se på ett og ett tegn i teksten med en løkke

```
1    alfabet = 'abcdefghijklmnopqrstuvwxyzæøå'
2    print(alfabet[0:3])
3    print(alfabet[:3])
4    print(alfabet[10:20])
5    print(alfabet[10:])
```

```
abc
abc
klmnopqrst
klmnopqrstuvwxyzæøå
```

- Vi kan aksessere tegn i teksten ved å bruke indeksering som om tekststrengen var en liste.

```
 9    ordliste = {}
10    nøkkel = 'navn'
11    verdi = 'Ragnhild'
12    ordliste[nøkkel] = verdi
```

```
In [86]: print(ordliste['navn'])
Ragnhild
```

```
14    ordliste = {nøkkel:verdi}
```

Ordlister (dictionaries)

- Ordlister er en annen type datastruktur enn lister og arrays
- Elementer i ordlister består av en nøkkel som viser til en verdi, og verdien
- For å få tilgang til verdien bruker vi nøkkelen

- Alternativ syntaks for å lage en ordliste er vist til venstre

Ordlister (dictionaries)

```
16  nødnummer = {'brann': 110, 'politi': 112, 'ambulanse': 113}
```

```
In [104]: for nøkkel in nødnummer:
     ...:     print(nødnummer[nøkkel])
     ...:
110
112
113
```

- Man kan aksessere verdiene i ordlisten med en løkke som vist til venstre

```
16  nødnummer = {'brann': 110, 'politi': 112, 'ambulanse': 113}
17  print(nødnummer.keys())
```

```
dict_keys(['brann', 'politi', 'ambulanse'])
```

- Funksjonen keys() gir alle nøklene i ordlisten.

```python
import pandas as pd

df = pd.read_csv('covid-19.csv')
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 230 entries, 0 to 229
Data columns (total 13 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   Id               230 non-null    int64
 1   Country          230 non-null    object
 2   TotalCases       230 non-null    int64
 3   TotalDeaths      230 non-null    int64
 4   NewDeaths        230 non-null    int64
 5   TotalRecovered   230 non-null    int64
 6   ActiveCases      230 non-null    int64
 7   SeriousCritical  230 non-null    int64
 8   TotCases1Mpop    230 non-null    int64
 9   Deaths1Mpop      230 non-null    int64
 10  TotalTests       230 non-null    int64
 11  Tests1Mpop       230 non-null    int64
 12  Population       230 non-null    int64
dtypes: int64(12), object(1)
memory usage: 23.5+ KB
None
```

Pandas

- Pandas er et statistikkbibliotek som kan brukes til å jobbe med tabeller
- Pandas bruker en ny type datastruktur, dataframes.
- Dokumentasjon ligger på

https://pandas.pydata.org/docs/index.html

- Funksjonen info() gir informasjon om alle kolonnene i tabellen.

```python
import pandas as pd

df = pd.read_csv('covid-19.csv')
print(df.info())
```

```
In [147]: df['Country']
Out[147]:
0                  USA
1                India
2               France
3               Brazil
4              Germany
             ...
225                Niue
226        Vatican City
227              Tuvalu
228      Western Sahara
229          MS Zaandam
Name: Country, Length: 230, dtype: object
```

Pandas

- Vi kan hente ut en spesifikk kolonne ved å bruke kolonne-navnet som en indeks

```python
import pandas as pd

df = pd.read_csv('covid-19.csv')
print(df.info())
```

```
In [139]: df.head()
Out[139]:
   Id  Country  TotalCases  ...  TotalTests  Tests1Mpop  Population
0   1      USA    98166904  ...  1118158870     3339729   334805269
1   2    India    44587307  ...   894416853      635857  1406631776
2   3   France    35342950  ...   271490188     4139547    65584518
3   4   Brazil    34706757  ...    63776166      296146   215353593
4   5  Germany    33312373  ...   122332384     1458359    83883596

[5 rows x 13 columns]
```

```
In [142]: df.describe()
Out[142]:
               Id    TotalCases  ...    Tests1Mpop    Population
count  230.000000  2.300000e+02  ...  2.300000e+02  2.300000e+02
mean   115.500000  2.705969e+06  ...  2.050888e+06  3.484620e+07
std     66.539462  8.779899e+06  ...  3.366370e+06  1.383153e+08
min      1.000000  9.000000e+00  ...  5.091000e+03  7.990000e+02
25%     58.250000  2.364900e+04  ...  1.968250e+05  5.476582e+05
50%    115.500000  2.037110e+05  ...  1.061616e+06  5.889248e+06
75%    172.750000  1.256286e+06  ...  2.072495e+06  2.546516e+07
max    230.000000  9.816690e+07  ...  2.200494e+07  1.448471e+09

[8 rows x 12 columns]
```

Pandas

- Andre nyttige pandas-funksjoner for en dataframe df er:
- df.head(n) returnerer de første n radene (hvis parentesen er tom er n = 5)
- df.describe() returnerer et sammendrag av noen egenskaper i hver kolonne, som minimumsverdi, maksimumsverdi, standardavvik, gjennomsnittsverdi, osv.

```python
import pandas as pd

df = pd.read_csv('covid-19.csv')
print(df.info())
```

```
In [148]: df.sort_values(by='TotalDeaths')
Out[148]:
      Id                Country  TotalCases  ...  TotalTests
Tests1Mpop  Population
228  229         Western Sahara          10  ...    31370107
2050888      626161
210  211           Cook Islands        6389  ...       19690
1120596       17571
215  216                  Nauru        4611  ...       20509
1881042       10903
218  219  Saint Pierre Miquelon        3188  ...       24902
4324015        5759
229  230             MS Zaandam           9  ...    31370107
2050888    34846200
..   ...                    ...         ...  ...         ...
```

```
In [154]: df['TotalCases'].mean()
Out[154]: 2705968.9260869566

In [155]: df['TotalCases'].sum()
Out[155]: 622372853
```

Pandas

- df.sort_values(by = 'kolonnenavn') lar oss sortere dataene ut i fra verdiene i en spesifikk kolonne
- df.sum(), df.mean(), df.median() er andre nyttige funksjoner