

Oppgaver: Løkker, lister og filbehandling

Oppgave 1 *Sjekke filtyper*

I denne oppgaven skal du bruke **in**-operatoren for å identifisere .txt filer i en liste av forskjellige filnavn. La oss si at du har en liste med filnavn:

```
1 filnavnliste = ['logg020320.txt', 'resultater1_45.csv',  
    , 'programvare.exe', 'logg030320.txt', 'Screenshot-  
    2020-09-15.png', 'logg040320.txt']
```

- a) Opprett en liste ved å kopiere kodelinjen over.
- b) Print ut det første filnavnet fra listen.
- c) Bruk en **for**-løkke sammen med **in**-operatoren og en betingelse til å skrive ut kun de filnavnene som inneholder teksten **'.txt'**.

Oppgave 2 *Lese .txt filer*

En .txt-fil er en av de enkleste typene filer vi har på en datamaskin, det er simpelthen bare ren tekst som er lagret i filen. Denne oppgaven ser på hvordan vi kan lese inn teksten som er i en .txt-fil.

- a) Lag en txt-fil og lagre denne i samme mappe som du har koden din. Åpne txt-filen og skriv inn noen ord i filen. For eksempel:

```
1 Eple
2 Appelsin
3 Banan
4 Nektarin
5 Fersken
6 Fiken
```

- b) Lag et nytt Python skript. I skriptet, åpne txt-filen med kommandoen `with open` og bruk kommandoen `readline` for å lese ut den første linjen med tekst i filen. Print ut resultatet til terminalen.
- c) For å lese alle linjene i fila kan vi bruke funksjonen `readlines` (legg merke til flertallsendelsen her), den gir oss en liste, der hvert element i listen er en linje i filen. Bruk dette sammen med en `for`-løkke til å printe ut alle linjene i fila.

For oppgave 3. og 4. behøver du filene som ligger i mappen:

oppgaver/oppgaver_dag_3_ressurser som kan hentes fra:

<https://github.com/kodeskolen/simula-kodekurs-v23>

Oppgave 3 *Behandle bestillinger fra fil*

- a) Last ned fila “bestillinger_mat.txt” og lagre den i samme mappe som du har Python koden din.
- b) Koden nedenfor åpner fila og lagrer innholdet i en variabel `tekstfil_innhold`. Kopier koden inn i kode-editoren din og kjør den. Hva printes ut?

```

1 with open("bestillinger_mat.txt", mode="r",
    encoding='utf-8') as tekstfil:
2     tekstfil_innhold = tekstfil.read()
3
4 print(tekstfil_innhold)

```

OBS: hvis du får `FileNotFoundError` bør du dobbeltsjekke at fila ligger i samme mappe som Pythonfila.

- c) Bruk `tekstfil_innhold.splitlines()`-funksjonen og lagre resultatet i en variabel `tekstfil_linjer` og print ut denne nye variabelen. Forklar hva denne variabelen inneholder.
- d) Kodesnutten nedenfor bruker en `for`-løkke for å iterere over alle linjene i "bestillinger_mat.txt" og printe ut litt informasjon for hver linje. Kopier koden til slutten av ditt program og kjør koden. Hva tror du `deler_av_linjen`-variabelen inneholder?

```

1 for linje in tekstfil_linjer:
2     deler_av_linjen = tekstfil_linje.split(',')
3
4     print(deler_av_linjen)

```

- e) Det første elementet i `deler_av_linjen`-variabelen inneholder navnet på bestillingen og det andre elementet inneholder prisen. Modifiser koden slik at du for hver runde i løkka har en variabel `bestilling` som inneholder bestillingen og en variabel `pris` som inneholder prisen. Bruk så en *flettestreng* sammen med `print` til å skrive ut navnet og prisen pent i samme linje. Prøv å gjenskape følgende output:

```
Bestilling: Hummus, Pris: 60,-  
Bestilling: Scampi wok, Pris: 179,-  
Bestilling: Gulrotkake, Pris: 79,-  
Bestilling: Veggie Burger, Pris: 138,-  
...
```

- f) Bruk en tellevariabel til å summere alle prisene i fila og skriv ut totalprisen i slutten av koden. Hvor mye kostet alle bestillingene til sammen? (**HINT:** Bruk `int`-funksjonen for å gjøre om prisen til en tallvariabel).
- g) Kjør programmet med fila “bestillinger_møbler.txt” som inputfil istedenfor “bestillinger_mat.txt”. Hvor mye kostet disse bestillingene til sammen?

Oppgave 4 *Ansatte*

I denne oppgaven ser vi hvordan vi kan få bruk for programmering på en litt mer praktisk måte. Vi skal undersøke en bedrift som har veldig mange ansatte og se at programmering effektiviserer og løser oppgavene mye raskere enn vi kunne gjort for hånd.

- a) Last ned filen *ansatte.csv* og lag et nytt Python-program. Sørg for at begge filene ligger i samme mappe.
- b) Ta en titt på csv-filen ved å åpne den og se hvordan den ser ut. Da ser vi at den øverste delen av filen ser slik ut:

```
1 navn , alder , inntekt , kontor
2 Monica , 27 , 470303 , Stavanger
3 Marianne , 46 , 485394 , Oslo
4 Erik , 44 , 713323 , Stavanger
5 ....
```

Her ser vi at listen inneholder informasjon om de ansatte i en veldig stor bedrift. Den første linjen forteller oss hvordan dataene i filen er ordnet. Vi ser altså at hver linje forteller navnet på den ansatte, alder, inntekt og hvilket kontor de jobber ved.

- c) Vi kan nå begynne å analysere dataene i filen. Les inn alle linjene i filen til en liste i Python. Kall listen for `ansatte`.
- d) Ved hjelp av en for-løkke kan vi gå gjennom alle elementene i `ansatte`. Print ut hver linje for å sjekke at du har fått lest inn filen riktig.
- e) Hvor mange ansatte er det i bedriften? (Hint: Hvor mange linjer er det i fila?)
- f) På hver linje har vi **teksten** eller strengen `"Monica, 27, 470303, Stavanger"`. For å dele denne strengen opp i den informasjonen den inneholder kan vi bruke kommandoen `split(",")` som deler opp denne linjen med tekst ved hvert komma. Da får vi en liste for den ansatte der det første elementet er navnet, den neste alderen også videre. Skriv nå bare ut navnet på den ansatte for hver ansatt i bedriften.
- g) Nå som vi har klart å hente ut de fire bitene med informasjon vi har på hver ansatt kan vi løse de følgende spørsmålene ved programmering. Velg et par av de du synes er mest interessant og prøv å finne svaret. Vanskelighetsgraden øker nedover lista.
 - Hva er gjennomsnittsalderen på de ansatte?
 - Hvor mye bruker bedriften på lønninger?
 - Hva er den høyeste lønningen?
 - Hvor gammel er den yngste ansatte?