

Bonus-pppgaver

God koding!

Oppgave 1 *Kjøpe telefon på kreditt*

Hallgeir har veldig lyst på en ny telefon som koster 2999 kroner. Problemet er bare at han har brukt opp sparepengene sine på andre ting. For å få kjøpt telefonen skaffer Hallgeir et kredittkort som har 30% rente. Etter å ha kjøpt telefonen går det tre år før Hallgeir betaler tilbake kreditten. I denne oppgaven skal vi undersøke hvor mye Hallgeir blir nødt til å betale da.

- a) Opprett variablene renter, originalpris og antall_år og gi med verdiene 30, 2999 og 3
- b) Regn ut vekstfaktoren til renta og lagre den i en variabel rentevækstfaktor
- c) Opprett en variabel, lån. Denne variabelen skal holde orden på hvor stort lånet til Hallgeir er. Til å begynne med er lånet like stort som prisen på telefonen. Sett altså lån-variabelen til å ha verdien 2999
- d) Bruk en **for**-løkke til å simulere hvordan lånet vokser for hvert år. Hint: for hvert år skal lånet ganges med vekstfaktoren du regnet ut i oppgave b)
- e) Oppdater programmet ditt til å skrive ut størrelsen på lånet for hvert år. Hvor mye skylder Hallgeir etter tre år?
- f) Hvor mye ekstra kostet telefonen i forhold til originalprisen?
- g) Gå inn på <https://kredittkort.com/> og se hvilket kort som kommer øverst og noter deg renta. Endre renta i programmet ditt til å matche denne renta. Gå inn på <https://www.prisjakt.no/category.php?k=103> og se hvilken telefon som er mest populær. Endre originalprisen i programmet ditt til å matche prisen til den telefonen. Kjør programmet nå. Hva ville denne telefonen og dette kredittkortet kostet Hallgeir?

Løsning oppgave 1 *Kjøpe telefon på kreditt*

a)

```
1 renter = 30
2 originalpris = 2999
3 antall_år = 3
```

b)

```
1 rentevekstfaktor = 1 + renter/100
```

c)

```
1 lån = originalpris
```

d)

```
1 for år in range(antall_år):
2     lån *= rentevekstfaktor
3
4 print(f"Hallgeir skylder {lån:.2f} kroner etter {
    antall_år} år")
```

```
Hallgeir skylder 6588.80 kroner etter 3 år
```

- e) Det kostet 3589.80 kroner mer enn originalprisen å kjøpe telefonen på kreditt.
- f) Det kredittkortet med lavest rente har en rente på 23,1%
- g) Den vanligste telefonen er en iPhone 11 64GB og koster 7990 kroner. Om vi bruker disse tallene i koden får vi at telefonen koster 14904,62 kroner, det er nesten dobbelt så mye som originalprisen!

Oppgave 2 *Wokoppskrift*

I denne oppgaven skal vi øve på å bruke en funksjon vi ikke har sett på før, nemlig `choice` fra `random`-biblioteket. Det å finne ut og forstå hva nye funksjoner er en viktig ferdighet for en programmerer. Derfor er det lurt å øve på å slå opp i dokumentasjonen.

Nå skal vi lage et program som kan autogenerere wok-oppskrifter, inspirert av denne nettsiden: <https://www.frenchguycooking.com/stir-fry-generator>. For å lage en god wok kan man kombinere en karbohydratkilde, grønnsaker og en proteinkilde. La oss derfor starte med å lage en liste over mulige karbohydratkilder.

- a) Lag en liste, karbohydratkilder med minst tre karbohydratkilder, f.eks. ris, eggnuddele og risnuddele.
- b) Importer funksjonen `choice` fra `random` biblioteket ved å skrive `from random import choice`.
- c) Bruk `help` funksjonen for å lese dokumentasjonen til `choice` ved å skrive `help(choice)` i terminalvinduet. Hva tror du denne funksjonen gjør?
- d) Bruk `choice`-funksjonen for å hente ut en tilfeldig karbohydratkilde fra karbohydratkilder-lista. Lagre karbohydratkilden i en variabel du kaller `karbohydratkilde` og skriv denne karbohydratkilden ut til brukeren
- e) Kjør programmet mange ganger, får du samme utskrift hver gang? Hvorfor får du denne oppførselen?
- f) Lag en liste grønnsaker som inneholder minst fem grønnsaker du kunne tenkt deg i en wok. For eksempel brokkoli, gulrot, spinat, vårløk og erter. Lag også en liste proteinkilder som inneholder noen proteinkilder, f.eks. tofu, kylling, storfe og scampi.
- g) Bruk `choice` funksjonen til å hente ut to tilfeldige grønnsaker og en proteinkilde, lagre disse grønnsakene i variablene `grønnsak1`, `grønnsak2` og proteinkilde.
- h) Skriv ut en fin beskjed til brukeren som beskriver woken du skal ha til middag. F.eks. `eggnuddele-wok med spinat, gulrot og tofu`

Løsning oppgave 2 Wokoppskrift

a)

```
1 karbohydratkilder = ["ris", "eggnuddele", "
```

```
risnuddel"]
```

b)

```
1 from random import choice
```

c)

```
1 help(choice)
```

d)

```
1 karbohydratkilde = choice(karbohydratkilder)
2 print(karbohydratkilde)
```

e) Vi får forskjellig resultat siden choice-funksjonen henter ut tilfeldige element fra karbohydrat-lista.

f)

```
1 grønnsaker = ["brokkoli", "gulrot", "spinat", "vå",
                "røt", "erter"]
2 proteinkilder = ["tofu", "kylling", "storfe", "scampi"]
```

g)

```
1 grønnsak1 = choice(grønnsaker)
2 grønnsak2 = choice(grønnsaker)
3 proteinkilde = choice(proteinkilder)
```

h)

```
1 from random import choice
2
3 karbohydratkilder = ["ris", "eggenuddel", "risnuddel"]
4 grønnsaker = ["brokkoli", "gulrot", "spinat", "vå",
                "røt", "erter"]
5 proteinkilder = ["tofu", "kylling", "storfe", "scampi"]
```

```

6
7
8 karbohydratkilde = choice(karbohydratkilder)
9 grønnsak1 = choice(grønnsaker)
10 grønnsak2 = choice(grønnsaker)
11 proteinkilde = choice(proteinkilder)
12
13 print(f"{karbohydratkilde}-wok med {grønnsak1}, {
    grønnsak2} og {proteinkilde}")

```

Oppgave 3 *Slå opp i dokumentasjonen*

I denne oppgaven skal du teste ut en smakebit på hvordan man kan lage sin helt egne funksjon i Python. Hvis du vil vite mer om dette kan du lese kapittel 8 i kompendiet (https://github.com/kodeskolen/tekna_agder_h20_1/blob/main/kompendium.pdf) Fra før av har du kanskje brukt ferdiglagde funksjoner som print, input og float, men man kan også lage egne funksjoner.

- a) Lim inn koden under inn i et Python script og kjør den. Hva får du ut i terminalen?

```

1 def hils_på(navn)
2     """funksjon som tar inn et navn og skriver ut en
      hilsen til det navnet"""
3     print(f'Hei på deg, {navn}!')
4
5 hils_på("Sofie")

```

- b) Endre 'Sofie' til 'Adla' i hils_på('Sofie'). Hva får du ut til terminalen nå?
- c) Basert på disse forsøkene, hva tror du funksjonen gjør?
- d) Hva tror du er poenget med teksten inne i """ på linje 2?
- e) Bruk `help(hils_på)` til å skrive ut dokumentasjonen til funksjonen

- f) Modifiser koden for å lage en funksjon `si_farvel(navn)` som tar inn et navn og skriver ut en farvelbeskjed (f.eks `f'Ha en fin dag, navn!'`).

Løsning oppgave 3 *Slå opp i dokumentasjonen*

- a) Vi får ut `Hei pådeg, Sofie!`.
- b) Vi får ut `Hei pådeg, Adla!`.
- c) Funksjonen tar inn et navn og skriver ut hilsenen “Hei på deg, NAVN”, hvor NAVN byttes ut med input-navnet.
- d) `"""`-strengen på linje 2 brukes for å lage dokumentasjonen til funksjonen. En slik kommentarstreng kalles gjerne for en dokumentasjonsstreng, eller en docstring.
- e) Når vi bruker `help`-funksjonen på en funksjon så printes dokumentasjonsstrengen til funksjonen ut.

f)

```
1 def si_farvel(navn):
2     """Funksjon som tar inn et navn og skriver ut
   en farvell-beskjed til det navnet."""
3     print(f'Ha en fin dag, {navn}')
```