IN1000: Innlevering 1

Introduksjon

Velkommen til første innleveringsoppgave i IN1000!

Før du går i gang med å løse oppgaven bør du ha jobbet deg gjennom relevante Trix-oppgaver. Om du står fast kan du se nærmere på (og eventuelt få hjelp med) Trix-oppgavene det lenkes til under hver oppgave. Gruppene starter 30. januar 2025.

Du skal løse tre (fire) deloppgaver. (Hvis denne innleveringen var obligatorisk, noe den ikke er om våren, måtte oppgave 0, 1 og 2 være besvart og koden måtte kjøre uten feil.) Les gjennom hver oppgave før du begynner å programmere, og forsøk å løse oppgavene på papir først!

For hvert program du skriver, skal du legge inn en kommentar i toppen av filen som forklarer hva programmet gjør. Videre forventes det at du kommenterer koden underveis, så det blir tydelig hva du har tenkt. Dersom oppgaven sier du skal endre programmet, trenger du kun å levere den endrede løsningen. Andre viktige krav til innleveringen og beskrivelse av hvordan du leverer finner du nederst i dette dokumentet.

NB: Hvis du ikke vet hvordan du bruker en teksteditor (f.eks. VS Code) eller kjører et program i terminalen, er det viktig at du går gjennom forkurset i informatikk her.

Læringsmål

Målet for disse oppgavene er at du skal ha kommet i gang med programmeringen. Du skal dessuten vise at du kan lagre variabler med verdier og gjøre beslutninger ved hjelp av *if*-setninger.

Oppgave 0: Filsystem og kjøring av Python-programmer

Ta bilde av skjermen din som viser navigering i filsystemet i terminalen.
Filnavn: *filsystem.jpg*

1. Åpne terminalen din og start i roten av filsystemet.
2. Finn frem til IN1000-mappen din ved hjelp av kommandoene cd, ls og pwd.
3. Bruk ls-kommandoen i terminalen til å skrive ut innholdet i IN1000-mappen din.
4. Ta skjermbilde av terminalvinduet ditt som viser kommandoene og resultatene av kommandoene, og lever skjermbildet.

Syntes du oppgaven var vanskelig? Ta en titt på eksemplene fra forkurset.

Oppgave 1: Utskrift og innlesing med variabler

Filnavn: *variabler.py*

1. Lag en fil ved navn *variabler.py*.

2. Skriv et program i denne filen som skriver ut ”Hei Student!” til terminalen.

3. Legg til kode der du ber brukeren om å oppgi et navn i form av en tekststreng ved hjelp av funksjonen *input()*, og lagre denne verdien i en variabel *navn*. Skriv så ut “Hei “ og variabelen *navn*.

4. Definer to variabler (velg variabelnavn selv) og gi hver av dem en heltallsverdi (som du også velger selv). Skriv ut variablene på hver deres linje i terminalen.

5. Beregn differansen mellom de to variablene (den første minus den andre) og legg resultatet inn i en ny variabel. Skriv ut “Differanse:” etterfulgt av den tredje variabelen.

6. Be brukeren om å skrive inn et nytt navn, og legg svaret i en ny variabel. Lag enda en variabel ved navn *sammen*, og gi den verdien av det første navnet etterfulgt av det andre navnet. Skriv ut *sammen* på en ny linje.

7. Du skal nå endre verdien av variabelen *sammen*. Endre den ved å slå sammen de to navnene som i forrige deloppgave, men denne gangen skal du legge til “og” med et mellomrom på hver side mellom navnene. For eksempel: Dersom *sammen* først hadde verdien “OlaKari” skal den nå ha verdien “Ola og Kari”. **Viktig:** Du skal utvide programmet ditt ved å endre verdien av variabelen *sammen* på en ny linje, ikke endre linjen der du først definerte variabelen.

Syntes du denne oppgaven var vanskelig? Se Trix-oppgave 1.01, 1.04, 1.06, 1.07 og 1.10
Syntes du denne oppgaven var enkel? Se Trix-oppgave 1.05, 1.12 og 1.13

Oppgave 2: Beslutninger

Filnavn: *beslutninger.py*

1. Skriv et program som ber brukeren om å svare “ja” eller “nei” på om vedkommende har lyst på en brus. Lagre svaret i en variabel.
2. Skriv en if-setning som gjør en beslutning ut fra hva brukeren har skrevet inn:

    a. Hvis brukeren svarer “ja”, skal “Her har du en brus!” skrives ut.

    b. Hvis brukeren svarer “nei”, skal “Den er grei.” skrives ut.

    c. Hvis brukeren svarer noe annet, skal “Det forsto jeg ikke helt.” skrives ut.

Syntes du oppgaven var vanskelig? Se Trix-oppgave 1.08, 1.09 og 1.11
Syntes du oppgaven var enkel? Se Trix-oppgave 1.13, 1.14 og 1.16

Oppgave 3: Problemløsing

Hvis du er usikker på hva oppgaven ber om, gjør antakelser ut fra hva som virker sannsynlig (og gjennomførbart for deg). Beskriv i så fall antakelsene du gjør i en kommentar i programmet.

Filnavn: *meny.py*

1. Lag et program som skriver ut en meny i terminalen. Menyen skal **minst** inneholde **tre** hovedretter, hvorav én vegetarrett (f.eks. biff, torsk og salat). I tillegg skal menyen inneholde **to** tilbehør; ett grønnsakstilbehør (f.eks. gulrøtter) og ett saustilbehør (f.eks. bearnaise).

2. Be brukeren velge én hovedrett og deretter ett tilbehør fra menyen. Les hovedretten og tilbehøret inn i to variabler; én variabel for hovedretten og én for tilbehøret.

3. Hvis brukeren har valgt en hovedrett uten grønnsakstilbehør (f.eks. biff med bearnaise), skal programmet skrive ut: “Du spiser ikke nok grønnsaker!”. Hvis brukeren har valgt både hovedrett og tilbehør med grønnsaker (f.eks. salat med linser), skal programmet skrive ut: “Du har valgt et vegetarmåltid”. Hvis brukeren har valgt en kombinasjon der **enten** hovedretten **eller** tilbehøret (men ikke begge) inneholder grønnsaker, skal programmet skrive ut brukerens valgte kombinasjon av hovedrett og tilbehør (f.eks. “Du har valgt biff med gulrøtter”).

4. Selv et ganske lite program som dette har mange (riktige) løsninger. Skriv et nytt program som løser punkt 3 i oppgaven på en annen måte, men som fortsatt bruker det som ble gjennomgått i ukens forelesning. Kall det andre programmet *meny2.py*.

Syntes du oppgaven var vanskelig? Se Trix-oppgave 1.09 og 1.11
Syntes du oppgaven var enkel? Se Trix-oppgave 1.14, 1.15, 1.16 og 1.18

Krav til innlevering

- Du skal kun levere inn *.jpg*-filen (skjermbildet fra oppgave 0) og alle *.py*-filene.

- Koden din skal inneholde gode kommentarer som forklarer hva programmene gjør.

- Programmene skal ha gode utskriftsetninger som gjør det enkelt for bruker å forstå.

Hvordan levere i Devilry

1. Du skal IKKE lage en .zip-fil

2. Du **skal** også besvare noen spørsmål i kommentarfeltet i Devilry:
    a. Hvordan syntes du innleveringen var? Hva var enkelt og hva var vanskelig?
    b. Hvor lang tid (ca.) brukte du på innleveringen?
    c. Var det noen oppgaver du ikke fikk til? Hvis ja:
        i. For hvilke(n) oppgave(r) fungerer ikke det innleverte programmet ditt?
        ii. Hvorfor tror du programmet ikke fungerer?
        iii. Hva ville du gjort for å få programmet til å fungere hvis du hadde mer tid?

3. Logg inn i Devilry.

4. Last opp *.jpg*-filen (skjermbildet fra oppgave 0) og alle *.py*-filene i samme innlevering. Fyll i tillegg inn kommentarfeltet med svar på spørsmålene under punkt 2.

5. Husk å trykke **lever** og sjekk deretter at innleveringen din er komplett. Du kan levere så mange ganger du vil frem til fristen.

6. Den obligatoriske innleveringen med tilhørende Trix-oppgaver er minimum av hva du bør ha programmert i løpet av en uke. Du finner flere oppgaver for denne uken her.