

# INF1000 - Obligatorisk innlevering 1

Temaer denne uka: *Utskrift til terminal, enkel innlesing, beslutninger og void-metoder.*

## Mål for uken

Mål for uken er at studentene skal beherske utskrift til terminal, variabler av typene String og int, lesing av input fra bruker i terminal, beslutninger basert på innhold i variabler (if/else), og void-metoder uten parametere.

## Oppgaver

For hver av oppgavene (1-5) i denne obligatoriske oppgaven ønsker vi at du skal levere én fil per oppgave. Filnavnet står spesifisert under hver enkelt oppgave. Husk at navnet på klassen inni filen må stemme overens med navnet på filen. Rent praktisk vil dette si at Oppgave 1 under skal inneholde en klasse *Beslutninger*, og ligge i filen med navn *Beslutninger.java*. Deloppgaver (a,b,c osv.) skal ikke ha egne filer.

### Oppgave 1.1)

**Tema:** *Enkel utskrift til terminal, enkel innlesning og beslutninger*

**Filnavn:** *Beslutninger.java*

- a) Lag en variabel *alder* og gi den en verdi (f. eks. alderen din).
- b) Lag en test som printer ut "Du er myndig" hvis *alder* har en verdi større enn eller lik 18, eller printer "Du er ikke myndig" hvis *alder* er mindre enn 18.
- c) Endre programmet slik at *alder* blir lest inn fra terminalen

Synes du denne oppgaven var vanskelig? Se øvingsoppgaver 1.05 og 1.06.

Synes du denne oppgaven var lett? Se utfordringsoppgave 1.07.

## Oppgave 1.2)

**Tema:** Enkel utskrift til terminal, enkel innlesning og beslutninger

**Filnavn:** Billettpris.java

Lag et program som avgjør, basert på alderen til en person, om personen kan få reise med trikken til halv pris. Programmet skal skrive ut hvor mye personen må betale for billett og vi antar at fullpris på en billett er 50kr. Reglene for billettpris er slik at alle under 12 år og alle over 67 år får reise for halv pris, mens alle andre må betale full pris. Programmet skal lese personens alder fra terminal.

Synes du denne oppgaven var vanskelig? [Se øvingsoppgaver 1.08.](#)

Synes du denne oppgaven var lett? [Se utfordringsoppgaver 1.09 og 1.10.](#)

## Oppgave 1.3)

**Tema:** Enkel utskrift til terminal, enkel innlesning og beslutninger

**Filnavn:** LikeVerdier.java

- Lag et program som inneholder to heltallsvariabler, c og d. Gi variablene verdier som du selv velger.
- Sjekk om c og d er like, og skriv ut til skjerm enten "c og d er like" eller "c er ikke lik d".
- Endre programmet slik at c og d leses inn fra terminalen.

Synes du denne oppgaven var vanskelig? [Se øvingsoppgaver 1.11.](#)

## Oppgave 1.4)

**Tema:** Void-metoder uten parametere.

**Filnavn:** Metoder.java

- Lag et program som kommuniserer med brukeren slik at det tar inn et navn og et bosted fra terminalen. Eksempel på hvordan en kjøring av programmet skal se ut:

```
> java Metoder
Skriv inn navn:
> Espen Askeladd
Skriv inn bosted:
> Oslo
Hei, Espen Askeladd! Du er fra Oslo.
```

- b) Flytt koden som leser inn informasjon og skriver ut en hilsen i en egen metode. Kall denne metoden 3 ganger slik at du får lest inn og skrevet ut informasjon om 3 personer. Til denne oppgaven trenger du kun å levere inn oppgave b). Eksempel på kjøring av programmet:

```
> java Metoder
Skriv inn navn:
> Espen Askeladd
Skriv inn bosted:
> Oslo
Hei, Espen Askeladd! Du er fra Oslo.
```

```
Skriv inn navn:
> Snurre Spret
Skriv inn bosted:
> Bergen
Hei, Snurre Spret! Du er fra Bergen.
```

```
Skriv inn navn:
> Minnie Mus
Skriv inn bosted:
> Trondheim
Hei, Minnie Mus! Du er fra Trondheim.
```

Synes du denne oppgaven var vanskelig? [Se øvingsoppgaver 2.01.](#)

## Oppgave 1.5)

**Tema:** *Egen oppgave.*

**Filnavn:** MinOppgave1.java

- a) Skriv en egen oppgavetekst som tar utgangspunkt i temaet beslutninger (if/else). Dette kan for eksempel være et quiz-program som stiller spørsmål og sjekker om svarene er korrekte. I oppgaven din må du gjerne også ta i bruk metoder uten returverdi, men dette er valgfritt.
- b) Løs oppgaven du skrev i a).

NB! Husk at du skal levere både løsningen og oppgaveteksten du skrev i a).

## Krav til innleveringen

1. I obligen skal man levere én fil per oppgave - husk at klassenavnet og filnavnet skal være identisk. Deloppgaver (a,b,c osv.) skal ikke ha egne filer.
2. Klassenavn skal skrives med stor forbokstav.
3. Variabelnavn skal ha liten forbokstav.
4. Oppgaven må kunne kompilere og kjøre på IFI sine maskiner.
5. Kun .java-filen skal innleveres.
6. Ikke bruk æ, ø eller å i innleveringen.
7. Filene skal inneholde gode kommentarer som forklarer hva programmet gjør.
8. Programmet skal inneholde gode utskriftssetninger som gjør det enkelt for bruker å forstå.

## Fremgangsmåte for innleveringer i INF1000

1. Lag en fil som heter README.txt. Følgende spørsmål skal være besvart i filen:
  - (a) Hvordan synes 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:
    - Hvilke(n) oppgave er det som ikke fungerer i innleveringen?
    - Hvorfor tror du at oppgaven ikke fungerer?
    - Hva ville du gjort for å få oppgaven til å fungere hvis du hadde mer tid?
2. Logg inn på Devilry.
3. Lever de 5 .java-filene(ikke .class-filene) samt README.txt i *samme innlevering*.
4. Husk å trykke lever og sjekk deretter at innleveringen din er komplett.

Den obligatoriske innleveringen er minimum av hva du bør ha programmert i løpet av en uke. Du finner flere oppgaver for denne uken her og flere utfordringsoppgaver her.