

# Prosjekt 2023 – TDT4145

Selve oppgaven er beskrevet i eget dokumentet. Dette dokumentet beskriver kravene til leveransene. Dere velger selv om dere vil skrive på norsk eller engelsk, men ikke bland dem.

## Innlevering, del 1: ER-modell og relasjonsdatabaseskjema

**DB1: Frist 13. mars 08:00 (08:00 am) på Blackboard.**

- a) En ER-modell som viser deres fullstendige datamodell. Dere står fritt til å bruke alle ER- og EER-konsepter som er gjennomgått i emnet. Dokumenter de forutsetningene dere har gjort og eventuelle restriksjonene som ikke (kan) uttrykkes gjennom ER-modellen.
- b) ER-modellen oversatt til relasjonsdatabaseskjema (tabeller). Spesifiser nøkler og fremmednøkler for hver tabell. For hver tabell skal du forklare hvorfor tabellen er på fjerde normalform, eventuelt forklar hvorfor du velger å ha den på en lavere normalform (du må forklare hvilken).
- c) Et SQL-script som konstruerer databasen med tabellene. Husk å spesifiser primær- og fremmednøkler, samt andre nødvendige restriksjoner. Dokumenter restriksjoner som ikke uttrykkes i relasjonsdatabaseskjemaet og derfor må håndteres i applikasjonsprogrammene.

Lever besvarelsen som PDF-fil. SQL-scriptet skal legges ved som en .sql-fil. Dokumentet skal være oversiktlig og konsist, og figurene skal være enkle å forstå. Husk å ta med navn på alle gruppemedlemmene og lever på BlackBoard.

## Innlevering, del 2: Realisert databasesystem

**DB2: Frist 24. mars 23:59 (11:59 pm) på Blackboard**

**TogDB implementert i Python med bruk av sqlite3.**

Databaseapplikasjonen skal implementeres i Python (noen brukerhistorier skrives som SQL-script, se beskrivelsen) basert på skjemaet fra første delinnlevering. Brukerhistoriene må være realisert og tilfredsstilt. Enkleste løsning er å lage et tekstbasert grensesnitt som kjører i et terminalvindu (eksempelvis cmd, bash, o.l.). Husk at poenget med oppgaven er å lage modeller, skrive SQL og gjøre databaseaksess fra Python. Følgende skal leveres:

- a) Python kildekode med SQL og SQL-scripts pakket i en zip-fil.
- b) Databasefilen til prosjektets SQLite-database.
- c) De tekstlige resultatene fra brukerhistorienes spørringer.

Dokumentet skal være konsist og evt. figurer enkle å forstå. Inkluder navn på alle gruppemedlemmene og lever på BlackBoard.

## Evalueringskriterier

Følgende kriterier ligger til grunn for vurderingen:

### DB1

1. Bruk av entiteter, relasjoner og attributter. Disse konseptene skal anvendes på en hensiktsmessig måte, dvs. det skal være lett å forstå hva de modellerer.
2. Bruk av nøkler, herunder naturlige og genererte.
3. Bruk av restriksjoner, f.eks. kardinaliteter, i modellen. Er disse anvendt på korrekt vis?
4. Riktig oversetting til relasjonsmodellen. Korrekt beskrivelse av normalformer.
5. Oversetting til SQL-tabeller. Korrekt bruk av SQL, herunder attributtdomener, (fremmed-)nøkkelrestriksjoner og UNIQUE.
6. Dokumentene skal være konsise og figurene enkle å forstå.

### DB2

1. Korrekt bruk av SQL i Python.
2. Forståelig og lesbar kode.
3. Det skal være mulig å reprodusere de leverte resultatene ved hjelp av programmet og databasen som er levert.

Karaktergivningen baserer seg på de generelle beskrivelsene man finner her:

<https://i.ntnu.no/wiki/-/wiki/Norsk/Karakterbeskrivelser+for+teknologiske+fag>