

Første sett obligatoriske oppgaver i INF3100 V2013

Oppgaven skal løses og leveres individuelt.

Gjennomføring og innlevering av oppgaven skal skje i henhold til gjeldende retningslinjer ved Institutt for informatikk, se

<http://www.mn.uio.no/ifi/studier/admin/obliger/>

Enhver innlevering av besvarelse på en obligatorisk oppgave tas som en bekreftelse på at retningslinjene er lest og forstått.

Innleveringsfrist: Fredag 8. mars kl. 24:00.

Fristen er absolutt, og det blir ikke gitt utsettelse. Alle spørsmålene må besvares for å få godkjent besvarelsen.

Oppgave 1

Nedenfor følger et begrepsmessig skjema for en liten relasjonsdatabase som brukes til å registrere utlån fra biblioteket i en bedrift til de ansatte:

Avdelling(AvdKode, AvdNavn, Adresse)

Låner(LånerNr, Navn, AvdKode, Telefon)

Bok(ISBN, Eksemplar, Forfatter, Tittel, Utgivelsesår, Kjøpsår)

Utlån(ISBN, Eksemplar, Lånedato, LånerNr, Returdato)

Primærnøkler er understreket. Dessuten er Telefon kandidatnøkkel i Låner. AvdKode i Låner er fremmednøkkel til Avdeling. (ISBN, Eksemplar) i Utlån er fremmednøkkel til Bok. LånerNr i Utlån er fremmednøkkel til Låner. Nullverdier er tillatt for Returdato i Utlån. I tillegg bestemmer ISBN i Bok entydig forfatter, tittel og utgivelsesår. Og for hver kombinasjon av ISBN, LåneNr og Lånedato i Utlån er det maksimalt ett eksemplar.

Databasesystemet håndterer datoer slik at differansen mellom to datoer gir antall dager mellom de to datoene.

Eksempler: 18.6.2012 - 8.6.2012 = 10 og 1.1.2013 - 31.12.2012 = 1

Dessuten finnes det en systemfunksjon CURRENT_DATE som gir dagens dato.

A. For hver av relasjonene Bok og Utlån skal du besvare/gjøre følgende:

- i. Hvilke FDer gjelder i relasjonen?
- ii. Hvilke kandidatnøkler har relasjonen?
- iii. For hver av FDene under (i), angi hvilken normalform FDen er på. (Skriv først om FDene slik at det er nøyaktig ett attributt i hver av høyresidene.)
- iv. Hva er den høyeste normalformen til relasjonen? Begrunn svaret.
- v. Hvis relasjonen ikke alt er på BCNF, dekomponer relasjonen tapsfritt til et antall relasjoner som alle er på Boyce-Codd-normalform (BCNF).
- vi. Angi hvilke FDer som gjelder for hver av relasjonene i dekomposisjonen.

I resten av oppgaven skal du ta utgangspunkt i det opprinnelige skjemaet og ikke ta hensyn til hva du måtte ha kommet frem til i punkt A.

- B. Skriv et relasjonsalgebrauttrykk og en SELECT-setning som hver for seg lager en liste over alle utlån som er eldre enn fire uker uten at boken er returnert. Listen skal inneholde navn og adresse til låner, og forfatter og tittel på bøker. Listen skal være sortert på avdelingsadresse og, for de med lik avdelingsadresse, på låners navn innenfor hver avdeling.
- C. Skriv et relasjonsalgebrauttrykk og en SELECT-setning som hver for seg finner antall bøker pr. forfatter som har vært utlånt til (de ansatte i) hver avdeling. Sorter resultatet på forfatter og avdeling.

Oppgave 2

Gitt relasjonen \mathcal{R} (A,B,C,D,E,F,G,H,I).

La $\mathcal{F} = \{ACE \rightarrow F, BCDE \rightarrow G, CI \rightarrow E, FI \rightarrow B, A \rightarrow I\}$ være gitt.

- A. Hvilke kandidatnøkler har \mathcal{R} ?
- B. Finn den høyeste normalformen som \mathcal{R} tilfredsstill.

La $\mathcal{D} = \{ABCD, BCDE, ACEFG, GHI\}$ være en dekomposisjon av \mathcal{R} .

- C. Vis at \mathcal{D} ikke er tapsfri med hensyn på \mathcal{F} .
- D. Dekomponer \mathcal{F} tapsfritt til BCNF. Er dekomposisjonen FD-bevarende?
- E. Vis at $CI \rightarrow B$ ikke holder.

Utvid \mathcal{F} med MVDen $CE \twoheadrightarrow F$.

- F. Vis at $CI \rightarrow B$ nå følger fra \mathcal{F} .

Oppgave 3

I denne oppgaven skal dere i utgangspunktet bruke den *fulle* versjonen av filmdatabasen¹ til å besvare spørsmålene under (men se også egen kommentar til deloppgave E). Besvarelsen skal for hvert punkt inneholde (i) SQL-spørringen og (ii) resultatet fra en kjøring av SQL-spørringen.

- A. Finn ut hvor mange filmer som tilhører begge genrene Fantasy og Animation og som samtidig ikke tilhører noen andre genre. (Det er mindre enn 100 slike filmer.)
- B. Skriv ut serietittel, første produksjonsår og antall episoder for alle TV-serier som startet opp det første året databasen har registrert TV-serier fra (dvs. de eldste TV-seriene).
- C. Finn for- og etternavn på alle skuespillere som har deltatt i mer enn 50 kinofilmer og der det for hver film er slik at etternavnet deres kommer sist i alfabetet blant skuespillerne i filmen. Bruk Postgres' innebygde funksjon for sammenlikning av tekststrenger til å bestemme alfabetisk ordning av etternavnene; det betyr at bokstaver med aksenter, som f.eks. Á og Ô, kommer sent i alfabetet. (Det er mindre enn tyve slike personer.)
- D. Finn for- og etternavn på alle personer som har vært regissør og skuespiller i nøyaktig like mange filmer, men som ikke har vært skuespiller i noen av de filmene de har regissert. Ta bare med de som har regissert mer enn ti filmer. Sorter resultatet alfabetisk etter etternavn. (Det er mellom ti og tyve slike personer.)
- E. Finn for- og etternavn på alle mannlige regissører som har laget mer enn 50 filmer, og der det er minst én skuespiller som har vært med i samtlige filmer av denne regissøren. (Det er mellom ti og femten slike personer.)

Denne siste spørringen tar potensielt så lang tid² at vi godtar at du bare kjører spørringen mot det nedskalerte segmentet av filmdatabasen som ligger på ditt lokale arbeidsområde, men da må du i stedet finne de som har lagd mer enn 10 slike filmer for å få ut noe svar. (I det lokale segmentet er det færre enn fem slike.)

¹ Se informasjon om filmdatabasen og om bruk av Postgres på <http://www.uio.no/studier/emner/matnat/ifi/INF3100/v13/undervisningsmateriale/filmdatabasen/filmdatabasen.html>

² Vi kjenner til én spørring som tar i størrelsesorden ett minutt mot den fulle databasen, mens andre, tilsynelatende fornuftig formulerte spørringer, har vist seg å ta mangfoldige timer. Også mot det nedskalerte segmentet kan noen av spørringene i 3E i verste fall ta flere timer.