

UNIVERSITETET I OSLO

Det matematisk-naturvitenskapelige fakultet

Eksamen i:	INF1300 — Introduksjon til databaser
Eksamensdag:	30. november 2015
Tid for eksamen:	09.00 – 15.00
Oppgavesettet er på:	6 sider
Vedlegg:	ORM 2 Graphical Notation, 5 sider
Tillatte hjelpemidler:	Halpin & Morgan: <i>Information Modeling and Relational Databases</i> . 2 A4-ark (4 sider) håndskrevne notater.

Kontroller at oppgavesettet er komplett før du begynner å besvare spørsmålene.

Del A (vekt 60 %)

For å bestå eksamen, må del A sensureres til karakteren E eller bedre.

Lag en ORM2-modell for databasen til et nettsted (kalt FileShare) som har som formål å gjøre det mulig for brukere å lagre, konvertere og dele filer.

- *Du kan dele opp ORM-modellen over flere sider, men det beste er å bruke en side pr ORM-oppgave.*
- *Les igjennom hele del A før du begynner å modellere.*
- *Ha gjerne med kommentarer i modellen. Om du mener noe er uklart i oppgaveteksten, kan du spesifisere med kommentarer i modellen hvilke antakelser du har lagt til grunn.*
- *Tegn og skriv tydelig.*

Oppgave 1: *Lag en ORM2-modell som fanger opp informasjon om brukere og brukergrupper på FileShare:*

En viktig funksjon for nettstedet FileShare er å kunne holde orden på informasjon om brukerne som er registrert på nettsiden. For enkelhets skyld identifiserer FileShare sine brukere internt med unike IDer (f.eks. «324»). Noe av informasjonen om brukere på FileShare er obligatorisk, mens annen informasjon er valgfri. Av obligatorisk informasjon kreves e-postadresse, passord (kryptert, selvfølgelig), og tidspunktet da brukeren registrerte seg på FileShare. FileShare tillater ikke brukere å registrere mer enn én e-postadresse, og den samme adressen kan ikke benyttes av flere brukere. Valgfri informasjon som kan legges inn er for- og etternavn, organisasjonen brukeren arbeider for, stedet der brukeren bor, brukerens telefonnumre, samt en lenke til brukerens private nettside. Merk at hvert telefonnummer som registreres på FileShare bare kan høre til nøyaktig én bruker.

En annen viktig funksjon på FileShare er muligheten til å lagre informasjon om brukergrupper. Når en bruker blir medlem av en brukergruppe, lagrer FileShare tiden dette medlemskapet startet. Dersom et medlemskap avsluttes, så lagrer FileShare når det skjedde. En bruker kan selvsagt være medlem av flere grupper på en gang.

Oppgave 2: *Utvid modellen med informasjon om filer og hvordan de lagres på FileShare:*

En av kjernefunksjonene til FileShare er å lagre informasjon om filer lastet opp av brukere på FileShare. Filer kan lastes opp av brukere, og blir lagret i filsystemet til FileShare. Filene identifiseres internt av en unik kombinasjon av mappen der filen er lagret i filsystemet (f.eks. «/home/jon/xls/») og filens navn (f.eks. «lønn.xls»). FileShare vil lagre informasjon om hvilke brukere som lastet opp hvilke filer, samt når hver fil ble lastet opp til FileShare.

FileShare tillater brukere å laste opp to filtyper: datafiler (dvs. filer som inneholder data slik som regneark og dokumenter) og eksekverbare filer (dvs. programmer – filer som inneholder kjørbare kode). I tillegg til navn og lokasjon må FileShare for datafiler også lagre informasjon om språket til datafiler (f.eks. «test.pdf» er et dokument skrevet på engelsk), og for eksekverbare filer må programmeringsspråket filene ble skrevet i lagres (f.eks. “run.exe” ble skrevet i C). En fil som er lastet opp til FileShare må enten være en datafil eller en eksekverbar fil (dvs. kan den ikke være både en datafil og en eksekverbar fil på en gang).

Oppgave 3: *Utvid modellen med informasjon om konvertering av filer på FileShare:*

Støtte for filkonvertering (og lagring av informasjon om denne konverteringen) er en annen viktig funksjon for FileShare. FileShare tillater sine brukere å konvertere datafiler (f.eks. konvertere fra xls-filer til pdf-filer) ved hjelp av programmer lagret på FileShare (f.eks. en eksekverbar fil «/home/jon/programmer/xls2pdf» tar som input en datafil «/home/jon/xls/lønn.xls» og produserer som utputt en annen datafil «/home/jon/pdfs/lønn.pdf»). Slike eksekverbare filer kan ta et ulikt antall av datafiler som input, men konverteringen resulterer alltid i en enkelt datafil. For eksempel, den eksekverbare filen «/home/jon/programmer/xls2pdf» tar input «home/jon/xls/lønn-jan.xls» og «/home/jon/xls/lønn-feb.xls» som henholdsvis inputparameter en og to, og kombinerer og konverterer disse filene til filen «/home/jon/pdfs/lønn-jan-feb.pdf». Hvis en datafil er et resultat av en konvertering så skal FileShare lagre informasjon om hvilken eksekverbar fil (program) som genererte denne, nøyaktig hvilken input som ble benyttet, samt tidspunkt for når konverteringen fant sted. En bruker kan selvsagt kjøre den samme eksekverbare filen med samme input flere ganger, hvilket resulterer i en ny fil hver gang (selv om innholdet kanskje er det samme).

Brukere kan manuelt oppdatere datafiler (f.eks. for å korrigere tekst eller verdier i datafiler), men kun de filene som ble generert av en konverteringsprosess som beskrevet over kan bli manuelt oppdatert. En bruker som manuelt oppdaterer en datafil må også ha generert denne datafilen (f.eks. en bruker som endrer en datafil må ha generert den samme filen via en konverteringsprosess, og den filen kan ikke genereres eller endres av andre brukere). FileShare holder styr på hvilke brukere som modifiserer hvilke filer.

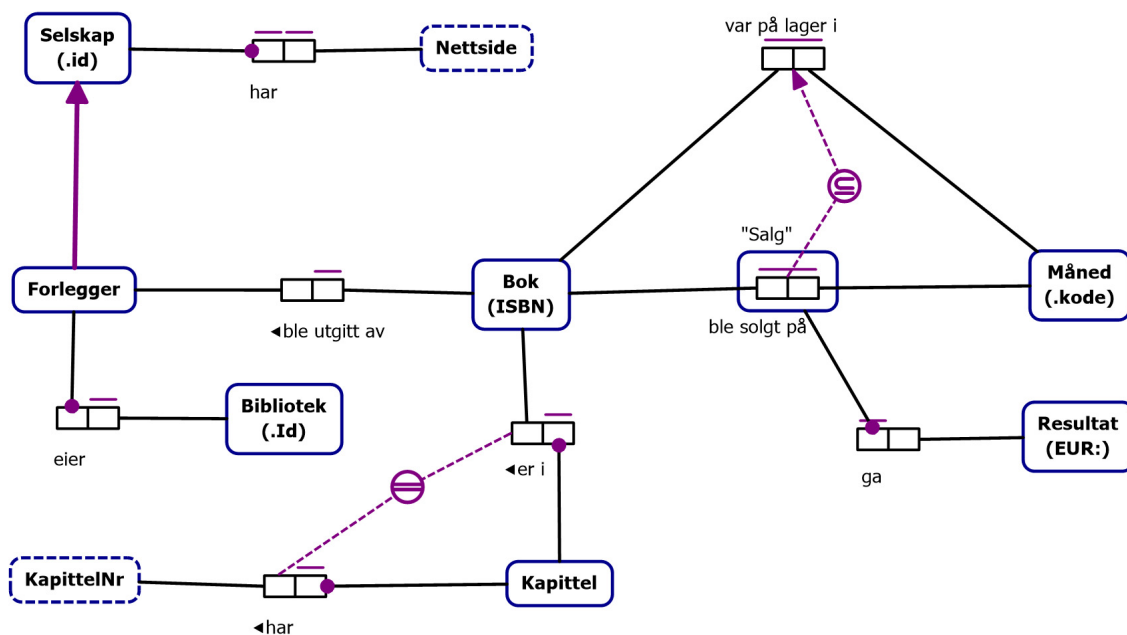
Oppgave 4: *Utvid modellen med informasjon om fildeling hos FileShare og andre skranker som følger:*

- a) Filer skal ha tilgangsbegrensninger. For hver fil holder nøyaktig én av følgende påstander: en fil er delt med en brukergruppe; en fil er privat; en fil er offentlig.

- b) En bruker som lastet opp en fil kan ikke dele denne filen med en gruppe brukeren selv ikke er medlem av. Så, for hver lastet opp fil og brukergruppe så er filen delt med denne brukergruppen hvis og bare hvis denne filen ble lastet opp av en bruker som er medlem av denne brukergruppen. (Merk at denne skranken gjelder bare filer som ble lastet opp, ikke filer som ble generert gjennom konvertering).
- c) Brukere kan følge aktiviteten til andre brukere. (f.eks. bruker med id «234» følger bruker «456» for aktiviteter slik som filopplasting og konverteringer). En bruker kan selvsagt ikke følge sin egen aktivitet. FileShare lagrer informasjon om hvilke brukere som følger hvilke brukere.
- d) Brukere kan endre programmer (f.eks. forandre kildekoden til en eksekverbar fil for en konvertering), men hver gang en eksekverbar fil endres, lagres den som en ny fil (f.eks. en endring av filen «/home/jon/programmer/xls2pdf» kan resultere i filen «/home/jon/programmer/xls2pdf_v01»). FileShare lagrer informasjon om hvilke brukere som genererer hvilke eksekverbare filer, og hvor disse filene stammer fra (f.eks. bruker «234» genererer «/home/john/programs/xls2pdf_v01» ved endring av «/home/john/programs/xls2pdf»). Legg til en passende ringskranke på relasjonen som lagrer informasjon om hvilke eksekverbare filer stammer fra hvilke eksekverbare filer.

Oppgave 5: Grupper den følgende ORM2-modellen til et relasjonsdatabaseskjema.

Relasjonsdatabaseskjemaet skal være korrekt (tilsvare ORM2-modellen), effektivt (unngå redundans og begrenset antallet tabeller), og tydelig (lett å forstå). Bruk grupperingsalgoritmen for å danne et slikt relasjonsdatabaseskjema. For hver relasjon, spesifiser relasjonens navn og navnet til hvert attributt. Du skal ikke spesifisere datatyper/domener for attributtene, og ikke benytte SQL i denne oppgaven. Marker primærnøkler med én strek. Dersom en relasjon har flere kandidatnøkler, marker alle kandidatnøkler med én strek, og primærnøkkelene med to streker. Marker fremmednøkler med piler. Merk: Realiser underbegreper slik at attributtene deles mellom overbegrepet og underbegrepet.



Del B (vekt 40 %)

Oppgave 6

- Skriv create-setninger for tabellene i relasjonsdatabaseskjemaet du laget i oppgave 5.
- Hvilken normalform er den tabellen med flest attributter på? (Er det flere velger du en av dem).

Oppgave 7

I denne oppgaven skal vi bruke følgende relasjon:

Filmgenre (filmid, title, prodyear, genre)

Primærnøkkelen i tabellen er kombinasjonen av filmid og genre (filmid, genre). Videre vet vi også at filmid bestemmer tittel og produksjonsår for en film.

- Bestem alle supernøkler i relasjonen Filmgenre. Skriv ned alle.
- Bestem alle FDene i relasjonen Filmgenre.
- Hvilken normalform er relasjonen Filmgenre på? Begrunn svaret ditt.

I resten av oppgavene skal du skrive SQL-setninger mot filmdatabasen. Her er tabellinfo som du kan få bruk for:

fdb=> \d Film

Column	Type	Modifiers
filmid	integer	
title	text	not null
prodyear	integer	

prodyear har verdien NULL, eller er et årstall større enn 1887.

fdb=> \d Filmparticipation

Column	Type	Modifiers
partid	integer	
personid	integer	not null
filmid	integer	not null
parttype	text	not null

Attributtet parttype angir funksjon i filmen og kan ha følgende verdier:

cast
composer
costume designer
director
editor
producer
writer

I denne oppgaven får du bruk for regissør (director) og skuespiller (cast).

```
fdb=> \d Person
```

Column	Type	Modifiers
personid	integer	
lastname	text	not null
firstname	text	
gender	character(1)	

```
fdb=> \d filmcountry
```

Column	Type	Modifiers
filmid	integer	
country	text	

I de neste oppgavene skal du skrive select-setninger og views. Oppgavene er formulert ut fra hva resultatet av setningen skal være. Definer gjerne views først for å gjøre spørringene mer letteste. Disse kan også komme til nytte i senere oppgaver. Der det står at du skal lage et view, blir det litt trekk dersom du bare skriver select-setningen.

Oppgave 8

En tabell med filmid, filmtittel og produksjonsår for alle filmer som har en tittel som inneholder ordet 'Exam' eller 'Exams'. Unngå og få med filmer med titler som inneholder ord som 'Example', 'Examen', 'Examination', osv. Du kan anta at filmtitler ikke kan inneholde andre tegn enn bokstaver, tall og skilletegnet blank (space).

Oppgave 9

En tabell med titler på film med antall deltagere per film. Definer tabellen som et et view. Kall viewet Oppg9. Attributtene i viewet skal hete tittel og ant_delt.

Oppgave 10

En tabell med navn på alle filmer regissert av Joachim Trier.

Oppgave 11

En tabell med alle land som det er produsert film i, ordnet i stigende alfabetisk rekkefølge.

Oppgave 12

Lag et view med tittel og produksjonsår for alle filmer produsert i Kambodsja ('Cambodia' i Filmcountry).

Oppgave 13

Samme oversikt, men få med en kolonne med navn på regissør og antall skuespillere (film participation.parttype = 'cast') for hver film.

Oppgave 14

Et view med filmer (filmid) som har kun én regissør.

Oppgave 15

Oversikt over titler på filmene med flest regissører.

Oppgave 16

Ser vi på filmproduksjonen pr. år, ser vi stort sett en stigning fra år til år. Men i 1936 var produksjonen 3865 filmer. Da tok det noen år før den igjen var over det antallet.

Skriv en select-setning som skriver ut det første året etter 1936 hvor filmproduksjonen var større enn 3865.

Oppgave 17

Lag et view som for hvert årstall har antall filmer som ble produsert året før.

Oppgave 18

En oversikt over alle produksjonsår hvor filmproduksjonen sank sammenlignet med produksjonen året før. Sorter kronologisk (årstall i stigende rekkefølge).

Oppgave 19

Skriv om denne setningen, slik at resultatet blir samme tabell, men bare med én select. Du trenger ikke ta hensyn til tuplenes rekkefølge.

```
(select F.filmid
  from Film F natural join Filmcountry C)
union
(select filmid
  from Film F
  where not exists (select filmid from Filmcountry
                    where F.filmid = filmid))
```

Oppgavesett slutt. Lykke til!
Dumitru Roman og Stein Michael Storleer