

## Eksamen DB1102 H2021

Hjemmeeksamen 7.desember - 8. desember 2021.

Teller 100% av karakteren i emnet. Vurdering: Bestått / Ikke bestått.

Alle hjelpemidler tillatt. Ikke tillatt å samarbeide med andre.

Når du leverer eksamen, skal du samle besvarelsen din i ett pdf-dokument. Dokumentet inneholder svar på alle oppgavene.

### Oppgave 1 – modellering (30%)

Robinsonekspedisjonen er endelig tilbake på TV-skjermen. Det begynner å bli så mange sesonger nå, at vi trenger en database som kan holde orden på alle episodene, deltakerne og noen hendelser som har skjedd. Hvis du ikke vet hva Robinsonekspedisjonen er, så er det et av de første realityprogrammene på norsk TV. Et kjapt Google-søk vil gi deg mer informasjon.

Vi må ha oversikt over alle sesongene; årstall og hvem som var programleder. Det har vært tre programledere i den norske versjonen (og det er den norske versjonen vi modellerer): Nils Ove Oftebro, Christer Falck og Silje Torp. For en sesong må vi vite hva premien var for å vinne. Vi må også kunne lagre en tekstlig, overordnet beskrivelse av sesongen som helhet. For programleiderne ønsker vi å vite når de er født, og hva de heter.

Vi ønsker å lagre data om hver enkelt episode. En episode hører naturlig nok til en sesong. Vi ønsker å lagre premieredato for alle episodene. Vi trenger nok også å kunne lagre en beskrivelse av episoden.

La oss ikke glemme deltakerne. Vi ønsker å lagre navn, fødselsdato, bosted og yrke (da de var med i Robinson) og hvilken personlig eiendel de valgte å ha med på øya.

En episode har gjerne noen faste hendelser, blant annet:

- En Robinsonkamp der deltakerne i kampen kan vinne immunitet i neste øyråd.
- Øyråd: En avstemming der den med flest stemmer blir stemt ut av ekspedisjonen.

**Robinsonkamp:** Robinsonkampene er individuelle. Vi ønsker å lagre hvem som deltok i kampen, hvilken episode kampen foregikk og hvem som vant immunitet.

**Øyråd:** Vi ønsker å vite hvilke deltakere som *mottok* stemmer i Øyråd. Eks: Kåre Hansen 3 stemmer, Else Jensen 1 stemme og Balder Haukedal 4 stemmer. Vi har ikke mulighet til å lagre *hvem som stemte på hvem*, for det er som oftest ukjent. Vi må også vite i hvilken episode øyrådet ble avholdt.

En sesong avsluttes med en siste avstemming for å kåre en vinner. Noen av deltakerne stemmer på den de mener fortjener seieren, og den med flest stemmer vinner ekspedisjonen. Vi ønsker å ha oversikt over alle disse avsluttende avstemningene som har vært i Robinsonekspedisjonene; hvem som ga sin stemme til hvilken kandidat. For her kan du forutsette at vi kjenner til hvem som stemte på hvem. I siste sesong fikk en av deltakerne anledning til å gi to stemmer. Du må derfor ta høyde for at en deltaker kan gi flere stemmer i denne avstemmingen.

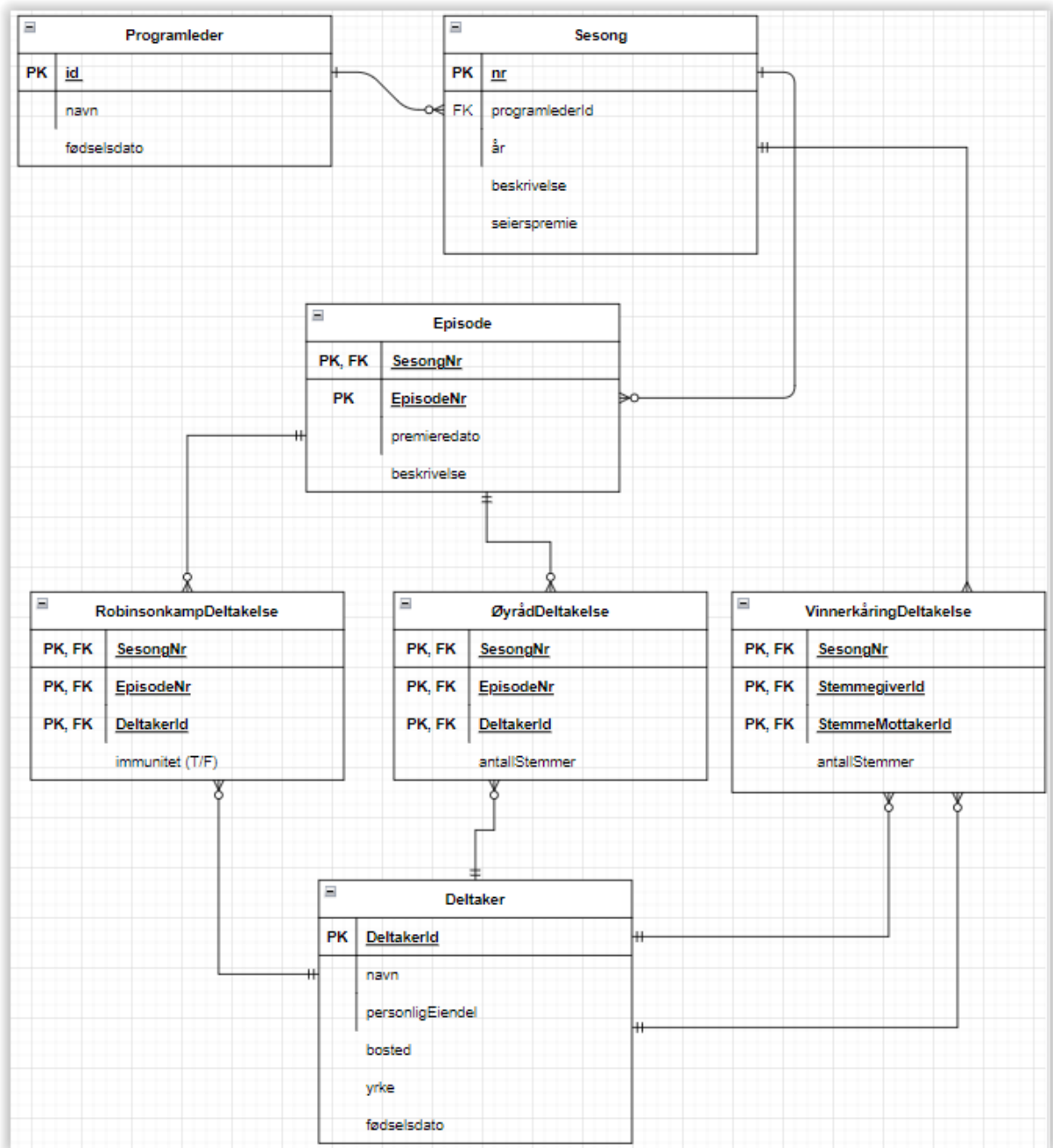
**Oppgave:** Tegn en modell for din foreslåtte løsning. Du kan selv velge om du vil benytte kråkefot eller UML notasjon. Velger du kråkefot trenger du ikke skille mellom identifiserende og ikke-identifiserende forhold. (UML notasjon har uansett ikke skille på dette.) Modellen din skal inneholde:

- Entitetene og deres attributter.
- Primærnøkler og fremmednøkler.
- Relasjonene mellom entitetene.
- Multiplisiteten (deltagelse og kardinalitet) for relasjonene.
- Hvis nødvendig, koblingsentiteter.

Hvis du synes noe er uklart, så gjør dine egne antagelser. Husk i så fall å gjøre rede for disse.

Vi anbefaler å benytte et program, som for eksempel Lucidchart, til å tegne modellen. Du kan også velge å tegne for hånd og lime inn bildet av tegningen din, men det kan da være vanskeligere for sensor å tyde den.

### Løsningsforslag:



Vi må forvente en del forskjellige modeller. Oppgaven er ganske omfattende, og det vil dukke opp noen forskjellige tolkninger.

## Oppgave 2 – SQL (50%)

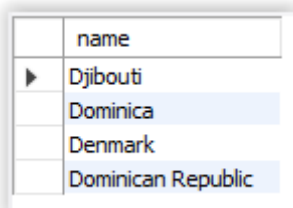
I eksamensoppgaven vil du finne et SQL-script (lotto.sql) som opprettet en database (lotto) og fyller tabeller med data. Kjør SQL-scriptet mot din egen MySQL-server. Deretter løser du oppgavene som beskrevet nedenfor. Svaret på hver deloppgave er todelt:

1. SQL som løser oppgaven.
2. En skjermdump som viser resultatet ditt når SQL er kjørt.

### Eksempel:

Hvis jeg hadde hatt en oppgave fra world-databasen: «Hent ut navn på alle land som begynner på bokstaven D», så ville et svar kunne sett slik ut:

```
SELECT name from country WHERE name LIKE 'D%';
```



	name
▶	Djibouti
	Dominica
	Denmark
	Dominican Republic

VIKTIG! At studentene tar skjermdump er ment for å gjøre det lettere for sensor. Hvis noen ikke har med skjermdump som dokumentasjon av svar så skal det IKKE trekke ned.

Men du skal altså benytte **lotto-databasen** som er vedlagt eksamensoppgaven. Husk at det er bedre å gjøre et forsøk på å besvare en oppgave enn å svare blankt.

- a) Lag en spørring som gir informasjon om hvilken av de registrerte kommunene som har størst innbyggertall, og hva denne kommunen heter.
- b) Lag en spørring som gir informasjon om hvilke registrerte spillere som bor i en kommune som heter Herøy.
- c) Lag en spørring som gir navn på alle registrerte kommuner som har samme navn, men ligger i forskjellige fylker.
- d) Lag en spørring som viser hvor mange registrerte kommuner som har en liten 'u' i navnet sitt. Navngi kolonnen i svaret: Antall U-Kommuner.
- e) Lag en spørring som viser hvilket fylke som samlet sett har hatt flest premievinnere til nå. Resultatet skal vise fylket, og antall vinnere. Hvis noen har vunnet flere ganger, så skal de telles for hver gang de vinner.
- f) Lag en spørring som viser hvilke trekninger som ikke har hatt noen vinnere. Resultatet skal vise trekningens dato, og navnet på hvem som var trekningsansvarlig.
- g) Lag en spørring som viser navn på spillere har vunnet flere enn en gang, hvor mange ganger de har vunnet, og hvilken kommune de bor i.
- h) Legg inn en ny kolonne Areal i kommunetabellen. Legg inn fornuftige verdier i den nye kolonnen for de eksisterende kommunene. Velg datatype du selv mener er passende. Arealet skal oppgis i antall kvadratkilometer, med to desimaler.
- i) Det har vært en ny trekning. Legg inn følgende informasjon i databasen: Trekningen ble avholdt 4. desember 2021. Det var nøyaktig 11 millioner i utbetaling. Det var en ny

trekningsansvarlig: Jens Jensen, som bor i Oslo. Det var to vinnere som delte utbetalingen: Lars Andersen, som bor i Ålesund (Lilliveien 56) og Line Jensen som bor på Elverum (Blåklukkaleen 4).

- j) (Vanskelig) Lag et view som viser hvilke fylker som har vunnet hvor mye penger. Viewet skal inneholde fylkets navn, og totale utbetalinger til fylkets spillere, sortert slik at fylket som har vunnet mest kommer først.

**Løsningsforslag: Se `lotto_sensor.sql`**

### Oppgave 3 – Normalisering (20%)

En bedrift leier ut diverse verktøy og utstyr til privatmarkedet. De har en tabell i en database som holder oversikt over utleien. Kolonnenavn i uthevet (bold) er primærnøkler. Et utdrag av tabellen vises nedenfor:

<b>Kundenr</b>	<b>UtstyrId</b>	Kundenavn	Utstyr	Utlånt	Innlevert
8	37	Morten Hansen	Batteridrill	2021-01-12	2021-01-13
9	15	Lene Jenssen	Sementblander	2021-03-22	2021-04-22
11	24	Hans Hansen	Tilhenger	2021-02-02	2021-02-03
12	15	Andre Jensen	Sementblander	2021-12-06	
12	24	Andre Jensen	Tilhenger	2021-12-06	
12	37	Andre Jensen	Batteridrill	2021-12-06	
13	11	Kari Normann	Batteridrill	2021-11-06	2021-11-08

- a) (5%) Slik tabellen fremstår, hvilke svakheter og begrensninger har denne tabellen?  
En kunde kan ikke leie samme utstyr flere ganger grunnet valgt PK.  
Det er redundans (Kundenavn og Utstyr).
- b) (10%) Normaliser tabellen til 3. normalform. Gjør rede for egne antagelser om dataene der du trenger det. Du kan velge å introdusere nye kolonner om ønskelig.  
Kunde: Kundenr (PK), Kundenavn  
Utstyr: UtstyrId (PK), Utstyr  
Utleie: Id (ny kolonne) PK, Kundenr (FK), UtstyrId (FK), Utlånt, Innlevert  
Kan også velge å ikke introdusere id-kolonne og heller innlemme Utlånt i PK.
- c) (5%) Begrunn hvorfor løsningen din oppfyller kravene til 3. normalform.  
Her kan man forvente en god del forskjellige svar, men det bør refereres til reglene for 2. og 3. normalform og en tydeliggjøring med bruk av dette konkrete caset.

**Lykke til!**