

# Oblig 1 – Databaser vår 2016.

## Hovedformål: øvelse i relativt enkle SQL-spørringer.

Denne oppgaven baserer seg på databasen til tutorial 2 og 3 hos SQLZoo ([www.sqlzoo.net](http://www.sqlzoo.net))

- Pga en forelesning som utgikk, endrer vi fristen til 01.02, men **skulle du få problemer med leveringen er absolutt siste frist 3.02. Leveres på Fronter**. Du kan alternativt levere oppgavene til gruppelærer i timene, og dermed få det godkjent direkte. Dersom dere får problemer med innleveringen, ta kontakt med stud.ass. umiddelbart.
- Det er mange oppgaver og en mange timers arbeid – beregn 15 timer. Bruk lærebøker, eksempler m.m.!
- Du skal ha forsøkt på minst 75% av oppgavene.  
Det aller meste skal være riktig for å få godkjent. Det er jo stort sett enkelt å se at det er korrekt ved å se på hvilket svar spørringen gir.  
Vær forberedt på at det blir muntlig spørsmål alene eller i grupper hvor du forklarer hvorledes du tenkte da du laget de ulike SQL-setningene.
- Pass på at du får hjelp underveis!
- Oppgaven innleveres helst individuelt, evt. i grupper på max. 3. Samarbeid gjerne, men pass å gjøre alt selv. Hvis dere samarbeider, skal grad av samarbeide stå på innleveringen – det blir for dumt hvis vi skal rette to nesten helt like besvarelser hver for seg.  
**Hvis dere har samarbeidet mye, leverer dere inn felles besvarelse (Bruk Leverer for nederst på innleveringen)**
- Del gjerne oppgavene i mindre deler, f.eks. slik at du får noe til å fungere først, deretter legger til f.eks. order by eller andre krav til oppgaven. Se også mye på lærebøker, kompendium og egne notater mens du jobber med å løse oppgaven.
- Tell timer du bruker underveis, ellers kan du ikke svare på siste oppgave!

## Oppgave 1.

(Kan løses ut fra forelesningen 11.01/14.01, samt enkel AND/OR. Noen AND/OR er “lurespørsmål”)

Under følger de spørringene som skal lages. Bruk en av boksene på siden til **tutorial 2** for å testkjøre spørringen, men vær da klar over at sqlzoo vil si at spørringen ikke er korrekt, fordi den er beregnet på en annen spørring enn den du lager. Dessuten vises bare de første 50 selv om resultatet skulle vist mange flere. Selve spørringen skal med i besvarelsen. Det kan også være en fordel for deg, bl.a. til senere repetisjon å ta med svar-tabellen (PrintScreen-paste el.l.), selv om det ikke er noe krav.

- a) List hele tabellen (og med alle kolonnene), sortert fra de største til de minste landene målt i antall innbyggere.
- b) Hent landene som ligger i Europa. Kun landsnavn og innbyggertall skal være med. Det skal være sortert alfabetisk på landsnavn.
- c) Hent ut alle land som ligger i Europa som har over 50 millioner mennesker, sortert fra de største til de minste. Landsnavn og innbyggertall skal med.

- d) Hent ut navn på alle land i Europa, bortsett fra landene Norge, Sverige og Danmark. Hint: en mulig løsning inneholder bl.a. **where** ...=..... **and** name <> «Norway» **and** name <> «Swe.....
- e) En annen løsning er noe med **where** ..... = ..... **and not** (name = «Norway» or name = «Swed  
NB! Pass på parenteser her!
- f) (fortsett at du har lært **in**) Finn en enklere løsning enn i forrige, og hvor du bruker **in**.
- g) Hent ut alle land som ligger i Asia som har over 99 millioner mennesker.
- h) Hent ut navn, befolkningstetthet (personer pr. km<sup>2</sup>) og gjennomsnittlig nasjonalprodukt pr. innbygger for hvert land i Europa. Utskriften skal ha alle tre kolonnene på samme rad. De «fattigste» landene skal først.
- i) Som over (dvs. kun Europa), men sortert slik at de med høyest befolkningstetthet kommer øverst.
- j) Skriv ut alt om landene Norge og Danmark.
- k) Skriv ut alle kontinentene. (Men noen av de kommer jo mange ganger(!), se neste spørsmål)
- l) Som over, men hvert kontinent skal bare skrives ut en gang.
- m) Skriv ut alle land i Amerika. (Hint: se resultatet fra forrige). Kom gjerne med flere ulike løsninger, bl.a. en hvor du bruker **in**.
- n) Hvilke land mangler beskrivelse av antall mennesker i landet?
- o) Hvilke land mangler beskrivelse av enten areal, innbyggertall eller brutto nasjonalprodukt?
- p) Vis navn og befolkning i antall millioner for landene i Afrika. Hint: divider med 1 mill.
- q) Som over, men tallet skal befolkningen i millioner skal skrives med en desimal. Hint: bruk round(<tall>, 1).
- r) Hva er feil med Select \* from world where population = NULL og Select \* from world where population <> NULL? Formuler disse så det blir riktig.

## Oppgave 2.

Her bruker vi tutorial 3, om nobelprisvinnere.

- a) Finn nobelprisvinnere i kjemi, sortert slik at de som fikk prisen første kommer øverst.
- b) Finn nobelprisvinnere som har navn som begynner på samme forbokstav som navnet ditt (hvis navnet ditt begynner på Æ, Ø, Å velger hhv. E, O, A i stedet).
- c) Finn nobelprisvinnere som er «Sir « (i starten av navnet), **og** hvor navnet slutter på «son». Ha gjerne to forskjellige alternativ.
- d) Finn nobelprisvinnere som er «Sir « (i starten av navnet), **eller** har et navn som slutter på «son».
- e) Finn nobelprisvinner som har et navn som inneholder «von» som selvstendig del av navnet, f.eks. Ole von Nes. Hvis von finnes, men ikke som en selvstendig del av navnet, f.eks. Ole Vonlaus, skal det ikke være med.

- f) Hvilke priser (typer) finnes? De skal vises bare en gang.

## Oppgave 3: Mer om land, Tutorial 2.

- a) Skriv ut land navn, folketall og areal for Norge, Sverige og Danmark. Bruk OR
- b) Kan gjøres på en annen måte enn du gjorde i forrige oppgave. Bruk in-operatoren.
- c) Hvilke land bor det mellom 4 og 8 millioner eller over 100000000 mennesker i? Oppgi navn og befolkning. Bruk between.
- d) Som over, men uten between.
- e) Hent ut land i Europa og Asia har mindre enn 10 mill mennesker. Bruk ikke Between
- f) Hent ut land i Asia og Europa som har mindre enn 10 mill mennesker, dessuten alle land i Afrika. Lag en variant med IN, en uten.
- g) Finn hvilke land i Europa som har flere innbyggere enn Norge, uten å vite hvor mange innbyggere som finnes i Norge. Landsnavn og innbyggertall skal med, og det skal sorteres på minkende innbyggertall. **NB! Spørring i spørring, noe med select ... where ... > (select ....). Denne er vanskelig.**
- h) Finn hvilke land som ligger i samme verdensdel som Australia og Kuba / Cuba? Dette skal du finne ut uten å vite hvilken verdensdel Australia og Kuba ligger i. NB! Her må du bruke **select ... where continent in (select continent ...)**. **NB! Denne er vanskelig.**
- i) Hvorfor går det ikke med name = (select name ...)? (Referer til punkt h)
- j) Hva gjør setningen:  
select \* from world  
where continent = 'Africa' and (name like 'M%' or name like '%M')?
- k) Omform denne setningen slik at du ikke bruker parentes (til gjengjeld blir setningen lenger). (Dette kalles en omforming ved å bruke den distributive lov).
- l) Lag deretter SQL-varianten av denne, og kjør den på nytt. Får du samme resultat?
- m) Hva gjør denne setningen?  
SELECT name FROM world WHERE NOT (continent='Europe' OR continent='Africa')
- n) Dette kan omformes slik at du bruker <> (ekvivalent: !=) og AND i stedet. Dette kalles omforming ved Bruk deMorgans lover for å omforme dette uttrykket
- o) Lag tilsvarende SQL-setning – kjør den på nytt og sjekk om du får det samme som over. Du kan gjerne i slutfasen forandre f.eks. not continent = 'Africa' til continent <> Africa.

## Oppgave 4:

Kjør disse setningene – dette blir forberedelse til videre forelesninger. Hva gjør de (1-2 linjers forklaring)?

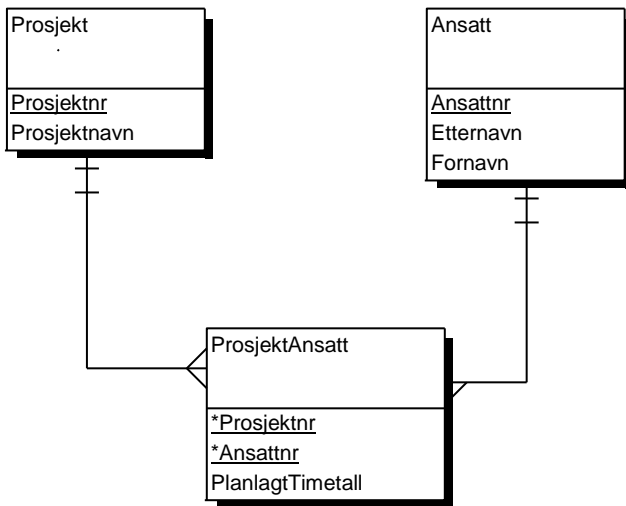
- a) `SELECT SUM(population) FROM world;`
- b) `SELECT count(population) FROM world;`
- c) `SELECT COUNT(*) FROM world;`
- d) `SELECT continent, COUNT(*), sum(population),  
FROM world  
GROUP by continent;`
- e) `SELECT continent, avg(population),  
FROM world  
GROUP by continent;`
- f) `SELECT max(population)  
FROM world`
- g) `SELECT name, max(population)  
FROM world` blir feil. Hvorfor?
- h) `SELECT name, population  
FROM world  
where population in (Select max(population) from world)` Hva gjør denne? Forklar hvorfor det blir korrekt.

## Oppgave 5:

(Kan løses ut fra forelesningen 19/21.01, evt. 26/28.01, samt litt lesning videre i kompendiet.)

**Hvis dårlig tid kan dette ventes med til oblig. 2 (helt eller delvis)**

Det følgende er en datamodell med bl.a. prosjekter og ansatte og hvilke prosjekter de er med på, samt et PlanlagtTimetall (det antall timer vi regner med at den enkelte har planlagt å bruke på dette prosjektet). Et reelt system her naturligvis utrolig mye mer med enn dette!



- Lag CREATE TABLE-setninger for å opprette disse tabellene. Legg inn primærnøkler sammen med CREATE-table-setningen. Sett NOT NULL der det er naturlig.
- Legg til ALTER TABLE-setninger for å legge inn fremmednøkler.
- Prosjektnavn skal dessuten være unikt. Lag en ALTER TABLE for å få til dette.
- Legg inn minimum 5 rader i hver av tabellene (via INSERT-setninger). Et par av prosjektene skal det ikke være medarbeidere på enda.
- Kobl sammen data fra de 3 tabellene som er nevnt over ved å sette betingelser i WHERE-setningen. Du får neppe med alle data fra alle tabeller. Kommenter!
- Gjør det samme, men "glem" å ta med WHERE-setningen. Forklar hvorfor dette blir dumt!
- Skriv alt i Prosjekt koblet med ProsjektMedarbeider ved bruk av INNER JOIN
- Gjør det samme, men ved bruk av LEFT OUTER JOIN
- Du fikk forskjellig resultat. Forklar!

## Oppgave 6:

Hvor mange timer brukte du på oppgavene? Svar ærlig – det spiller ingen rolle for godkjent/ikke-godkjent.