



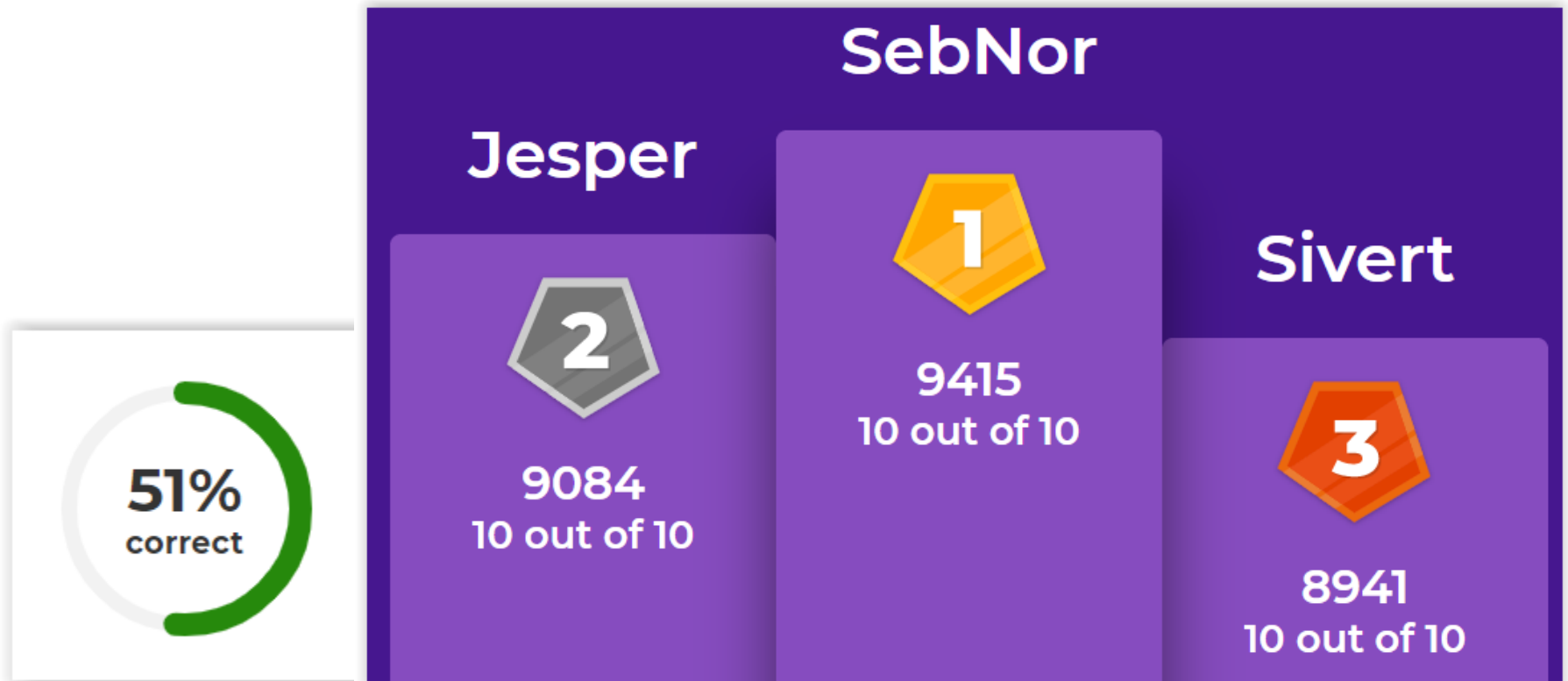
Økt 3 (av 12)

DB1102 Databaser

Per Lauvås / per.lauvas@kristiania.no

Yuan Lin / yuan.lin@kristiania.no

Kahoot



Hvilken egenskap er ikke tilfelle for en tabell?

	Den må inneholde minst én rad.	✓ 	51
	Den skal ikke ha noen like rader.	✗ 	51
	Hver celle skal inneholde én gyldig verdi (eller NULL).	✗ 	93
	Rekkefølgen på kolonnene skal være uten betydning.	✗ 	59
<input type="checkbox"/>	No answer	✗ 	27

Dagens temaer

Dagens tema: [Lage og bruke tabeller.](#)

– Dagens pensum: [Læreboka, kapittel 3.](#)

- Litt repetisjon fra forrige økt.
- Nytt innhold:
 - Datatyper
 - Løpenummer
 - Oppdatering og Endring av tabeller
- Ny SQL i denne sammenheng:
 - CONSTRAINT, INSERT INTO, UPDATE, DELETE FROM, CREATE/ALTER/DROP TABLE, AUTO_INCREMENT, CASCADE

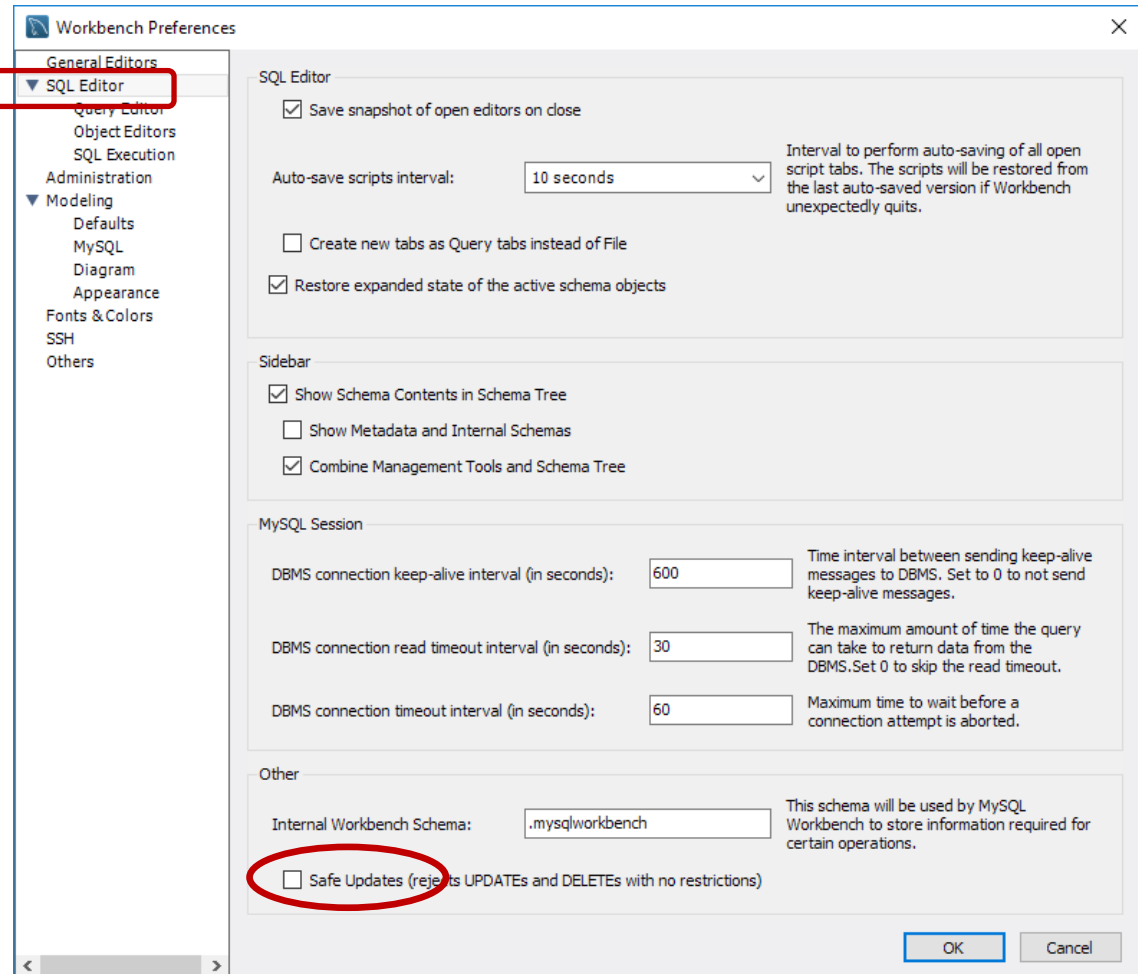
Om øvingene

- «Jeg skjønner ikke dette før jeg ser løsningsforslaget.»
 - Be gjerne om et hint, i motsetning til en løsning.
- Gjør ditt ypperste for å holde tempoet -> Ikke utsette til uken etter.

En innstilling til i MySQL Workbench

VIKTIG!

- Edit -> Preferences...
- **FJERN kryss** i "Safe Updates ..." feltet.
- Hindrer oss fra å gjennomføre enkelte av dagens oppgaver!
- (Mulig du må scrolle ned for å se denne innstillingen i ditt preferences vindu.)



Kort repetisjon, nøkler

- **Primærnøkkel** (PK, primary key) er den unike identifikatoren for radene i en tabell. Kan *aldri* være NULL.
- **Fremmednøkkel** (FK, foreign key) er en (eller flere) kolonne(r) i en tabell som viser til (har samme verdi som) en primærnøkkel i en annen (evt. samme) tabell. Kan være NULL.

Primærnøkkel i
Person tabellen



Person:

Pers_ID	Surname	First_Name	City
0	Miller	Paul	London
1	Ortega	Alvaro	Valencia
2	Huber	Urs	Zurich
3	Blanc	Gaston	Paris
4	Bertolini	Fabrizio	Rom

no relation

Car:

Car_ID	Model	Year	Value	Pers_ID
101	Bentley	1973	100000	0
102	Rolls Royce	1965	330000	0
103	Peugeot	1993	500	3
104	Ferrari	2005	150000	4
105	Renault	1998	2000	3
106	Renault	2001	7000	3
107	Smart	1999	2000	2

Primærnøkkel
i Car tabellen



Fremmednøkkel
i Car tabellen





Mer nøkler

- Primærnøkkel kan være sammensatt av flere kolonner.

LEIEKONTRAKT				
EIENDOMNR	FRADATO	TILDATO	LNR	LEIE

1	01-JUN-05	31-MAY-10	3	6500
1	01-JUN-08		3	7500
2	01-AUG-98	31-JAN-06	4	11000

 PK

 FK

Nøkler, oppgaver – skriv ned svarene dine!

Eier:

Id	Navn	Adresse	Tlf	Epost
1	Ola Olsen	Liksomveien 2	22222222	ola@online.no
2	Ola Olsen	Januarveien 2	33333333	ola@is.com
3	Ina Jensen	Juliveien 3	44444444	ina@is.com

Husdyr:

Id	Dyr	Navn	Eier
1	Katt	Mia	2
2	Hund	Passop	2
3	Papegøye	Polly	1
4	Katt	Kitty	3

1. Hvilke tabeller har vi?
2. Hvilken kardinalitet og grad har tabellene?
3. Hvilke kolonner kan være primærnøkler?
4. Hvilke kolonner kan være fremmednøkler?

Nøkler - oppgaver

City

ID	Name	CountryCode	District	Population
1	Kabul	AFG	Kabul	1780000
2	Qandahar	AFG	Qandahar	237500
3	Herat	AFG	Herat	186800
4	Mazar-e...	AFG	Balkh	127800
5	Amster...	NLD	Noord-H...	731200
6	Rotterd...	NLD	Zuid-Holl...	593321
7	Haag	NLD	Zuid-Holl...	440900
8	Utrecht	NLD	Utrecht	234323

Countrylanguage

CountryCode	Language	IsOfficial	Percentage
ABW	Dutch	T	5.3
ABW	English	F	9.5
ABW	Papiamentu	F	76.7
ABW	Spanish	F	7.4
AFG	Balochi	F	0.9
AFG	Dari	T	32.1
AFG	Pashto	T	52.4
AFG	Turkmenian	F	1.9

Country

Code	Name	Continent	Region	SurfaceArea	IndepYear	Population	LifeExpectancy	GNP	GNPOld	LocalName	GovernmentForm	HeadOfState	Capital	Code2
ABW	Aruba	North America	Caribbean	193.00	NULL	103000	78.4	828.00	793.00	Aruba	Nonmetropolitan T...	Beatrix	129	AW
AFG	Afghani...	Asia	Souther...	652090.00	1919	22720000	45.9	5976.00	NULL	Afganistan/A...	Islamic Emirate	Mohammad Omar	1	AF
AGO	Angola	Africa	Central ...	1246700.00	1975	12878000	38.3	6648.00	7984.00	Angola	Republic	JosÃ© Eduard...	56	AO
AIA	Anguilla	North America	Caribbean	96.00	NULL	8000	76.1	63.20	NULL	Anguilla	Dependent Territor...	Elisabeth II	62	AI
ALB	Albania	Europe	Souther...	28748.00	1912	3401200	71.6	3205.00	2500.00	ShqipÃ«ria	Republic	Rexhep Mejdani	34	AL
AND	Andorra	Europe	Souther...	468.00	1278	78000	83.5	1630.00	NULL	Andorra	Parliamentary Copri...		55	AD
ANT	Netherl...	North America	Caribbean	800.00	NULL	217000	74.7	1941.00	NULL	Nederlandse...	Nonmetropolitan T...	Beatrix	33	AN

1. Hvilke kolonner kan være primærnøkler?
2. Hvilke kolonner kan være fremmednøkler?

Kort repetisjon, SQL spørringer

- Skal vi hente data fra en tabell gjøres det med SELECT.
- Eksempel, opppg. 2-10 fra forrige økt:
 - *"Hent ut kontinent og antall land i dette kontinentet, for de kontinentene som har 30 eller fler land. Sorter på antall land i synkende rekkefølge."*

```
select Continent, count(*) as NumberOfCountries
from country
group by Continent
having NumberOfCountries >= 30
order by NumberOfCountries desc;
```

SQL DDL m.m: endring tabeller

- DDL = Data Definition Language. Del av SQL. Vi har følgende SQL kommandoer for å modifisere tabeller:
 - CREATE TABLE
 - ALTER TABLE
 - DROP TABLE
- Og følgende SQL kommandoer for å modifisere innhold:
(NB – ”vanlig” SQL, ikke del av DDL)
 - INSERT INTO
 - UPDATE
 - DELETE FROM

SQL – opprette tabell

- Skal vi lage en ny tabell, gjøres det med følgende syntaks: (et utvalg av mulighetene)

```
CREATE TABLE tabellnavn
(
kolonnenavn datatype [UNIQUE|NOT NULL|DEFAULT|...],
kolonnenavn datatype [UNIQUE|NOT NULL|DEFAULT|...],
...,
[PRIMARY KEY (kolonnenavn)],
[FOREIGN KEY (kol.navn) REFERENCES tabell_navn (kol.navn)]
);
```

Datatyper i MySQL

- Navn og syntaks for datatyper varierer litt fra database til database.
- Blant MySQLs vanligere datatyper er:
 - `char` og `varchar` ← tekst
 - `[tiny/small/medium/big] int` ← heltall
 - `float` ← desimaltall
 - `enum` ← spesiell, selvdefinert datatype
 - `date` ← datoformat, på formen: 'YYYY-MM-DD'
 - `timestamp` ← auto-tidspunkt (v/insert el. update)
- For utfyllende oversikt over datatyper i MySQL, se f.eks.:
 - http://www.w3schools.com/sql/sql_datatypes.asp

SQL – opprette tabell (forts.)

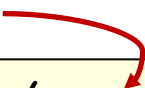
- Eksempel:

```
CREATE TABLE countryIsh (  
  Code CHAR(3) NOT NULL DEFAULT '',  
  Name CHAR(52) NOT NULL UNIQUE,  
  Continent ENUM('Asia','Europe' ) NOT NULL,  
  Region CHAR(26) NOT NULL DEFAULT '',  
  SurfaceArea FLOAT(10,2) NOT NULL DEFAULT 0.00,  
  IndepYear SMALLINT (6) DEFAULT NULL,  
  Population INT(11) NOT NULL DEFAULT 0,  
  PRIMARY KEY (Code)  
)
```

- NOT NULL betyr at kolonnen må fylles inn.
- UNIQUE hindrer to like verdier (får feilmelding ved innleggelse av duplikat).
- DEFAULT brukes for å angi standardverdier.
- Kjør gjerne denne SQL-en!

Løpenummer

- Løpende rekke av tall, benyttes gjerne som primærnøkkel.
- Sikrer at alle verdier i en kolonne er forskjellige.
 - Ergo også unik primærnøkkel om den benyttes som dette.
- Vi kan angi at en kolonne skal benytte et løpenummer v.h.a. ordet AUTO_INCREMENT.



```
CREATE TABLE Persons (  
  ID int NOT NULL AUTO_INCREMENT,  
  LastName varchar(255) NOT NULL,  
  FirstName varchar(255),  
  Age int,  
  PRIMARY KEY (ID)  
);
```


SQL – opprette tabell (forts.)

- Eksempel #2: en tabell med fremmednøkkel til countryIsh tabellen:

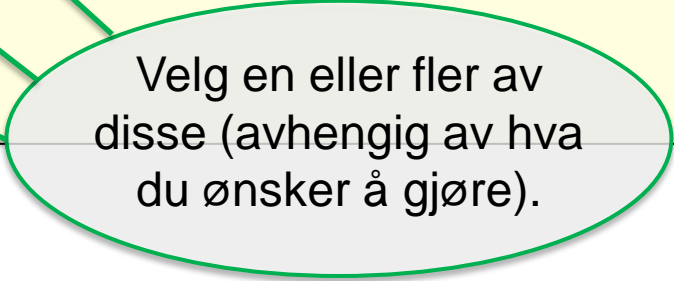
```
CREATE TABLE cityIsh
(
    id INT(11) NOT NULL AUTO_INCREMENT,
    name CHAR(35) NOT NULL DEFAULT '',
    countryCode CHAR(3) NOT NULL,
    district CHAR(20) NOT NULL DEFAULT '',
    population INT(11) NOT NULL DEFAULT 0,
    PRIMARY KEY (id),
    CONSTRAINT fk_city_countryIsh FOREIGN KEY
        (countryCode) REFERENCES CountryIsh(Code)
)
```

- (AUTO_INCREMENT betyr, som alt nevnt, autogenerated løpenummer.)

SQL – endre tabell

- Eksisterende tabeller kan endres med følgende syntaks: (MySQL spesifikt)

```
ALTER TABLE tabellnavn  
ADD kolonnenavn datatype,  
CHANGE kolonnenavn_nå kolonnenavn_ny datatype_ny,  
DROP COLUMN kolonnenavn,  
...
```



Velg en eller fler av disse (avhengig av hva du ønsker å gjøre).

- Eksempel:

```
ALTER TABLE person  
ADD foedselsdato date;
```

SQL – slette tabell

- Å slette en tabell er kort og greit:

```
DROP TABLE tabellnavn;
```

- NB: Men vær sikker på at du sletter rett tabell!
Den blir borte for evig og alltid. ;-)

- Eksempel:

```
DROP TABLE epost;
```

SQL – legge til innhold i tabell

- Å legge til rader i en tabell kan gjøres på to måter.
 - Med eller uten spesifisering av kolonnene:

```
INSERT INTO tabellnavn  
VALUES (value1, value2, value3, ..., valueN);
```

eller

```
INSERT INTO tabellnavn (kolonnenavn1, kolonnenavn2, ...)  
VALUES (value1, value2, ...),  
       (value1, value2, ...),  
       ...
```

- Merk: Tekstfelt (som varchar) skal stå i enkelfnutter.
- Husk: innhold i primærnøkkelceller må være unike!

INSERT INTO – eksempel

```
INSERT INTO countryIsh (Code, Name, Continent)
VALUES ('WWW', 'Wlandet', 'Asia');
```

```
INSERT INTO cityIsh (name, countryCode, population)
VALUES ('Byen min', 'WWW', 12345);
```

- Legg merke til at vi ikke legger inn informasjon i alle kolonner, selv om de ikke tillatter nullmeker.
 - Vi kan gjøre det slik fordi de får satt en DEFAULT-verdi.
- Spørsmål: Er det tilfeldig at det legges inn rader i countryIsh før radene med koplinger til disse legges inn i cityIsh?

SQL – oppdatere innhold i tabell

- Oppdatering av innhold i en tabell kan gjøres for alle rader, eller med en where-clause for et utvalg rader:

```
UPDATE tabellnavn  
SET kolonnenavn1 = value1, kolonnenavn2 = value2, ...  
[WHERE ...]
```

- Eksempel:

```
UPDATE countryIsh  
SET Region = 'Wregion', Population = 74  
WHERE Code = 'WWW';
```

SQL – fjerne innhold i tabell

- Fjerning av innhold i en tabell kan gjøres for alle rader, eller med en where-clause, for enkelte rader: (som for update)

```
DELETE FROM tabellnavn  
[WHERE ...]
```

- Eksempel:

```
DELETE FROM countryIsh  
WHERE Code = 'WWW';
```

- NB: Som en sikkerhetsfeature har MySQL Workbench som default en sperre mot delete from uten where-clause som inneholder key referanse. (Dette kan endres i Preferences.)

What? DELETE fungerte ikke!?!

Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint f... 0.000

- Vi får ikke til å slette fra countryIsh.
 - Vi får ikke lov fordi byen vi la inn har en fremmednøkkel til dette landet.
- Vi kan fikse dette ved å slette byen først. Men la oss forsøke noe annet:
 - Vi vil legge inn en DELETE CASCADE (pensumboka kap. 3.2.7).
 - Da må vi fjerne FK, og legge den inn på nytt (neste side).

CASCADE DELETE, eksempel

- Først fjerne eksisterende FK:

```
ALTER TABLE cityIsh  
DROP FOREIGN KEY fk_city_countryIsh;
```

- Deretter opprette den på nytt med DELETE CASCADE:

```
ALTER TABLE cityIsh  
ADD CONSTRAINT fk_city_countryIsh  
FOREIGN KEY (countryCode) REFERENCES  
CountryIsh(Code) ON DELETE CASCADE;
```

- Forsøker så å slette på nytt:

```
DELETE FROM countryIsh  
WHERE Code = 'WWW';
```

SQL – fjerne hele tabellen

- Det er altså en viktig forskjell på å slette **innhold** i en tabell (rader i en tabell) og å fjerne selve tabellen (inclusive alle rader).
- Hvis vi vil rydde opp etter oss i dette eksempelet, og **fjerne** testtabellene våre:

```
DROP TABLE cityIsh;
```

```
DROP TABLE countryIsh;
```

Lagre/angre

- Når vi endrer data og/eller tabeller i SQL, er det to måter dette kan håndteres på.
- Endringene kan gjøres gjeldende fortløpende.
 - Dette kalles **auto-commit**, og er default settingen i MySQL Workbench.
- Endringene kan gjøres gjeldende først når vi ber om det (ergo kan vi gjøre fler samlet, og de kan angres).
 - Vi styrer dette med SQL-kommandoene `START TRANSACTION`, `COMMIT` og `ROLLBACK`.
 - Finner du knappen for å toggle autocommit i MySQL Workbench?

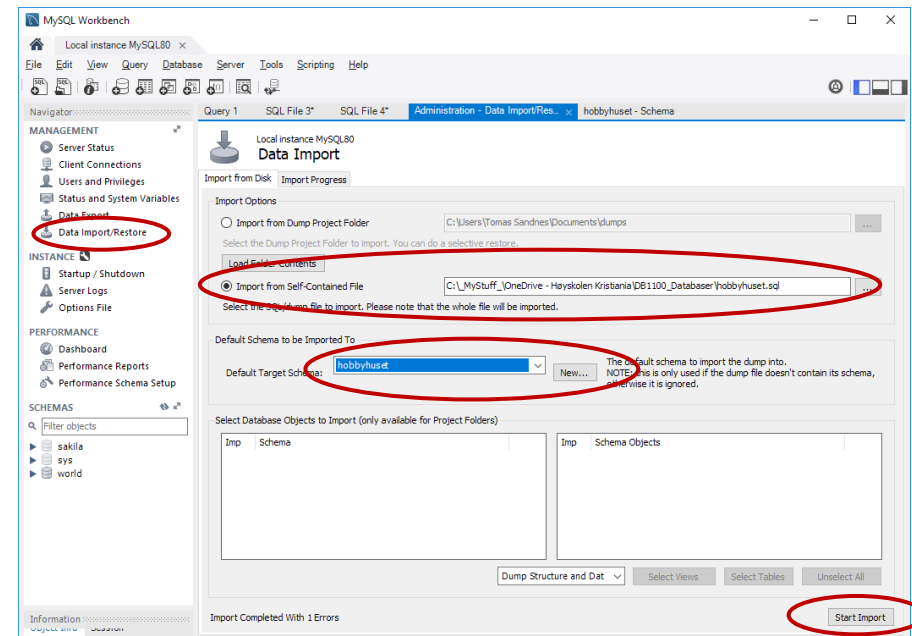
Reinstallere world

- Det kan jo være at man sletter (eller endrer) innhold i world databasen som ikke skulle vært slettet (eller endret).
- Fortvil ikke! For å reinstallere world:
 - Kjør SQL kallet: `DROP SCHEMA world;`
 - Klikk File->Open SQL Script, velg world-scriptet lastet ned fra Canvas, under «Eksterne ressurser".)
 - Klikk lynet!
- FERDIG – (the) world is restored! ;-)



Legge inn Hobbyhuset db

- For de som vil leke seg med oppgavene i boka i tillegg til oppgavene på Canvas, trenger man Hobbyhuset databasen. Fremgangsmåte:
 - Gå til: dbsys.info/Databasesystemer/1_Datasett/sqlskript.html (nettsiden til læreboka). Høyreklikk [HobbyHuset med Eksempeldata] (sjekk hvilket format du bør velge), "Save link as...", lagre på din maskin.
 - I MySQL Workbench, klikk "Data Import/Restore".
 - Velg "Import from Self-Contained File", og finn fila du lastet ned.
 - Klikk [New...] knappen for Default Target Schema, skriv inn "hobbyhuset".
 - Trykk [Start Import] knapp.



SQL – beskrivende kommandoer

- For å se hvilke tabeller vi har i en database/schema:

```
SHOW TABLES  
FROM <db name/schema>;
```

eller:

```
SELECT table_name  
FROM information_schema.tables  
WHERE table_schema = '<db name/schema>;'
```

(NB: Disse to har MySQL spesifikk syntaks.)

- For å se beskrivelse av en tabell:

```
DESC <table name>;
```

Lykke til med ukens oppgaver!

- Neste gang:
 - Kapittel 4: Spørringer mot flere tabeller.