

Gruppetime uke 5 - Mer SQL

Av Katrine :)

Før vi begynner

- Så lenge streiken foregår, vil jeg ha på zoom samtidig som det holdes fysisk
- De som vil være med på zoom trenger ikke melde seg opp



Ting jeg glemte å nevne forrige gang :P

- INNER JOIN
- FROM tabell1 INNER JOIN tabell2 ON (tabell1.kolonne = tabell2.kolonne)
- Ligner på implisitt, men at kolonnene som skal slås sammen på blir satt i FROM
- Husk på at kolonner må ha tabellen den kommer fra foran. Det joines på matchende rader, alle rader som ikke matcher fjernes
- Nested spørring kan også gjøres i SELECT
- Eksempel: `SELECT fn, ln, order = (SELECT count(id) FROM order WHERE o.cid = c.id) FROM c;`

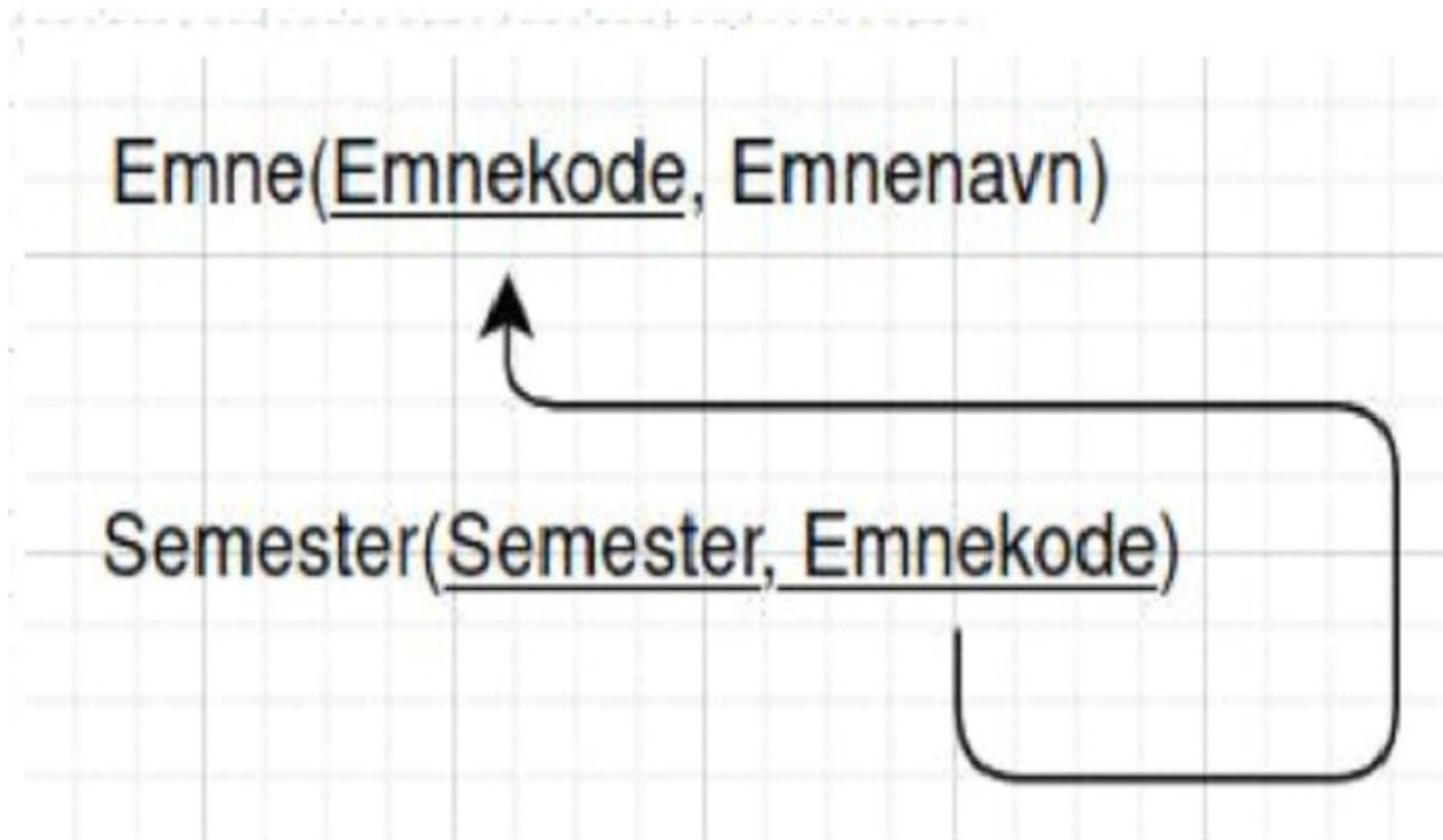


SQL spørring -Typer

- Select - hente data fra tabell
- Create - opprette tabeller, databaser, views
- Insert - sette inn rader i tabell
- Update - oppdatere data i tabell
- Delete - slette rader i tabell
- Drop - slette tabeller, databaser, views



CREATE



- CREATE TABLE Strudent (SID int UNIQUE NOT NULL, StdName NOT NULL, StdBd date);
- NOT NULL hvis det er krav om at alle verdier i kolonnen må ha en verdi, som navn
- UNIQUE hvis det er krav om at alle verdier i kolonnen må være unike, som SID
- MEN! UNIQUE kan fortsatt være NULL
- UNIQUE NOT NULL = PRIMARY KEY
- CONSTRAINT hvis vi vil ha skranker over flere kolonner
- CONSTRAINT name_bd UNIQUE (StdName, StdBd)
- REFERENCES for verdier som bare finnes i en annen tabell
- Course (SID int REFERENCES Students (SID))
- Relasjonsmodell av REFERENCES på bildet



INSERT

- INSERT INTO [tabell] VALUES ([fn], [ln], [SID]);
- Eksempel: INSERT INTO Students VALUES ('Katrine', 'Feten', 123);
- Kan også bruke SELECT i stedet for values
- Select kolonnene som er med i tabellen, som fn, ln og SID over
- Kan også lage tabeller med SELECT på samme måte, men da blir de uten skranker, som må bli lagt til i etterkant
- CREATE TABLE [tabell] AS SELECT
- Med DEFAULT kan det bli oppgitt en verdi om det ikke blir lagt til noen
- fn text DEFAULT 'john', ln text DEFAULT 'doe'
- Med SERIAL kan vi automatisk generere unike heltall, fra 1 til n. Nyttig for primærnøkler som består av heltall
- SID SERIAL PRIMARY KEY
- Hvordan lese inn rader fra fil i forelesningsfoilene
- Overtredelser gir ERROR



DROP og DELETE

- Med DROP TABLE [tabell] sletter vi tabellen
- DROP TABLE [tabell] CASCADE slettes også alle tabeller som er avhengig av denne tabellen
- Med DELETE sletter vi rader
- Samme syntaks som SELECT, bare uten SELECT [kolonner]
- DELETE FROM Students WHERE SID = 123;



ALTER og UPDATE

- Med alter kan vi endre selve tabellen
- `ALTER TABLE Students ADD CONSTRAINT name_db UNIQUE (StdName, StdBd)`
- Med UPDATE endrer vi verdi på kolonner som oppfyller krav
- `UPDATE Students SET StdBd = '1996-04-21' WHERE fn = 'Katrine' AND ln = 'Feten'`



Casting

- Data typen til en kolonne blir lagt til ved opprettelse, så om du skriver et tall med klammer vil den caste det til den oppgitte datatypen
- Casting kan også gjøres eksplisitt med "cast ('1' AS int)" og "int '1'"
- Vi snakket om aggregering forrige gang, men det er også mulig å bruke numeriske typer
- floor, sqrt, cos, random() osv
- datatypene varchar(n) og char(n) har et begrenset antall tegn, så bruk dem som skranker
- Vi har også tidstyper, som timestamp, date, time og interval
- Med EXTRACT kan man hente ut år, måned, etc fra tiden
- EXTRACT (year from date) = 2020 etc
- Og OVERLAPS kan brukes mellom par
- (fromDate1, toDate1) OVERLAPS (fromDate2, toDate2)
- Til slutt vil skranken CHECK la oss bare sette inn verdier som er godkjente av sjekken
- bd date CHECK (bd < '2020-01-01') <- bare lov til å sette inn fødselsdatoer før 1. januar 2020 eller NULL



VIEW - en navngitt spørring

- Blir beregnet hver gang den blir kalt på, i motsetning til lagrede tabeller
- CREATE VIEW [navn] (kolonner datatyper) SELECT-spørring
- Fordi de ikke er lagrede verdier, men endrer seg for hvert kall...
- Kan vi bruke dem for de utledende attributtene fra ER-diagrammene, som alder!
- Hvis de brukes ofte kan de lagres med CREATE MATERIALIZED VIEW [navn] AS SELECT-spørring
- Men det krever at det refreshes med REFRESH MATERIALIZED VIEW [navn];



SQL Scripts

- Se på filen på emnesiden som inneholder et script for å oppretter tabellen ordboken
- Dere vil se at den starter med `DROP TABLE IF EXISTS ordbok_raw CASCADE`
- Hvor den sletter den gamle versjonen av seg selv hvis den allerede finnes
- Det gjør at dere bare kan kjøre filen hvis det er en oppdatering :)
- Så vil dere se at selve opprettelsen av tabellen skjer innenfor `BEGIN COMMIT`



Liten quiz

:D



CREATE TABLE tabell (a int, b int, c int); INSERT INTO tabell
VALUES (1, '2', tre); gir...



SELECT *
FROM tabell
= (1, 2, tre)



SELECT *
FROM tabell
= (1, '2', tre)



Error

Hvis jeg vil fjerne en tabell fullstendig, hvilken spørring bør jeg gjøre det med?



DROP TABLE
tabell;



DROP TABLE
IF EXISTS
tabell
CASCADE;



DROP TABLE
tabell
CASCADE;

ALTER TABLE tabell ADD COLUMN a int; A T tabell ADD
COLUMN b int; A T tabell ALTER a SET DEFAULT '0'; INS I
tabell VAL (NULL, NULL); SEL * FROM tab



(NULL,
NULL)



(NULL, 0)



(NULL, '0')

Leaderboard

No results yet

Top Quiz participants will be displayed here once there are results!



Tid for
eggetarbeid :)

