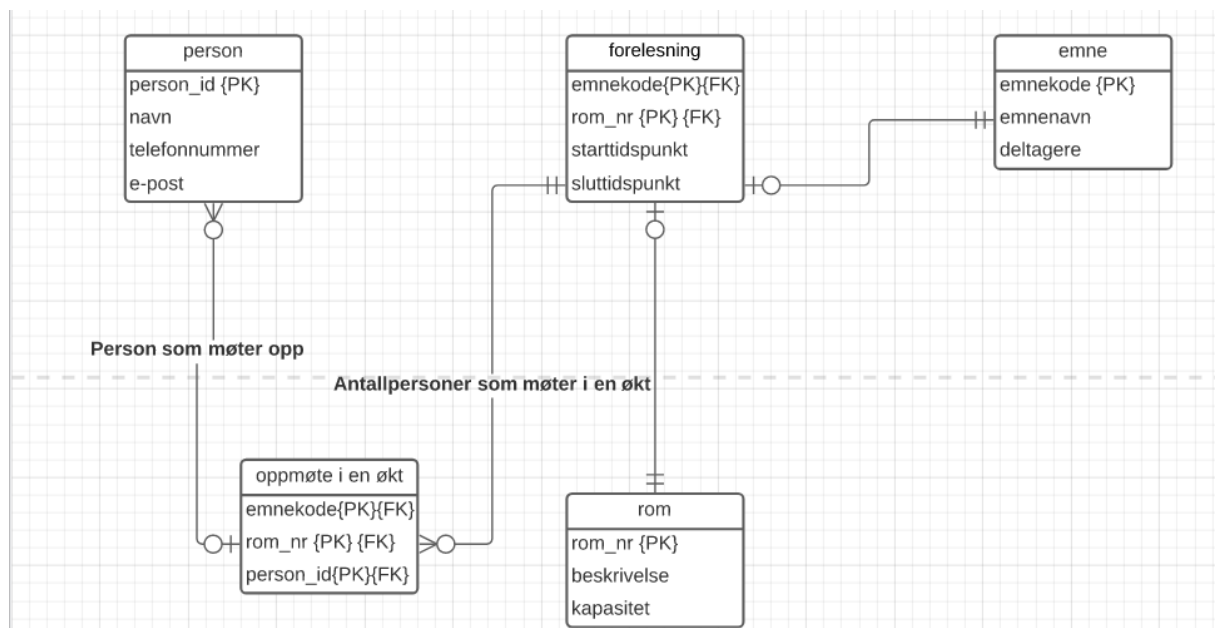


Eksamen DB1102 H2020

Oppgave 1 – modellering



Beskrivelse av modellen:

Ender opp med 5 entiteter, person forelesning, emne, rom og oppmøte i en økt (koplings-entitet) utifra oppgave teksten. Kopling entiteten en økt' tar utgangspunkt i at en person kan kun møte opp i en økt om gangen, mens en forelesning kan ha flere oppmøter per forelesning.

Valgte å bruke en rullerende surigatnøkkel som primærnøkkel for person tabellen i stedet for en naturlignøkkel telefonnummer, synes det er en ryddigere møte å gjøre det på.

Har også satt opp enda en surigatnøkkel (rom_nr) for rom tabellen, siden det et rom lan ha lik beskrivelse og samme kapasitet, da kan disse attributtene brukes som primærnøkkel.

Oppgave 2 – SQL

a)

SELECT * FROM deltaker ORDER BY etternavn ASC, fornavn ASC;

	DNr	Fornavn	Etternavn	EPost
►	15	Anders	Andersen	anders@andersen.no
	6	Benny	Ball	benny@benny.no
	10	Billy	Betong	billy@ppbb.no
	4	Eva	Dahl	eva@dahl.no
	12	Frida	Frosk	frida@ppbb.no
	1	Hans	Hansen	hans@hansen.no
	18	Svetlana	Iversen	svetlana@olsen.no
	8	Hans	Jensen	hj@jensen.no
	3	Jens	Jensen	jens@jensen.no
	16	Julie	Jensen	julie@jensen.no
	7	Oline	Jensen	o-j@jensen.no
	13	Leon	Latex	leon@ppbb.no
	2	Kari	Normann	kari@normann.no
	17	Igor	Olsen	igor@olsen.no
	5	Ole	Olsen	ole@olsen.no
	11	Pelle	Parafin	pelle@ppbb.no
	14	Ragna	Rekkverk	ragna@ppbb.no
	9	Sandra	Salamander	sandra@ppbb.no

b)

SELECT fornavn, etternavn FROM deltaker WHERE epost LIKE '%@ppbb.no';

	fornavn	etternavn
►	Sandra	Salamander
	Billy	Betong
	Pelle	Parafin
	Frida	Frosk
	Leon	Latex
	Ragna	Rekkverk

c)

SELECT dagnr AS Konferansedag,

SUM(måltidpris) AS Måltidpris

FROM måltid

GROUP BY dagnr;

	Konferansedag	Måltidpris
►	1	278
	2	278

d)

Velger å bruke subquery her.

```
SELECT * FROM deltaker WHERE dnr NOT IN(SELECT dnr FROM forfatter);
```

	DNr	Fornavn	Etternavn	EPost
►	15	Anders	Andersen	anders@andersen.no
	16	Julie	Jensen	julie@jensen.no
	17	Igor	Olsen	igor@olsen.no
	18	Svetlana	Iversen	svetlana@olsen.no

e)

```
SELECT fornavn, etternavn, COUNT (temanr) AS temaer FROM deltaker
```

```
LEFT JOIN deltakertema ON deltaker.DNr = deltakertema.DNR
```

```
GROUP BY deltaker.DNr
```

```
ORDER BY temaer DESC;
```

	fornavn	etternavn	temaer
►	Hans	Hansen	5
	Billy	Betong	5
	Oline	Jensen	4
	Hans	Jensen	3
	Pelle	Parafin	3
	Frida	Frosk	3
	Ole	Olsen	3
	Kari	Normann	2
	Eva	Dahl	2
	Leon	Latex	2
	Benny	Ball	1
	Sandra	Salamander	1
	Jens	Jensen	1
	Ragna	Rekkverk	0
	Anders	Andersen	0
	Julie	Jensen	0
	Igor	Olsen	0
	Svetlana	Iversen	0

f)

Legger inn måltidene

```
INSERT INTO måltidbestilling(DNr,Måltidtype,Dagnr) VALUES (4,'Middag',1);
```

```
INSERT INTO måltidbestilling(DNr,Måltidtype,Dagnr) VALUES (6,'Lunsj',2);
```

```
INSERT INTO måltidbestilling(DNr,Måltidtype,Dagnr) VALUES (5,'Middag',2);
```

```
INSERT INTO måltidbestilling(DNr,Måltidtype,Dagnr) VALUES (7,'Lunsj',1);
```

```
INSERT INTO måltidbestilling(DNr,Måltidtype,Dagnr) VALUES (3,'Middag',2);
```

Sjekker at alt er lagt inn

```
SELECT * FROM måltidbestilling;
```

	DNr	MåltidType	DagNr
▶	4	Middag	1
	3	Middag	2
	5	Middag	2
	7	Lunsj	1
	6	Lunsj	2
✱	NULL	NULL	NULL

g)

```
UPDATE deltaker SET EPost = 'svetlana@iversen.no' WHERE Fornavn = 'Svetlana';
```

```
SELECT EPost FROM deltaker WHERE fornavn = 'Svetlana';
```

	EPost
▶	svetlana@iversen.no

h)

```
SELECT etternavn, COUNT(Etternavn) AS forekomst
```

```
FROM deltaker GROUP BY etternavn HAVING forekomst > 1;
```

	etternavn	forekomst
▶	Jensen	4
	Olsen	2

i)

```
CREATE TABLE transport (
    DNr int(11) NOT NULL,
    Dag_nr int(11) NOT NULL,
    Fra_plass ENUM('konferanse','flyplass') NOT NULL,
    Til_plass ENUM('konferanse','flyplass') NOT NULL,

    CONSTRAINT transport_pk PRIMARY KEY (DNr,Dag_nr),
    CONSTRAINT transport_fk FOREIGN KEY (DNr) REFERENCES deltaker(DNr)
);
```

```
INSERT INTO transport(DNr, Dag_nr,Fra_plass,Til_plass) VALUES(8,1,'konferanse','flyplass');
```

```
INSERT INTO transport(DNr, Dag_nr,Fra_plass,Til_plass) VALUES(4,2,'flyplass','konferanse');
```

```
SELECT * FROM transport;
```

	DNr	Dag_nr	Fra_plass	Til_plass
▶	4	2	flyplass	konferanse
	8	1	konferanse	flyplass
●	NULL	NULL	NULL	NULL

j)

Lager spørringen først:

```

SELECT p.RomNr,
CONCAT(DATE_FORMAT(starttid, "%d. %M kl %H.%i-"), DATE_FORMAT(DATE_ADD(starttid,
INTERVAL 20 MINUTE),"%H.%i")) AS tidspunkt,
p.tittel,
CONCAT( d.fornavn, ' ', d.etternavn) AS navn,
t.temanavn, r.antplasser
FROM presentasjon p
LEFT JOIN deltaker d ON p.DNr = d.DNr
LEFT JOIN tema t ON p.TemaNr = t.TemaNr
LEFT JOIN rom r ON p.RomNr = r.RomNr
ORDER BY r.Romnr ASC, tidspunkt ASC;

```

Lager viewet:

```

CREATE VIEW presentasjoner
AS
SELECT p.RomNr,
CONCAT(DATE_FORMAT(starttid, "%d. %M kl %H.%i-"), DATE_FORMAT(DATE_ADD(starttid,
INTERVAL 20 MINUTE),"%H.%i")) AS tidspunkt,
p.tittel,
CONCAT( d.fornavn, ' ', d.etternavn) AS navn,
t.temanavn, r.antplasser
FROM presentasjon p
LEFT JOIN deltaker d ON p.DNr = d.DNr
LEFT JOIN tema t ON p.TemaNr = t.TemaNr
LEFT JOIN rom r ON p.RomNr = r.RomNr
ORDER BY r.Romnr ASC, tidspunkt ASC;

```

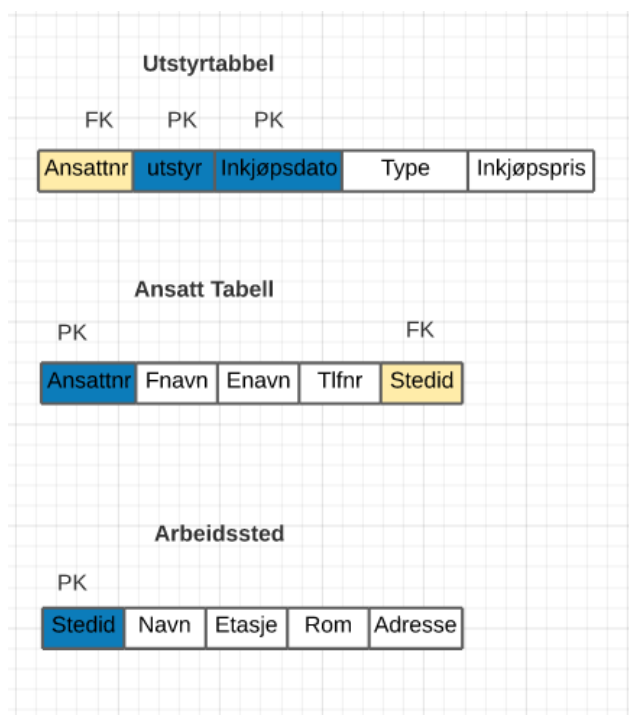
Tester view:

```
SELECT * FROM presentasjoner;
```

	RomNr	tidspunkt	tittel	navn	temanavn	antplasser
►	A1	24. November kl 09.45-10.05	Feasibility of Optimizations Requiring Bounded ...	Hans Hansen	Performance and Optimization	100
	A1	24. November kl 10.15-10.35	Evaluation of graph algorithm frameworks for m...	Kari Normann	Performance and Optimization	100
	F1	24. November kl 09.45-10.05	IT students perceptions of mandatory coursework	Jens Jensen	IT didactics	50
	F1	24. November kl 10.15-10.35	Introducing ePortfolios to IT students: The sup...	Eva Dahl	IT didactics	50
	F1	24. November kl 10.45-11.05	Teaching AI Ethics: Observations and Challenges	Ole Olsen	IT didactics	50
	F1	24. November kl 11.15-11.35	The Live Programming Lecturing Technique: A S...	Benny Ball	IT didactics	50
	F2	25. November kl 09.45-10.05	INERTIA AND CHANGE IN TRANSFORMATION ...	Oline Jensen	Digital transformation	40
	F2	25. November kl 10.15-10.35	DIGITAL TRANSFORMATION UNDER A PANDEM...	Hans Jensen	Digital transformation	40
	F2	25. November kl 10.45-11.05	Exploring the Impact of Mob Programming on th...	Sandra Salamander	Digital transformation	40
	F2	25. November kl 11.15-11.35	Exploring the Hiring Process of a Norwegian Mu...	Billy Betong	Digital transformation	40

Oppgave 3 – Normalisering

Tabell 1:



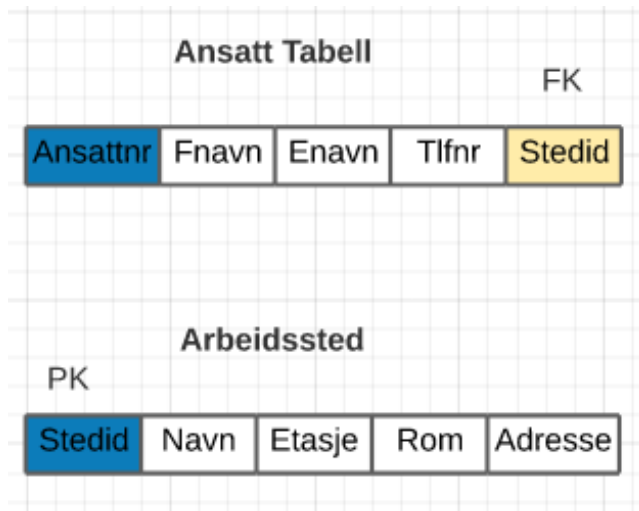
Forklaring til tabell 1:

Tabellen ovenfor er på 3NF fordi attributtene innkjøpspris og type er avhengig av begge primærnøkklene. Eks, for å finne innkjøpspris på type så må man vite både dato og utstyr.

Har tatt ut informasjon om ansatt og lagt dette inn i en egen tabell, siden de attributtene kun er avhengig av ansatnr attributt.

Tillegg har jeg laget en tabell for arbeidssted tabell som har fremmednøkkel Stedid i ansatt tabellen, dette for å gjøre tabellene mer oversiktlige.

Tabell 2:



Oppgave 4 – Diverse

- For å optimalisere databasen kan vi bruke noe som heter for *indekser*, indekser er brukes for å kjapt hente ut kolonner med gitt kolonne verdier. Uten hjelp av indekser så leter spørringen gjennom hele tabellen, noe som kan ta vesentlig lang tid hvis tabellen inneholder mye data.
- Minimere bruken av «krevende» views, siden et view kun er en pre-definert spørringen, så kan den bli tung å kjøre avhengig av kompleksiteten til spørringen.