

Øving 2: Databaser

Oppgave 1: Relasjoner

a) Forklaring av relasjoner

- 1:1 Relasjon:
Dette er når en rad i en tabell er relatert til nøyaktig én rad i en annen tabell
 - Eksempel 1: En person har kun én pass-id, og pass-id er unikt for hver person.
 - Eksempel 2: Et universitet har kun én offisiell rektor, og rektoren har kun ett universitetsid.
- 1:n Relasjon:
Dette er når en rad i en tabell er relatert til flere rader i en annen tabell
 - Eksempel 1: En lærer kan ha flere studenter, men en student har kun én lærer.
 - Eksempel 2: En forfatter kan skrive flere bøker, men hver bok har kun én forfatter.
- n:m Relasjon:
Dette er når flere rader i en tabell er relatert til flere rader i en annen tabell
 - Eksempel 1: Studenter kan melde seg på flere kurs, og ett kurs kan ha flere studenter.
 - Eksempel 2: Skuespillere kan medvirke i flere filmer, og en film kan ha flere skuespillere.

Øving 2: Databaser

Oppgave 2: Tabell "Auditorium"

a) Normalformer

- 1NF: Tabellen oppfyller første normalform (1NF) siden hver celle inneholder bare en verdi, og hver kolonne er av samme datatype.
- 2NF: Tabellen bryter med andre normalform (2NF) fordi det er noen ikke-nøkkelattributter (HAdr og HTlf) som er avhengige av en delmengde av nøkkelen (HSkole), og ikke hele nøkkelen (AKodeID).
- 3NF: Tabellen bryter med tredje normalform (3NF) fordi det er transitive avhengigheter. HAdr og HTlf er avhengige av HSkole, som igjen er avhengig av AKodeID.

b) Lage tabeller

- Del opp tabellen Auditorium i to tabeller:
- **Auditorium:**
 - AKodeID - Entydig kode for hvert enkelt auditorium (eks: AudMax, AudMin, AudG, ...)
 - AntPlass - Antall sitteplasser i auditoriet
 - VKanon - Informasjon om hvorvidt auditoriet har videokanon installert
 - PC - Informasjon om hvorvidt auditoriet har en pc installert
 - HSkole - Informasjon om hvilken høyskole auditoriet tilhører (eks: HiA, ...)
- **Høgskole:**
 - HSkole - Informasjon om hvilken høyskole auditoriet tilhører (eks: HiA, ...)
 - HAdr - Høgskolens adresse
 - HTlf - Høgskolens telefonnummer (her ett telefonnummer pr høyskole)
- Her er AKodeID og HSkole primærnøkklene i Auditorium og Høgskole tabellene henholdsvis. Dette oppfyller 2NF og 3NF fordi det ikke er noen delvis eller transitive avhengigheter.

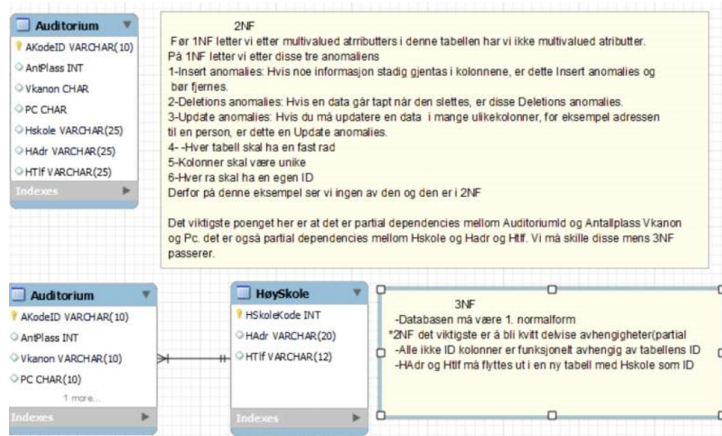


Figure 1: a og b

c) Splitt opp tabell for 3NF

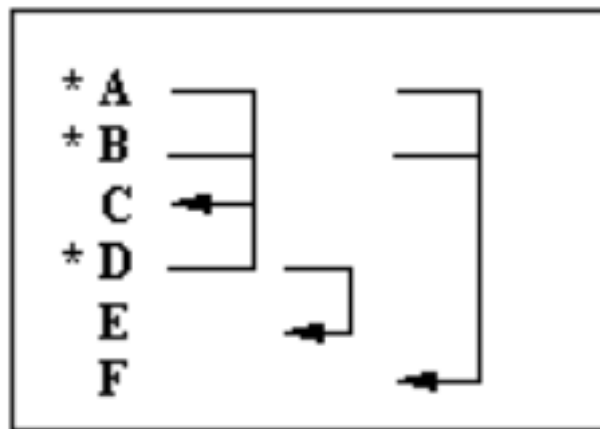


Figure 2: fra oppgave

- For å oppfylle tredje normalform (3NF), kan tabellen splittes opp i følgende måte:
- **Tabell 1:**
 - Kolonner: A*, F
 - Her er A primærnøkkelen.
- **Tabell 2:**
 - Kolonner: B*, C
 - Her er B primærnøkkelen.
- **Tabell 3:**
 - Kolonner: D*, E
 - Her er D primærnøkkelen.
- **Tabell 4:**
 - Kolonner: A*, B*, D*

- Her er A, B, og D primærnøkklene.
- I denne strukturen er det ingen transitive avhengigheter, og hver ikke-nøkkelattributt er fullstendig funksjonelt avhengig av hele nøkkelen. Derfor oppfyller denne strukturen tredje normalform (3NF).

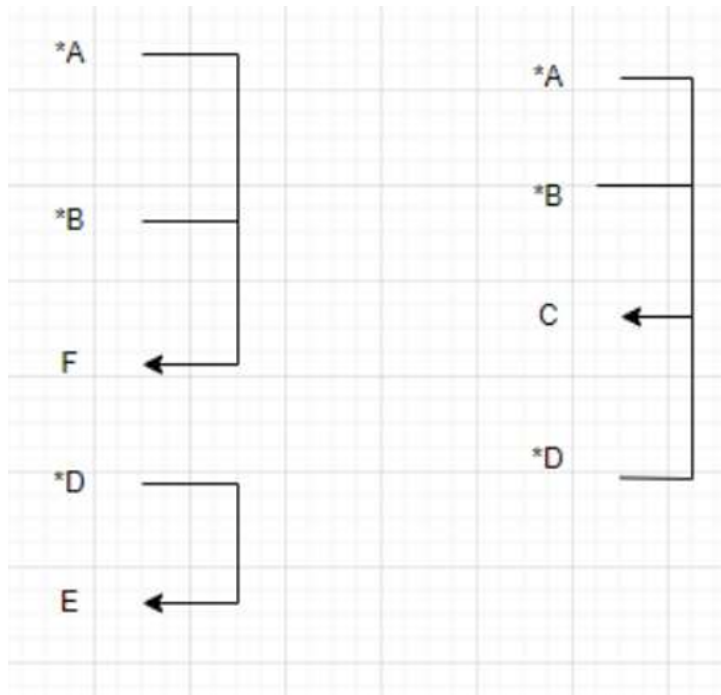


Figure 3: svar

Oppgave 3: Datamodell for legekonsultasjoner

a) Datamodell

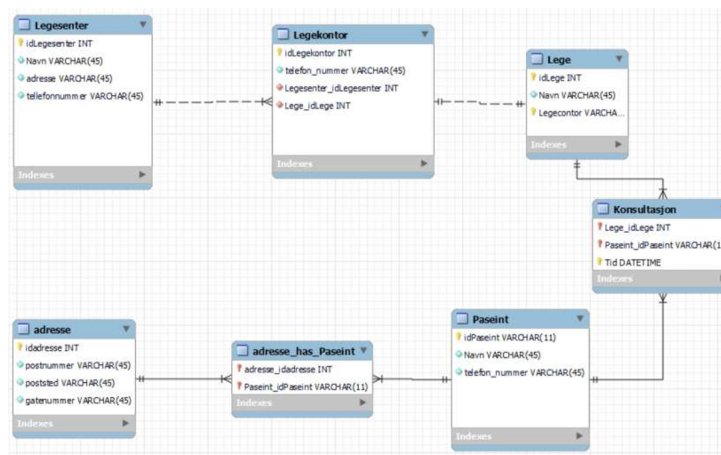


Figure 4: a og b

- Legesenter (LegesenterNavn, Adresse, Telefonnummer)
- Legekantor (KontorNummer, Telefonnummer, *LegesenterNavn*)
- Lege (LegeKode, Fornavn, Etternavn, *KontorNummer*)
- Pasient (Personnummer, Fornavn, Etternavn, Adresse, Telefonnummer)

b) Forklaring av tabeller

- Legesenter (LegesenterNavn)
- Legekontor (KontorNummer, *LegesenterNavn*)
- Lege (LegeKode, *KontorNummer*)
- Pasient (Personnummer)

Oppgave 4: SQL-script

a) MySQL-script for datamodellen i oppgave 3

```
CREATE TABLE Lege (  
    LegeId      Integer NOT NULL,  
    Navn        Char(20),  
    KontoId     int,  
    PRIMARY KEY (LegeId),  
    FOREIGN KEY (KontoId)  
    REFERENCES legeKonto(KontoId)  
    ON DELETE RESTRICT  
);  
  
CREATE TABLE Pasient(  
    Personnummer varChar(11) NOT NULL,  
    Navn         Char(20),  
    Telefonnummer VarChar(12),  
    PRIMARY KEY (Personnummer)  
);
```

Figure 5: svar

```

1 • create database Legekonsultasjon;
2
3 • use legekonsultasjon;
4
5 • CREATE TABLE Legesenter (
6     LegesenterId int,
7     Navn Char(25) NOT NULL,
8     Adresse VarChar(20),
9     Telefonnummer VarChar(12),
10    PRIMARY KEY (LegesenterId)
11 );
12 • CREATE TABLE LegeKonto (
13     KontoId Integer NOT NULL,
14     Telefonnummer VarChar(12),
15     LegesenterId int,
16     PRIMARY KEY (KontoId),
17     FOREIGN KEY (LegesenterId)
18         REFERENCES Legesenter(LegesenterId)
19         ON DELETE RESTRICT
20 );

```

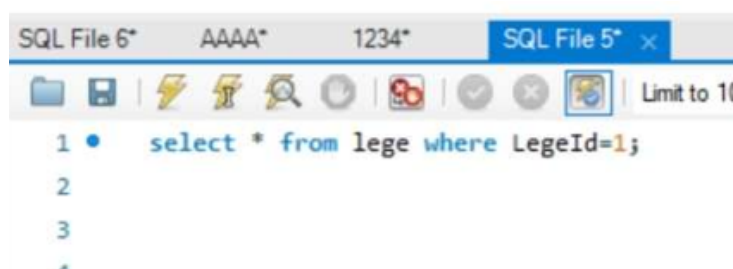
Figure 6: svar

b) MySQL-script for å legge inn data

```
1 • use legekonsultasjon;
2 • INSERT INTO address
3     VALUES ("ObereStr57", "Berlin", 12209, "03040076545");
4
5 • INSERT INTO address
6     VALUES ("AvdaConstitución 222", "Mexico", 05021, "55553745346");
7
8 • INSERT INTO address
9     VALUES ("120 Hanover Sq.", "London", 52066, "51715557788");
10
11 • INSERT INTO address
12     VALUES ("54, rue Royale", "Nantes", 44000, "40332212190");
13
14 • INSERT INTO address
15     VALUES ("C/ Moralarzal, 86", "Madrid", 28034, "91555944467");
16
17 • INSERT INTO pasient
18     VALUES ("03040076545", "Maria Anders", "5035557555");
19
20 • INSERT INTO pasient
21     VALUES ("55553745346", "Lino Rodriguez", "055509876");
22
23 • INSERT INTO pasient
24     VALUES ("51715557788", "Helen Bennett", "6045553392");
25
26 • INSERT INTO pasient
27     VALUES ("40332212190", "Daniel Tonini", "5095557969");
28
29 • INSERT INTO pasient
30     VALUES ("91555944467", "Isabel de Castro", "0372-035188");
31
32
33 • INSERT INTO legesenter
34     VALUES (1, "Wilman Kala", "Keskuskatu 45", "90224885890");
35
36 • INSERT INTO legesenter
37     VALUES (2, "Vaffeljernet", "Smagsloget 45", "06223344556");
38
39 • INSERT INTO legesenter
40     VALUES (3, "Santé Gourmet", "Boulevard Tirou, 255", "1715555646");
41
42 • INSERT INTO legesenter
43     VALUES (4, "Queen Cozinha", "Alamedados 891", "9075552880");
44
```

Figure 7: svar

c) Egendefinerte SELECT-spørringer



The screenshot shows a MySQL query editor window with a toolbar at the top. The toolbar includes icons for file operations (new, open, save, print), editing (undo, redo, find, replace), and execution (run, stop, limit). The main text area contains the following SQL query:

```
1 • select * from lege where LegeId=1;
2
3
4
```

Figure 8: spørring 1

Result Grid			
	LegeId	Navn	KontoId
▶	1	Liu Wong	2
●	NULL	NULL	NULL

Figure 9: spørring 1

```

SQL File 6*  AAAA*  1234*  SQL File 5* x  Query 1
Limit to 1000 rows
9
10 • select * from Lege where Navn="Pascale Cartrain";
11

```

Figure 10: spørring 2

Result Grid			
	LegeId	Navn	KontoId
▶	2	Pascale Cartrain	3
●	NULL	NULL	NULL

Figure 11: spørring 2

```

SQL File 6*  AAAA*  1234*  SQL File 5* x  Query 1
Limit to 1000 rows
9
10 • select * from Lege where not Navn="Pascale Cartrain" and kontoid>2;
11

```

Figure 12: spørring 3

```

SQL File 6*  AAAA*  1234*  SQL File 5* x  Query 1
Limit to 1000 rows
9
10 • select * from Lege where not Navn="Pascale Cartrain" and kontoid>2;
11

```

Figure 13: spørring 3