IN2090
HØST 2021

Obligatorisk innlevering 3 – Databasedesign

Oppgave 1

Lager først tabell for tog, og velger her å bruke UNIQUE NOT NULL for TogNr, da dette er primærnøkkelen i denne tabellen, men også fremmednøkkel i de neste tabellene som skal opprettes.

```
CREATE TABLE tog(
  TogNr INT PRIMARY KEY UNIQUE NOT NULL,
  StartStasjon TEXT NOT NULL,
  EndeStasjon TEXT NOT NULL,
  AnkomstTid TIME NOT NULL
);
```

For TogNr i togTabell bruker jeg også UNIQUE NOT NULL for TogNr, da TogNr her referer til et til toget som kjører på en bestemt rute, og at toget kjører fra en bestemt stasjon.

```
CREATE TABLE togTabell(
  TogNr INT UNIQUE NOT NULL,
  AvgangsTid TIME NOT NULL,
  Stasjon TEXT UNIQUE NOT NULL,

  PRIMARY KEY(TogNr, AvgangsTid)
  FOREIGN KEY(TogNr) REFERENCES tog(TogNr)
);
```

I tabellen for plass, valgte jeg å sette at dato også er UNIQUE NOT NULL, fordi dato attributtet beskriver plass på en bestemt dag.

```
CREATE TABLE Plass(

  Dato DATE UNIQUE NOT NULL,

  TogNr INT UNIQUE NOT NULL,

  VognNr INT NOT NULL,

  PlassNr INT NOT NULL,

  Vindu BOOLEAN NOT NULL,

  Ledig BOOLEAN NOT NULL,


  PRIMARY KEY(Dato, TogNr, VognNr. PlassNr)

  FOREIGN KEY(TogNr) REFERENCES tog(TogNr)

);
```

Oppgave 2

Vi har gitt relasjonen R(A, B, C, D, E, F, G) med FD'er:

1. C D E → B
2. A F → B
3. B → A
4. B C F → D E
5. D → G

a) Hvilke kandidatnøkler har R?
Dersom vi har funksjonell avhengighet fra X til Y, vil det si at for at X skal være en kandidatnøkkel (må ha unik verdi), så må X → Y, der Y finnes i R.
Gitt FD'ene over kan vi sortere de slik:

Aldri på høyresider: C, F

Bare på høyresider: G

Altså begynner vi med CF og forsøker å utvide med: A, B, D, E, siden det er slik at hvis et attributt ikke forekommer i noen høyreside i R, må det være med i alle kandidatnøklene.

X = C, F

X^+ = C, F

Dermed er {C, F} ikke en kandidatnøkkel

Utvider deretter X med A;

X = C, F, A

Siden vi har F, A kan vi også si at vi har B fordi A F $\rightarrow$ B:

X^+ = C, F, A, B

På samme måte kan vi da legge til

X^+ = C, F, A, B, D, E, G

Og vi ser da at {C, F, A} er en kandidatnøkkel.

Vi forsetter å prøve på samme måte videre.

X = C, F, B

X^+ = C, F, B, A, D, E, G

{C, F, B} er en kandidatnøkkel.

X = C, F, D

X^+ = C, F, D, G

{C, F, D} er ikke en kandidatnøkkel.

X = C, F, E

X^+ = C, F, E

{C, F, E} er ikke en kandidatnøkkel.

X = C, F, E, D

X^+ = C, F, E, D, B, A, G

{C, F, E, D} er en kandidatnøkkel.

Vi får da at kandidatnøklene til R er: {C, F, A}, {C, F, B}, og, {C, F, E, D}.

b) Finn den høyeste normalformen som R tilfredsstiller:

Vi kan gå ut ifra at alle FD'er oppfyller 1NF. Vi må da se videre på hvilke som oppfyller 2F. I oppgave 2a ser vi at A,B, C, D, E, og, F er nøkkelattributter. Siden G er den eneste komponenten i R som da ikke er et nøkkelattributt må dette bety at D er et nøkkelattributt, fordi vi har FD'en D → G. Siden dette er et brudd på 2NF kan vi da konkludere med at hele relasjonen vår R er på nivå 1NF.

c) For at tabellen skal oppfylle BCNF må alle attributter kun være avhengige av en kandidatnøkkel.

For å dekomponere R tapsfritt til BCNF, så må vi sikre at dersom R(X, Y, Z) dekomponeres til S_1(X, Y), S_2(X, Z) så er X → Y.

Vi vil da for hver Y → A i R(X) beregne Y^+, og dekomponere R til $S_1(Y^+)$ og $S_2(Y,X/Y^+)$

Hvis vi tar utgangspunkt i C, D, E → B:

Y = C, D, E

Y^+= C, D, E, B, G, A

Dermed får vi at S_1 = (C, D, E, B, G, A) og

S_2 = (C, D, E, A, B, C, D, E, F, G)/(C, D, E, B, G, A)

S_2 = (C, D, E, F)

Vi vil nå fortsette rekursivt over S_1 og S_2 til vi ikke har noen brudd på BCNF.

FD'ene til S_1 er nå:

B → A og D → G

Ettersom at A er avhengig av B i denne FD'en, og B ikke er en kandidatnøkkel, oppfyller ikke dette BCNF. Vi forsetter da å dekomponere S_1 på samme måte, og får:

S_{11}(B, A) og S_{12}(B, C, D, E, G).

Av disse to strider kun S_{12} med BCNF, dette fordi at G er avhengig av D som vil være en delmengde av sin egen kandidatnøkkel {C, D, E} .

Vi fortsetter dermed med S_{12} og får:

S_{121}(D, G) og S_{122}(C, D, E, B) ingen av disse strider med BCNF, og vi kan da gå videre og se på S_2(C, D, E, F).

Siden S_2 heller ikke strider med BCNF kan vi dermed si at R kan dekomponeres til fire tabeller:

S_{11}(B, A)

S_{121}(D, G)

S_{122}(C, D, E, B)

S_2(C, D, E, F).