

```

CREATE TABLE Tog(
  togNr INT PRIMARY KEY, -- setter TogNr som primærnøkkel og data type int
  startStasjon TEXT NOT NULL, --Setter stasjonene som text og not null.
  endeStasjon TEXT NOT NULL,
  ankomstTid TEXT NOT NULL --Setter ankomstTid sin datatype like time, slik at det skal bli lettere å lage tog
);

CREATE TABLE TogTabell(
  togNr INT,
  avgangstid TIME,
  stasjon text NOT NULL,
  PRIMARY KEY (togNr, avgangstid), --Setter kombinasjonen av togNr og avgangstid som primærnøkkel
  FOREIGN KEY (togNr) REFERENCES Tog (togNr) -- Setter opp togNr som fremmednøkkel
);

CREATE TABLE Plass(
  dato DATE, --Tar i bruk data typen DATE
  togNr INT,
  vognNr INT,
  plassNr INT,
  vindu BOOLEAN NOT NULL, --Bruker boolean pga det må være en av to tilstander
  ledig BOOLEAN NOT NULL, --Setter NOT NULL for at man må fylle denne attributten
  PRIMARY KEY (dato, togNr, vognNr, plassNr),-- Setter primær nøkkel
  FOREIGN KEY (togNr) REFERENCES Tog (togNr)-- Setter fremmednøkkel med referanse til togNr til Tog
);

```

Oppgave 2

Vi har relasjonen $R(A,B,C,D,E,F,G)$

La $Q = \{$

$CDE \rightarrow B,$

$AF \rightarrow B,$

$B \rightarrow A,$

$BCF \rightarrow DE,$

$D \rightarrow G$

$\}$

Barer venstre side	Bare høyre side	Begge sider
--------------------	-----------------	-------------

C og F	G	A, B, D, E
--------	---	------------

Prøver C, F, A: Kandidatnøkkel

Prøver C, F, B: Kandidatnøkkel

Prøver C, F, D: Ikke kandidatnøkkel

Prøver C, F, E: Ikke kandidatnøkkel

Prøver C, F, D, E: Kandidatnøkkel

Vi gjenstår derfor med kandidatnøkklene: (A, C, F), (B, C, F) og (C, D, E, F)

2b)

$CDE \rightarrow B$	3NF
$AF \rightarrow B$	3NF
$B \rightarrow A$	3NF
$BCF \rightarrow DE$	BCNF
$D \rightarrow G$	1NF

Siden FD'en « $D \rightarrow G$ » bare tilfredsstiller 1NF, får vi at R tilfredsstiller normalformen 1NF.

2c)

$CDE \rightarrow B$	3NF
$AF \rightarrow B$	3NF
$B \rightarrow A$	3NF
$BCF \rightarrow DE$	BCNF
$D \rightarrow G$	1NF

Nøkklene til R: (A, C, F) og (B, C, F), (C, D, E, F)

CDE bryter med BCNF

Vi får:

S1: (C,D,E,B,A,G)
S1 har FDene 1, 2, 5

S1 bryter BCNF og vi gjør algoritmen på S1:

1. $CDE \rightarrow B$
2. $B \rightarrow A$
3. $D \rightarrow G$

Kandidatnøkkel: CDE

$B \rightarrow A$ bryter BCNF

Vi får S11: (A, B)

S11 har FDene: 2

S11 bryter ikke med BCNF

Vi får S12: (B, C, D, E, G)

S12 har FDene: 1 og 3

S12 bryter med BCNF

S12 bryter med BCNF og vi gjør algoritmen:

1 $CDE \rightarrow B$

2 $D \rightarrow G$

CDE er kandidatnøkkel

$D \rightarrow G$ bryter med BCNF

S121: (D,G)

S121 har FDene 2

S121 bryter ikke med BCNF

S122: (B,C,D,E)

S122 har FDene 1

S122 bryter ikke med BCNF

S2 brøt ikke med bcnf

S2: (C,D,E,F)

S2 har FDene: Ingen

S2 bryter ikke med BCNF

Det dekomponeres derfor til:

S11(A, B), S121(D,G), S122(B,C,D,E), S2(C,D,E,F)