

Indholdsfortegnelse

Indledning:.....	1
Krav:.....	1
Konklusion:.....	2
Løsningsforslag og Anvendte teknologier:.....	2
Dokumentation:.....	20
Excel.....	20

Indledning:

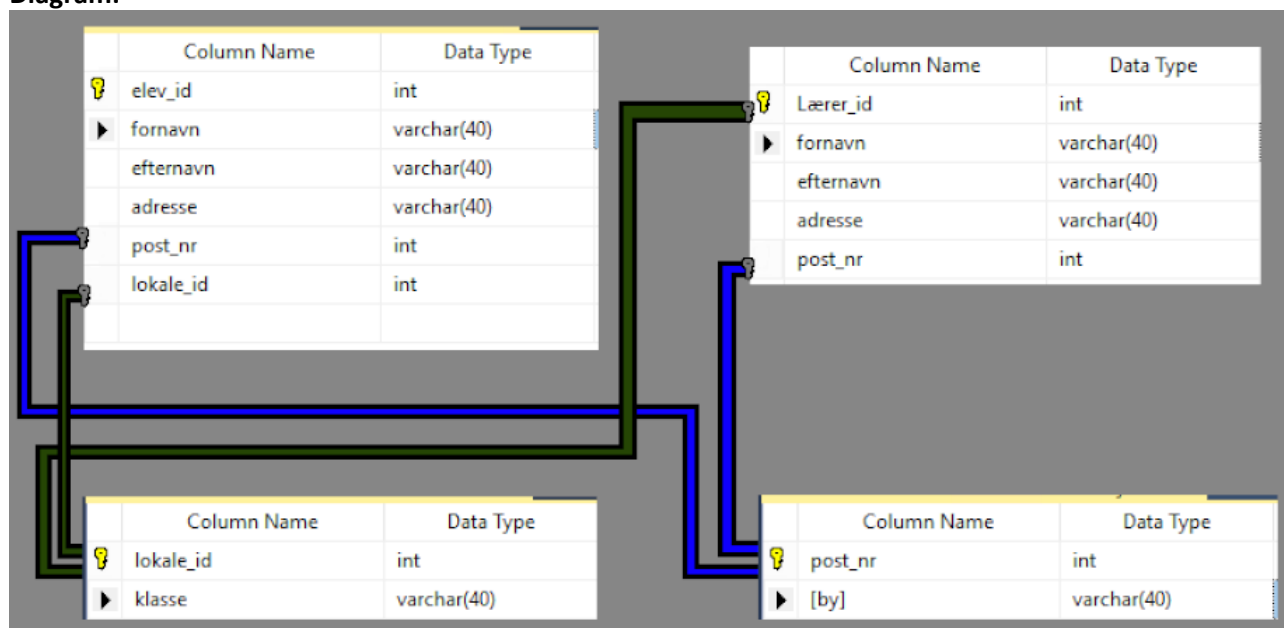
Opgaven går ud på at en OS (Windows Server) skal oprettes på en virtuel maskine i Hyper-V, derudover skal der installeres MS SQL relaterede programmer som Azure Data Studio og MS SQL Server 2019.

De skal bruges til at oprette en skole database som vil indeholde lærer og elev oplysninger, samt en post nr-by og klasse lokale databaser.

Der skal demonstreres hvordan data (navn, id tal) kan blive indført i databasen med SQL kode og hvordan man kan søge i databasen med SQL queries.

Krav:

Diagram:



Konklusion:

Jeg lærte at arbejde med MS SQL Server Management Studio og en smule om SQL syntaks

Løsningsforslag og Anvendte teknologier:

Anvendte teknologier:

Hyper-V Manager

Azure Data Studio

Microsoft SQL Server 2019 configuration Manager

Primary og foreign keys

Primary keys er vigtige fordi de identificer unikke rækker (og celler i rækkerne) i en tabel.

Foreign keys peger til en anden tabel

Løsning:

Gå til computer, åbn "Fællesdrev (X:)", åbn "ISO Filer" folder, åbn "Windows" folder, kopier Windows_Server_2019.iso til et eller andet lokalt sted på computeren.

Skab en ny Virtuel Maskine i Hyper-V.

Tryk på ny i hovedmenu.

Inden du starter

Tryk på næste

Angiv navn og placering

Jeg vil kalde min "WindowsServer2019Case2SQLDemonstration"

Angiv generation

Jeg vil bruge Generation 1

Tildel hukommelse

Jeg giver den 4096 MB

Konfigurer netværk

Ikke tilsluttet er fint for nu, det vil være nødvendigt at downloade nogle opdateringer senere.

Opret forbindelse til virtuel harddisk

Vælg, "Opret en virtuel harddisk", lad den være som den er. 127 GB er rigeligt.

Indstillinger for installation

Vælg, "Installer et operativ fra en cd/dvd-rom, der kan bootstartes", under Medie, vælg "billedfil (.iso):" og peg til iso filen kopieret fra TEC's server.

Oversigt

Tjek at alting er i orden og tryk på "udfør".

Dobbelt tryk på dens navn i "virtuelle maskiner" listen i Hyper-V hovedmenu.

Klip på start i den menu der dukker op.

Windows installationen vil nu begynde, under "Keyboard og input method", vælg "Danish".

Tryk på næste og på "Install Now".

Nu vælg "Windows Server 2019 Standard Evaluation (Desktop Experience)"

Klik på kassen ved siden af "I accept the license terms", next

Vælg "Custom:Install Windows Only", next.

"Where do you want to install Windows?" Next

Vent

I "customize settings", jeg indtaster "P45s_W0rdD", som kodeord.

Log ind.

Luk Server Manager

Skriv "computer management" ind i Windows search, vælg "Local Users and groups" og højre tryk på users, venstre tryk på "new user". Skab bruger1, bruger2 og bruger3.

Download til hovedmaskinen.

SQL Server 2017 Dev herfra:

<https://go.microsoft.com/fwlink/p/?linkid=853016>

SSMS herfra:

<https://aka.ms/ssmsfullsetup>

Flyt alle filene til en enkelt folder, jeg flyttede dem hertil: C:\Users\allgaw\Downloads\SQLCase2

Skab en Excel Fil her og udfyld den

	A	B	C	D	E	F
1	elev_id	fornavn	efternavn	adresse	post_nr	lokale_id
2	1	Bo	Andersen	Gammel Byvej 12	2650	1
3	2	Frederikke	Hansen	Amager Boulevard 5	2300	2
4	3	Jens	Mikkelsen	Lily Brogbergs Vej 17	2500	3
5	4	Philip	Mortensen	Brunevang 90	2610	3
6	5	Kasper	Frederiksen	Bryggertorvet32	3650	2
7	6	Milla	Jørgensen	Virum Torv 25	2830	1
8	7	Fie	Knudsen	Allen 85	2770	1
9	8	Henrik	Madsen	Lily Brogbergs Vej53	2500	2
10						
11						
12						
13						
14						
15						
16						
17						
18						

Brug hver ark for sin tabel (dvs. elev, lærer, postnr), vælg rette navne.

Nu er det tid til at kopiere de downloadede filer til vores virtuelle maskine.

Bring powershell op

Og tast ind denne kommando

```
$s = New-PSSession -VMName <ViruelMaskineNavn> -Credential (Get-Credential)
```

I min tilfælde: \$s = New-PSSession -VMName WindowsServer2019Case2SQLDemonstration - Credential (Get-Credential)

En menu dukker op, jeg indtaster "Administrator" og "P45s_W0rdD"

SQLServer2017-SSEI-Dev:

```
Copy-Item -ToSession $s -Path C:\Users\allgaw\Downloads\SQLCase2\SQLServer2017-SSEI-Dev.exe -Destination C:\Users\Administrator\Downloads\
```

SSMS:

```
Copy-Item -ToSession $s -Path C:\Users\allgaw\Downloads\SQLCase2\SSMS-Setup-ENU.exe -  
Destination C:\Users\Administrator\Downloads\
```

Denne her kan tage et par forsøg, muligvis en genstart af powershell. Du kan også give op på dette og lade VMen forbinde til internettet og bruge dens internet explorer til at downloade denne fil.

Excel filen

```
Copy-Item -ToSession $s -Path C:\Users\allgaw\Downloads\SQLCase2\SQLCase2-Tables.xlsx -  
Destination C:\Users\Administrator\Downloads\
```

Kør hver .exe

Internettet er nødvendig her.

SQL Server 2017, default install location er fint, vælg basic, vent.

SSMS, default install location er fint, vent.

Brug Windows search og indtast "SQL Server Management Studio Management Studio 19"

Der burde allerede være en valgt server, yderligere servere kan blive skabt i "Azure Data Studio"

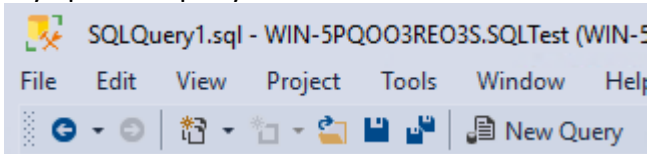
Klik på "Connect".

I Object Exploreren, højre klik på folderen der hedder "Databases", vælg "ny database".

Indtast "TEC" som database navn.

SQL kode

Tryk på new query:



Flere tabeller vil blive skabt nu, rækkefølgen de bliver skabt i er vigtig fordi, elev (nedeunder) for eksempel peger på post_nr og den eksisterer jo ikke lige nu. Som regel kan computere ikke lide referencer til ting der ikke findes og det gælder også her, vi vil få en fejlbesked. Dette er en mindre betragtning hvis man skaber tabellerne først og marker primary og foreign keys bagefter.

Denne kode vil skabe en tabel der hedder dbo.PostNrBy:

```
CREATE TABLE PostNrBy (  
  post_nr int NOT NULL,  
  [by] varchar(40),  
  PRIMARY KEY (post_nr),  
);
```

Post_nr bliver primary Key

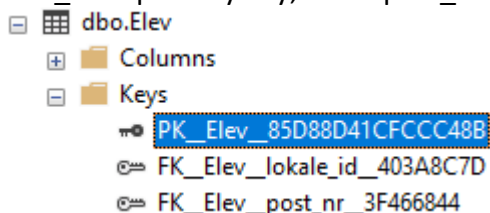
Denne kode vil skabe en tabel der hedder dbo.Skole_klasse med lokale_id som primary key:

```
CREATE TABLE Skole_klasse (  
  lokale_id int NOT NULL,  
  klasse varchar(40),  
  PRIMARY KEY (lokale_id),  
);
```

Denne kode vil skabe en tabel der hedder dbo.Elev:

```
CREATE TABLE Elev (  
  elev_id int NOT NULL,  
  fornavn varchar(40),  
  efternavn varchar(40),  
  adresse varchar(40),  
  post_nr int,  
  lokale_id int,  
  PRIMARY KEY (elev_id),  
  FOREIGN KEY (post_nr) REFERENCES PostNrBy(post_nr),  
  FOREIGN KEY (lokale_id) REFERENCES Skole_klasse(lokale_id)  
);
```

Elev_id er primary key, mens post_nr og lokale_id er foreign keys



Denne kode vil skabe en tabel der hedder dbo.Lærer:

```

CREATE TABLE Lærer (
  Lærer_id int NOT NULL,
  fornavn varchar(40),
  efternavn varchar(40),
  adresse varchar(40),
  post_nr int,
  PRIMARY KEY (Lærer_id),
  FOREIGN KEY (post_nr) REFERENCES PostNrBy(post_nr),
  FOREIGN KEY (Lærer_id) REFERENCES Skole_klasse(lokal_id)
);

```

Lærer er både primary og foreign key, den peger mod klasse lokalet.

Post_nr er foreign key.

Vores Tabeller er nu oprettet men de mangler data.

Jeg vil nu bruge følgende query for at tilføje data ind i PostNrBy:

```

InsertIntoPostNrBy....Administrator (53)) -> X
INSERT INTO PostNrBy (post_nr, [by])
VALUES ('1824', 'Frederiksberg, Sankt Thomas Alle'),
       ('2300', 'København Sydhavnen'),
       ('2500', 'Valby'),
       ('2610', 'Rødovre'),
       ('2650', 'Hvidovre'),
       ('2740', 'Skovlunde'),
       ('2750', 'Ballerup'),
       ('2770', 'Kastrup'),
       ('2830', 'Virum'),
       ('3650', 'Ølstykke');

```

Jeg brugte den samme query for at tilføje data ind i Skole_klasse:

```

InsertIntoSkole_kla...Administrator (54)) -> X
INSERT INTO Skole_klasse (lokal_id, klasse)
VALUES ('1', '1A'),
       ('2', '8B'),
       ('3', '4D');

```

Elev:

```

InsertIntoElev.sql -...Administrator (53)) -> X
INSERT INTO Elev (elev_id, fornavn, efternavn, adresse, post_nr, lokale_id)
VALUES
  ('1', 'Bo', 'Andersen', 'Gammel Byvej 12', '2650', '1'),
  ('2', 'Frederikke', 'Hansen', 'Amager Boulevard 5', '2300', '2'),
  ('3', 'Jens', 'Mikkelsen', 'Lily Brogbergs Vej 17', '2500', '3'),
  ('4', 'Philip', 'Mortensen', 'Brunevang 90', '2610', '3'),
  ('5', 'Kasper', 'Frederiksen', 'Bryggertorvet32', '3650', '2'),
  ('6', 'Milla', 'Jørgensen', 'Virum Torv 25', '2830', '1'),
  ('7', 'Fie', 'Knudsen', 'Allen 85', '2770', '1'),
  ('8', 'Henrik', 'Madsen', 'Lily Brogbergsvej53', '2500', '2');

```

Lærer:

```

InsertIntoElev.sql -...Administrator (53))
INSERT INTO Lærer (lærer_id, fornavn, efternavn, adresse, post_nr)
VALUES
    ('1', 'Tom', 'It', 'Sankt Thomas Alle 3', '1824'),
    ('2', 'Lars', 'Henriksen', 'Nisserdalen 76', '2740'),
    ('3', 'Mia', 'Hansen', 'Østervej 16', '2750');

```

Datene er inde.

Tid for queries:

Select Top 50:

Elev

```

SelectTop50Elev.sql...Administrator (55))
SELECT TOP (50) [elev_id]
, [fornavn]
, [efternavn]
, [adresse]
, [post_nr]
, [lokale_id]
FROM [Skolen].[dbo].[Elev]

```

	elev_id	fornavn	efternavn	adresse	post_nr	lokale_id
1	1	Bo	Andersen	Gammel Byvej 12	2650	1
2	2	Frederikke	Hansen	Amager Boulevard 5	2300	2
3	3	Jens	Mikkelsen	Lily Brogbergs Vej 17	2500	3
4	4	Philip	Mortensen	Brunevang 90	2610	3
5	5	Kasper	Frederiksen	Bryggerortvet 32	3650	2
6	6	Milla	Jørgensen	Virum Torv 25	2830	1
7	7	Fie	Knudsen	Allen 85	2770	1
8	8	Henrik	Madsen	Lily Brogbergs Vej 53	2500	2

Select Join

1/3

SELECT Skole_klasse.lokale_id, elev.fornavn

FROM Skole_klasse

INNER JOIN elev on Skole_klasse.lokale_id = elev_id;

```

SelectJoin01.sql -...Administrator (53))
SELECT Skole_klasse.lokale_id, elev.fornavn
FROM Skole_klasse
INNER JOIN elev on Skole_klasse.lokale_id = elev_id;

```

	lokale_id	fornavn
1	1	Bo
2	2	Frederikke
3	3	Jens

2/3

SELECT Skole_klasse.lokale_id, Lærer.fornavn, Lærer.efternavn

FROM Skole_klasse

INNER JOIN Lærer on Skole_klasse.lokale_id = lærer_id;

```

SelectJoin02.sql -...Administrator (53))
SELECT Skole_klasse.lokale_id, Lærer.fornavn, Lærer.efternavn
FROM Skole_klasse
INNER JOIN Lærer on Skole_klasse.lokale_id = lærer_id;

```

	lokale_id	fornavn	efternavn
1	1	Tom	It
2	2	Lars	Henriksen
3	3	Mia	Hansen

3/3

```
SELECT PostNrBy.post_nr, lærer_id, Lærer.fornavn, Lærer.efternavn  
FROM PostNrBy  
INNER JOIN Lærer on PostNrBy.post_nr = Lærer.post_nr;
```

SelectJoin03.sql - ...Administrator (53)

```

SELECT PostNrBy.post_nr, lærer_id, Lærer.fornavn, Lærer.efternavn
FROM PostNrBy
INNER JOIN Lærer on PostNrBy.post_nr = Lærer.post_nr;

```

Results

Messages

	post_nr	lærer_id	fornavn	efternavn
1	1824	1	Tom	It
2	2740	2	Lars	Henriksen
3	2750	3	Mia	Hansen

Excel Import

Download Microsoft Access Database Engine 2016 Redistributable:

<https://www.microsoft.com/en-us/download/details.aspx?id=54920>

Kopier Microsoft Access Database Engine 2016 Redistributable over:

Copy-Item -ToSession \$s -Path C:\Users\allgaw\Downloads\SQLCase2\accessdatabaseengine_X64.exe -Destination C:\Users\Administrator\Downloads\

Installer AccessDatabase, default install location er fint.

I Windows search indtast "SQL Server Import and Export Wizard" åbn programmet.

Klik på next, ændr "Data Source:" til "Microsoft Excel"

Klik på "browse" og find Excel filen.

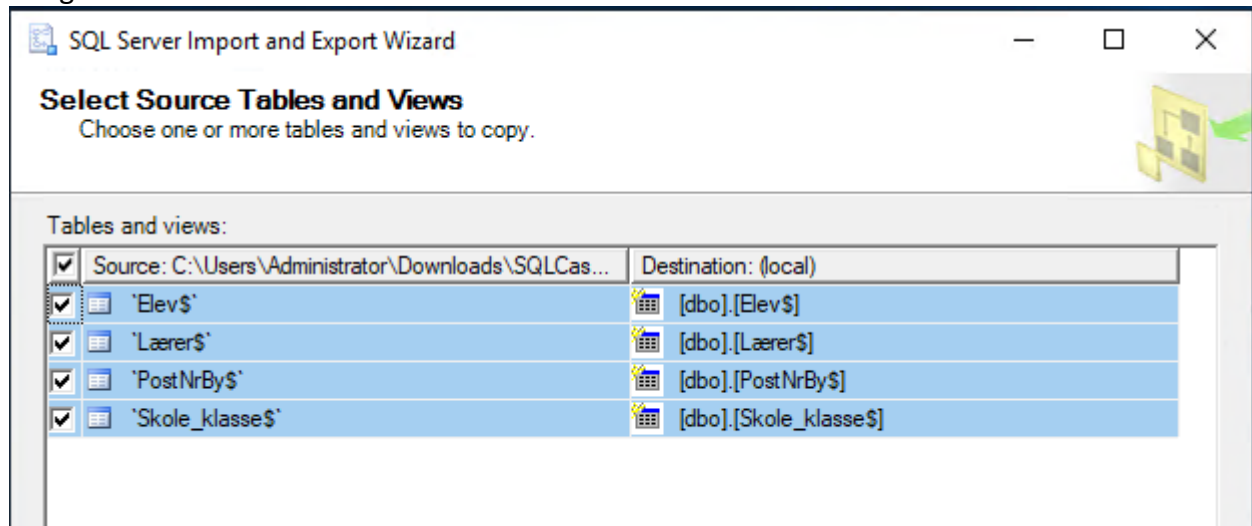
Under "Excel Version" vælg den ser mest ret ud, next.

I den næste menu ved "Destination" vælg "SQL Native Client 11.0"

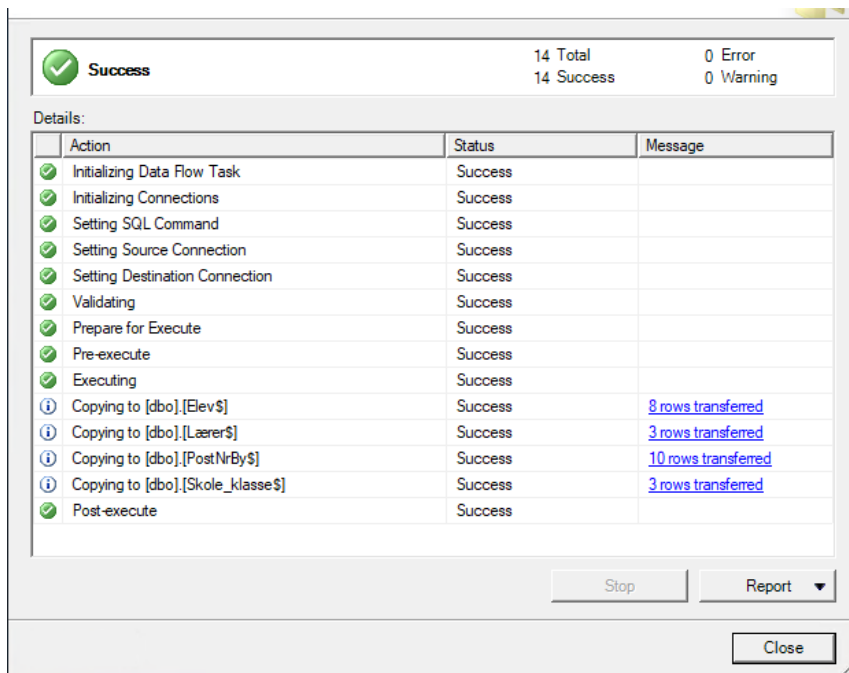
Under "Database" vælg "TEC", next.

"Copy data from one or more tables or views", Next.

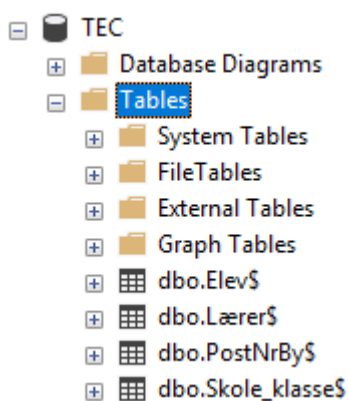
Vælg alle:



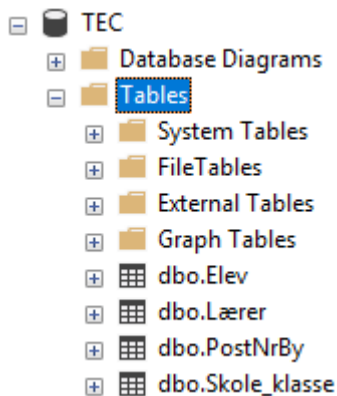
Next, next, finish, vent på import, og close.



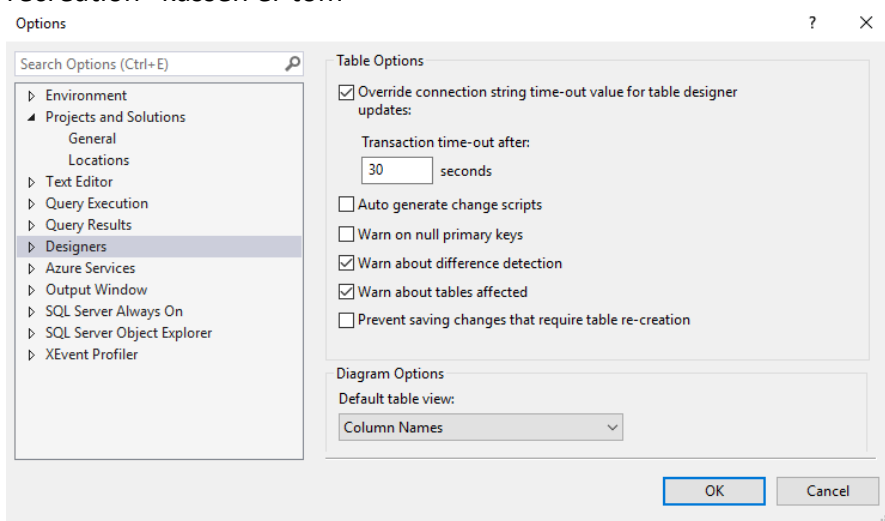
I Server management studio højre klik og tryk på refresh.



Vores data er nu inde i vores database.
 Dollar tegnene kan fjernes fra dbo navnene.



Gå ind i tools/options/designers og gør sikker at "Prevent saving changes that require table recreation" kassen er tom



Jeg højre trykker på dbo'ene og vælger design. Så højre tryk på kolumnerne der skal være primary keys.

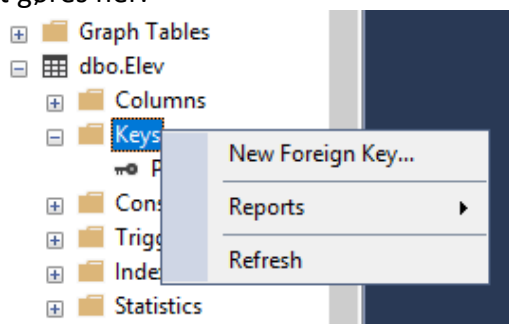
I Elev er det elev_id, i lærer er det lærer_id, i PostNrBy er det post_nr og i skole_klasse er det lokale_id.

Data typene kan ændres, ingen af tallene bruger komma så de kan ændres til int.

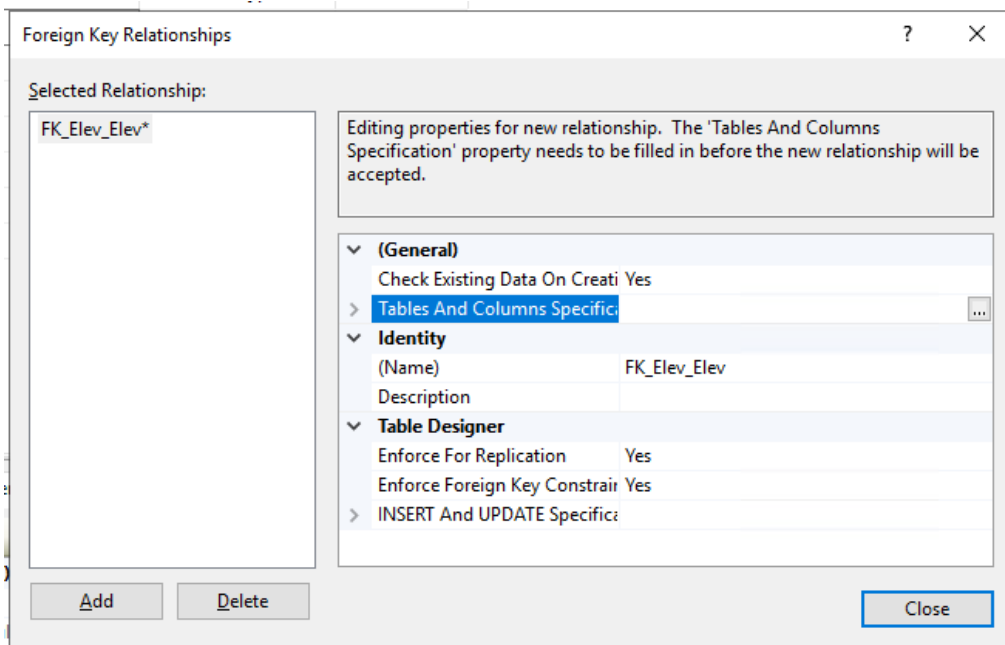
Jeg har valgt at ændre tekst til varchar(40) fra nvarchar(255).

Nu er det tid til at lave foreign keys:

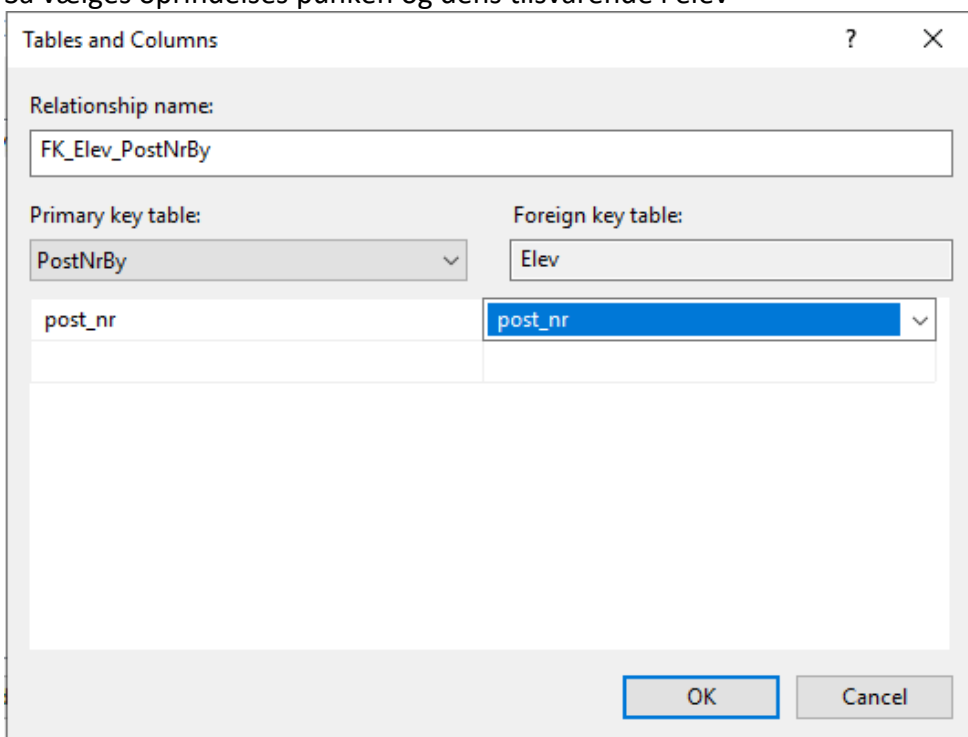
Det gøres her:



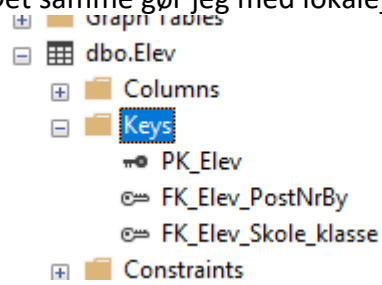
Foreign keys stiftes på bruger siden, for eksempel bruges post_nr i elev og lærer.



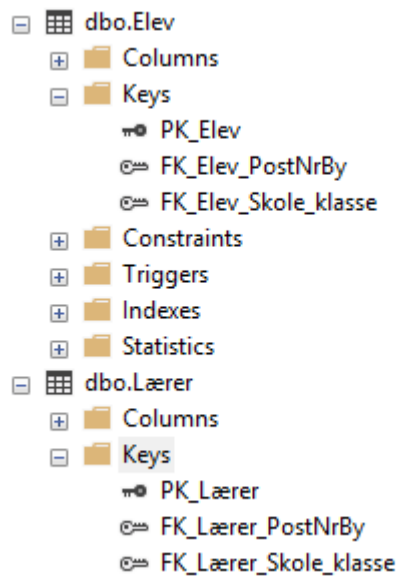
Vi klikker på '...' ikonen ved siden af "Tables and Columns Specific"
Så vælges oprindelses punkten og dens tilsvarende i elev



Det samme gør jeg med lokale_id



Det samme gør jeg for læreren



Nu er både primary og foreign keys sættet op.

Dokumentation:

Excel

Elev

elev_id	Fornavn	Efternavn	adresse	post_nr
1	Bo	Andersen	Gammel Byvej 12	2650
2	Frederikke	Hansen	Amager Boulevard 5	2300
3	Jens	Mikkelsen	Lily Brogbergs Vej 17	2500
4	Philip	Mortensen	Brunevang 90	2610
5	Kasper	Frederiksen	Bryggertorvet32	3650
6	Milla	Jørgensen	Virum Torv 25	2830
7	Fie	Knudsen	Allen 85	2770
8	Henrik	Madsen	Lily Brogbergs Vej53	2500

PostNrBy

post_nr	by
1824	Frederiksberg, Sankt Thomas Alle
2300	København Sydhavnen
2500	Valby
2610	Rødovre
2650	Hvidovre
2740	Skovlunde
2750	Ballerup
2770	Kastrup
2830	Virum
3650	Ølstykke

Lærer

Lærer_id	fornavn	efternavn	adresse	post_nr
1	Tom	It	Sankt Thomas Alle 3	1824
2	Lars	Henriksen	Nisserdalen 76	2740
3	Mia	Hansen	Østervej 16	2750

Skole_klasse

lokale_id	Klasse
1	1A
2	9B
3	4D