

SQL Rapport

Navn - Jonas Barigo Østergaard

Uddannelse - Datatekniker med speciale i Programmering

Instruktør - Simon Nicolas El Hanafi

The image shows the splash screen of Microsoft SQL Server Management Studio. It features a dark blue background with a lighter blue abstract design on the right side. A dark gray rectangular box is centered on the screen, containing the text "Microsoft SQL Server Management Studio" in white. Below this, in smaller white text, are the version number "v18.10", the copyright notice "© 2021 Microsoft.", and "All rights reserved." data-bbox="204 442 767 500"/>

Microsoft
SQL Server Management Studio

v18.10

© 2021 Microsoft.
All rights reserved.

Indhold

Indledning	3
Dokumentation.....	4
Konklusion	21
Logbog	22

Indledning

I dette projekt har jeg skulle beskæftige mig med windows server, MS SQL database samt MS SQL server management. Ud over disse har jeg også skulle oprette diagrammer der beskriver databasen.

Til selve serveren skulle jeg benytte hyper-v for at oprette en virtuel maskine hvorpå serveren lå. På serveren skulle jeg derefter oprette 3 brugere, installere SQL database og server management, hvortil jeg også skulle oprette 3 brugere.

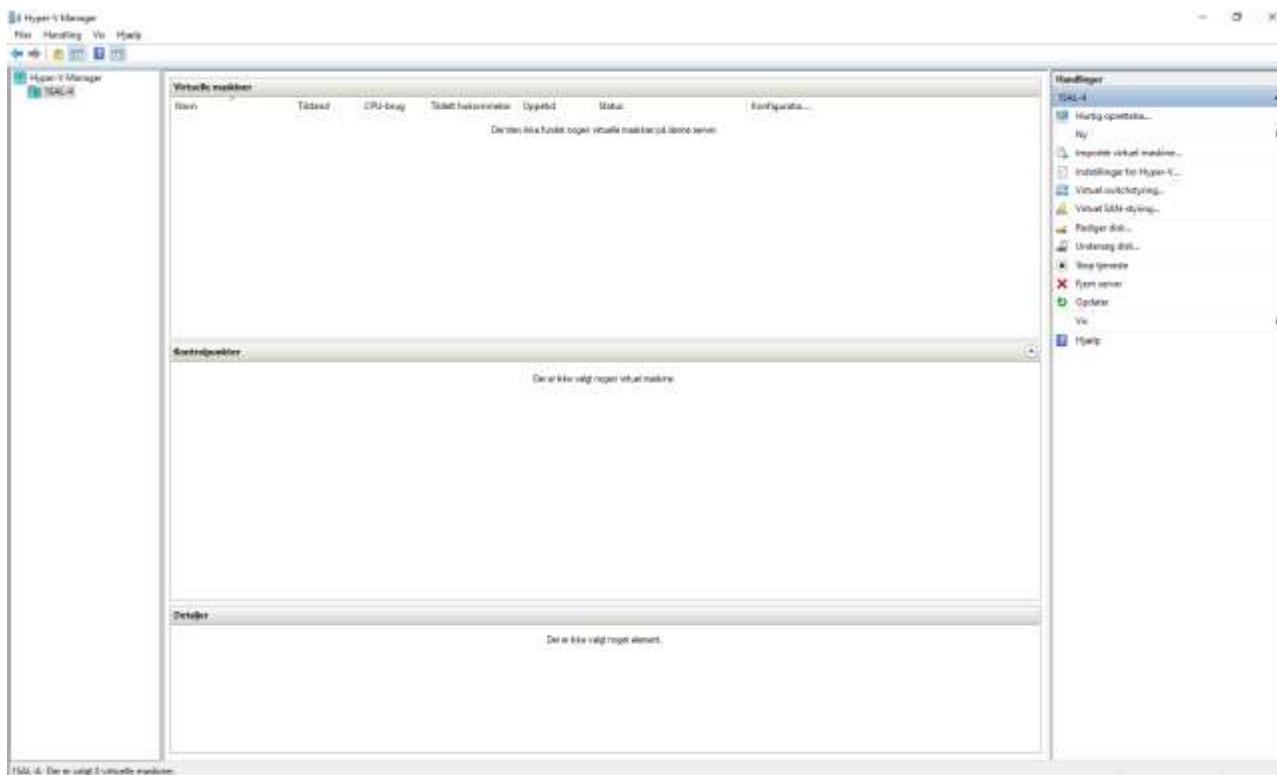
Til sql databasen oprettede jeg desuden et script som bestod af flere tabeller, der tilsammen skulle simulere en database over en skole. Herunder var elever, lærere, klasser og postnumre afbillede.

Derudover skulle jeg importere et excel ark til en tabel, hvorfra jeg kunne finde postnumrenes tilsvarende bynavne.

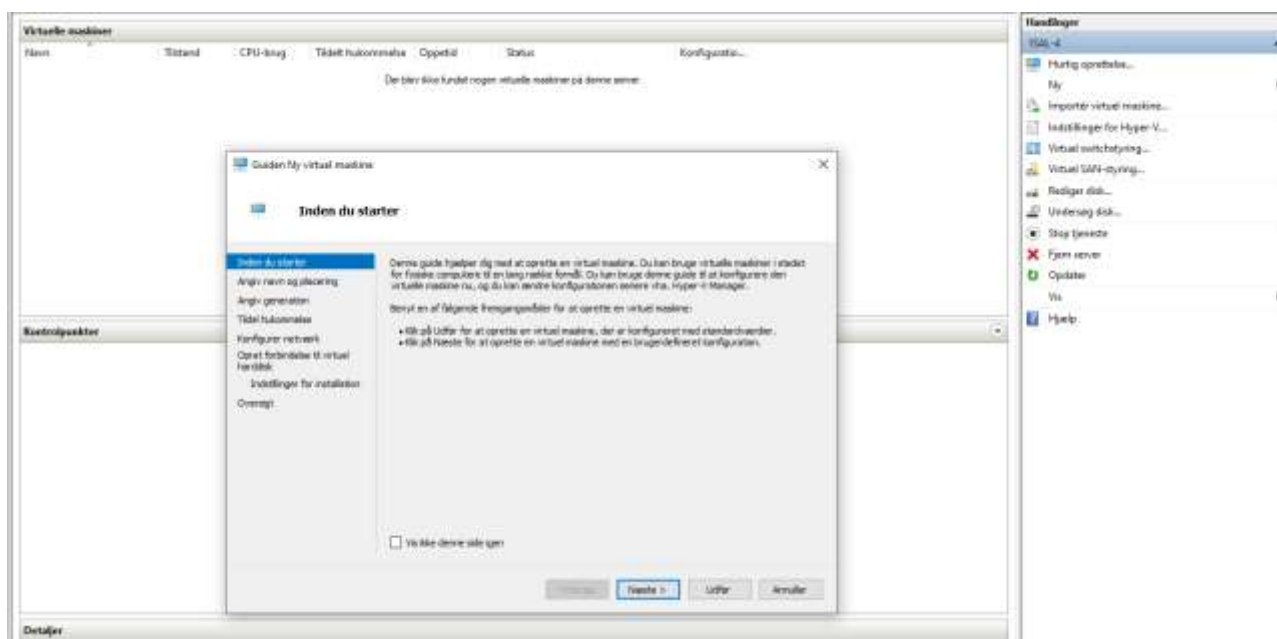
Desuden fik jeg også til opgave at mit script skulle kunne køre flere gange, uden at der opstår fejl, altså gøre mit script dynamisk.

Dokumentation

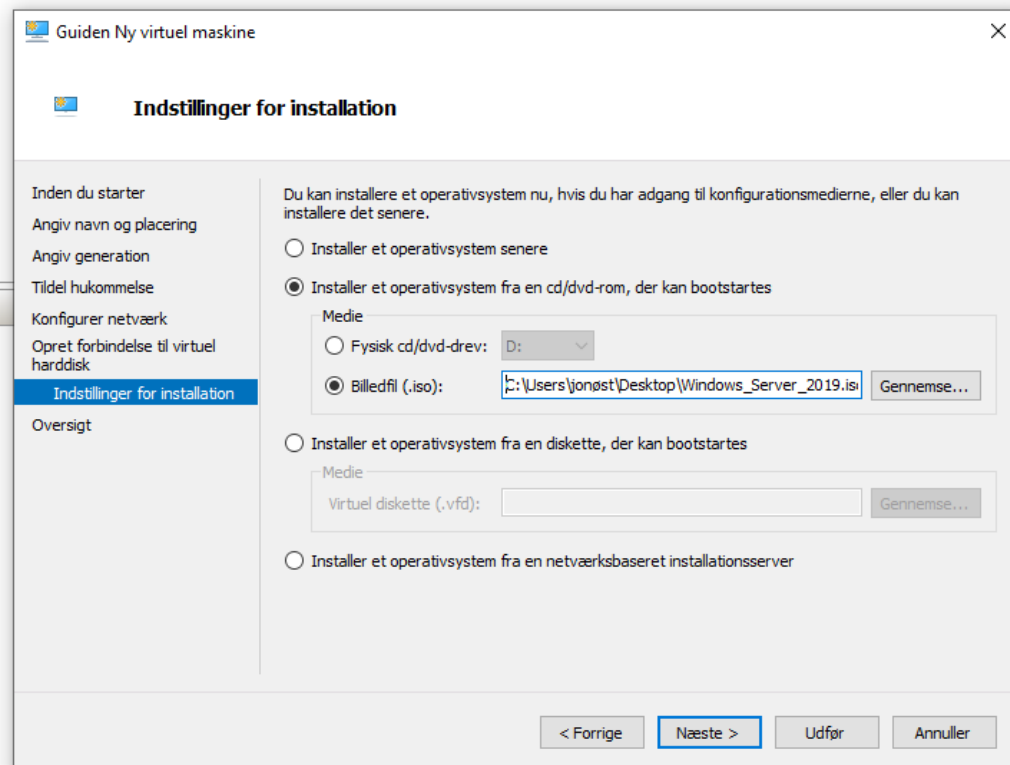
Opsætning af windows server på hyper-v



Først åbnes hyper-v og man forbinder til en lokale pc, hvis man ønsker dette. Dernæst opretter man en ny virtuel maskine ved at klikke ny. Dette vil åbne en setup wizard som man skal følge.

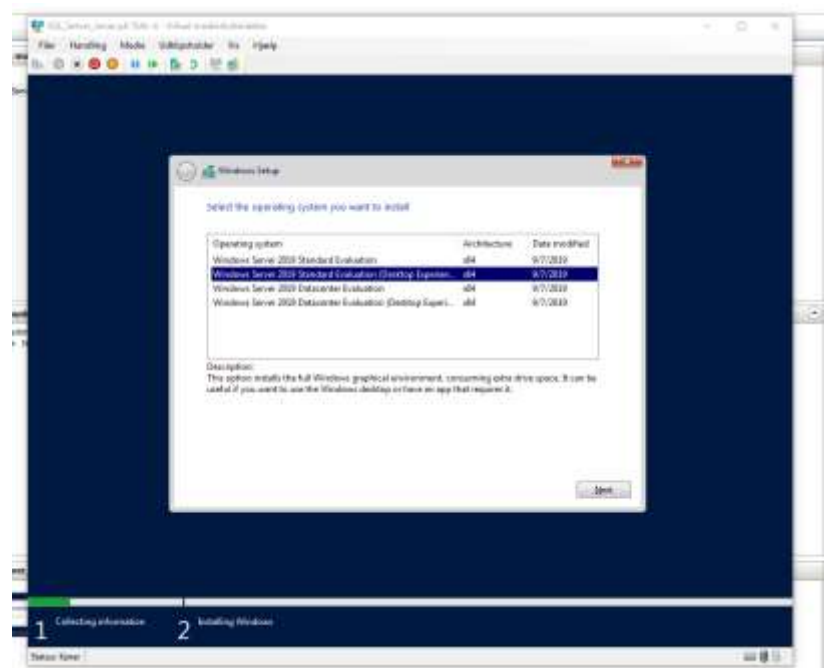


Her skriver man pc'ens navn, generation, angivelse af hukommelse, netværk, angivelse af virtuel harddisk og hvordan man vil installere sit operativ system, hvilket i dette tilfælde vil være en billedfil af en windows server.



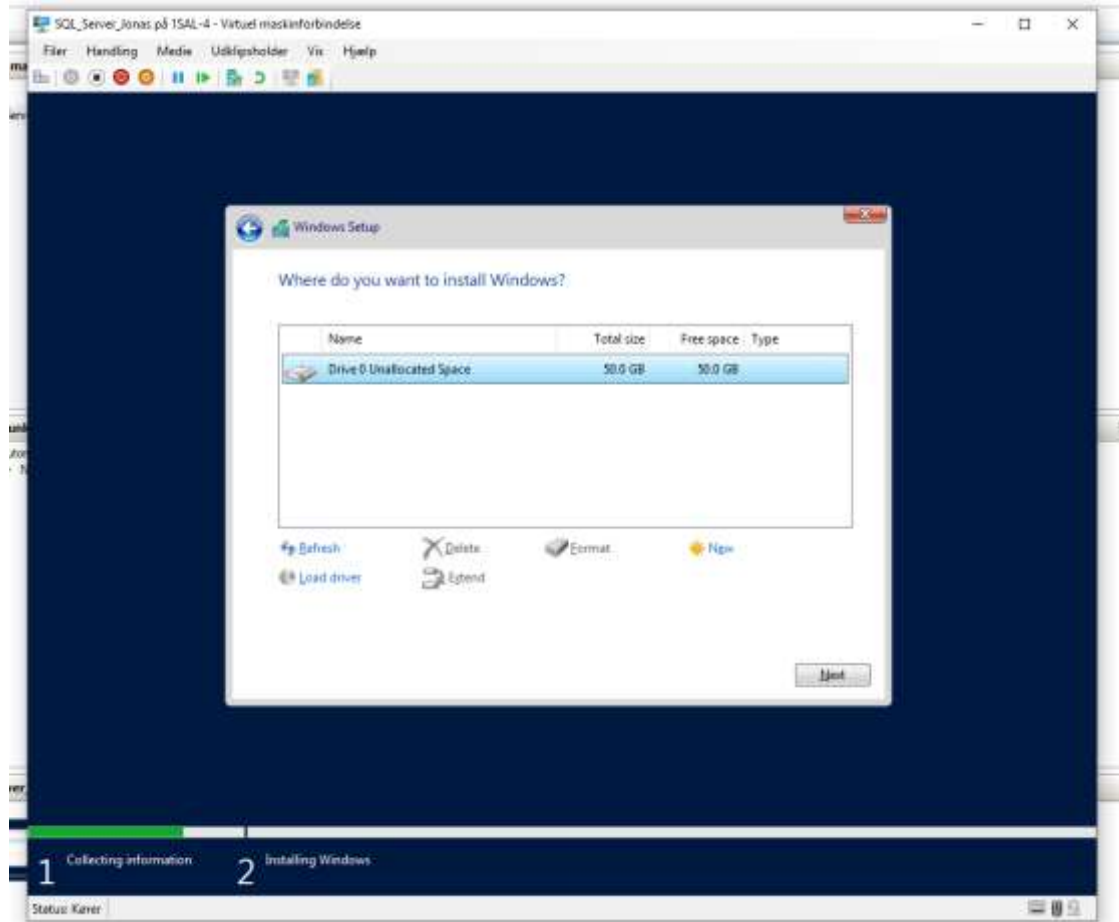
Der er ikke valgt noget element.

Efter man har valgt billedfilen(iso) går man videre og godkender sine indstillinger. Dermed er computeren oprettet, men selve operativ systemet er ikke konfigureret endnu. Når man åbner serveren, vil configurationen gå i gang af sig selv. Her vælger man ønsket sprog, tid og valuta format samt keyboard layout. Derefter skal man vælge sin windows server version man ønsker at installere.

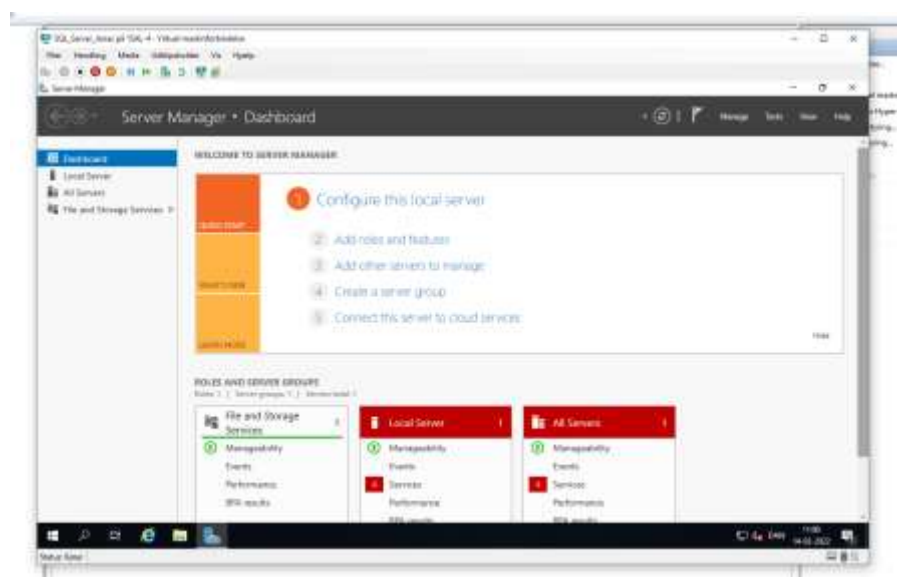


Her vælges windows server 2019 standard evaluation (desktop Experience...). Denne version har et gui som gør det nemmere for brugere navigere rundt på serveren.

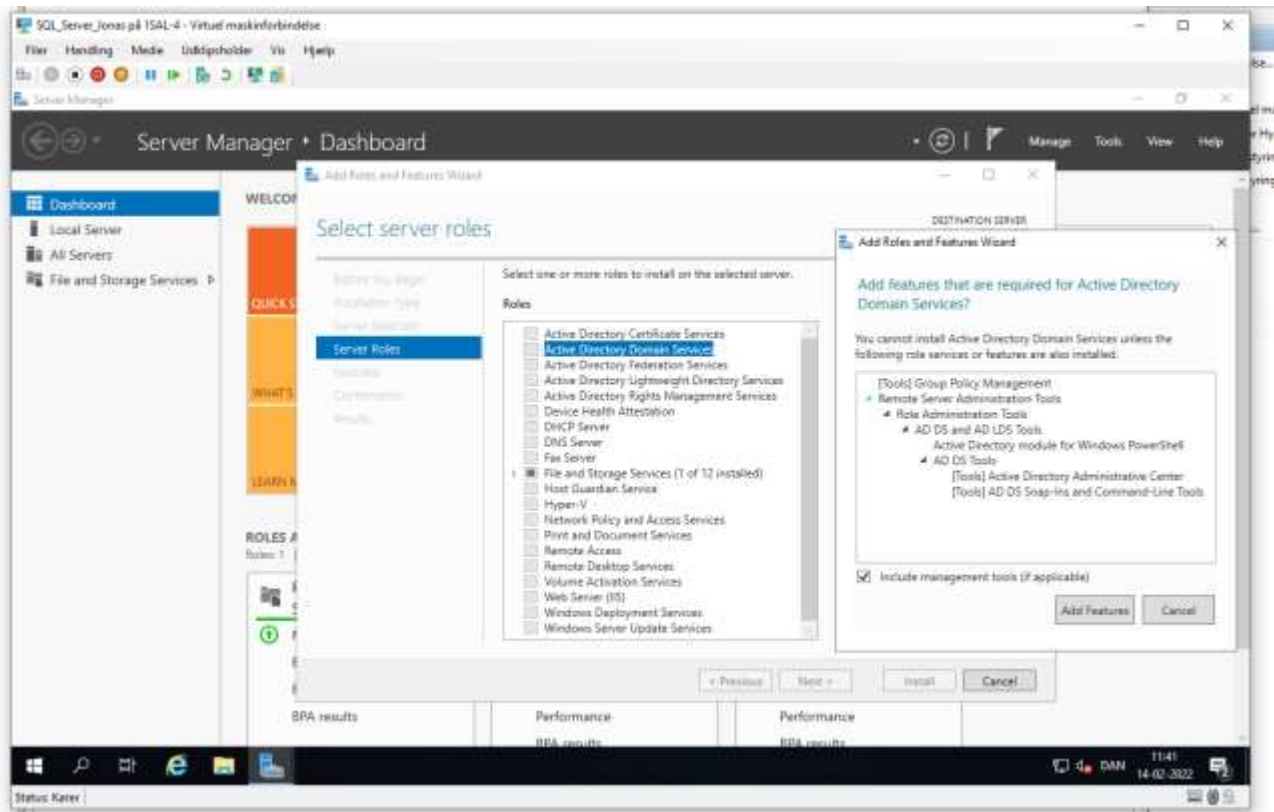
Derefter skal man acceptere betingelserne, vælge custom: install windows only (advanced) og vælge den virtuelle harddiske man oprettede sammen med installationen af den virtuelle maskine.



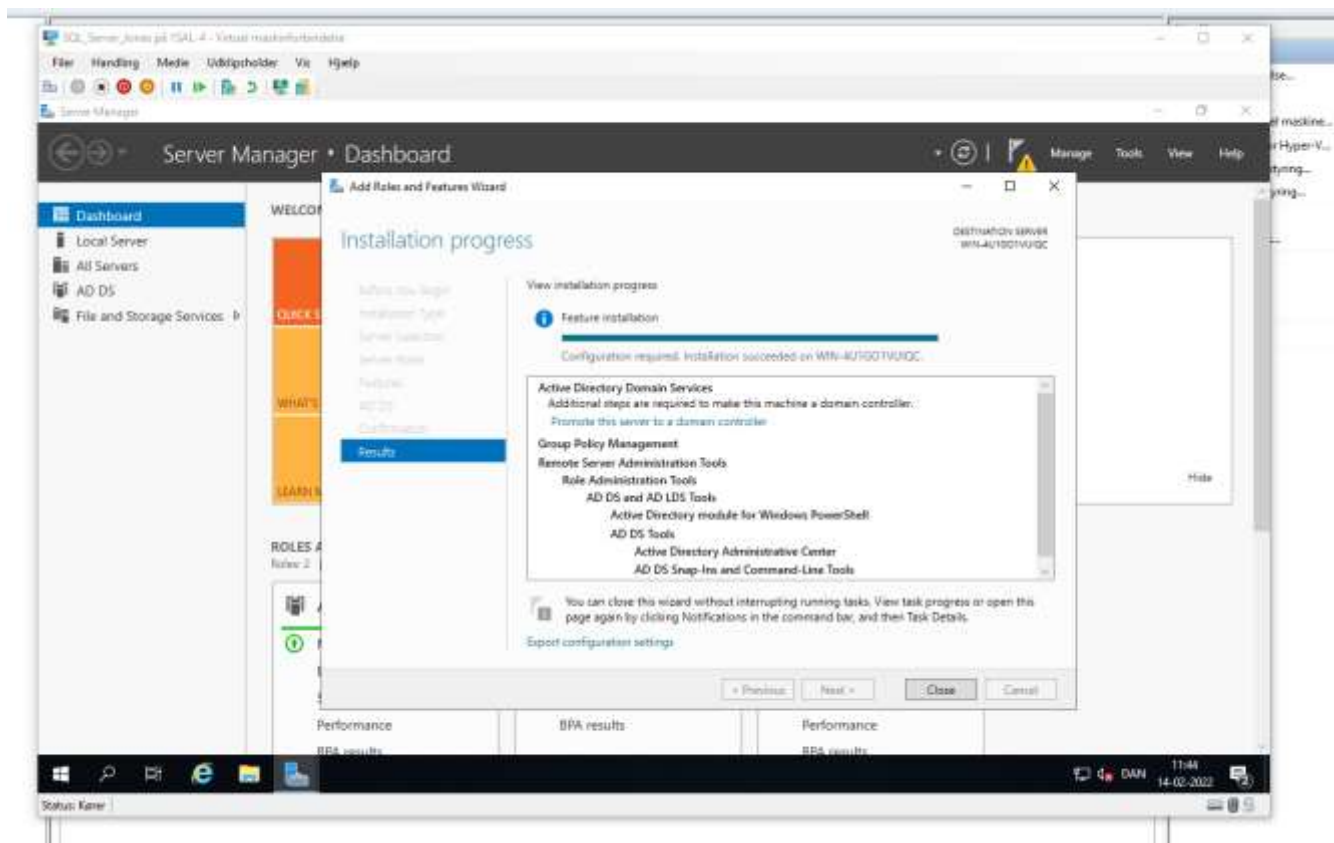
Derefter installeres operativ systemet endeligt og man skal vente i lidt tid. Efter dette bliver man bedt om at oprette et password til brugeren administrator. Dette password kræver en hvis længde, store og små bogstaver, samt tal eller speciel-tegn. Dermed er installationen fuldstændt og man vil logge ind på serveren. Her vil man blive mødt af en server manager der åbner automatisk ved opstart.



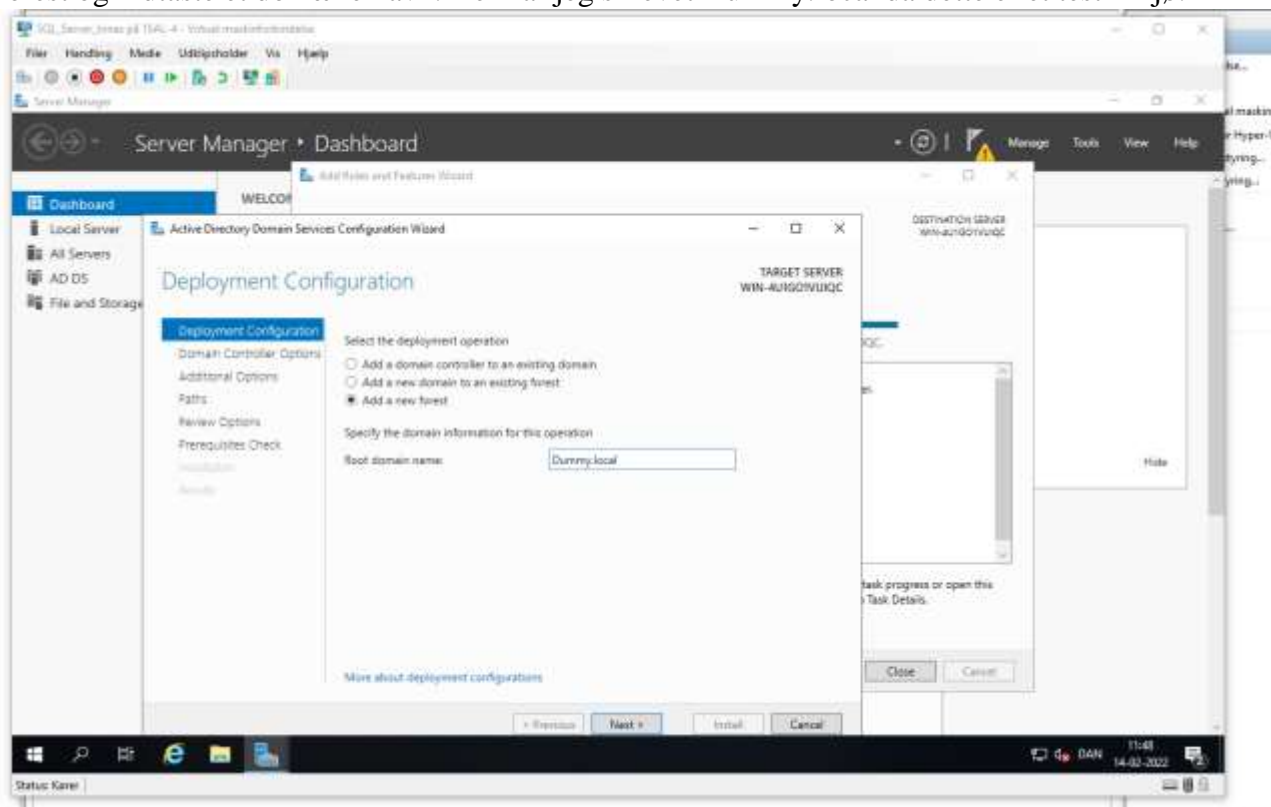
Her kan man vælge add roles and features under configure this local server eller manage. Dette vil åbne en setup wizard, hvor man kan vælge og installere features og roller. Her skal man vælge Active Directory Domain Services (AD DS) under roller og Group Policy Management under Features.



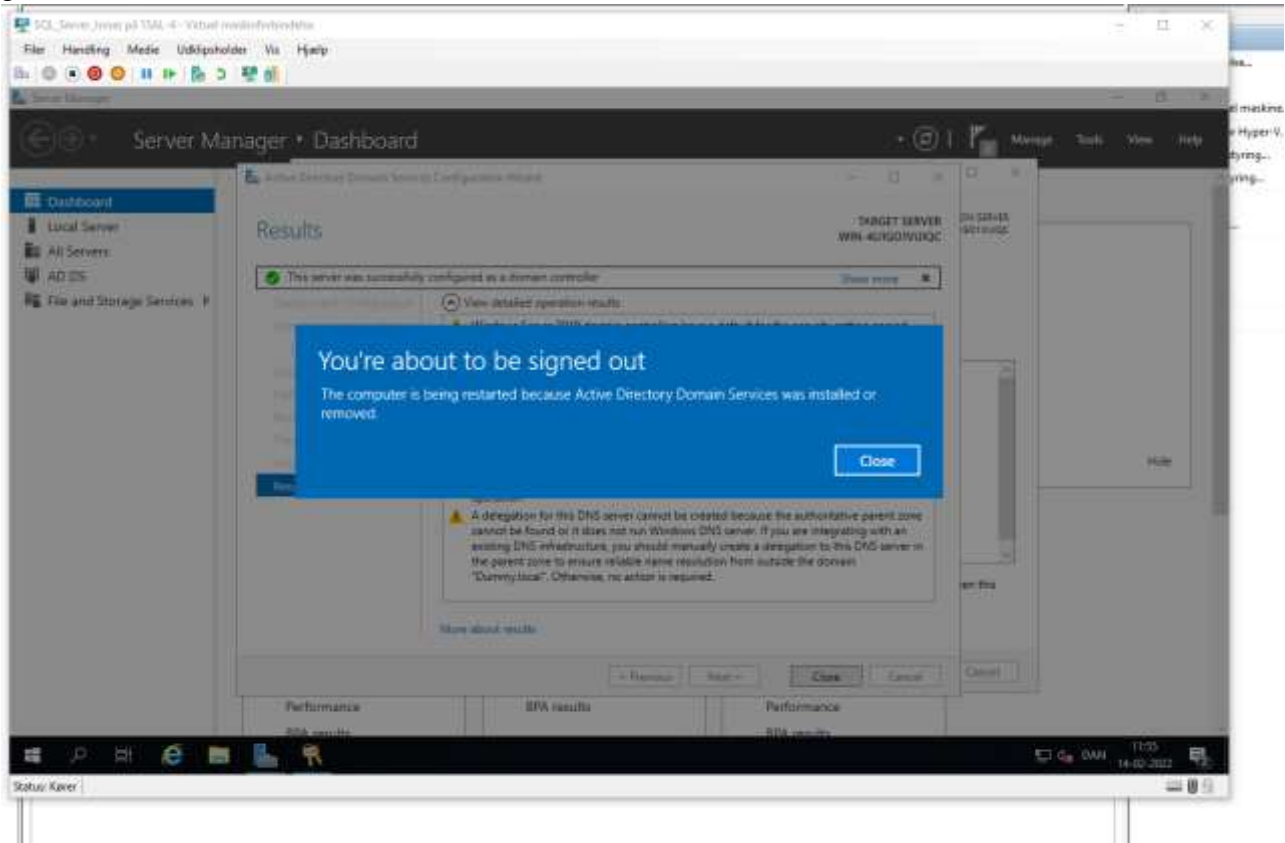
Efter dette fortsætter man setup wizarden, det er en god ide at sætte checkmark under genstart af computeren hvis nødvendigt. Når installationen er fuldført skal man trykke på den blå tekst "Promote this server to a domain controller"



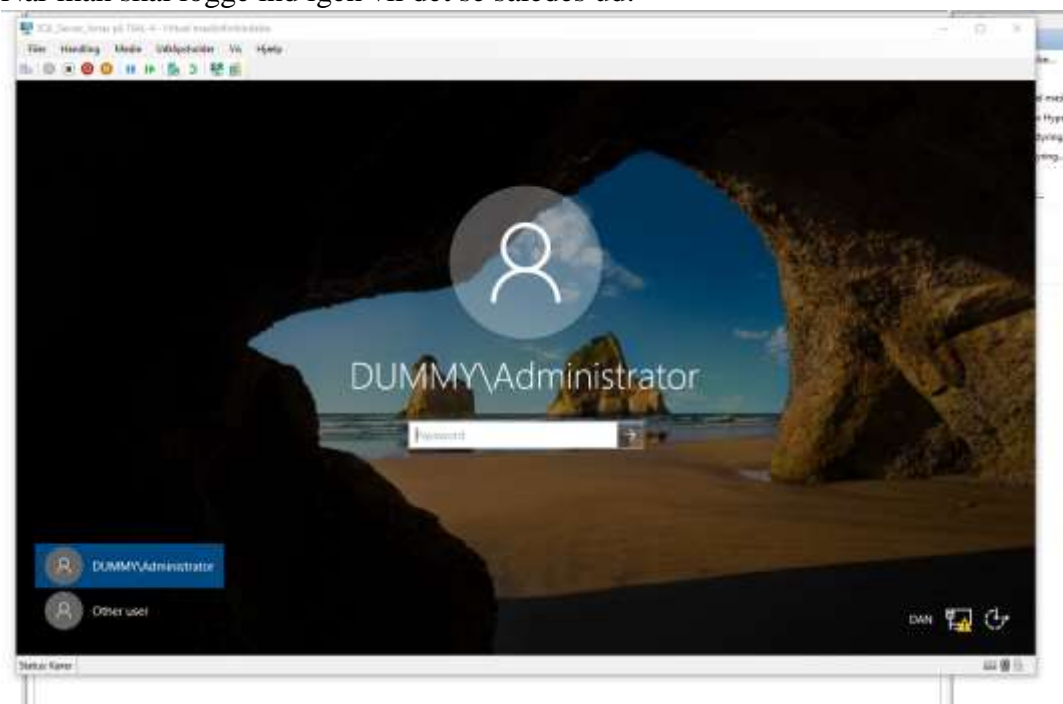
Herefter vil et nyt vindue åbne for at konfigurere domain controller. Her skal man vælge add a new forest og indtaste et domæne navn. Her har jeg skrevet Dummy.local da dette er et test miljø.



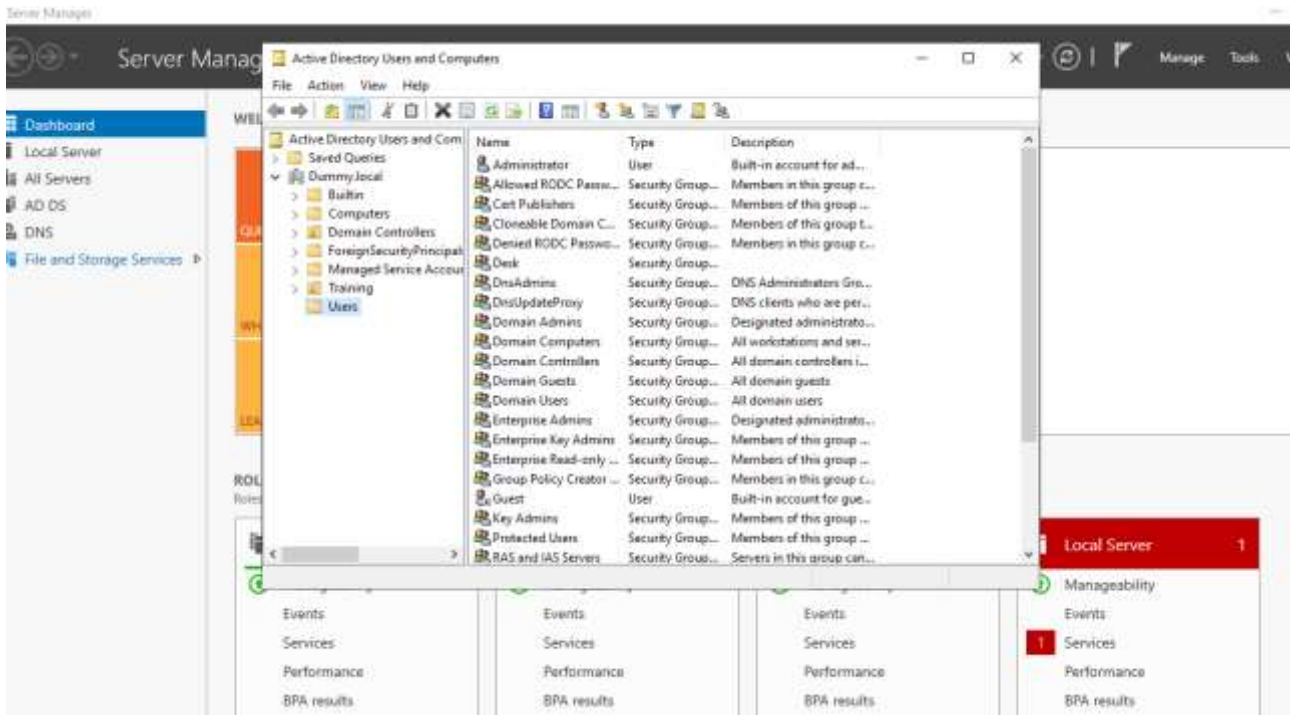
Derefter skal man følge wizarden færdig, her vil man skulle oprette et password, vælge om man vil oprette en DNS, verificere ens domæne navn, vælge hvor ens databases mapper skal være og til sidst bekræfte ens valg og installere. Når programmet er installeret skal den virtuelle maskine genstarte.



Når man skal logge ind igen vil det se således ud:



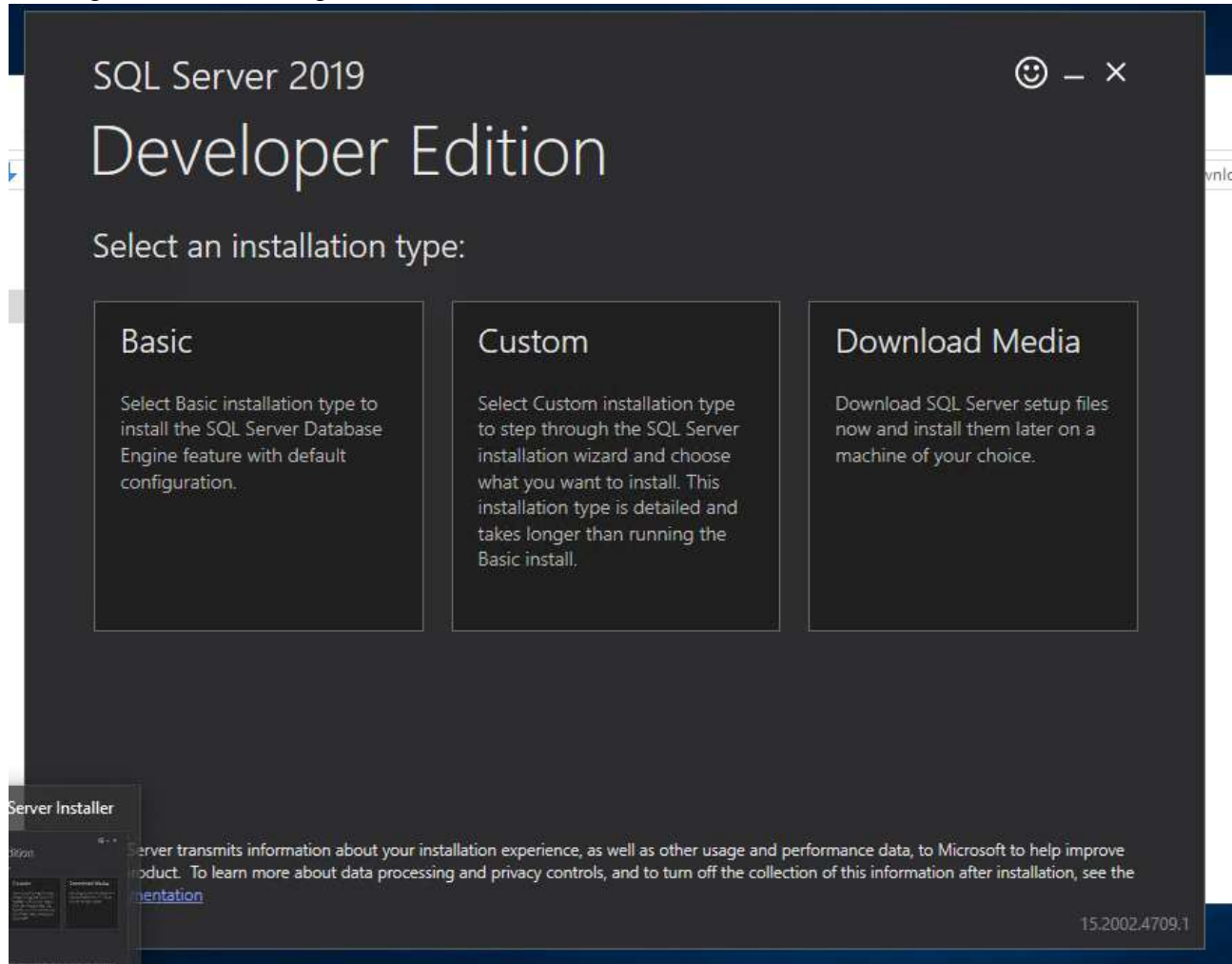
Nu skal man altså logge ind på domænet. Efter dette kan man oprette nye brugere på maskinen ved at gå ind i server management → Tools → AD DS → Users. Her kan man højre klikke og vælge ny user.



Her skal man skrive user logon name og oprette et password. Der skulle desuden oprettes 3 brugere ifølge opgaven. Jeg har valgt at kalde mine oprettede brugere User1, User2, User3 alle med adgangskoden "Password!".

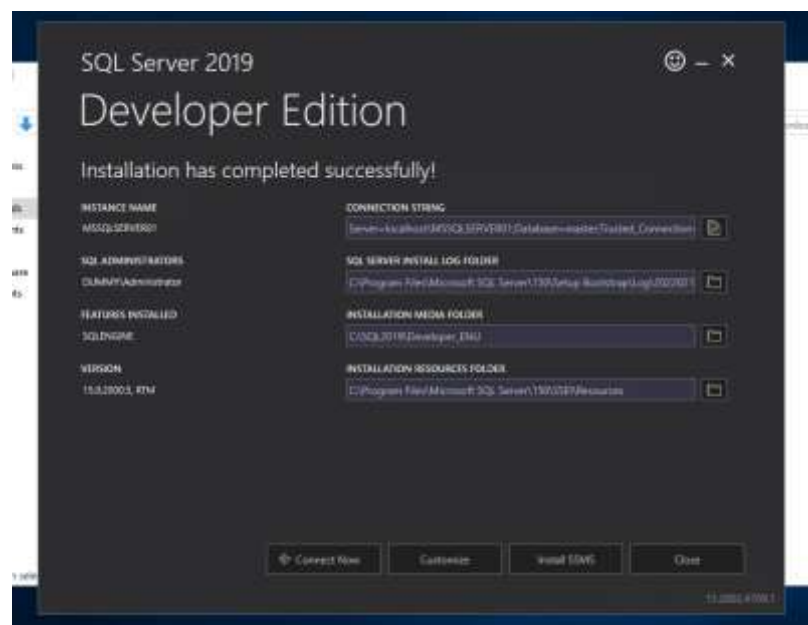
Installation af MS SQL DEVELOPER

Først skal man download Ms Sql Developer's exe fil og køre den, her vil en installations wizard åbne og den skal man følge



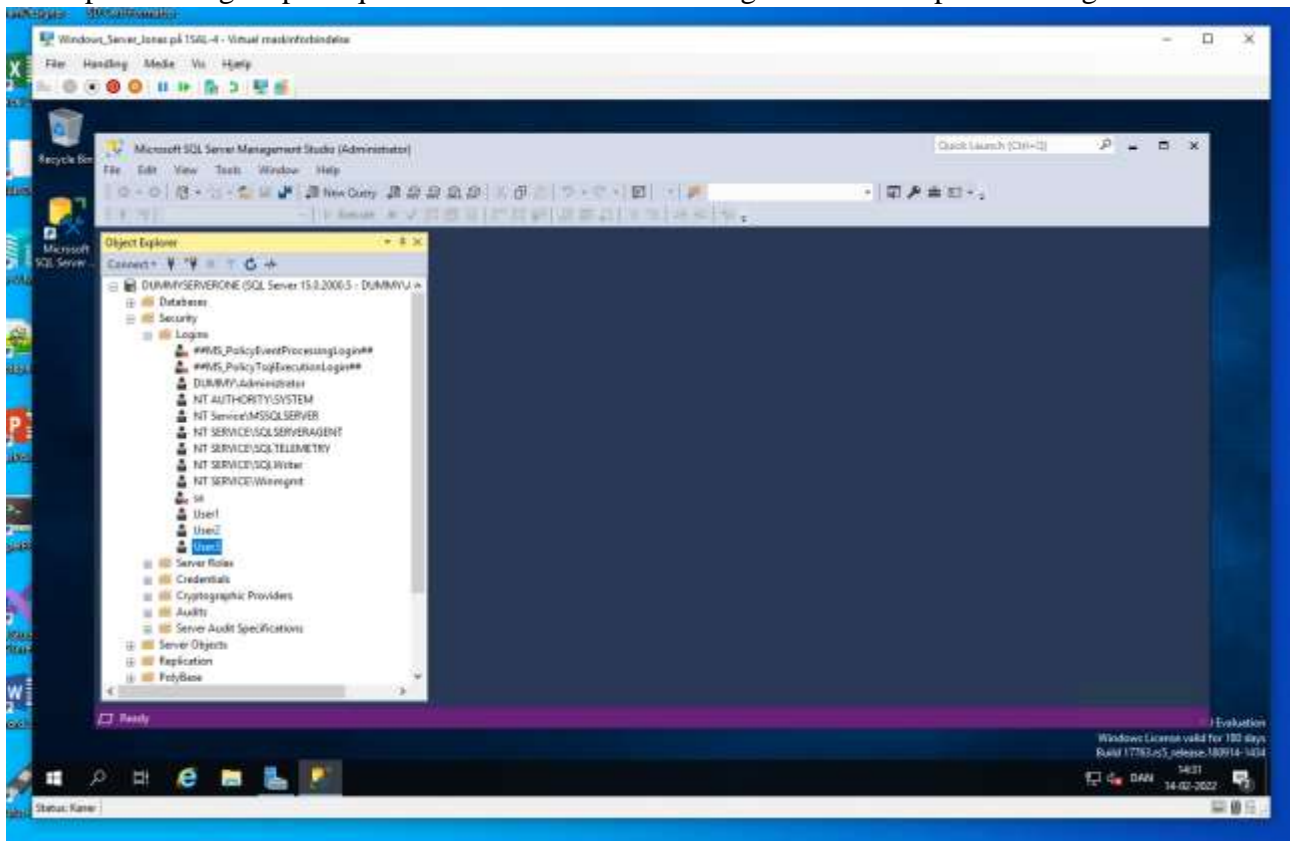
Her vælger man basic og fortsætter installationen. Nu vil man skulle vælge placering hvor databasen skal installeres, hvorefter installationen går i gang. Når installationen så er færdig får man denne skærm frem.

Her kan man desuden installere SSMS direkte, hvilke også er et krav for denne opgave. Dette vil åbne en hjemmeside hvorfra man kan downloade installations filen.

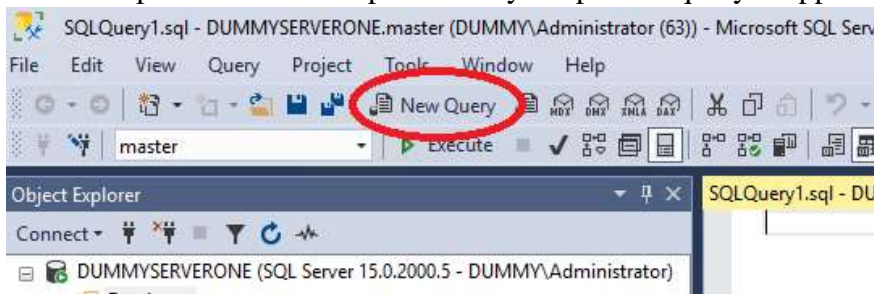


Her ses installations wizarden af ssms, når man trykker install vil installationen begynde automatisk og afslutte når den er færdig. Her skal man blot trykke close, hvorefter man kan starte med at benytte sql databasen.

I sql databasen skal man som det første forbinde til sin windows server, dette kan enten gøres ved windows authentication eller ved sql authentication, men da vi ikke har oprettet nogle sql brugere endnu benyttes windows authentication. Derefter navigere man til security → logins in object explorer. Hvorefter man kan højre klikke og på logins og vælge new login, hvilket er måden man opretter brugere på i sql databasen. Det er desuden også et krav at oprette 3 brugere her.



Derefter opretter man et script ved at trykke på new query i toppen af programmet.

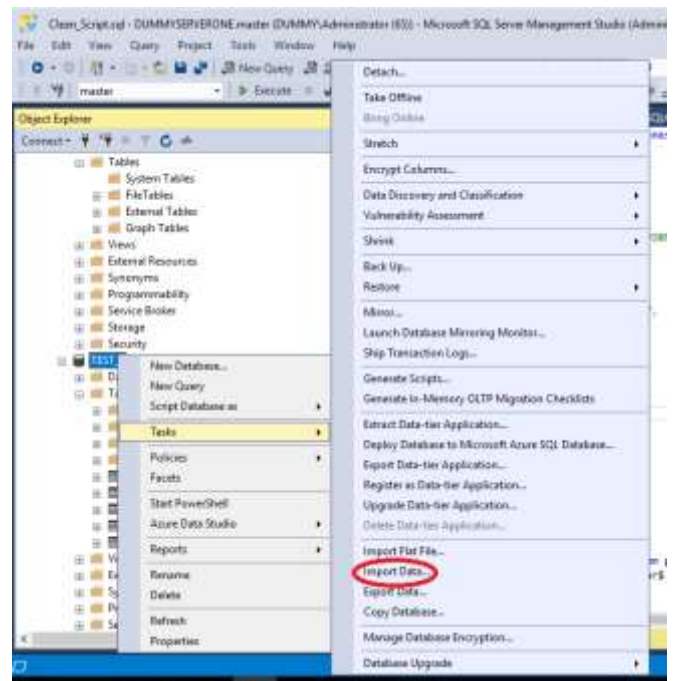


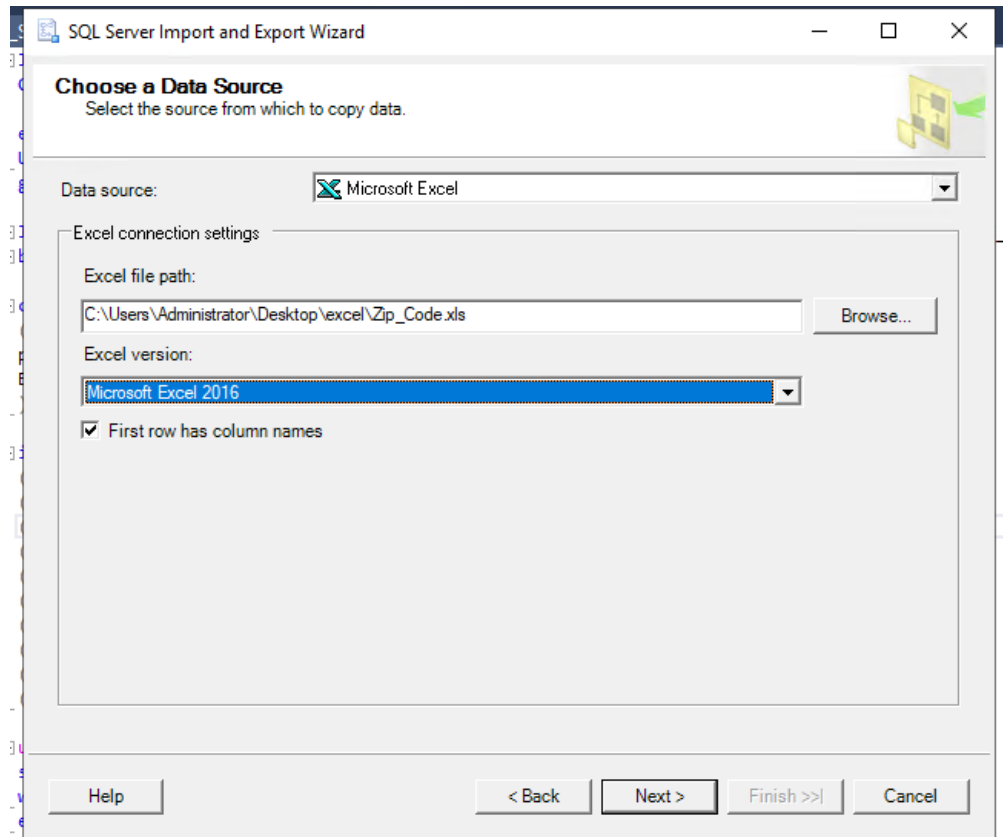
Herefter vil man få en tom side hvori man kan skrive sine queries, som tilsammen vil danne et script.

Desuden skal man også importere et excel ark og oprette det som en tabel. Før man kan importere som tabel skal man først have oprettet en database, dette gøres enten ved at højre klikke på databases og trykke new database, eller ved at køre et query med koden "CREATE DATABASE database_name;".

Når det er gjort højre klikker man på sin nyoprettede database, trykker task og vælger import data.

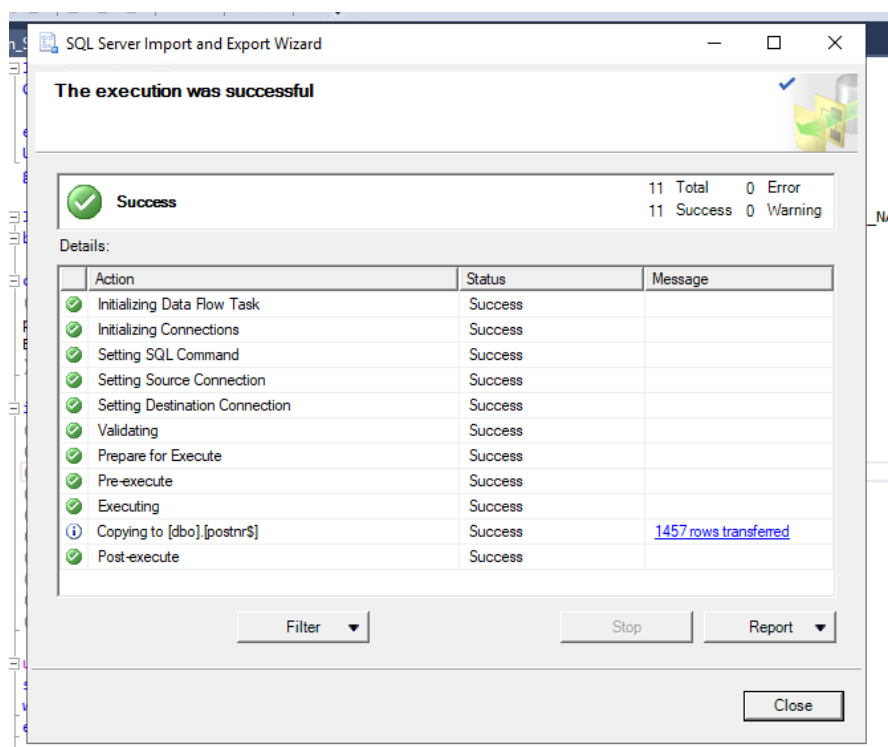
Derefter vil en wizard åbne, som man skal følge for at gennemføre sin import. Her skal man starte med at vælge sin importerede data-type, i dette tilfælde vil det være en excel fil. Udover datatypen skal man også vælge filens placering, samt version af excel. Wizarden auto vælger version 97-2003, men hvis du bruger en nyere version kan dette resultere i en fejl. Jeg fandt ud af på den hårde måde at jeg skulle bruge 2016 versionen.





Derefter skal man vælge en destination på importen, hvilket vil være serveren vi arbejder på, denne hedder "SQL Server native Client", derudover skal man også vælge hvilken database man vil importere til. Derefter skal man vælge copy data... of trykke næste.

Her kan du vælge at gennemse din tabel eller redigere i dine kolonner. Derefter skal man trykke næste indtil man kan fuldføre sin import.



Dokumentation af kode

```
-- IF NOT EXISTS (SELECT name FROM master.dbo.sysdatabases WHERE name = N'TEC') -- ser om databasen TEC eksistere, hvis den ikke gør det oprettes databasen TEC
CREATE DATABASE TEC;

else
Use TEC -- hvis databasen eksistere, bruges denne database.
go

-- hvis tabellen post_nr_by ikke eksistere startes nedenstående
IF NOT EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_SCHEMA = 'dbo' AND TABLE_NAME = 'post_nr_by')
begin

-- tabel post_nr_by oprettes med primary key post_nr og varchar by_navn
create table post_nr_by
(
post_nr int NOT NULL PRIMARY KEY,
by_navn varchar(255)
)

-- indsætte værdier i tabellen post_nr_by
insert into post_nr_by values
(2650, 'temp'),
(2300, 'temp'),
(2500, 'temp'),
(2610, 'temp'),
(3650, 'temp'),
(2830, 'temp'),
(2770, 'temp'),
(1824, 'temp'),
(2740, 'temp'),
(2750, 'temp');

end

--hvis tabellen postnr$ eksistere køres følgende
IF EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_SCHEMA = 'dbo' AND TABLE_NAME = 'postnr$')
begin

update post_nr_by -- opdatere tabellen post_nr_by
set by_navn = postnr$.by_navn from postnr$ -- by_navn fra post_nr_by sættes lig med by_navn fra tabellen postnr$
where post_nr_by.post_nr = postnr$.post_nr; -- dette sker kun hvor post_nr fra post_nr_by er lig med post_nr fra postnr$, altså der hvor postnumrene stemmer overens

end
```

```
-- ser om tabellen klasse eksistere, hvis den ikke gør det køres følgende
IF NOT EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_SCHEMA = 'dbo' AND TABLE_NAME = 'klasse')
begin

create table klasse -- tabel klasse oprettes med værdierne klasse_id primary key og klasse varchar
(
    klasse_id int NOT NULL PRIMARY KEY,
    klasse varchar(255) -- 255 indikere hvor mange tegn der kan være i hver enkelte element i klasse.
)

--indsætter følgende ind i tabellen klasse
insert into klasse values
(1, '1A'),
(2, '9B'),
(3, '4D');

end

-- ser om tabellen elev findes, hvis den ikke gør det køres følgende
IF NOT EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_SCHEMA = 'dbo' AND TABLE_NAME = 'elev')
begin

create table elev -- opretter tabellen elev med elev_id som primary key, fornavn som varchar, efternavn som varchar, adresse som varchar
-- , klasseid som foreign key der referere til klasse_id i tabellen klasse og post_nr foreign key der referere til post_nr i tabellen post_nr_by
(
    elev_id int NOT NULL PRIMARY KEY,
    fornavn varchar(255),
    efternavn varchar(255),
    adresse varchar(255),
    klasse_id int foreign key references klasse(klasse_id),
    post_nr int foreign key references post_nr_by(post_nr)
)

-- indsætter data ind i tabellen elev
insert into elev values
(1, 'Bo', 'Andersen', 'Gammel Byvej 12', '1', 2650),
(2, 'Frederikke', 'Hansen', 'Amager Boulevard 5', '2', 2300),
(3, 'Jens', 'Mikkelsen', 'Lily Brogbergs Vej 17', '3', 2500),
(4, 'Philip', 'Mortensen', 'Brunevang 90', '1', 2610),
(5, 'Kasper', 'Frederiksen', 'Bryggetorvet 32', '2', 3650),
(6, 'Milla', 'Jørgensen', 'Virum Torv 25', '3', 2830),
(7, 'fie', 'Kudsen', 'Allen 85', '1', 2770),
(8, 'Henrik', 'Madsen', 'Lily Brogbergs Vej 53', '2', 2500);

end
```



```

end

-- gør det samme som tabellen til elever, her er navnet bare lærer i stedet for elev.
IF NOT EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_SCHEMA = 'dbo' AND TABLE_NAME = 'lærer')
begin
create table lærer
(
    lærer_id int NOT NULL PRIMARY KEY,
    fornavn varchar(255),
    efternavn varchar(255),
    adresse varchar(255),
    klasse_id int foreign key references klasse(klasse_id),
    post_nr int foreign key references post_nr_by(post_nr)
)

insert into lærer values
(1, 'Tom', 'it', 'Sankt Thomas alle 3', '1', 1824),
(2, 'Lars', 'Henriksen', 'Nissedalen 76', '2', 2740),
(3, 'Mia', 'Hansen', 'Østervej 16', '3', 2750);

end

-- vælger fornavn og efternavn fra tabellen elev og klasse fra tabellen klasse her sammensættes tabellerne fra elev og klasse
-- dette sker kun hvor elevens klasse_id stemmer overens med klassens klasse_id
-- eg hvis elev1's klasse_id er 1 og klassen 1B's klasse_id er 1 vil vi få elev1 9b. (jeg har udeladt efternavn her i eksemplet)
select elev.fornavn, elev.efternavn, klasse.klasse
from elev
join klasse on elev.klasse_id = klasse.klasse_id;

-- vælger lærers fornavn og efternavn, samt bynavn
-- dette gøres ud fra tabellen lærer der bliver joinet af post_nr_by
-- dette sker kun hvor lærers postnummer er det samme som postnummeret fra post_nr_by
-- output vil være fornavn , efternavn , by_navn
select lærer.fornavn, lærer.efternavn, post_nr_by.bynavn
from lærer
join post_nr_by on lærer.post_nr = post_nr_by.post_nr;

-- vælger fornavn og efternavn fra elev, samt fornavn og efternavn fra lære
-- joiner elev og lære hvor elevens og lærers efternavn er det samme
select elev.fornavn, elev.efternavn, lærer.fornavn, lærer.efternavn
from elev
join lærer on elev.efternavn = lærer.efternavn;

```

Output af Join statements

	fomavn	eftemavn	klasse
1	Bo	Andersen	1A
2	Frederikke	Hansen	9B
3	Jens	Mikkelsen	4D
4	Philip	Mortensen	1A
5	Kasper	Frederiksen	9B
6	Milla	Jørgensen	4D
7	fie	Kudsen	1A
8	Henrik	Madsen	9B

	fomavn	eftemavn	by_navn
1	Tom	it	Frederiksberg C
2	Lars	Henriksen	Skovlunde
3	Mia	Hansen	Ballerup

	fomavn	eftemavn	fomavn	eftemavn
1	Frederikke	Hansen	Mia	Hansen

Det skal siges, at noget af koden er afhængig af, at man har importeret excel arket fra postnord og redigeret det således at tabel navnene stemmer overens med det jeg har kaldt dem. Hvis man ikke har det vil man få værdierne "TEMP" i bynavn.

Snake Case

I sql bruger man snake case, fordi sql ikke tager højde for store eller små bogstaver. Derudover kan der også være databaser som konvertere store bogstaver til små, hvilket kan gøre det uoverskueligt hvis man ikke brugte snake case.

Tabeller og Kolonner

I sql opretter man objekter med create, oftest er disse objekter tabeller, eller tables. Tabeller indeholder kolonner og disse kolonner indeholder hver især deres egen værdier. Når man opretter en tabel skal man også oprette mindst en kolonne. Når man opretter en kolonne skal man give den et navn og en variabel-type f.eks. int eller varchar. Efter at man har oprettet en tabel med kolonner kan man også indsætte værdier ind i kolonnerne. Disse værdier skal stemme overens med den type som kolonnen er defineret som. Man kan desuden ændre tabeller og derved tilføje eller fjerne kolonner. Dette gøres med 'ALTER TABLE tabel_navn ADD kolonne_navn variabel-type'. Man kan desuden også opdatere værdierne i kolonnerne med 'UPDATE' statement.

I ens tabeller, kan man også definere hvilken kolonne der er ens primary key, og hvilke der er foreign keys. En Primary key er en kolonne hvori alle værdierne er unikke. En Primary Key bruges som refererings punkt af andre tabeller. Foreign key kan man se som det omvendte af Primary Keys. Disse bruges af en tabel til at referere til andre tabellers Primary Keys. Med andre ord kan en tabel indeholde kolonnen elev_id som er en foreign key, der referere til kolonnen elev_id i en anden tabel, hvor den er en Primary Key.

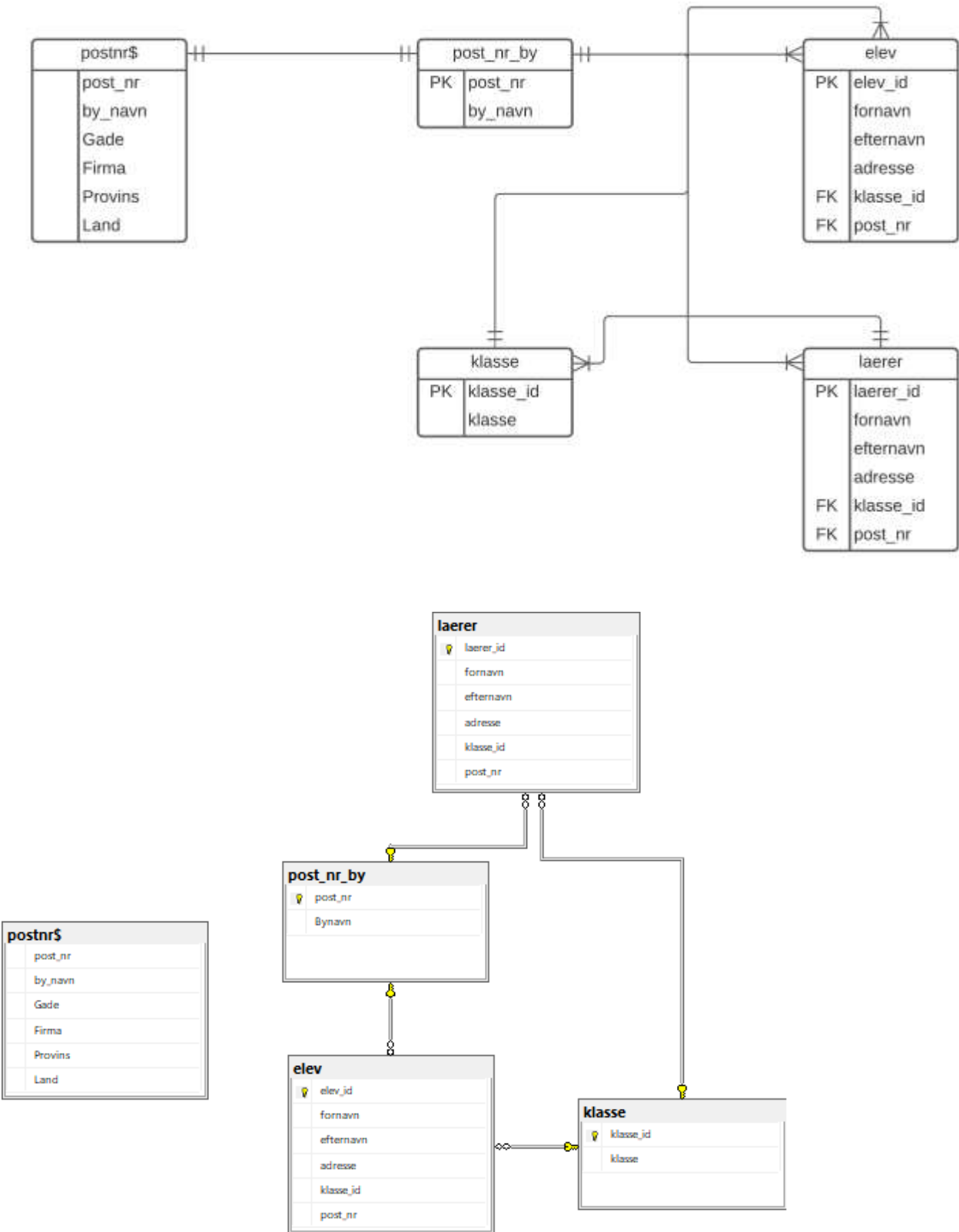
Når man har oprettet tabeller med kolonner og værdier, kan man begynde at analysere dem til at vise udvalgte kolonner. Dette gøres med et 'SELECT' statement, hvor man udvælger hvilke kolonner man vil benytte, efterfulgt af et 'FROM' statement, hvor man vælger hvilke tabeller der skal benyttes.

Man kan også benytte sig af et 'JOIN' statement, her tager man to tabeller og benytter Primary- og Foreign Keys til at sammenkoble data udfra et boolsk udtryk. Her vælger man først de kolonner man vil have vist med select, herefter et from statement, dernæst et Join statement hvor man, udvælger den tabel man ikke valgte efter sit from statement og til sidst et on statement, der beskriver hvordan de skal sammenkobles. F.eks.

```
'SELECT elev.fornavn, klasse.klasse_navn FROM elev JOIN klasse JOIN klasse on elev.klasse_id =
klasse.klasse_id'
```

Her bruger vi tabellerne elev og klasse. I tabellen elev findes kolonnerne fornavn og klasse_id, her er klasse id en foreign key. I tabellen klasse findes kolonnerne klasse_id og klasse_navn, her er klasse_id en primary key. Til sidst kobler vi dem sammen således at elevens navn kommer til at stå ved siden af den klasse som svare til elevens klasse_id. Hvis Hans har et klasse_id på 1 og klassen 9B har et klasse_id på 1, vil der komme til at stå 'Hans' i den første kolonne og '9B' i den anden kolonnen.

E/R Diagram og UML/Klasse Diagram



Konklusion

Idet jeg har arbejdet med en sql database liggende på en virtuel windows server oprettet på hyper-v, har jeg fået en forståelse for alle disse elementer. Af disse har jeg lært hvordan man opsætter en windows server fra en iso fil på en virtuel maskine, samt hvordan man opretter brugere til windows server og hvordan disse kan bruges til at verificere ens adgang til en sql database. Derudover har jeg lært at bruge en SQL database, hvori jeg har benyttet et dynamisk script der kan køre flere gange uden at slå fejl, samt hvordan man udvikler henholdsvis et E/R diagram og et UML/Klasse diagram til at afbillede en database. Derudover har jeg lært at anvende SQL-kodesproget til, at udføre oprette et script. I dette script har jeg brugt forskellige metoder til at oprette en database, oprette tabeller, indsætte værdier i tabellerne, opdatere en tabel, samt oprette join statements der viser sammenkoblede dele af forskellige tabeller.

Logbog

14-02-2022

I dag gik jeg i gang med SQL opgaven. Jeg startede ud med at læse mini guide til SQL og starten af Casen. Allerede her stødte jeg på problemer da der ikke var givet nogen forklaring på hvordan man installere windows server og ms sql databasen, hverken med eller uden hyper-v manager. Til sidst fandt jeg nogle iso filer, gemt afvejen på et fællesdrev. Herefter oprettede jeg en windows server 2019 og tilføjede AD DS til den og med dette tilføjede 3 yderligere brugere.

Efter dette skulle jeg også installere en MS SQL server, da jeg prøvede dette fandt jeg en mappe blandt iso filerne, som man kunne importere til hyper-v. Da jeg endelig fik importeret det fandt jeg ud af at der medfulgte 2 brugere med passwords, som jeg ikke kendt. Derfor slettede jeg den importerede virtuelle maskine, og prøvede at oprette en ny virtuel maskine med en iso fil tilhørende en ms sql server. Da jeg prøvede dette kunne den ikke gå i gang med opsætningen og sagde at den manglede en boot fil. Til sidst prøvede jeg at downloade en installationsfil til min windows server virtual machine. Det endte med at lykkes og prompte mig til også at installere SQL server management studio, hvilket jeg gjorde. Derudover fik jeg også oprettet mine 3 brugere til sql databasen.

Mine 3 brugere hedder desuden User1, User2, User3 og har alle det samme password "Password!"

15-02-2022

Jeg begyndte dagen med at lære om sql commands via codecademy.com. Her fuldendte jeg den gratis del af kurset om sql inden frokost.

Derefter begyndte jeg at opsætte min sql server med database. Her oprettede jeg også nogle scripts for at oprette min tabel, samt at indsætte noget data til nogle af tabellerne. Derefter brugte jeg resten af dagen på at importere et excel ark med postnumre på. Dette tog lang tid da instruktionerne fra opgaven ikke virkede. Jeg endte med at skulle køre import/export wizard fra ssms, fordi programmet ellers ikke ville modtage excel filer som input. Dette gjordes ved at højre klikke på databasen, vælge task, import data. Herfra fulgte jeg import wizarden og fik excel arket importeret som en tabel til min database. Derudover har jeg også set videoer om hvordan man opretter scripts i ssms <https://www.youtube.com/watch?v=RSIqWnP-Dy8&list=PLUaB-1hjhk8GT6N5ne2qpf603sF26m2PW> . *ssms betyder SQL Server Management Studio

16-02-2022

I dag startede jeg med at sammenkoble mit script, hvorefter jeg prøvede at indsætte tabellen fra excel arket ind i min tabel med post nummer og bynavn. Dette gav mange udfordringer da min første indsættelse returnere null. Jeg har stadig ikke fundet ud af hvorfor den gør det når jeg skriver

```
--insert into post_nr_by ( by_navn)
select postnr$.by_navn from postnr$, post_nr_by
where postnr$.post_nr = post_nr_by.post_nr
union
select post_nr_by.by_navn from post_nr_by;
```

følgende kode

	by_navn
1	NULL
2	Ballerup
3	Frederiksberg C
4	Hvidovre
5	Kastrup
6	København S
7	Rødovre
8	Skovlunde
9	Valby
10	Virum
11	Ølstykke

Fordi den første værdi er null kan jeg ikke bruge insert into. Der opstår en fejl når jeg prøver.

Jeg fandt senere ud af at der er en eksisterende row i min tabel by_navn der har værdierne Null, Null. Jeg ved ikke hvorfor eller hvordan den er opstået. Min primary key er post_nr og burde ikke kunne være null.

Jeg besluttede mig for at starte på frisk. Derfor oprettede jeg en ny database. Her brugte jeg powershell for at importere mit excel ark til den virtuelle maskine fra min host. Dette endte med at resultere i en fejl, der gjorde at en del af dataen blev slettet. Dette tog langtid for mig at opdage, men til sidst gjorde jeg. Desuden var fejlen at jeg importerede den som en 97-2003 version excel ark i stedet for 2016. Desuden oprettede jeg også min host's c-drev som en fælles mappe, for lettere at kunne tilgå filer. Derefter begyndte jeg at teste for om man kunne joine kolonnerne post_nr fra tabel1 og by_navn fra tabel2 sammen. Dette lykkedes med koden:

```
select postnr$.by_navn, post_nr_by.post_nr from postnr$
right join post_nr_by
on postnr$.post_nr = post_nr_by.post_nr
```

100 %

Results Messages

	by_navn	post_nr
1	Frederiksberg C	1824
2	København S	2300
3	Valby	2500
4	Rødovre	2610
5	Hvidovre	2650
6	Skovlunde	2740
7	Ballerup	2750
8	Kastrup	2770
9	Virum	2830
10	Ølstykke	3650

Jeg fandt dog aldrig ud af hvorfor en af mine kolonner havde null værdier før.

For at indsætte dataen i min tabel post_nr_by prøvede jeg at benytte insert into select i lang tid, men

```
update post_nr_by
set Bynavn = postnr$.by_navn from postnr$
where post_nr_by.post_nr = postnr$.post_nr;
```

fandt til sidst ud af at jeg skulle bruge update.

Her kan man se hvordan tabellen endte med at se ud.

Sidst på dagen begyndte jeg at skrive min kode mere dynamisk ved at bruge en masse if statements i sammenhæng med EXISTS. Dette gøres for at scriptet skal kunne køres igennem flere gange uden at der opstår fejl.

	post_nr	Bynavn
▶	1824	Frederiksberg C
	2300	København S
	2500	Valby
	2610	Rødovre
	2650	Hvidovre
	2740	Skovlunde
	2750	Ballerup
	2770	Kastrup
	2830	Virum
	3650	Ølstykke
*	NULL	NULL

17-02-2022

I dag skrev jeg resten af mit script færdigt, jeg manglede kun de 3 join queries. Derefter lavede jeg et E/R diagram og et uml/klasse diagram. Med dette var jeg færdig med opgaven og begyndte derfor på at skrive rapporten, samt at lave kommentarer til min sql kode.

Jeg kom meget langt med min rapport efter frokost, og mangler nu kun konklusionen, samt at indsætte denne logbog.

18-02-2022

I dag startede jeg med at skrive en sektion om kolonner og tabeller i min rapport.

Derefter indsatte jeg billeder af mit output fra min kode.

Med det blev jeg færdig med min rapport.