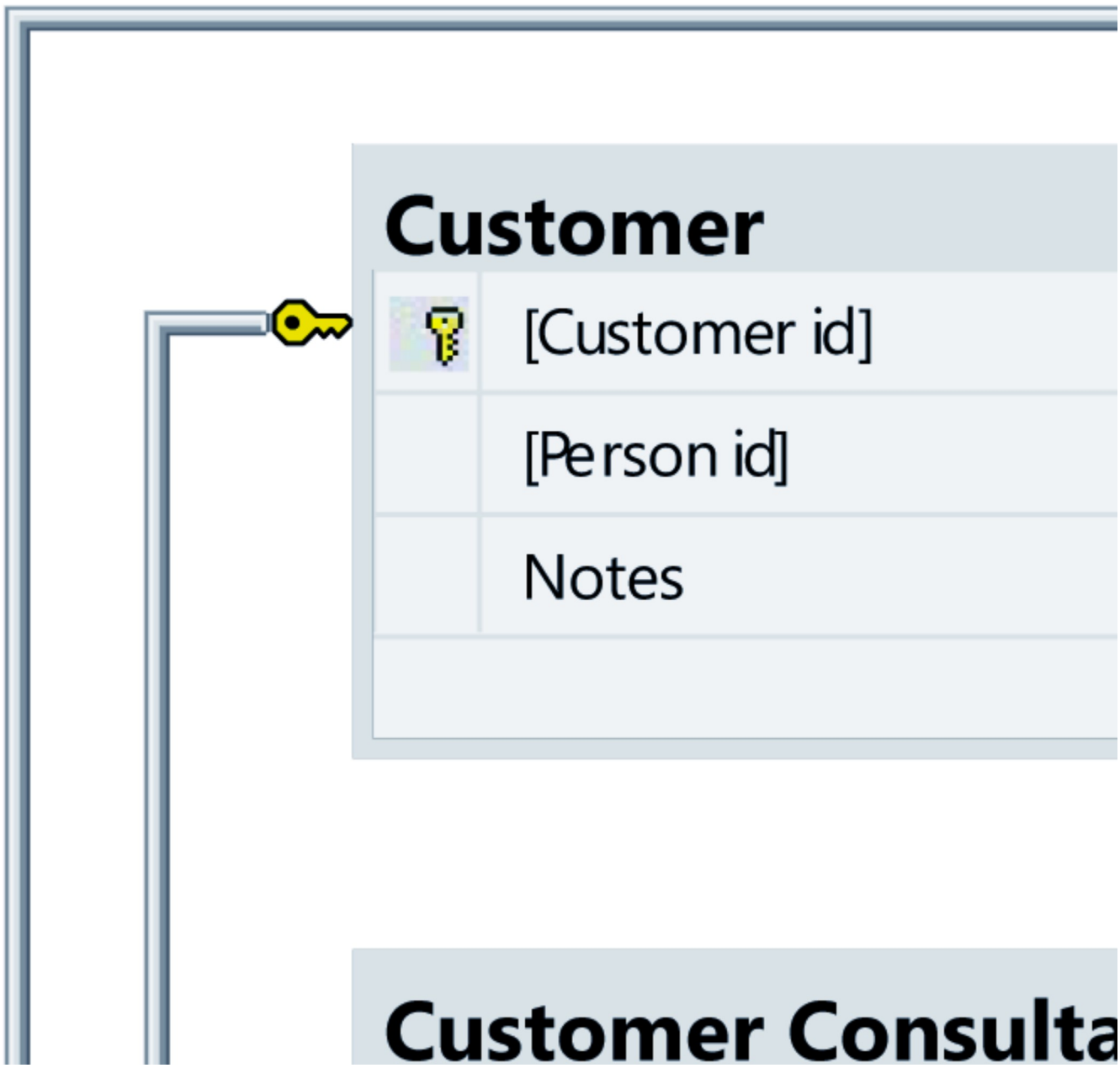
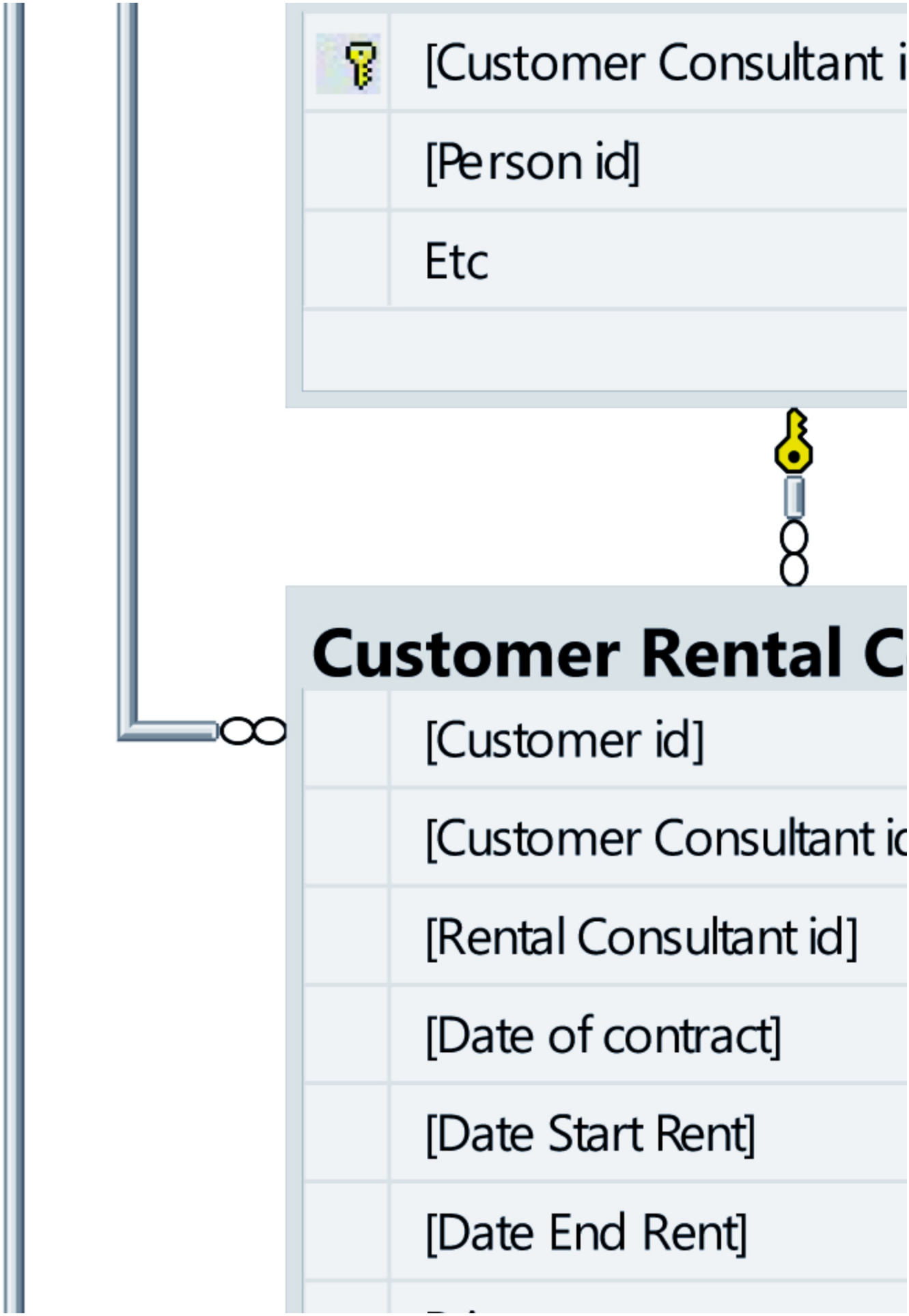
Wiki

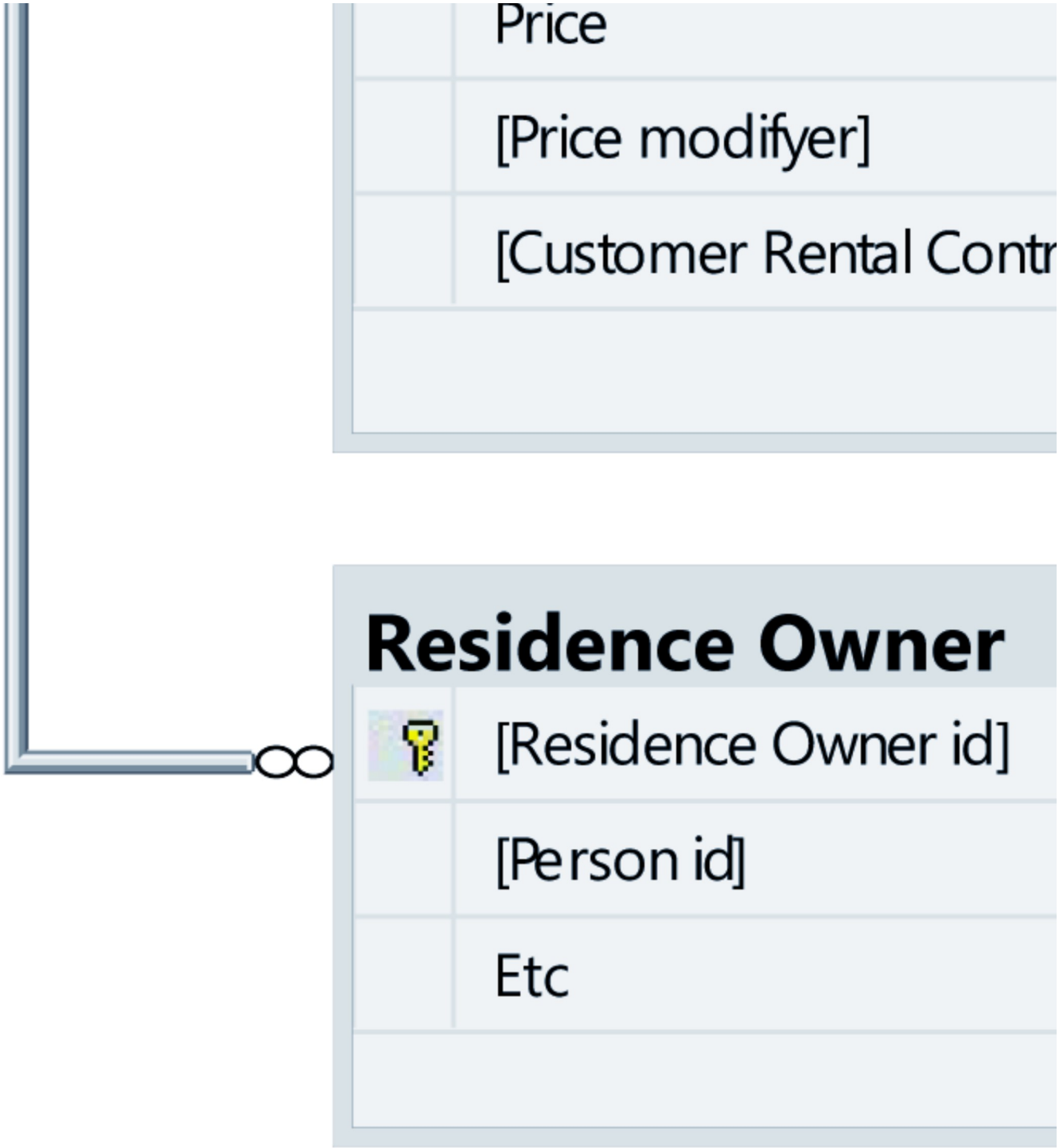
Dokumentation • Database • C# • Tests • Log, Tidsplan • Manual

Database

1. Entite-Relations Diagram




[Customer Consultant i
[Person id]
Etc
Customer Rental C
[Customer id]
[Customer Consultant id
[Rental Consultant id]
[Date of contract]
[Date Start Rent]
[Date End Rent]



2. Ms SQl server

2.1 Login

Da der ikke på dette H1 hovedforløb er snakket specielt meget om bruger styring af Microsoft SQL Server bruger konrol , er der heller ikke brugt nævneværdige resourser på dette forhold! Der er ikke lagt op til at vi skal håndtere dette emne. Ergo:

Har alle brugere det samme password og kan i praksis ca. lige lidt. på nær hvis man starter programmet med et user id og et password. Eller som de for tiden siger i USA: Sikkerhed er en by i Rusland.

3. Scripts

3.1 Opret database

```
USE master;
DROP DATABASE IF EXISTS "Sydvest-Bo";

print CHAR(13) + CHAR(10) + '             ************************************************'
print '             **  Databasen ''Sydvest-Bo'' oprettes på ny    **'
CREATE DATABASE "Sydvest-Bo";
GO
print '             **                                             **'
USE "Sydvest-Bo";
GO
print '             **  og er taget i bruge..                      **'
print '             ************************************************' + CHAR(13) + CHAR(10)
GO

CREATE TABLE "Address"
(
  "Address id" INT           NOT NULL IDENTITY(1,1),
  "ZipCode id" INT           NOT NULL,
  "Adresse"    nvarchar(127) NOT NULL DEFAULT '',
  CONSTRAINT "PK_Address" PRIMARY KEY ("Address id")
)
GO

CREATE TABLE Customer
(
  "Customer id" INT      NOT NULL IDENTITY(1,1),
  "Person id"   INT      NOT NULL,
  "Notes"       nvarchar NOT NULL DEFAULT '',
  CONSTRAINT "PK_Customer" PRIMARY KEY ("Customer id")
)
GO

CREATE TABLE "Customer Consultant"
(
  "Customer Consultant id" INT      NOT NULL IDENTITY(1,1),
  "Person id"              INT      NOT NULL,
  "Etc"                    nvarchar NOT NULL DEFAULT '',
  CONSTRAINT "PK_Customer Consultant" PRIMARY KEY ("Customer Consultant id")
)
GO

CREATE TABLE "Customer Rental Contract"
(
  "Customer id"                   INT      NOT NULL,
  "Customer Consultant id"        INT      NOT NULL,
  "Rental Consultant id"          INT      NOT NULL,
  "Date of contract"              DATE     NOT NULL DEFAULT GETDATE(),
  "Date Start Rent"               DATE     NOT NULL DEFAULT GETDATE(),
  "Date End Rent"                 DATE     NOT NULL DEFAULT GETDATE(),
  "Price"                         money    NOT NULL DEFAULT 0.0,
  "Price modifyer"                FLOAT    NOT NULL DEFAULT 1.0,
  "Customer Rental Contract Text" nvarchar NOT NULL DEFAULT ''
)
GO
```

```
CREATE TABLE District
(
  "Area id"              INT           NOT NULL IDENTITY(1,1),
  "Area name"            nvarchar(200) NOT NULL,
  "Rental consultant id" INT           NOT NULL,
  CONSTRAINT "PK_District" PRIMARY KEY ("Area id")
)
GO

CREATE TABLE "Independant overseer"
(
  "Independant overseer id" INT      NOT NULL IDENTITY(1,1),
  "Person id"               INT      NOT NULL,
  "Etc"                     nvarchar NOT NULL DEFAULT '',
  CONSTRAINT "PK_Independant overseer" PRIMARY KEY ("Independant overseer id")
)
GO

CREATE TABLE Person
(
  "Person id"  INT           NOT NULL IDENTITY(1,1),
  "Address id" INT           NOT NULL,
  "Name"       nvarchar(47)  NOT NULL DEFAULT '',
  "Last name"  nvarchar(79)  NOT NULL DEFAULT '',
  "Email"      nvarchar(127) NOT NULL DEFAULT '',
  "PhoneNo"    nvarchar(20)  NOT NULL DEFAULT '',
  CONSTRAINT "PK_Person" PRIMARY KEY ("Person id")
)
GO

CREATE TABLE "Rental Consultant"
(
  "Rental Consultant id" INT NOT NULL IDENTITY(1,1),
  "Person id"            INT NOT NULL,
  CONSTRAINT "PK_Rental Consultant" PRIMARY KEY ("Rental Consultant id")
)
GO

CREATE TABLE Residence
(
  "Residence id"                INT          NOT NULL IDENTITY(1,1),
  "Residence Owner Contract id" INT          NOT NULL,
  "Independant overseer id"     INT          NOT NULL,
  "Address id"                  INT          NOT NULL,
  "Area"                        INT          NOT NULL,
  "Size"                        INT          NOT NULL DEFAULT 0,
  "Rooms"                       INT          NOT NULL DEFAULT 0,
  "Number of beds"              INT          NOT NULL DEFAULT 0,
  "Rental quality"              nvarchar(47) NOT NULL DEFAULT '',
  "Etc"                         nvarchar     NOT NULL DEFAULT '',
  "Base Price"                  money        NOT NULL DEFAULT 0.0,
  "Residence Type"              nvarchar(31) NOT NULL DEFAULT '',
  CONSTRAINT "PK_Residence" PRIMARY KEY ("Residence id")
)
GO

CREATE TABLE "Residence Owner"
(
  "Residence Owner id" INT      NOT NULL IDENTITY(1,1),
  "Person id"          INT      NOT NULL,
  "Etc"                nvarchar NOT NULL DEFAULT '',
  CONSTRAINT "PK_Residence Owner" PRIMARY KEY ("Residence Owner id")
)
```

```
GO

CREATE TABLE "Residence Owner Contract"
(
  "Residence Owner Contract id"   INT      NOT NULL IDENTITY(1,1),
  "Residence Owner id"            INT      NOT NULL,
  "Rental consultant id"          INT      NOT NULL,
  "Date of contract"              DATE     NOT NULL DEFAULT GETDATE(),
  "Date Start Rent"               DATE     NOT NULL DEFAULT GETDATE(),
  "Date End Rent"                 DATE     NOT NULL DEFAULT GETDATE(),
  "Price"                         money    NOT NULL DEFAULT 0.0,
  "Residence Owner Contract Text" nvarchar NOT NULL DEFAULT '',
  CONSTRAINT "PK_Residence Owner Contract" PRIMARY KEY ("Residence Owner Contract id")
)
GO

CREATE TABLE Vacancy
(
  "year"                         DATE  NOT NULL,
  "week"                         DATE  NOT NULL,
  "Residence id"                 INT   NOT NULL,
  "Udlejnings pris"              money NOT NULL DEFAULT 0.0,
  "Residence Owner Contract id" INT   NOT NULL DEFAULT 0,
  CONSTRAINT "PK_Vacancy" PRIMARY KEY (YEAR, week, "Residence id")
)
GO

CREATE TABLE "ZipCode Town"
(
  "ZipCode id" INT  CHECK ("ZipCode id" > 99 AND "ZipCode id" < 10000) NOT NULL,
  "Town"       nvarchar(79)                                            NOT NULL DEFAULT '''',
  CONSTRAINT "PK_ZipCode Town" PRIMARY KEY ("ZipCode id")
)
GO

ALTER TABLE "ZipCode Town"
  ADD CONSTRAINT "UQ_ZipCode id" UNIQUE ("ZipCode id")
GO

ALTER TABLE Residence
  ADD CONSTRAINT "FK_District_TO_Residence"
    FOREIGN KEY (Area)
    REFERENCES District ("Area id")
GO

ALTER TABLE "Residence Owner"
  ADD CONSTRAINT "FK_Person_TO_Residence Owner"
    FOREIGN KEY ("Person id")
    REFERENCES Person ("Person id")
GO

ALTER TABLE "Customer Consultant"
  ADD CONSTRAINT "FK_Person_TO_Customer Consultant"
    FOREIGN KEY ("Person id")
    REFERENCES Person ("Person id")
GO

ALTER TABLE "Rental Consultant"
  ADD CONSTRAINT "FK_Person_TO_Rental Consultant"
    FOREIGN KEY ("Person id")
    REFERENCES Person ("Person id")
GO

ALTER TABLE District
```

```
  ADD CONSTRAINT "FK_Rental Consultant_TO_District"
    FOREIGN KEY ("Rental consultant id")
    REFERENCES "Rental Consultant" ("Rental Consultant id")
GO

ALTER TABLE "Customer Rental Contract"
  ADD CONSTRAINT "FK_Customer_TO_Customer Rental Contract"
    FOREIGN KEY ("Customer id")
    REFERENCES Customer ("Customer id")
GO

ALTER TABLE Customer
  ADD CONSTRAINT "FK_Person_TO_Customer"
    FOREIGN KEY ("Person id")
    REFERENCES Person ("Person id")
GO

ALTER TABLE "Independant overseer"
  ADD CONSTRAINT "FK_Person_TO_Independant overseer"
    FOREIGN KEY ("Person id")
    REFERENCES Person ("Person id")
GO

ALTER TABLE Person
  ADD CONSTRAINT "FK_Address_TO_Person"
    FOREIGN KEY ("Address id")
    REFERENCES "Address" ("Address id")
GO

ALTER TABLE "Address"
  ADD CONSTRAINT "FK_ZipCode Town_TO_Address"
    FOREIGN KEY ("ZipCode id")
    REFERENCES "ZipCode Town" ("ZipCode id")
GO

ALTER TABLE Residence
  ADD CONSTRAINT "FK_Residence Owner Contract_TO_Residence"
    FOREIGN KEY ("Residence Owner Contract id")
    REFERENCES "Residence Owner Contract" ("Residence Owner Contract id")
GO

ALTER TABLE Residence
  ADD CONSTRAINT "FK_Address_TO_Residence"
    FOREIGN KEY ("Address id")
    REFERENCES Address ("Address id")
GO

ALTER TABLE Residence
  ADD CONSTRAINT "FK_Independant overseer_TO_Residence"
    FOREIGN KEY ("Independant overseer id")
    REFERENCES "Independant overseer" ("Independant overseer id")
GO

ALTER TABLE "Residence Owner Contract"
  ADD CONSTRAINT "FK_Rental Consultant_TO_Residence Owner Contract"
    FOREIGN KEY ("Rental consultant id")
    REFERENCES "Rental Consultant" ("Rental Consultant id")
GO

ALTER TABLE Vacancy
  ADD CONSTRAINT "FK_Residence_TO_Vacancy"
    FOREIGN KEY ("Residence id")
    REFERENCES Residence ("Residence id")
GO
```

```
ALTER TABLE Vacancy
  ADD CONSTRAINT "FK_Residence Owner Contract_TO_Vacancy"
    FOREIGN KEY ("Residence Owner Contract id")
    REFERENCES "Residence Owner Contract" ("Residence Owner Contract id")
GO

ALTER TABLE "Customer Rental Contract"
  ADD CONSTRAINT "FK_Rental Consultant_TO_Customer Rental Contract"
    FOREIGN KEY ("Rental Consultant id")
    REFERENCES "Rental Consultant" ("Rental Consultant id")
GO

ALTER TABLE "Customer Rental Contract"
  ADD CONSTRAINT "FK_Customer Consultant_TO_Customer Rental Contract"
    FOREIGN KEY ("Customer Consultant id")
    REFERENCES "Customer Consultant" ("Customer Consultant id")
GO

ALTER TABLE "Residence Owner Contract"
  ADD CONSTRAINT "FK_Residence Owner_TO_Residence Owner Contract"
    FOREIGN KEY ("Residence Owner id")
    REFERENCES "Residence Owner" ("Residence Owner id")
GO

------ USER controle
USE "Sydvest-Bo"
GO

ALTER ROLE db_datareader ADD MEMBER tec
GO

ALTER ROLE db_datawriter ADD MEMBER tec
GO

ALTER ROLE db_ddladmin ADD MEMBER tec
GO

---- Insert from Resources
/*
BULK INSERT "ZipCode Town"
FROM 'd:\TEC.Datatekniker\H1\Versionering og Dokumentation\H1-Case-1\Resources\postnumre.csv'
WITH
(
    FORMAT = 'CSV',
    FIELDQUOTE = '"',
    FIRSTROW = 2,
    FIELDTERMINATOR = ';',  --CSV field delimiter
    ROWTERMINATOR = '\n',   --Use to shift the control to next row
    TABLOCK
)
GO
*/

/*
-- BULK INSERT "ZipCode Town"
BULK INSERT [dbo].[ZipCode Town]
FROM 'd:\TEC.Datatekniker\H1\Versionering og Dokumentation\H1-Case-1\Resources\postnumre.csv'
with (
    FIRSTROW = 2,
    FIELDTERMINATOR = ';',
    ROWTERMINATOR = '\n',
    BATCHSIZE = 250000,
    MAXERRORS = 2,
```

```
    CODEPAGE = 65001
);
GO
*/
```

4.

5.

6. Test Data

Til at test databasen, har vi lavet en række filer med tupler der kan indsættes i forskellige sammenhænge. Som base herfor har vi anvendt de ærede medlemmer af Folketinget som vi tidligere har arbejdet med. Oversigten findes i filen Sydvest-Bo_testfiles.txt. For nemheds skyld har alle personerne fået det samme password: Spassw0rd

Test DB Data	se filen: **Sydvest-Bo_testfiles.txt**	
Filnavn	**Tabel**	**Kriterie**
ZipCode_Town.csv	ZipCode Town	Danske postnumre fra post nord.
Sydvest-Bo_test_Customers.csv	Customers	Radikale Venstre
Sydvest-Bo_test_Customer_Consultants.csv	Customer Consultant	Det Konservative Folkeparti
Sydvest-Bo_test_Independant_Overseers.csv	Independant Overseers	Uafhængige og mindre partier
Sydvest-Bo_test_Rental_Consultants.csv	Rental Consultants	Socialistisk Folkeparti
Sydvest-Bo_test_Residence_Owners_Apartments.csv	Residence Owners (Apartments)	Liberal Alliance
Sydvest-Bo_test_Residence_Owners_Summerhouses.csv	Residence Owners (Summerhouses)	Dansk Folkeparti
Sydvest-Bo_test_Residence_Apartments.csv	Residence (Apartments)	Venstres adresser
Sydvest-Bo_test_Residence_Summerhouses.csv	Residence (Summerhouses)	Social Demokratiets adresser

6.1 ZipCode Town

Danske postnumre og Bynavne fra post nord, inc Grønland og nordatlanten.

6.2 Customers

De ærede medlemmer fra det Radikale Venstre optræder som feriegester og kunder i Sydvest-Bo.

6.3 Customer Consultant

De ærede medlemmer fra Det Konservative Folkeparti optræder som kunde konsulenter.

6.4 Independant_Overseers

De ærede medlemmer fra mindre partier og uafhængige optræder som rengærings assistentere.

6.5 Rental Consultants

De ærede medlemmer fra Socialistisk Folkeparti optræder som Udlejnings konsulenter

6.6 Residence Owners (Apartments)

De ærede medlemmer fra Liberal Alliance optræder som ejere af ferielejligheds komplekser.

6.7 Residence Owners (summerhouses)

De ærede medlemmer fra Dansk Folkeparti optræder som sommerhus ejere.

6.8 Residence_Apartments

De ærede medlemmer fra Venstre har lagt adresser til denne kategori.

6.9 Residence_Summerhouses

De ærede medlemmer fra Social demokratiet har lagt adresser til denne kategori.

7.

8.

9.

10.

11. OCDB
