

Database

Det første der bliver lavet i projektet, er databasen. Da det er den som applikationen bliver bygget efter. I databasen blev der kigget på vores case og snakkes omkring hvad der skal med i projektet og hvordan man kan danne en applikation ud fra databasen. Efter der blev snakket og planlagt hvordan projektet skal løses i projektet, blev der lavet en skitse af relationel model for at få overblik over hvordan man kan lave projektet. Efter den blev lavet giver den overblik over hvordan databasen skal se ud. Der startes med at lave en database som bliver kaldt for "project_commerce".

```
DROP DATABASE IF EXISTS project_commerce;  
CREATE DATABASE project_commerce;  
USE project_commerce;
```

Her vises der vores database. Der bliver skrevet op på denne for at man kan køre hele databasen selvom der er noget man skal have rettet på.

Efter man har oprettet databasen, bliver der lavet tabeller ud fra det der blev lært i timerne. Her vises der et eksempel på hvordan nogle af tabellerne vil se ud. Der bliver f.eks. vist hvad tabellen indeholder og hvilke constraints den har.

```

• ⊖ CREATE TABLE Kunder(
    Kunde_ID int NOT NULL AUTO_INCREMENT,
    Navn varchar(45) NOT NULL,
    Adresse varchar(45) NOT NULL,
    Postnr int NOT NULL,
    Bynavn varchar(45) NOT NULL,
    Telefon varchar(45) NOT NULL,
    Email varchar(45) NOT NULL,
    Password varchar(45) NOT NULL,
    Nyhedsbrev varchar(10) NOT NULL,
    Kundedato datetime,
    PRIMARY KEY(Kunde_ID)
);

• ⊖ CREATE TABLE Firmaby(
    Bynavn_ID int NOT NULL AUTO_INCREMENT,
    Postnr int NOT NULL,
    Bynavn varchar(45) NOT NULL,
    PRIMARY KEY(Bynavn_ID)
);

• ⊖ CREATE TABLE Firmaadresse(
    Firmaadresse_ID int NOT NULL AUTO_INCREMENT,
    Adresse varchar(45) NOT NULL,
    Telefon varchar(45) NOT NULL,
    Email varchar(45) NOT NULL,
    Bynavn_ID int NOT NULL,
    PRIMARY KEY(Firmaadresse_ID),
    FOREIGN KEY (Bynavn_ID) REFERENCES Firmaby(Bynavn_ID)
);

```

Derudover som vist i "Firmaadresse" er det ikke kun primary key men der bliver også tilføjet foreign key for at få relationerne til at passe med hinanden. Der bliver oprettet tabellerne: Produkter, hylder, lagerstatus, leveringsdato, reoler, lokation, firmaadresse, firmaby, ordre, kunder og ordrelinjer (ER-modellen vises nedenfor for et overblik over databasen).

Efter der er lavet tables vil der bagefter være muligt for at indsætte values. I disse values bliver der valgt at tage udgangspunkt i 10 forskellige varer som bliver solgt i 3 forskellige lokationer. Dette bruges til udgangspunkt til vise hvordan applikationen vil virke når man skal købe en ting. Derudover bliver der også lavet values i de andre tabeller såsom priser, hvor man kan finde varerne og hvor mange vare der er tilbage på lager.

Efter der er sat values ind bliver der dannet nogle triggere som er et lagret program (med forespørgsler), som udføres automatisk for at reagere på en specifik hændelse, såsom indsættelse, opdatering eller sletning, der forekommer i en tabel.

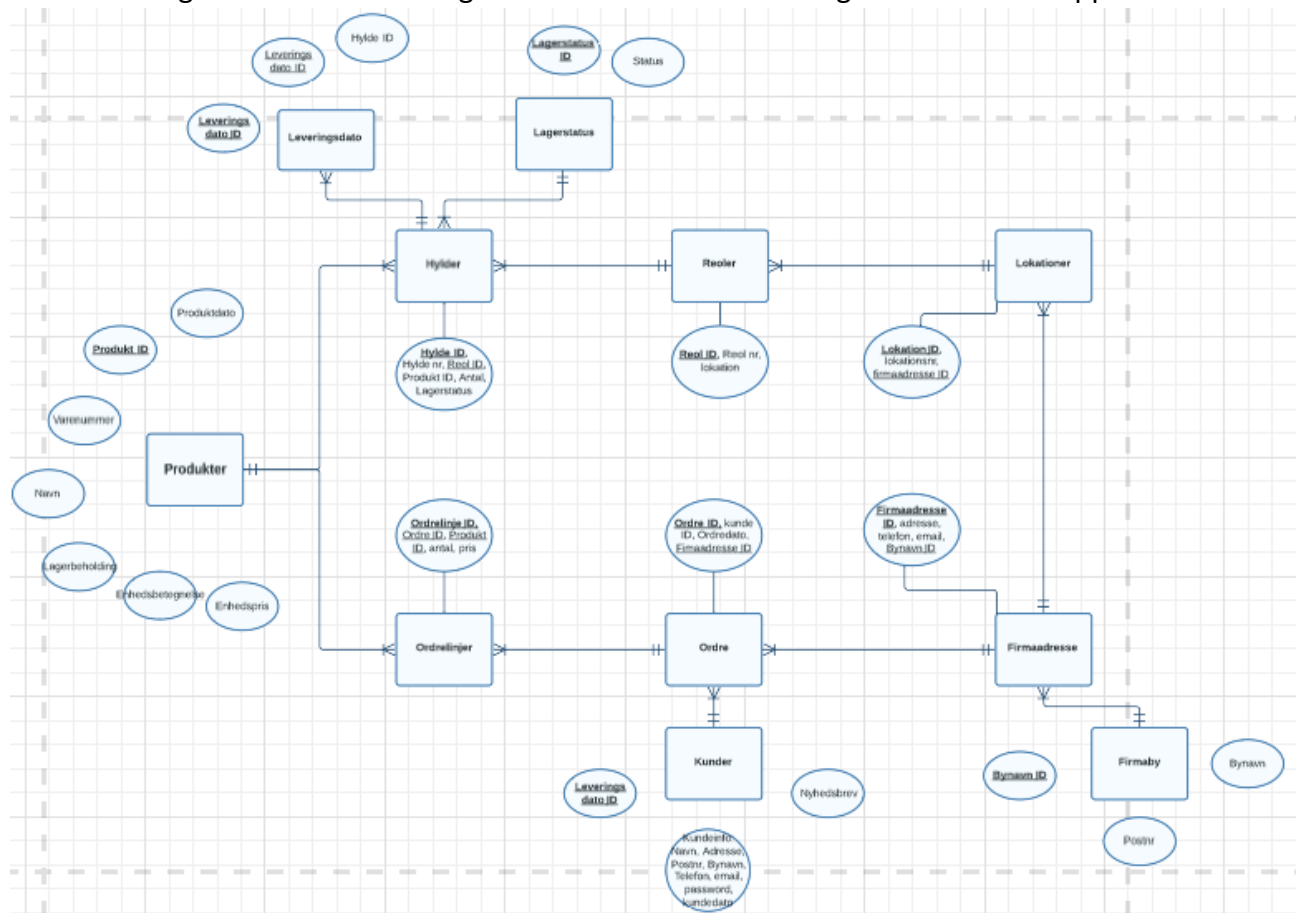
```
CREATE TRIGGER before_kunde_update
BEFORE UPDATE ON Kunder
FOR EACH ROW
INSERT INTO Kundelog
SET action = 'update', Kunde_ID=OLD.Kunde_ID, Navn=OLD.Navn, Adresse = OLD.Adresse, Postnr = OLD.Postnr, Bynavn = OLD.

CREATE TRIGGER before_kunde_delete
BEFORE DELETE ON Kunder
FOR EACH ROW
INSERT INTO Kundelog
SET action = 'delete', Kunde_ID=OLD.Kunde_ID, Navn=OLD.Navn, Adresse = OLD.Adresse, Postnr = OLD.Postnr, Bynavn = OLD.

CREATE TRIGGER before_kunde_insert
BEFORE INSERT ON Kunder
FOR EACH ROW
INSERT INTO Kundelog
SET action = 'insert', Kunde_ID=NEW.Kunde_ID, Navn=NEW.Navn, Adresse = NEW.Adresse, Postnr = NEW.Postnr, Bynavn = NEW.
```

Her er der et eksempel på create trigger hvor der bliver benyttet sig af alle de 3 hændelser “delete”, “update” og “insert” disse hændelser vil blive benyttet sig af i applikationen.

Ud fra databasen blev der lavet som beskrevet i opgaven en ER-model af vores database. Den blev lavet efter databasen fordi at gruppen bedømte at der var stort set brug for den relationelle model til at lave databasen. Dog bruges denne ER-model til at få et lettere overblik over sammenhængen over databasen og hvordan den vil kunne hænge sammen med applikationen.



Her vises der hvordan sammensætningen på vores ER-model ser ud. Den er stort set ikke brugt til lave databasen, men er dog nyttig for at danne sig en hurtig oversigt over strukturen for både for læseren men også for gruppen.

Efter dette kan man begynde med at lave applikationen til siden. som vil vises på den næste menu.