DBConnection klassen

Klassen DBConnection har ansvar for at håndtere forbindelsen til databasen. På den måde kan vi ved programmets start oprette et database- objekt, og dermed oprette forbindelse til databasen. Det betyder at hver gang vi skal hente eller indsætte noget i databasen, så kan vi bruge objektets metoder.

#### connect()

Metoden opretter forbindelse til de parametre som er medtaget fra constructoren (String db\_host, String db\_port, String db\_name, String db\_user, String db\_pw) og returnerer true eller false.

getData(String getdata)  
Medtoden bruger vi når vi skal hente data fra databasen. Man kalder funktionen med en sql-streng som så returnerer resultatet i et ResultSet.

#### setData(String getdata)

Metoden bruger vi når vi skal indsætte data i databasen. Man kalder funktionen ved at give den en sql-streng, der så skal bruges til indsættelsen.

DatabaseObjectHandler klassen

Klassen DatabaseObjectHandler håndterer de SQL-queries der laves i programmet. Den har forskellige metoder som kan hente modelobjekter ud af databasen, indsætte dem i databasen, redigere og slette dem.

Klassen er en overbygning af DBConnection klassen. Vi har lavet klassen så vi har alle metoder til at hente de førnævnte objekter ud af databasen i modsætning til at lave dem i hvert enkelt klasse hvor vi skal hive dem ud. Vi har valgt at dele SQL-queries op i lag, altså getKunde kalder getPostnummer, i stedet for at lave udtræk af flere tabeller. Det kunne vi gøre anderledes næste gang vi laver et projekt da de fleste get Metoder så kun ville kræve en SQL-streng/metode i modsætning til flere lag.

public ArrayList<Vare> getFiltreretVareListe(int grp\_nr, int minHøjde, int maxHøjde, int minBredde, int maxBredde, double minPris, double maxPris, int vareStatus)  
Metoden henter en ArrayList af Vare objekter ud af databasen, som er filtreret ud fra de parametre som constructoren er kaldt med. Der opbygges en SQL streng ud fra parametrerne ved at den tjekker om de forskelling parametre er 0, og hvis de ikke er lægges det paramter til SQL-strengen. De metoder der er beskrevet i denne klasse er udvalgt fordi de beskriver de forskellige slags metoder vi har, da vi har mange gengangere til de forskellige objekter.

***public String createOrdre(Ordre ordre)***Metoden bruges til at oprette en ordre i databasen ud fra det Ordre objekt som constructoren bliver kaldt med. Metoden starter med at kalde createKunde metoden med Ordre objektets Kunde objekt som derefter vil oprettet en kunde hvis den ikke findes, eller redigere den hvis den findes i forvejen. Derefter tjekker metoden om en en ordre med det valgte ordre nummer findes i forvejen, hvis det gør det smider den en fejl og stopper metode, hvis den ikke gør går den videre. Derefter tjekker metoden om der er et Kirkegård objekt i Ordre objektet, hvis der opretter den ordren i databasen med et kirkegård id, hvis der ikke er opretter den det uden. Til sidst tager metoden Vare\_Linje ArrayList’en fra Ordre objektet og opretter dem enkeltvist med Ordre objektets ordre\_nr og et linje nr, samt detaljerne om varelinjerne.

#### public int getMaxVareNr()

Metoden bruger vi til at finde det højeste varenummer der findes i databasen, som den så returnerer som en integer. Vi bruger den til at finde den sidste vare vi har oprettet så vi kan knytte den på andre objekter, i stedet for at søge på navn el.lign.

#### public void updateVareStatus(Vare vare)

Metoden bruger vi til at opdatere varestatus på en vare når vi opretter et ordre objekt, så databasen registrerer at varen ikke er på lager mere, eller omvendt hvis det er en annullering af en ordre. Vi bruger en update som finder varenummeret og sætter varestatus for den vare til den nye varestatus som er valgt.

#### public String getNextOrdreNr()

Den finder det det højeste ordre nummer i databasen og laver det om til en integer så vi kan lægge 1 til hvorefter vi sætter nuller på igen for at gøre det til det næste ordre nummer. Vi valgte at gøre ordrenummer til en String fordi at vi skulle have de ekstra nuller på, men efterfølgende har vi kunne se at det ikke var nødvendigt da vi bare kunne have sat nullerne på i GUI/PDF’er når vi skulle vise dem.

#### public void deleteOrdre(Ordre ordre)

Metoden bruger vi til at slette ordre i databasen. Vi bruger en delete til at slette alle entries der passer til det ordre nummer der er i ordre objektet.