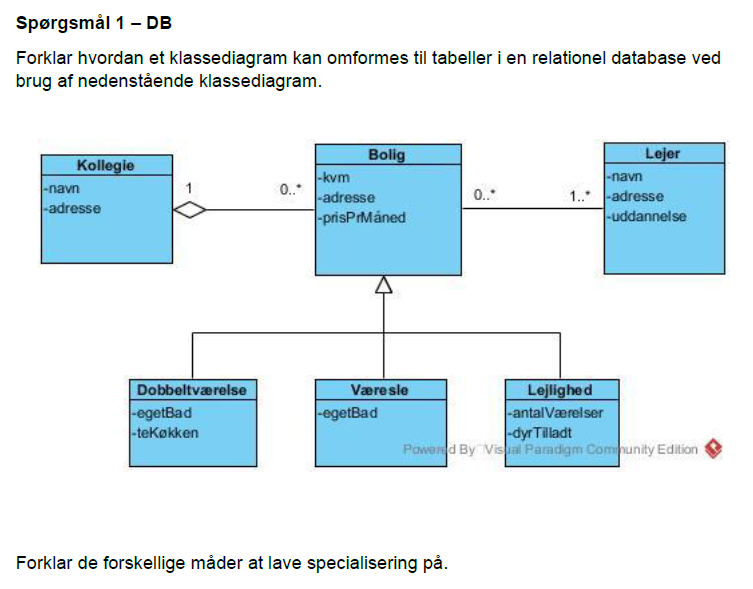
Spørgsmål 1



Det første du gør når du vil omforme et klassediagram til tabeller er at kigger på associationerne mellem klasserne.

Der er 3 typer associationer:

* 1-N. Hvilket er en til mange association.
* N-N. Hvilket er mange til mange association.
* 1-1. En til en, her ligger du bare de 2 klasser sammen og laver en tabel. Ellers laver du den som en en til mange association.

I en til mange association, tager man primary key fra 1 siden og sætter ind på mange siden som en foreign key.

I en mange til mange association, laver man en binde tabel, oftest ved at inkorporer begge navne ind i tabelnavnet. Denne tabel tager primary key fra begge mange til mange tabelerne og sætter ind som foreign key.

Dernæst tilføjer man alle de attributter de forskellige klasser har ind, og enten skaber en ID som primary key eller bruger en af attributterne som primary key.

Et eksempel på ovenstående klasse diagram ville være:

Kollegie har:

navn som primary key.  
addresse som attribut

Lejer har:  
navn som primary key

addresse som attribut

uddannelse som attribut.

Bolig har:

ID som primary key

kvm som attribut

prisprmåned som attribut

lejerNavn som foreign key

kollegieNavn som foreign key.

til ville være det basiske man skulle gøre.

I dette tilfælde har vi også nedarvninger.

Der er 3 måder at gøre det på.

Vi kan lave en tabel der indeholder superklassen og alle subklassernes attributter. Dette er oftest den letteste metode, men der kommer til at være meget spildplads i tabellen.

Vi kan lave 3 tabeller, en for hver subklasse der indeholder dens attributter og super klassens attributter. Dette er ofte den løsning man vil køre. Da der ikke er meget spild plads og ikke helt nær så mange klasser som løsning 3.  
  
Løsning 3 er at du lave en super klasse og alle subklasserne. Hvor du placer en attribut i subklassen der angiver hvorfor et object der vedrøreres i superklassen. Personligt kan jeg ikke lide denne løsning og vil ikke bruge den medmindre jeg er påkrævet da den er ofte forvirrende og kan let give fejl. Plus det er den der tilføjer flest tabeller til din database.

I henhold til aggregering, så omlægges de som almindelige associationer i ms sql.