**Normalformer**

Når man snakker om databaser findes der en række normalformer. En database kan være på 1., 2. eller 3. normalform, men hensyn til relationer og opbygning af tabeller. Når man laver en database vil man helst have at den skal være på 3. normalform, da det er det ”flotteste” og man vil nemt kunne undgå at gentage data. For at opnå 3. normalform skal 1. og 2. også være opfyldt. 1. Normalform har to krav, for det første skal hver tabel have en primær nøgle, da man ellers ikke kan slå op i en tabel uden. Andet krav er at alle attributter er en-værdi attributter, man må ikke have nogle sammensatte attributter da det bare gør det sværere at få fat i dele af attributter eller ændre i den. Hvis de krav er opfyldt har man som min sin database på 1. normalform. Ud over de 2 krav er der også snak om at ingen attributter må have værdien NULL, men da der er meget uenighed om emnet kalder man det for stærk 1.normalform, hvis der ingen attributter er som har værdien NULL. 2. normal gælder hvis databasen er i 1. normalform og derudover også gælder at der enten ikke er nogen sattesatte kandidatnøgler(som er en nøgle bestående af to eller flere attributter), eller at der er flere sammesatte kandidatnøgler men at de attributter der afhænger af kandidatnøglen afhænger af hele den sammensatte nøgle og ikke kun noget af den. 3 normal har kun, udover sevlfølgelig at være på 2. normalform, kun et krav og det er at ingen attributter indirekte afhænger af en kandidatnøgle, det kan dog løse ved at splitte tabellen op i 2, der begge er i 3. normalform.

**E/R diagrammer**

Når man vil opstille sin database på en simpel og overskuelig måde og så man kan se relationerne, så kan man bruge en E/R diagram (Entity/Relation)

**Databaser**