# Objektorientering VS Relationelle databaser

Data i relationelle databaser ligner meget den opbygning der findes i Objektorienteret og det betyder at hvis vi sammenligner dem step for step, vil vi se at det nærmest blot er navne der ændre sig:

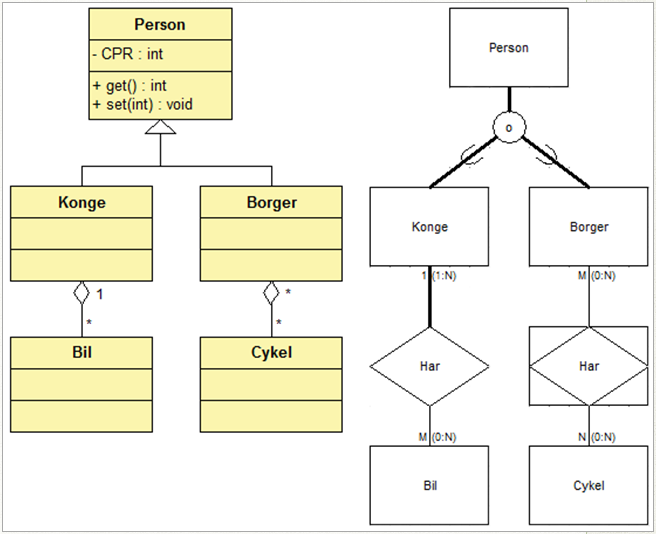
|  |  |
| --- | --- |
| **ObjektOrientering** | **Relationel Database** |
| Klasser | Entiteter |
| Data i attribut i objekter | Data i attribut i tabeller |
| Ændre data via metoder: get(), set() | Ændre data via DML(data Manipulation Language) INSERT,UPDATE,DELETE,SELECT |
| Relationer dannes mellem klasser med pointere og referancer | Relationer mellem entiteter dannes på baggrund af Keys |
| Beskrives med et Klassediagram | Beskrives med et ER Diagram |

Begge former tilbyder høj sikkerhed og et kraftigt query language.

Objektorienteret blev lavet for at skabe en database der var tættere på det system der havde brug for at opbevarer data. Det betyder at vi i objektorienteret kan gøre brug af

* Brugerdefineret data typer
* Nested objekter
* Containers, f.eks. sets, lists og bags

Det betyder også at der er stærk sammenhæng mellem grænsefladen til databasen fra ens applikation. Af samme grund tilbyder Objektorienteret ikke den matematiske fordele når vi skal lave operationer på dem.



Figur : forsekelle på diagrammerne

Den relationelle model gemmer data i et sæt af tables det består af rækker og kolonner. Hver kolonne repræsenterer en property og hver række repræsenterer en entitet.

Vi kan mappe et objekt til den relationelle form, hvis det ønskes.