



The Force presenterar

YrkesCo skoldatabas

En datamodellering för YH



Presenteras av Thorbjörn P Steive

The Force består av Efrem Daniel, Hampus Donnersten, Andreas Reinholdsson, Henrik Sjögren samt Thorbjörn P Steive



Kravspecifikation

Affärsregler

För var tredje år ett program blir beviljat måste det innehålla samma kurser

Adresser är kopplade till personnummer, adress plockas från folkbokföring

Man kan bara gå en klass åt gången i vår skola

Man kan bara ha en yrkesroll

Om en lärare vill börja studera så får hen göra det i en annan skola

En enda UL håller i alla fristående kurser per skola

Skolan är ny, således bara studenter i 24-klasserna

YrkesCo har identifierat att de vill ha en databas som har följande information

- om studenter, förnamn, efternamn, personnummer, email
- utbildare kan vara konsulter
- de planerar att anställa fasta utbildare
- utbildningsledare och deras personuppgifter
- utbildningsledare har hand om 3 klasser
- kurser med namn, kurskod, antal poäng, kort beskrivning av kursen
- program har ett antal kurser knutna till sig
- ett program blir beviljat i tre omgångar, dvs att det finns 3 klasser
- det finns även fristående kurser
- konsulter, deras företag, företagsinfo som organisationsnummer, har F-skatt, address, hur mycket de tar i arvode per timme
- YrkesCo har två anläggningar, en i göteborg och en i stockholm, i framtiden kanske de kommer expandera till flera orter

Konceptuell modell

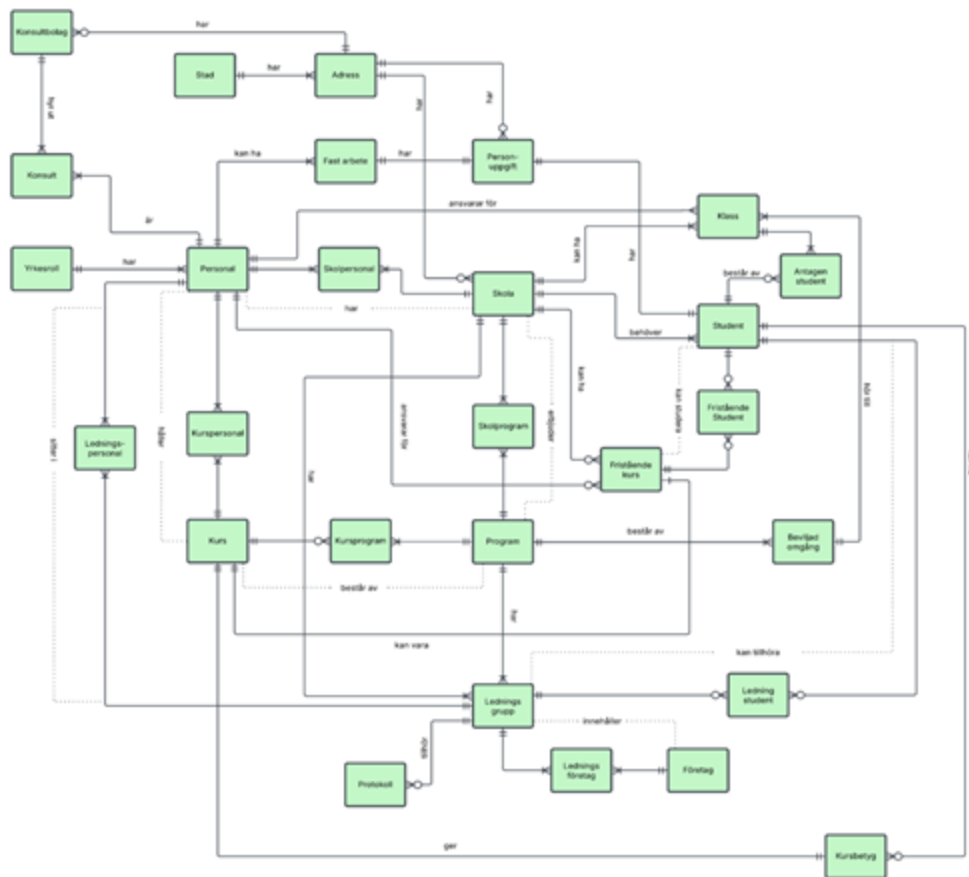
Fokus är på entitet och relation

Kardinalitet är förhållandet mellan entiter

Exempel kan vara förhållandet mellan skola och student

-> En Skola kan ha en eller flera studenter

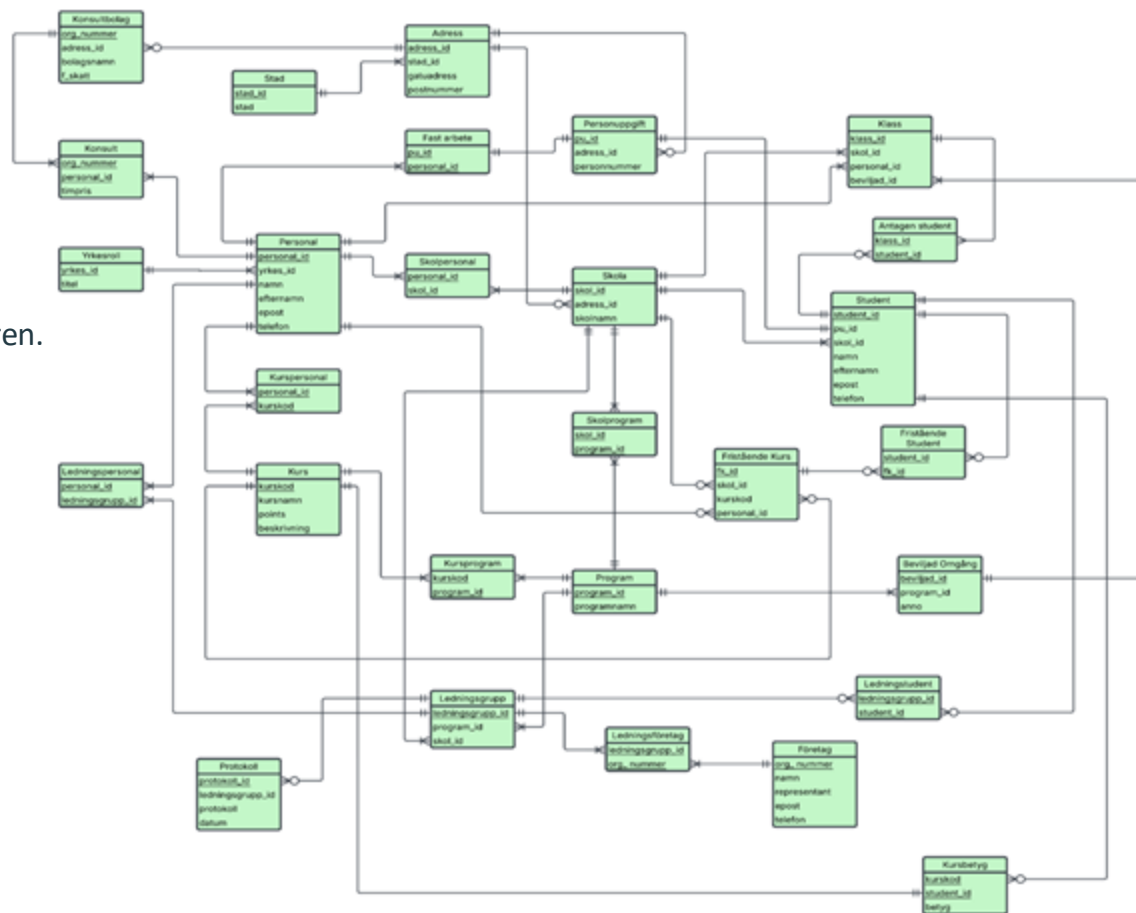
<- En student tillhör en och endast en skola



Logisk modell

Bygger vidare på den konceptuella modellen.

Vi lägger till attribut, primary key, foreign key och normaliserar strukturen.



Normalisering

Är processen att bryta ner databasen från stora tabeller och organisera data i olika normalformer för att minska dataredundans och förbättra dataintegriteten.

Första normalformen (1NF)

(1NF) innebär att radordningen inte spelar någon roll, varje tabell har en primary key, det inte finns några upprepande grupper av kolumner i en rad, och all data i en kolumn är av samma datatyp.

Andra normalformen (2NF)

(2NF) kräver att tabellen är i 1NF och att alla icke-primära attribut är funktionellt beroende av hela primärnyckeln och inte bara en del av den (gäller speciellt vid sammansatta primärnycklar).

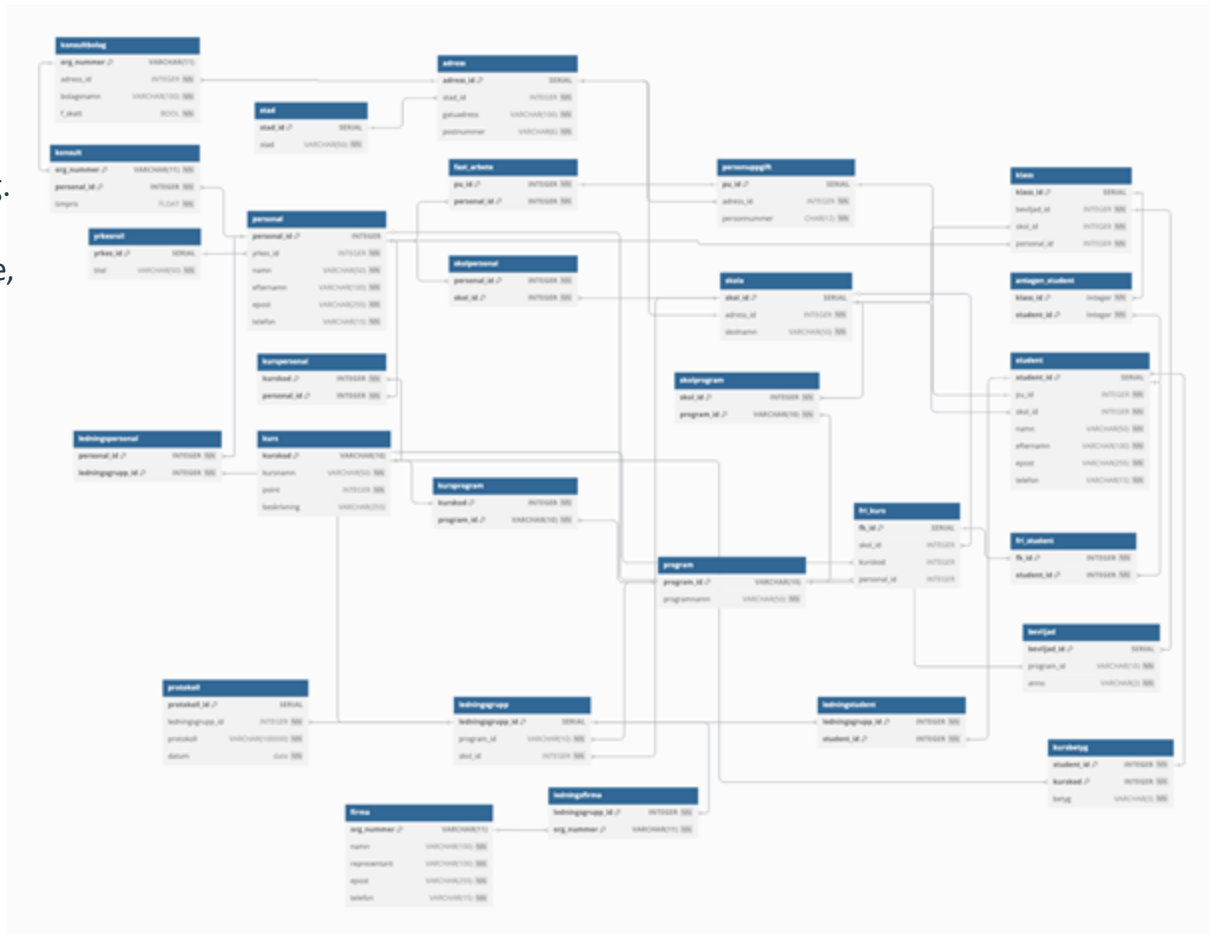
Tredje normalformen (3NF)

(3NF) kräver att tabellen är i 2NF och att inga icke-primära attribut har transitivt beroende till primärnyckeln (dvs., ett icke-primärt attribut beror på ett annat icke-primärt attribut som i sin tur beror på primärnyckeln).

Fysisk modell

Definierar hur data lagras i en specifikt databas och är klar för implementering.

Det innebär att vi väljer databas engine, definierar datatyper och constraints.



Implementation

Med hjälp av dbdiagram kunde vi snabbt göra en tabell implementation av tabeller i databasen

```
New YikesCo ERD Personal < Share < History < Import <

1 Table stad{
2   stad_id SERIAL [primary key]
3   stad VARCHAR(50) [not null]
4 }
5
6 Table adress{
7   adress_id SERIAL [primary key]
8   stad_id INTEGER [not null, refs > stad.stad_id]
9   gatadress VARCHAR(100) [not null]
10  postnummer VARCHAR(6) [not null]
11 }
12
13 Table skola{
14   skol_id SERIAL [primary key]
15   adress_id INTEGER [not null, refs = adress.adress_id]
16   skolnamn VARCHAR(50) [not null]
17 }
18
19 Table personuppgift{
20   pu_id SERIAL [primary key]
21   adress_id INTEGER [not null, refs > adress.adress_id]
22   personnummer CHAR(12) [not null]
23 }
24
25 Table program{
26   program_id VARCHAR(10) [primary key]
27   programnamn VARCHAR(50) [not null]
28 }
29
30 Table skolprogram{
31   skol_id INTEGER [not null, refs > skola.skol_id]
32   program_id VARCHAR(10) [not null, refs > program.program_id]
33   INDEXES ((skol_id, program_id) (pk))
34 }
35
```



Vi har valt att använda oss av PostgreSQL Databas i en Dockermiljö

Anslutningen görs via Visual Studio Code

Nu live demo!



Tack för mig!