

# YrkesCo

---

Databasen som möjliggör



# Data Modelleringprocessen

- prata med och förstå stakeholders
- skriva ner och ändra krav-specifikationer

- Definera Entiteter , Attribut och skriva ner relationer mellan dessa

- Grafiskt ta fram Conceptual model med relationsetiketter
- Gå tillbaka till tidigare steg (iterativ process)

- Skapa Logiska modellen
- Gå tillbaka till tidigare steg (iterativ process)

- Fysisk modell
- Gå tillbaka till tidigare steg (iterativ process)

- SQL - kod
- Implementera i postgresql

# Kravspecifikationer

Studenter (förnamn, efternamn, personnummer, email)

En utbildare kan vara fast anställd eller konsult

En konsult är kopplad till exakt en utbildare

Utbildningsledare (personuppgifter)

En utbildningsledare har hand om 3 klasser och 2 program

En student måste ingå i en kurs eller ett program

En utbildare kan ha en eller flera kurser

Kurser - (namn, kurskod, kurspoäng, kort beskrivning)

En kurs kan vara fristående eller ingå i ett program

Ett program har 3 olika startår / klasser

Konsulter - (företag, företagsinfo, organisationsnummer, F-skatt, adress och arvode/h)

YrkesCo har 2 anläggningar (Gbg och Sthlm)

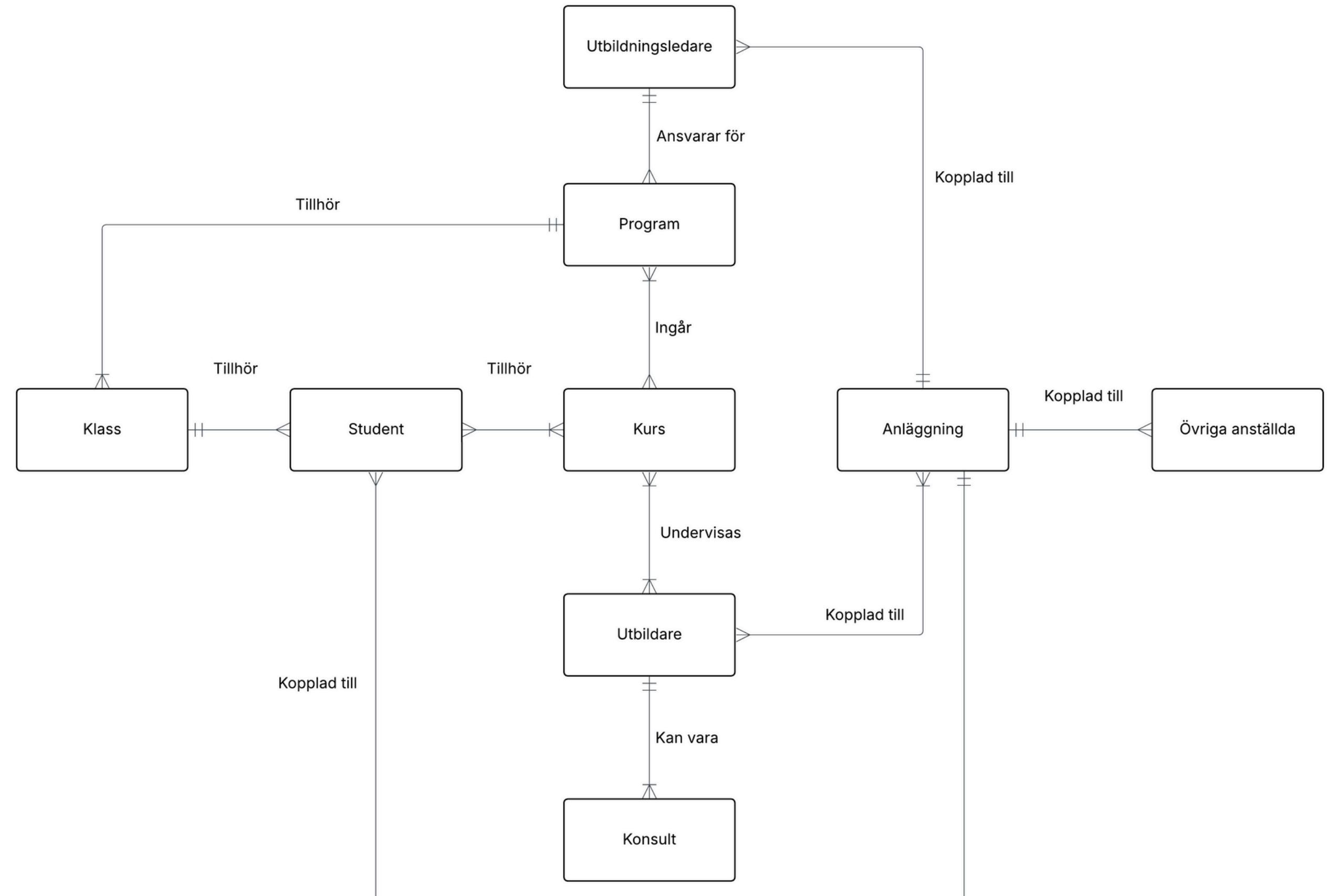
Känsliga uppgifter / Personuppgifter är inte synligt för alla i organisationen

Övriga anställda - här ingår övriga roller på YrkesCo - (Admin, VD, utbildningschef, teamleader, ekonomiansvarig, ledningsgrupper, kommunikationsansvariga)

# Steg 1

## Konceptuell modell (med relations-etiketter)

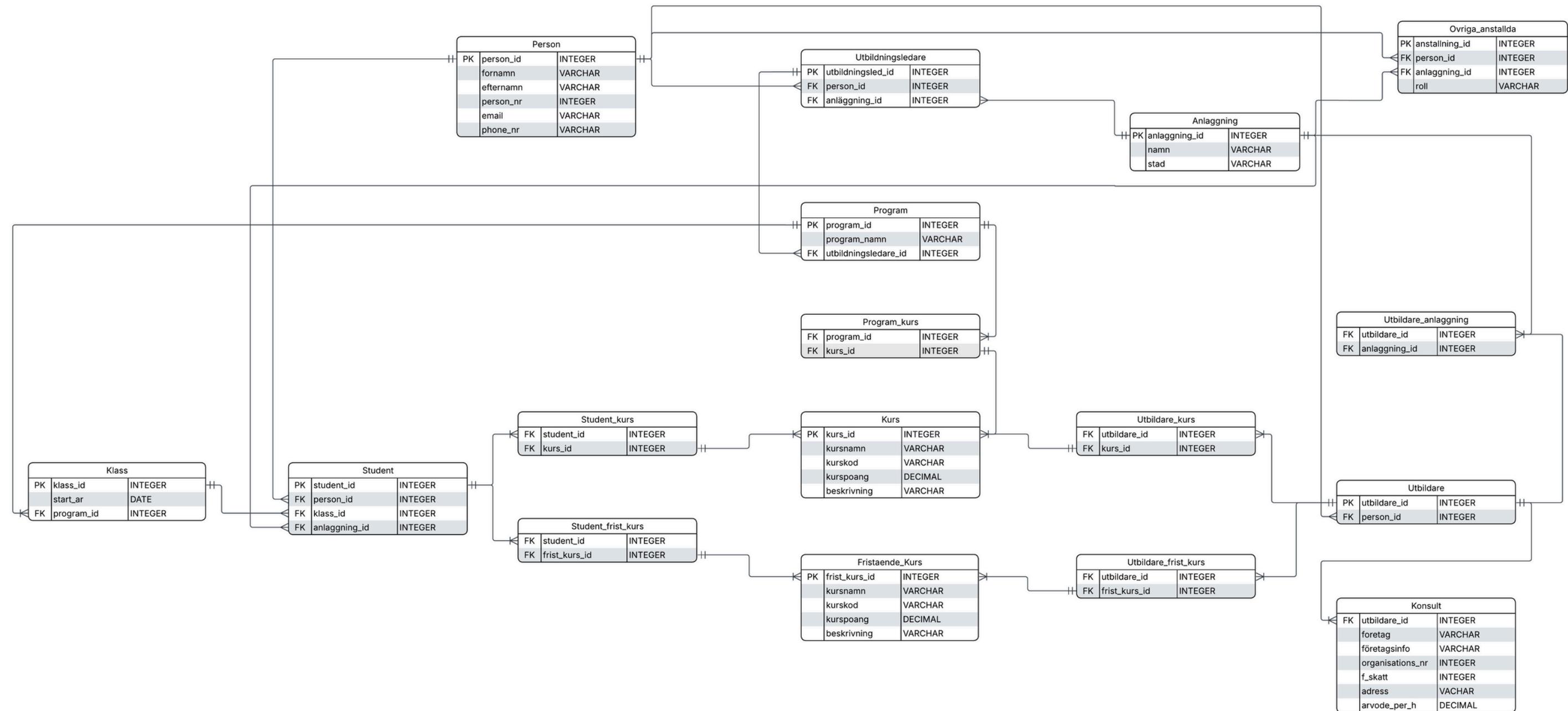
- Entiteter
- Crow foot notation (relationer)
- relationsetiketter



# Steg 2

## Logisk modell

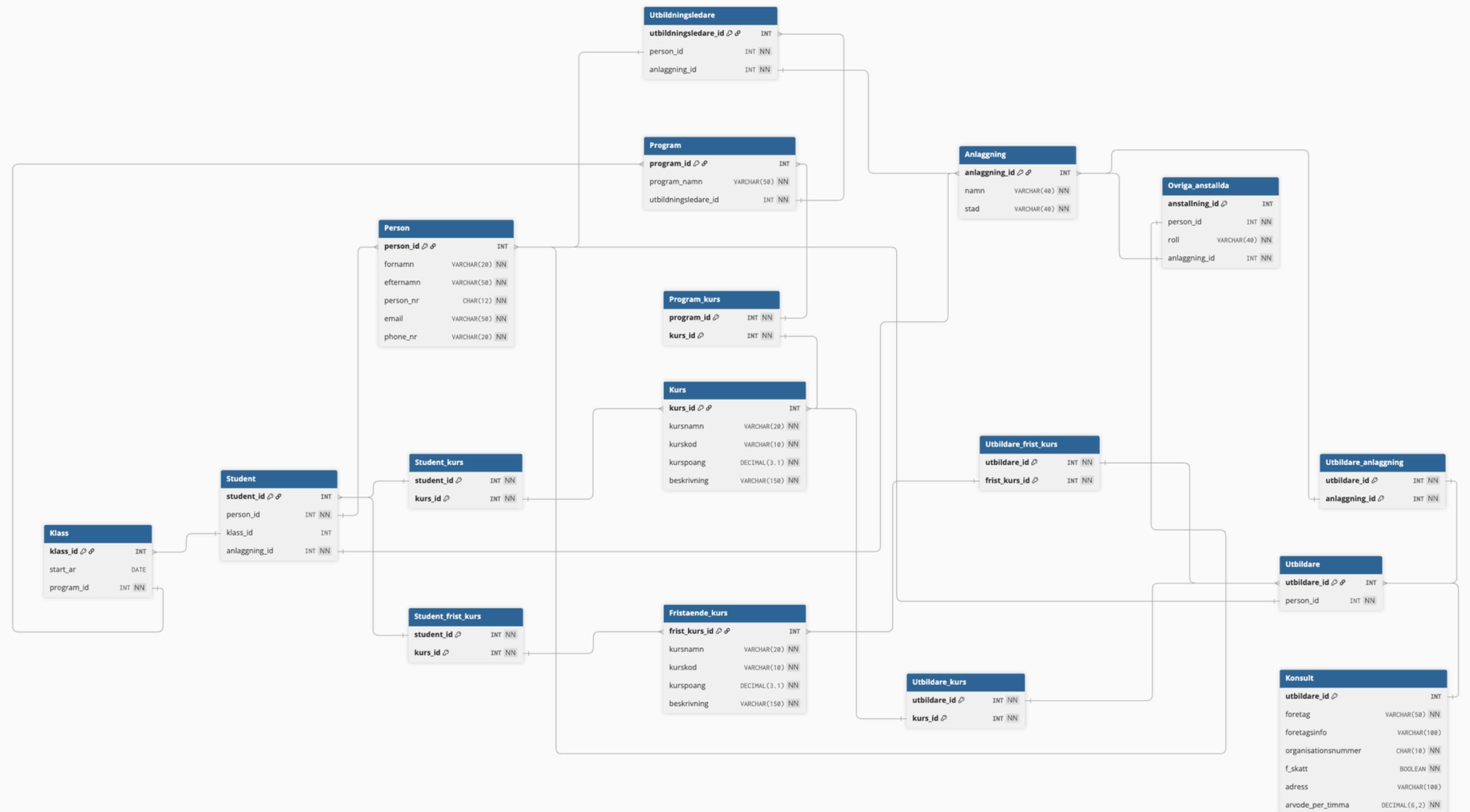
- Entiteter
- Attribut
- Nycklar
- Eliminerar Many to many
- Datatyp



# Steg 3

## Fysisk modell

- Referenser mellan Tables genom nyckelattribut
- Not null, datalängd



# Varför är modellen bra?



## 1NF

I varje table finns det Primary Keys

Datatyperna är en och samma i varje rad

Varje kolumn har ett unikt namn

Tabellerna innehåller endast atomära värden (ett värde per rad)



## 2NF

### 1NF

Attributen är kopplade och funktionellt beroende av PK



## 3NF

### 2NF

Attributen är kopplade och funktionellt beroende av PK

Icke-primära-attribut är inte beroende av andra icke-primära-attribut

# Användning

```
school_db=# \i sql/sql_queries.sql
          utbildare_namn | utbildare_id | kursnamn | kurs_typ
-----+-----+-----+-----+
Anna Svensson      | 1 | Matematik | Programkurs
Björn Lind          | 2 | Programmering | Programkurs
Caroline Ek          | 3 | Databaser | Programkurs
David Johansson     | 4 | Matematik | Programkurs
Elin Karlsson        | 5 | Programmering | Programkurs
Anna Svensson      | 1 | Kommunikation | Fristående Kurs
Björn Lind          | 2 | Projektledning | Fristående Kurs
Caroline Ek          | 3 | Ledarskap | Fristående Kurs
David Johansson     | 4 | Kommunikation | Fristående Kurs
Elin Karlsson        | 5 | Projektledning | Fristående Kurs
(10 rows)

          program_namn |      ul_namn
-----+-----+
Program A           | Fredrik Andersson
Program B           | Gustav Nilsson
Program C           | Fredrik Andersson
(3 rows)

          anläggning | antal_studenter | antal_utbildare | antal_kurser
-----+-----+-----+-----+
YrkesCo STHLM       | 4 | 2 | 4
YrkesCo GBG          | 3 | 3 | 4
(2 rows)

school_db=# 
```

Lätt att få fram information genom  
att JOIN olika Tables

Lätt att kunna se exempelvis  
anläggning , Utbildare och hur  
många kurser ,studenter denna har

# Utvecklingsmöjligheter

