

Modulopgave 1: Database design

Hold: Dat18C

Undervisere: Jarl Tuxen, Asger B. Clausen.

Uddannelsesinstitution: KEA, Københavns erhvervsakademi

Gruppe 6: Ali Al-Sharefi, Andreas Dan Petersen, Désirée Schüller, Daniel Blom, Thias Meyer Petersen.

Afleveringsdato: 01/03/2019

Antal sider: 7

Indholdsfortegnelse

Modulopgave 1: Database design	1
Indholdsfortegnelse	2
Indledning	3
Opgavens krav	3
Domæne model	3
Entity relation diagram	4
Primary keys og Foreign keys	5
Logical Architecture & Layers	6
Konklusion	7

Indledning

Til opgave har vi fået stillet at lave en database som tager udgangspunkt i at strukturere en oversigt over fraflytninger fra byen København samt Frederiksberg, og sætte data sammen i henhold til bevægelse i diverse byer.

Vi har i vores opgave fået udleveret 3 csv filer som er udarbejdet af Danmarks Statistik, hvor vi har fået til opgave at planlægge databasens struktur og design.

Opgavens krav

Opgaven skal indeholde følgende:

Domæne model

Entity relation diagram

Backend

Frontend

Programmet skal kunne:

- Søgning baseret på kriterie, der er indtastet af brugeren.
- Vis en liste over indhold fra en tabel baseret på et kriterium.
- Vis en liste som har en kombination af flere tabeller (INNER JOIN).
- Hvis tiden tillader søgning med mindre end, større end kriterier.

Domæne model

Efter vi har kigget på domæne modellen kan vi konkludere at der skal oprettes nye tabeller. Årsagen til vi valgte at lave ny tabeller, er på baggrund at man så kan søge på flere kriteriere.

Vi har derfor oprettet følgende:

migration(Alt pågældende data)

agespan (ageTo, ageFrom)

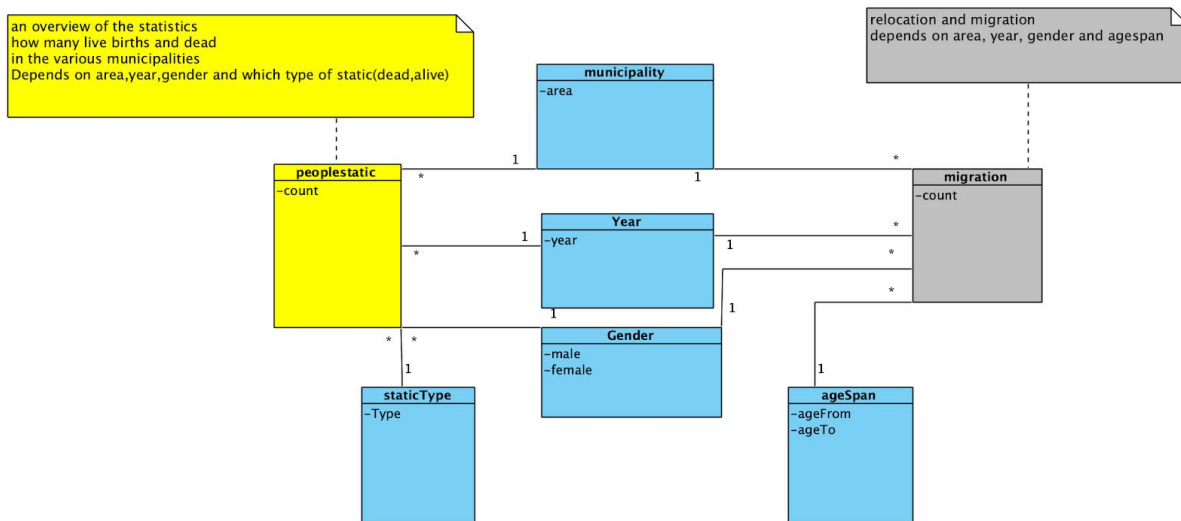
municipality(Kommuner)

year(Hvilket år)

gender (Specifikt køn, mand eller kvinde)

people Static(Alt pågældende data)

static Type(fraflyttede,døde osv)



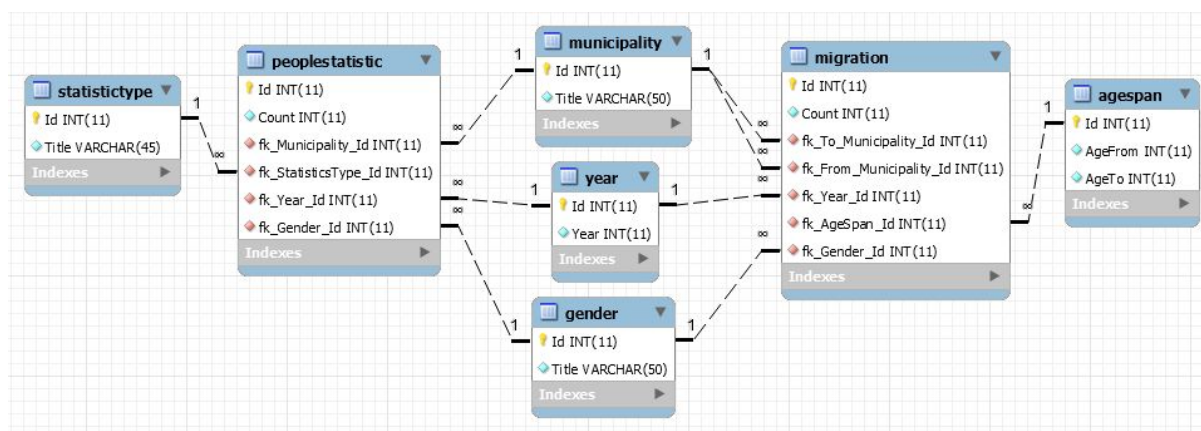
Entity relation diagram

I vores ER diagram (entity relation) har vi valgt at give vores tabeller et specificeret ID samt primary keys og foreign keys. Som kan anvendes i at f.eks. at finde antallet fraflytninger fra København og Frederiksberg.

Tanken i vores ERD er at man ud fra Municipality, Year og Gender kan få sig et overblik i Migration og Peoplestatic.

I Peoplestatic har man mulighed for at specificere sin søgning i levendefødte osv.

Nedenstående en oversigt over vores ERD;



Primary keys og Foreign keys

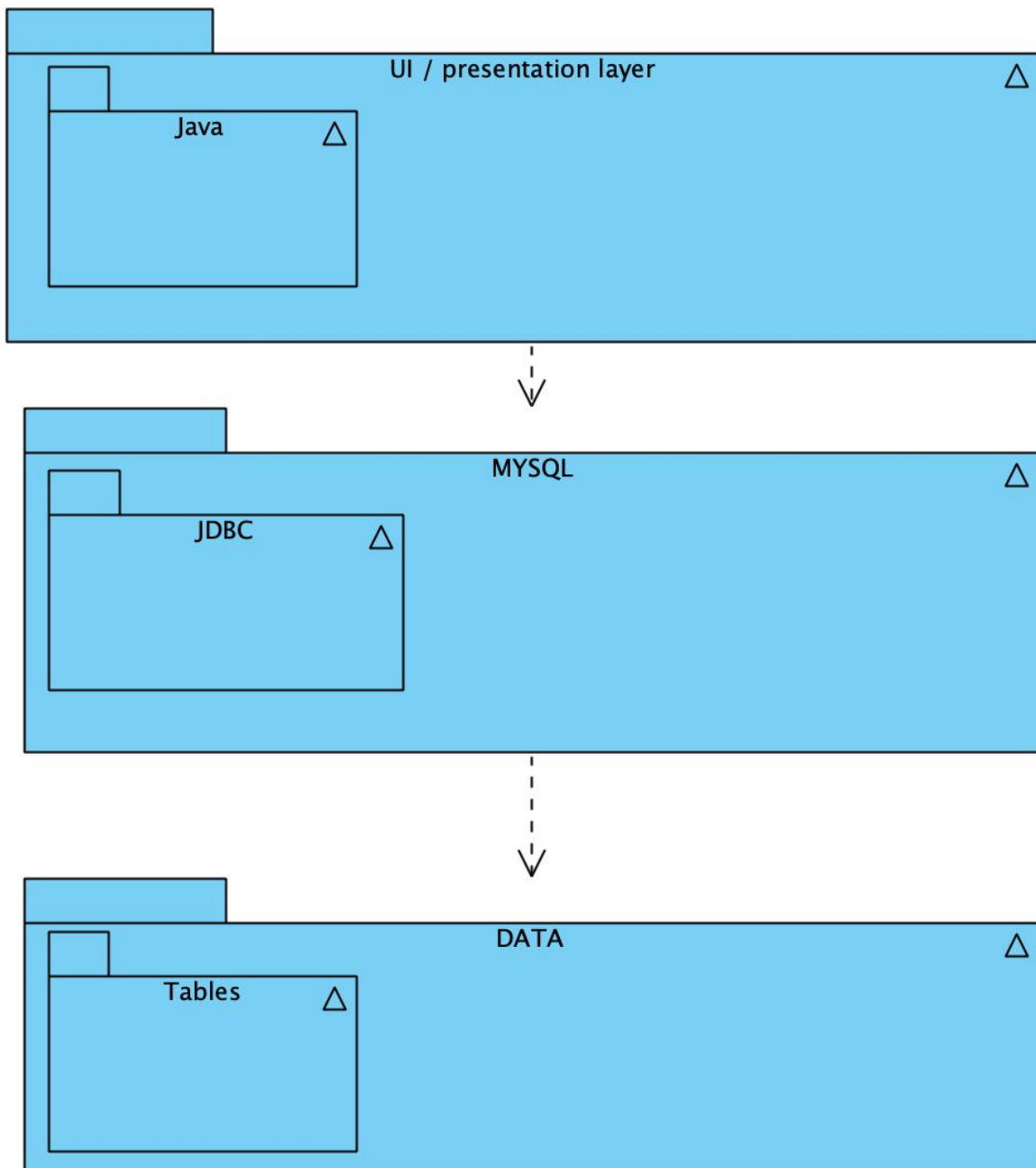
1NF: Vi har valgt at lægge year,gender og municipality i hver sin tabel

2NF: Vi har kun en enkelt primær nøgle(PK) i hver tabel

3NF: Alle tabeller har fået deres helt egen unikke nøgle

Alle primary keys, hedder id i hver sin tabel

Logical Architecture & Layers



Dette diagram viser hvordan vores programmer arbejder sammen.

Vi bruger Java som vores primære måde at køre programmet, hvor vi anvender JDBC til at connecte til mysql hvor alle vores tabeller og databaser ligger.

Konklusion

Vi kan konkludere at de tabeller vi fik udleveret var afhængig af hinanden.

Undervejs i processen fik vi formået at skabe et domæne model, hvor der ingen afhængigheder var.

Vi har skillet tabellerne ad og tildele dem unikke ID, samt gøre tabellerne unikke.

Tabellerne er skræddersyet således at man kan lave ændringer/tilføjsere at skal indflydelse på de andre tabeller. Slutproduktet lyder således på at man som bruger har muligheden for at søge på ét eller flere forskellige kriteriere, som viser listen af indhold af flere tabeller.