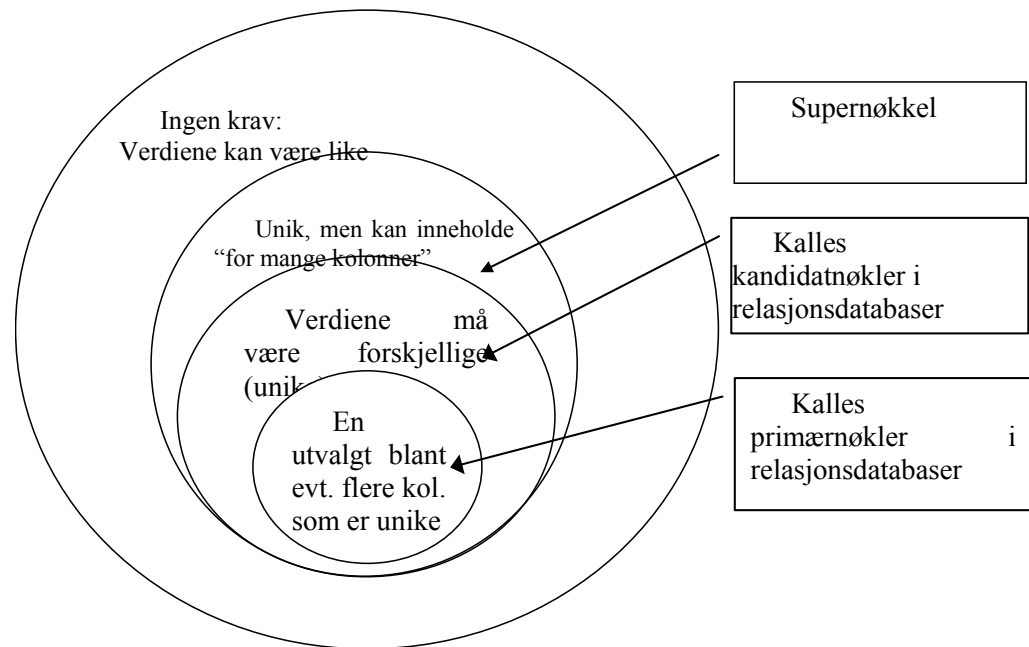


Skranker i databaser, hvilke er
viktige?

Vanlige skranker i databaser

- Ulike typer nøkler



I SQL i praksis: PRIMARY KEY og UNIQUE

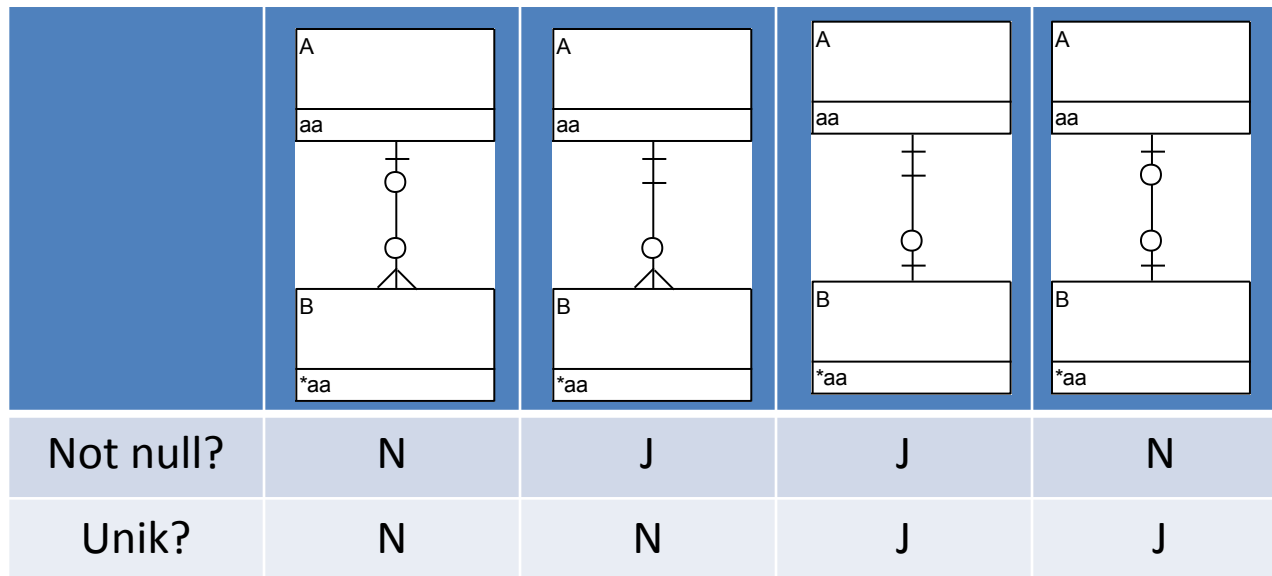
- Verdinødvendighet

I SQL i praksis: NOT NULL eller ikke

- Hvilken (evt. hvilke) av disse er viktige? (diskuter!)

Sammenhenger med datamodeller og realisering med fremmednøkler

- Vanlige strukturer innen datamodellering. Hva er egenskapen for FK?



- Konklusjon 1: Kombinasjonen av de 4 mulighetene av **Verdinødvendighet (J/N)** og **Unikhet (J/N)** gir de vanlige strukturene for datamodeller og det er en isometri mellom disse og datamodellstrukturene.
- Konklusjon 2: de viktigste egenskapene er **Verdinødvendighet** og **Unikhet**, ikke PK.

Unikhetsbegrepet

Unikhetsbegrepet er uklart, kan bety

- UNN (unik og NOT NULL), «sterk unikhet»
- NoU (NULL eller Unik), kan altså tillate mange NULL på unik (men: er det da unikt???), «svak unikhet».
- Unik, med et begrenset antall NULL, slik at entydighet fremdeles beholdes, «minimalkravs unikhet».
 - hvis enkel unik: en verdi kan være NULL
 - hvis unik kombinasjon av to: hvis vi har n rader, kan vi tillate $2n-1$ NULLer.
 - osv.

Hva er de viktigste begrepene?

- Min påstand: de viktigste er
 - NoU
 - NN (not null),
blant annet fordi disse er uavhengige (ortogonale),
og spenner ut de andre.
- $\text{NoU} \cap \text{NN} \rightarrow \text{UNN}$
- $\text{NoU} \cup \text{NN} \rightarrow \text{ingen krav}$

Trengs vi primærnøkkel-begrepet?

- I prinsippet ikke, en FK kan referere til andre «sterke unike» enn PK.
- Likevel praktisk, vi velger ut en blant de flere mulige kandidatnøkler (som må være sterke unike), slik at den ene «reperesenterer» tabellen, m.a.o. den vi kaller primærnøkkel.

Trenger vi fremmednøkler?

Alternativer:

- liste opp verdier fra mange-siden på 1-siden (Avdelingens ansatte: 1,4,7,3,..), men det blir ikke-atomisk, som ikke er tillatt i relasjonsdatabaser.
- En pekerkjede som peker på en og en verdi på mange-siden, men pekere er ikke tillatt i relasjonsdatabaser.

Konklusjon: Fremmednøkler er måten å implementere 1:m (og 1:1) på i relasjonsdatabaser. De implementerer også referanseintegritet.

Konklusjon: hva er viktig?