

Onsdag 7 Mars 2018



Databasteknik

20 YHP

Vad är en databas?



- En **databas** är en samling av information som hör ihop, dom finns nästan överallt i många olika former
- **Database management system** är ett mjukvarusystem som hanterar datan i en databas
- Oftast när vi inom systemutveckling använder *databas* så menar vi ett **databassystem**, det vill säga både datan och management systemet.

Komponenter i ett DBMS:



- **Procedurer**

- Definierar användningen av databasen

- **Vyer**

- Resultat av procedurer
 - oftast en delmängd av data i databasen

- **Schema**

- En beskrivning av databasens struktur
 - Man pratar ofta om hela databasens schema, men beroende på databasen kan det även vara relevant att prata om flera **scheman** i databasen.

Vad är en filbaserad databas?



- Enklaste formen av en databas är ett **filbaserat system**
- I nästan alla programspråk kan man skapa, redigera och ta bort filer.

Vad är en filbaserad databas?



- **Fördelar:**

- Enkelt att skapa och använda
- Snabbt vid enkla datamängder
- Stora mängder information kan sparas och processeras

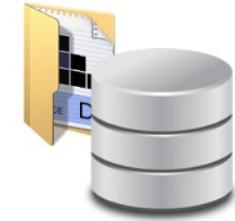
Vad är en filbaserad databas?



- **Nackdelar:**

- Svårt att skala
- Svårt att se kopplingar mellan information
- Applikationer blir DBMS, men utan tydliga Procedurer eller Vyer

Vad är en filbaserad databas?



- **Fördelar med ett "riktigt" databassystem:**
 - Kan bytas mellan olika applikationer.
Samma procedurer och samma vyer.
 - Tydligare relationer mellan data
 - Enklare att skala mellan små och stora program

Vad är en filbaserad databas?



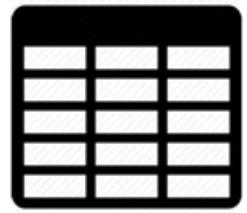
- **Nackdelar med ett "riktigt" databassystem:**
 - Högre komplexitet, kräver mer kunskap
 - Kan ge sämre prestanda om databasen dåligt utformad
 - Högre prestandakrav på hårdvaran

Hur designar man en databas?



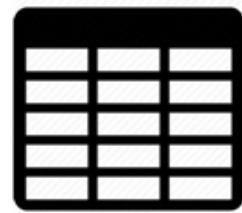
- **Viktigaste:** Man måste separera på applikationen (verb) och datan (substantiv). Data först, sen användningsområdet.
- Vi kommer gå igenom diagram i en framtida version och prata om hur man designar en databas steg för steg

Vad är en relationsdatabas?



- Helt klart den vanligaste typen av databaser, exempelvis MSSQL och MySQL.
- Grundar sig på en akademisk modell från 1970
- Lägger stort fokus på relationerna mellan datainlägg, och sparar dessa relationer som **data** istället för **metadata**
- Vad är en relation? 🤔

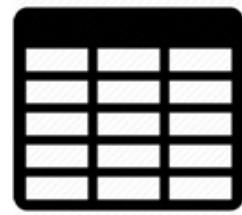
Tabeller i Relationsdatabas



- En relationsdatabas består av tabeller, med kolumner och rader.
- Tabeller ska namnges så här:

StorBokstavFörstOchInteNågraMellanrumEllerSymboler

Tabeller i Relationsdatabas



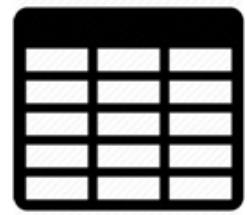
- Varje tabells namn bör beskriva innehållets relation till resten av databasen/applikationen:

"User", "Car", "RentableApartment", "Order"

- ...jämför med en *faktiskt* beskrivning av innehållet:

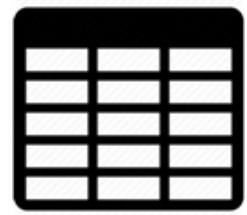
"Person", "Building", "Document"

Attribut i Relationsdatabas



- En kolumn i en tabell representerar ett attribut
attribut heter det också i php's klasser, och i xml.
- Anger oftast en egenskap hos det som är sparat i tabellen
- Jämförelsebart med en Nyckel i PHP

Attribut i Relationsdatabas



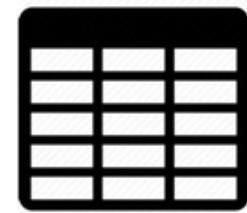
- Ett attribut namnges medLitenBokstavFörst
- Exempel på attribut:

"name", "age"

"serialNumber"

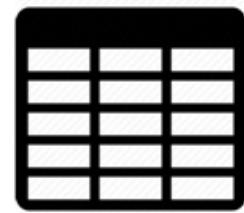
"orderDate", "nrOfItems"

Rader i Relationsdatabas



- Varje rad identifierar en instans av det tabellen definerar,
"En person som heter Gustav och är 63 år gammal"
- Varje cell i raden innehåller ett värde:
"Gustav", 63
- Ibland kallas en rad för en "Tupel"

Värden i Relationsdatabaser



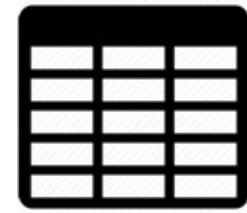
- Vilka värden som är tillåtna bestäms av kolumnens **domän**
- Domän kan vara ett eller flera värden och regler, exempelvis:

String, maxlength 30, minlength 3

Number, max 120, min 0

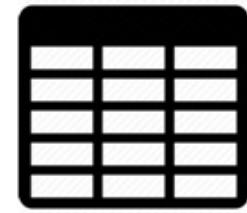
Null or Number or String

Regler i Relationsdatabaser



- Varje tabell har ett unikt namn
- Varje kolumn har ett unikt namn i den tabellen
- Varje cell innehåller exakt **1** värde
- Alla värden i en kolumn måste följa den kolumnens domän
- Varje rad är unik, **det finns inga dubblettter**
- Ordningen på kolumner spelar ingen roll
- Ordningen av rader spelar ingen roll

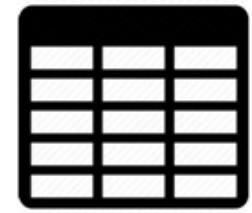
Nycklar i Relationsdatabaser



PHP Repetition:

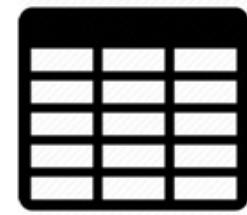
- Nycklar är i PHP något vi använder för att hämta ett specifikt värde ur en array.
- Det kan bara finnas en nyckel med ett visst namn i en array, annars skrivs det förra värdet över.

Nycklar i Relationsdatabaser



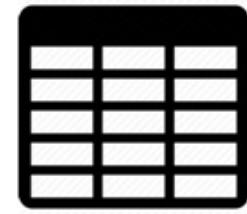
- I relationsdatabaser används nycklar för att hitta en viss rad i en tabell.
- Nycklar är metadata som finns i varje tabell, en tabell kan ha många nycklar.
- En nyckel består av en eller flera kolumner

Nycklar i Relationsdatabaser



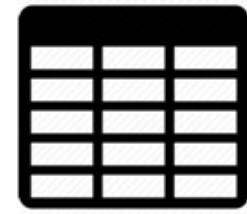
- En **supernyckel** är en nyckel, en eller flera kolumner, som unikt identifierar en rad i tabellen.
- Bra exempel på supernycklar med en kolumn är *personnummer*, *registreringsskylt* eller ett *ordernummer*.
- Ibland kan flera kolumner identifiera en unik rad, exempelvis *longitude* och *latitude*

Nycklar i Relationsdatabaser



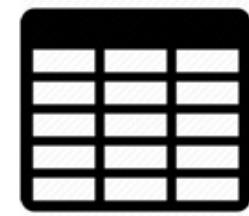
- En tabell kan ha flera olika supernycklar, en av dessa måste dock vara **primärnyckeln**.
- Primärnyckeln är den nyckeln som enklast kan identifiera en rad i tabellen.

Nycklar i Relationsdatabaser



- För att definiera relationer mellan tabeller kan man ange den primära nyckeln från en tabell i en annan tabell
- Denna kolumn kallas då en **främmmande nyckel** (foreign key)

Nycklar i Relationsdatabaser



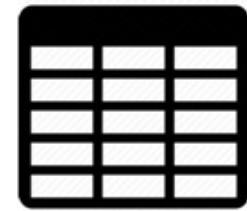
Owner

name	personNr (primary key)
janne	840212
david	564324
viktor	900731
anett	123435

Pet

name	id (primary key)	owner (foreign key)
maru	1231	900731

Relationsdatabaser & DBMS



- Varje tabells attribut, domäner och nycklar definierar **tabellens schema**
- Alla tabellers scheman definierar **databasens schema**
- **Processer** och **vyer** varierar mellan relationsdatabaser, men består nästan alltid av SQL

Tack

för

idag.