

# Informations- arkitektur och databasutveckling

Mikael Olsson  
v13 - Torsdag





# Socrative

<https://www.socrative.com/>

- Frågehanterare
  - Logga in som student
  - Ange rum “Emmio”
  - Få upp en vänta-skärm



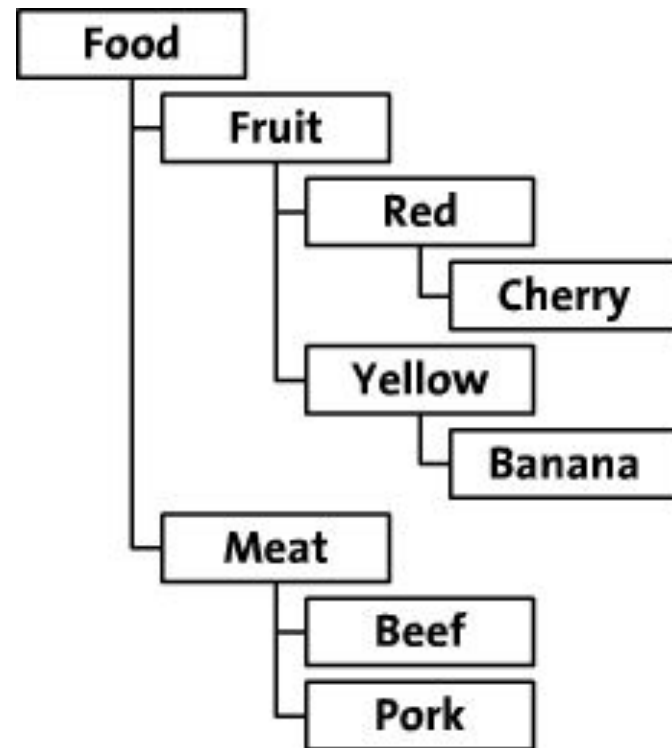
Waiting for the next activity to begin...



# Trädstruktur

- Adjacency List Model

- Begrunda denna trädstruktur.





# Trädstruktur

- Adjacency List Model

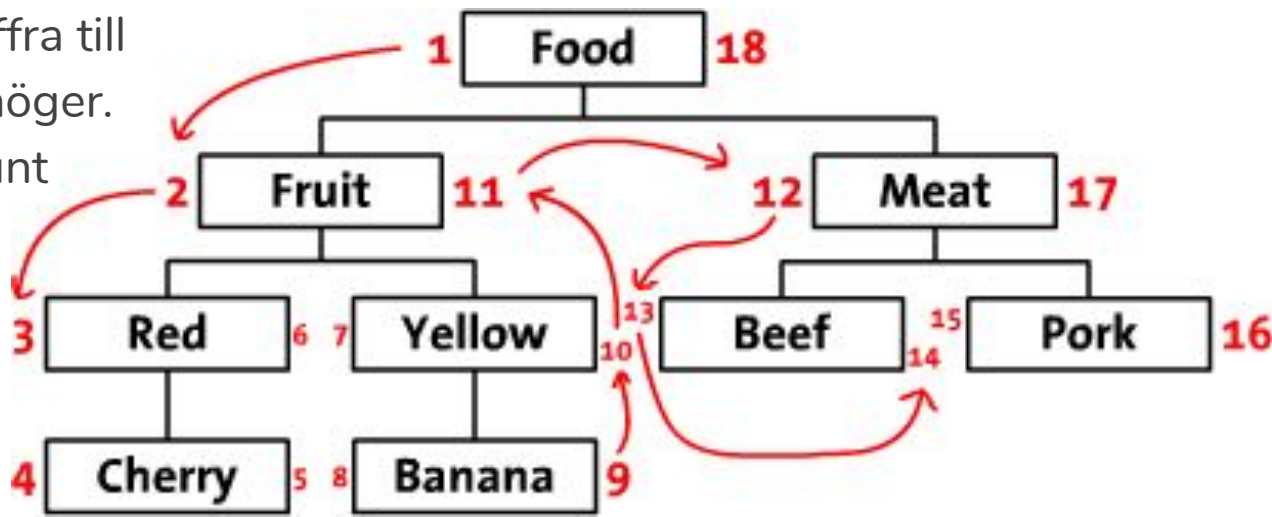
- Kan vi med en fråga hämta ut vilken förälder som Fruit har?
- Kan vi med en fråga hämta ut alla direkta barn som Food har?
- Kan vi med en fråga hämta ut alla poster som härstammar från Fruit?

parent	title
	Food
Food	Fruit
Fruit	Green
Green	Pear
Fruit	Red
Red	Cherry
Fruit	Yellow
Yellow	Banana
Food	Meat
Meat	Beef
Meat	Pork



# Trädstruktur

- Vi gör en modifiering som vi strax ska förklara.
- Varje nod har en siffra till vänster och en till höger.
- Man följer varvet runt moturs.





# Trädstruktur

- Kan vi med en fråga hämta ut vilken förälder som Fruit har?
- Kan vi med en fråga hämta ut alla direkta barn som Food har?
- Kan vi med en fråga hämta ut alla poster som härstammar från Fruit?

parent	title
	Food
Food	Fruit
Fruit	Green
Green	Pear
Fruit	Red
Red	Cherry
Fruit	Yellow
Yellow	Banana
Food	Meat
Meat	Beef
Meat	Pork



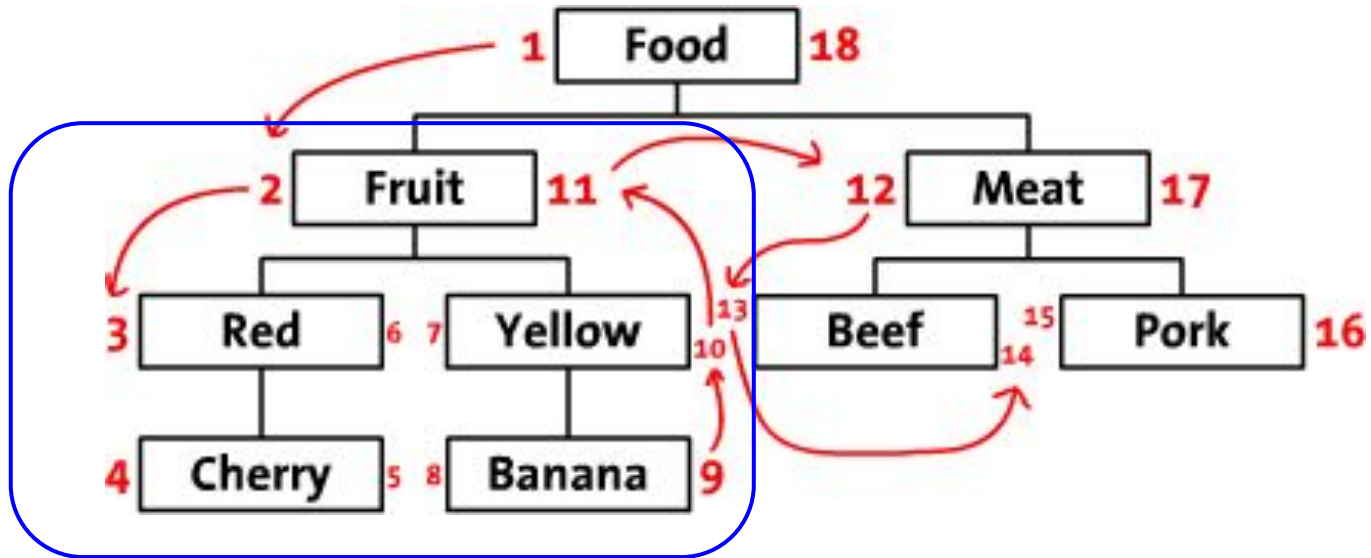
# Trädstruktur

- Vad har Fruit för värde för lft och rgt?
- Vad får vi ut av följande fråga?  
`SELECT *`  
`FROM tree`  
`WHERE lft BETWEEN 2 AND 11`

parent	title	lft	rgt
	Food	1	18
Food	Fruit	2	11
Fruit	Red	3	6
Red	Cherry	4	5
Fruit	Yellow	7	10
Yellow	Banana	8	9
Food	Meat	12	17
Meat	Beef	13	14
Meat	Pork	15	16



# Trädstruktur







# NoSQL

- Typiska RDBMS är antingen gjorda för små men ofta förekommande läs/skriv-transaktioner eller för stora batchar av skrivtransaktioner som körs sällan.
- NoSQL serverar tunga läs/skriv-arbeten.
- NoSQL ger ofta dåliga garantier för konsistens.



# Nyckel-värde-lagring

- Nyckel-värde (key-value) är en typ av lagring som låter applikationen lagra data.
- Data sparas ofta som en datatyp eller ett objekt och därför behöver man ingen fixerad datamodell.



# JSON - JavaScript Object Notation

- Inte en databas! Format, typ som XML.
- {} - objekt
- [] - array
- "" runt strängar, datum mm
- Varje key-value separeras med komma.

```
{  
  "firstName": "John",  
  "lastName": "Smith",  
  "age": 25,  
  "address": {  
    "streetAddress": "21 2nd Street",  
    "city": "New York",  
    "state": "NY",  
    "postalCode": "10021"  
  },  
  "phoneNumber": [  
    { "type": "home", "number": "212 555-1234" },  
    { "type": "fax", "number": "646 555-4567" }  
  ],  
  "newSubscription": false,  
  "companyName": null  
}
```



# Neo4j

- Graf-databas
- <https://www.youtube.com/watch?v=41qdmKlIMz0>



## Git - pull requests

- För att ha koll på sitt repo kan man ibland vilja spärta användare från att pusha upp ändringar.
- Istället kan contributern skapa en pull request, dvs be ägaren att hämta in kod.



# Labbar eller tinkar time för grupparbetena

- Fram till lunch
- Efter lunch: redovisning av grupparbeten

**Lunch!**





# Redovisning grupparbeten

Det jag förväntar mig att se från era webbshoppar är:

- Wireframes (vilka vyer ni har tänkt på, hur ni har tänkt att de ska se ut)
- Modeller
- Ett random sample queries som ni har tänkt att ni behöver för funktionerna, t ex att lägga till produkter i varukorgen, sätta ihop en plocklista eller vad ni nu har tänkt er.





# Slutprojekt

- Slutprojektet ligger i repot, instruktioner mm. Inga resultat ska läggas in här.
- Ni ska skapa ett eget repo. Det är detta repo som ni ska redovisa era resultat i.
- Ert repo ska heta förnamn och initialen i efternamnet, t ex MikaelO. Det är för att underlätta rättningen.
- Instruktioner finns i repot.



# Förberedelser inför nästa tillfälle

- Triggers

<https://www.youtube.com/watch?v=NCEGs4RwAiM>