

# Join

Inner Join, Left Join, Right Join

# Inner Join

- Tämä on hyvin saman tapainen tekniikka kuin Where-osan liitosehto
  - Käytämme samaa esimerkkiä kuin Where-osan liitosehtoon
- Left Join ja Right Join poikkeavat hieman idealtaan

# Inner Join

- Hakee Pulivarin omistajien etunimen ja sukunimen
- Aloitetaan kertaamalla meidän relaatiomallimme



- Omistaa-aulussa viiteavain sekä ankkalinnalainen-auluun että lemmikki-auluun

# Inner Join

- Huomataan, että halutut sarakkeet eli omistajien etunimi ja sukunimi ovat *ankkalinnalainen* taulussa
- Huomataan, että lähtötietomme eli lemmikin nimi Pulivari, löytyy *lemmikki*-taulusta
- Huomataan, että tieto siitä, kuka omistaa kenetkin, löytyy puolestaan *omistaa*-taulusta
- Johtopäätös: Tarvitsemme kaikki kolme taulua mukaan kyselyyn

# Inner Join

- Tämän jälkeen kyselyn tekeminen on mekaanista
- Muistamme, että viiteavain osoittaa aina toisen taulun pääavaimeen
- Joten kirjoitamme taulut yhdistävät liitosehdot
- Omistaa-aulun viiteavain ankkalinnalainen\_id yhdistyy ankkalinnalainen taulun pääavaimeen id
- Omistaa-aulun viiteavain lemmikki\_id yhdistyy lemmikki taulun pääavaimeen id
- Lopuksi täytyy muistaa vielä lisäehto, että haemme nimenomaan Pulivarin omistajia

# Inner Join

- Pulivarin omistajan/omistajien etunimi ja sukunimi:

select etunimi, sukunimi

from ankkalinnalainen

inner join omistaa on omistaa.ankkalinnalainen\_id = ankkalinnalainen.id

inner join lemmikki on omistaa.lemmikki\_id = lemmikki.id and lemmikki.nimi = "Pulivari";

```
MariaDB [ankkalinna]> select etunimi, sukunimi
-> from ankkalinnalainen
-> inner join omistaa on omistaa.ankkalinnalainen_id = ankkalinnalainen.id
-> inner join lemmikki on omistaa.lemmikki_id = lemmikki.id
-> where lemmikki.nimi = "Pulivari";
```

etunimi	sukunimi
Aku	Ankka
Tupu	Ankka

- Uusi opittava asia:
  - Syntaksi *inner join* ja *on*
  - Liitosehto sama kuin Where osan liitosehto tekniikassa
- Huom. Ei ole väliä, missä järjestyksessä taulut liittää toisiinsa
- Huom. Ei ole väliä, kummalle puolelle yhtäsuuruusmerkkiä kirjoittaa viiteavaimen ja pääavaimen
- Huom. Jos useammassa kyselyn taulussa on saman niminen sarake (meillä tässä id), niin aina täytyy muistaa tarkentaa, minkä taulun saraketta tarkoitetaan

# Left Join / Right Join

- Nämä molemmat ottavat toisesta taulusta mukaan kaikki arvot, vaikka liitosta ei olisikaan
  - Vrt. Where-osan liitosehto ja Inner Join, jotka ottavat mukaan vain ne, missä liitos on
- Left Join ottaa vasemmalle puolelle kirjoitetusta taulusta kaikki arvot
- Right Join ottaa oikealle puolelle kirjoitetusta taulusta kaikki arvot
  - Helppo huomata, että nämä ovat vain toistensa peilikuvia

# Left Join / Right Join

- Edellisissä esimerkeissä on haettu Pulivarin omistajaa
- Nyt pohjustetaan Left Join / Right Join esimerkkiä siten, että tehdään Inner Joinilla kysely, joka listaa kaikki ankkalinnalaiset etunimellä ja sukunimellä, ja heidän lemmikkinsä lemmikin nimellä



# Left Join / Right Join

- Kaikkien lemmikillisten ankkalinnalaisten etunimi ja sukunimi sekä heidän lemmikkinsä nimi:

```
select etunimi, sukunimi
from ankkalinnalainen
inner join omistaa on omistaa.ankkalinnalainen_id = ankkalinnalainen.id
inner join lemmikki on omistaa.lemmikki_id = lemmikki.id;
```

```
MariaDB [ankkalinna]> select etunimi, sukunimi, nimi
-> from ankkalinnalainen
-> inner join omistaa on omistaa.ankkalinnalainen_id = ankkalinnalainen.id
-> inner join lemmikki on omistaa.lemmikki_id = lemmikki.id;
```

etunimi	sukunimi	nimi
Aku	Ankka	Pulivari
Tupu	Ankka	Pulivari
Mikki	Hiiri	Pluto
Milla	Magia	Korri

- Huomataan, että Roope Ankka puuttuu listauksesta, koska Roope Ankalla ei ole lemmikkiä. Muistetaan aiemmalta kalvolta, että Inner Join ja Where-osan liitosehto ottaa mukaan vain ne, missä liitos on

```
MariaDB [ankkalinna]> select * from ankkalinnalainen;
```

ID	etunimi	sukunimi
1	Aku	Ankka
2	Roope	Ankka
3	Tupu	Ankka
4	Milla	Magia
5	Mikki	Hiiri

# Left Join / Right Join

- Nyt tehdään kysely, joka listaa kaikki ankkalinnalaiset etu- ja sukunimellä, riippumatta siitä, onko heillä lemmikkiä. Jos heillä on lemmikki, sen nimi mainitaan, muuten se on 'null'

# Left Join

- Kaikki ankkalinnalaiset etunimellä ja sukunimellä ja heidän mahdollinen lemmikkinsä:

select etunimi, sukunimi, nimi

from ankkalinnalainen

left join omistaa on omistaa.ankkalinnalainen\_id = ankkalinnalainen.id

left join lemmikki on omistaa.lemmikki\_id = lemmikki.id;

```
MariaDB [ankkalinna]> select etunimi, sukunimi, nimi
-> from ankkalinnalainen
-> left join omistaa on omistaa.ankkalinnalainen_id = ankkalinnalainen.id
-> left join lemmikki on omistaa.lemmikki_id = lemmikki.id;
```

etunimi	sukunimi	nimi
Aku	Ankka	Pulivari
Roope	Ankka	NULL
Tupu	Ankka	Pulivari
Milla	Magia	Korri
Mikki	Hiiri	Pluto

- Huomataan, että Roope Ankka tuli mukaan listaukseen
- Huomataan, että ankkalinnalainen-taulu on merkitty Left Join:n vasemmalle puolelle

# Right Join

- Kaikki ankkalinnalaiset etunimellä ja sukunimellä ja heidän mahdollinen lemmikkinsä:

select etunimi, sukunimi, nimi

from omistaa

right join ankkalinnalainen on omistaa.ankkalinnalainen\_id = ankkalinnalainen.id

left join lemmikki on omistaa.lemmikki\_id = lemmikki.id;

```
MariaDB [ankkalinna]> select etunimi, sukunimi, nimi
-> from omistaa
-> right join ankkalinnalainen on omistaa.ankkalinnalainen_id = ankkalinnalainen.id
-> left join lemmikki on omistaa.lemmikki_id = lemmikki.id;
```

etunimi	sukunimi	nimi
Aku	Ankka	Pulivari
Roope	Ankka	NULL
Tupu	Ankka	Pulivari
Milla	Magia	Korri
Mikki	Hiiri	Pluto

- Huomataan, että nyt sama tehtiin vastaavasti pelikuvana Right Join avulla, jolloin ankkalinnalainen taulu merkitään Right Joinin oikealle puolelle