

Tietokannan luominen ja datan lisääminen

Tietokannan luominen

- Tehdään tyypillisesti kerran tietokannan elinkaaren aikana
- Voidaan tehdä graafisessa käyttöliittymässä tai SQL-kielen tasolla
 - Opettelemme ja teemme tällä kurssilla SQL-kielen tasolla, koska:
 - SQL-kieltä upotetaan Python kielen sekaan
 - Saamme näin tietokannanluontiskriptin, mikä on hyödyllinen. Hätätilassa voimme vaikka tuhota koko kannan ja saamme sen hetkessä takaisin

Tietokannan luominen

- Itse tykkään kirjoittaa SQL-koodia Notepad++:lla
- Notepad++:sta kopioin koodin MariaDB-komentoriville
- Kalvojen kuvakaappaukset Notepad++:sta ja Command Line Clientistä



Notepad++

App



MySQL Client (MariaDB 10.4 (x64))

App

- Myös graafisessa käyttöliittymässä pystyy toiminaan SQL-tasolla, mikä on myös hyvä ratkaisu

Tietokannan luominen ja käyttöönotto

- create database –lause luo tietokannan
- use ankkalinna -lause ottaa juuri luodun tietokannan käyttöön

```
create database ankkalinna;  
use ankkalinna;
```

Taulujen luominen

- create table -lause luo taulun:
 - Lauseessa määritellään:
 - Pääavain
 - Mahdolliset viiteavaimet
 - Tietotyypit
 - Extroina mm. saako kolumni olla tyhjä ja auto_increment
 - auto_increment tuottaa automaattisesti numerosarjan, jossa ei ole vahingossa kahta kertaa samaa numeroa. Tämä on hyvä apu meille erehtyväisille ihmisille.
- Viiteavainten takia taulut tulee luoda järjestyksessä, jossa viitattava taulu on jo luotu

Taulujen luominen

```
create table ankkalinnalainen(  
  ID int not null auto_increment,  
  etunimi varchar(40),  
  sukunimi varchar(40),  
  primary key (id)  
);  
  
create table lemmikki(  
  ID int not null auto_increment,  
  nimi varchar(40),  
  primary key (id)  
);  
  
create table omistaa(  
  lemmikki_ID int,  
  ankkalinnalainen_ID int,  
  primary key (lemmikki_ID, ankkalinnalainen_ID),  
  foreign key (lemmikki_ID) references lemmikki(ID),  
  foreign key (ankkalinnalainen_ID) references ankkalinnalainen(ID)  
);
```

Datan lisääminen

- insert into -lause lisää dataa
 - auto_incrementin takia joudumme käyttämään pidempää syntaksia, missä luetellaan ne sarakkeet, joihin dataa lisätään
 - auto_increment sarake jätetään mainitsematta
 - auto_increment sarakkeeseen ei lisätä myöskään itse dataa

Datan lisääminen

```
insert into ankkalinnalainen(etunimi, sukunimi)
values("Aku", "Ankka"), ("Roope", "Ankka"),
("Tupu", "Ankka"), ("Milla", "Magia"), ("Mikki", "Hiiri");

insert into lemmikki(nimi)
values ("Pulivari"), ("Pluto"), ("Korri");

insert into omistaa(lemmikki_ID, ankkalinnalainen_ID)
values(1,1), (1,3), (2,5), (3,4);
```


Luontiskripti

- Tekstimuodossa ankkalinnan luontiskripti
 - Tämän avulla on todella nopeaa tehdä itselleen harjoittelua varten ankkalinna-tietokanta

```
create database ankkalinna;
use ankkalinna;
create table ankkalinnalainen(
ID int not null auto_increment,
etunimi varchar(40),
sukunimi varchar(40),
primary key (id)
);
create table lemmikki(
ID int not null auto_increment,
nimi varchar(40),
primary key (id)
);
create table omistaa(
lemmikki_ID int,
ankkalinnalainen_ID int,
primary key (lemmikki_ID,ankkalinnalainen_ID),
foreign key (lemmikki_ID) references lemmikki(ID),
foreign key (ankkalinnalainen_ID) references ankkalinnalainen(ID)
);
```

```
insert into ankkalinnalainen(etunimi, sukunimi)
values("Aku", "Ankka"),("Roope", "Ankka"),
("Tupu", "Ankka"), ("Milla", "Magia"), ("Mikki", "Hiiri");
```

```
insert into lemmikki(nimi)
values ("Pulivari"), ("Pluto"), ("Korri");
```

```
insert into omistaa(lemmikki_ID, ankkalinnalainen_ID)
values(1,1),(1,3),(2,5),(3,4);
```