

Tietokantasuunnittelusta

- Tietokantasuunnittelun pääperiaatteena on tiedon toiston välttäminen.
- Tiedon toistumiseen liittyy monenlaisia ongelmia
 - toistuva tieto vie tilaa
 - ylläpito muodostuu hankalaksi
 - ylläpito-operaatioilla voi olla odottamattomia sivuvaikutuksia.

Tietokantasuunnittelusta

- Esimerkki taulusta, joka ei käyttyä hyvin:
EMP_DEPT:

E_no	E_name	E_bdate	D_no	D_name	D_location
1	M.Seppä	1.3.59	3	Myynti	Helsinki
2	D.Leivo	4.10.40	3	Myynti	Helsinki
3	K.Koivu	30.1.66	4	Hallinto	Lahti
4	B.Oja	2.5.65	4	Hallinto	Lahti
5	O.Itä	10.2.55	6	Tuotanto	Helsinki

Avain: E_no

Jos O.Itä poistetaan,
häviää tieto tuotanto-
osastosta

Toistettava
jokaisen osaston
tt:n kohdalla

Jos Hallinto
muuttaa Espooseen
om muutettava
useita rivejä

Tietokantasuunnittelusta

- Tietokannan suunnittelun vaiheita ovat
 - tietosisällön kartoitus
 - luokkakaaviona (Johdatus sovellussuunnitteluun)
 - kartoittamalla attribuutit ja niiden väliset riippuvuudet
 - loogisten rakenteiden suunnittelu ja
 - teknisten rakenteiden suunnittelu.
- **Loogisen rakenteen suunnittelun tavoite on sijoittaa yhteenkuuluvat tiedot samaan tauluun**

Muunnos luokkakaaviosta relaatiokaavioon

- Järjestelmän määrittelyn yhteydessä järjestelmän tietosisältö voidaan määrittellä **luokkakaavion** avulla
- Luokkakaavio kuvaa tällöin kuvaa järjestelmän **pysyväisluonteisia** (persistent) tietoja - tietojen on säilyttävä ohjelman suorituskertojen välillä
- Luonteva paikka säilyttää olioiden tila on tietokanta:
 - oliorakenteinen tietokanta (oliotietokanta)
 - **relaatiotietokanta**

Muunnos luokkakaaviosta relaatiokaavioon

- relaatiotietokanta
 - standardoitu SQL ja liittymät perinteisiin ohjelmointikieliin
 - ei suoraa tukea olioiden välisille yhteyksille
 - ei luokkahierarkiaa eikä periytymistä
 - vain data ei toimintoja
 - uusimmissa tkhj:ssä mukana oliopiiireitä
 - **edellyttää muunnosta oliorakenteista relaatorakenteiksi**

Muunnos luokkakaaviosta relaatiokaavioon

- Lähtökohtana **normalisoitu** luokkakaavio = **tiettyjä sääntöjä noudattava kaavio**
 - Yhteydet näkyviä, ei attribuuteiksi piilotettuja
 - Kukin asia esitetään vain kertaalleen
 - Ei johdettavissa / pääteltävissä olevaa tietoja
 - Yhteydet määritelty oikeiden osapuolten välillä

Yhteydet piilotettu

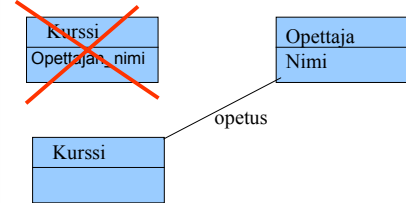
- Olioiden välinen kytkentä esitetään aina yhteytenä, sitä ei piiloteta ominaisuudeksi (attribuutiksi)



Kytkenä on piilotettuna kurssin attribuuttiin opettajan_nimi

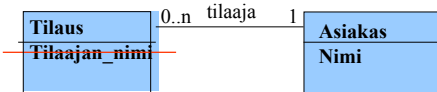
Yhteyden piilotus

- Olioiden välinen kytkentä esitetään aina yhteytenä, sitä ei piiloteta ominaisuudeksi (attribuutiksi)



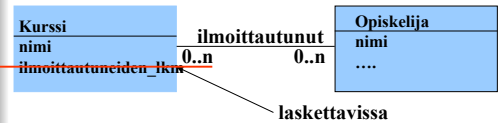
Asiat esitetään vain kertaalleen

- Samaa asiaa ei pidä esittää sekä yhteytenä että ominaisuutena

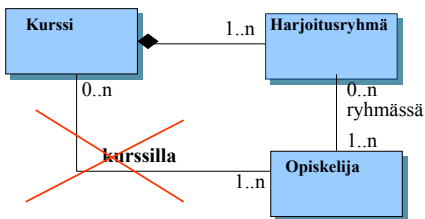


Ei esitetä johdettavissa olevaa tietoa

- tulisi selvittää mitkä tiedot ovat perustietoja ja mitkä voidaan päätellä (laskea) muiden tietojen perusteella
- attribuutin arvo tai yhteyden olemassaolo voi olla pääteltävissä



Ei esitetä johdettavissa olevaa tietoa

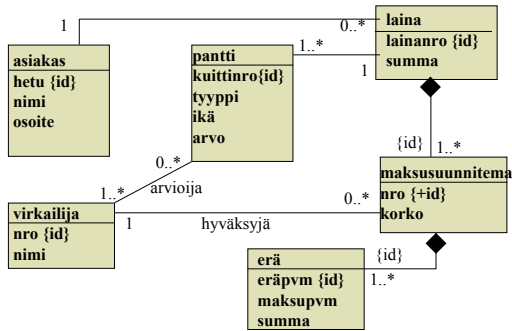


Kursilla-yhteys on pääteltävissä, jos jokaisen kursillaolijan on oltava jossain ryhmässä

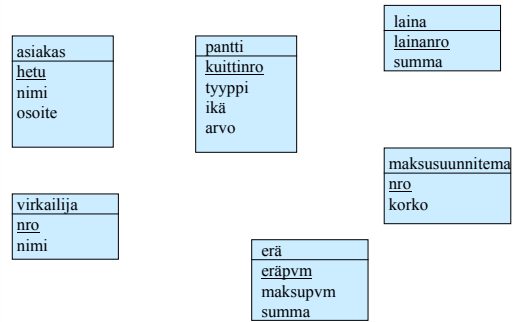
Muunnoksen perussäännöt: Luokat

- Kutakin olioluokkaa vastaa samanniminen taulu
- Kullakin luokan yksiarvoisella attribuutilla on samanniminen vastinsarake luokkaa vastaavassa taulussa
 - Luokan tunnistavia attribuutteja vastaavat sarakkeet kuuluvat luokkaa vastaavan taulun avaimen

Luokkakaavio: Vippi oy



Relaatiorakenteen ydin - jokaista luokkaa vastaa taulu



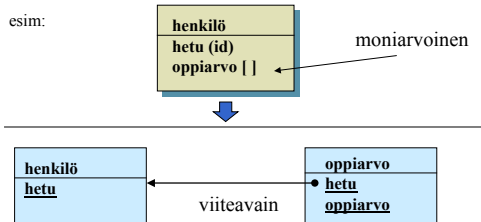
Muunnoksen perussääntö: Kokoelma-attribuutit

- Kutakin **kokoelma-attribuuttia** (moniarvoista) vastaa taulu, jonka sarakkeina ovat
 - viitevain kokoelmatyyppisen attribuutin sisältävää luokkaa vastaavaan tauluun,
 - sarake attribuutin arvo varten.
- Taulun kaikki sarakkeet kuuluvat avaimeen

Kokoelma-attribuutit

- Vippi oy:n esimerkissä ei ole kokoelma-attribuutteja

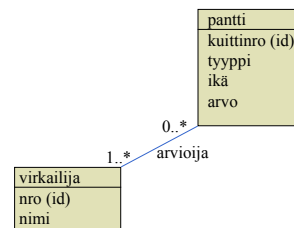
esim:



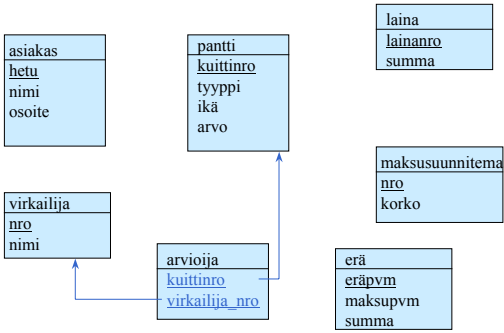
Muunnoksen perussääntö: Monen suhde moneen yhteydet

- Kutakin **monen suhde moneen** yhteyttä vastaa taulu
 - Taulun nimi = yhteyden nimi
 - Taulun sarakkeina ovat yhteyden osapuoliin osoittavat viiteavaimet.
 - Taulun kaikki sarakkeet kuuluvat taulun pääavaimeen
 - monen suhde moneen yhteys on yhteys, jossa kummankin osapuolen maksimiosallistumisrajoite on suurempi kuin 1.

Monen suhde moneen yhteydet



Monen suhde moneen yhteydet



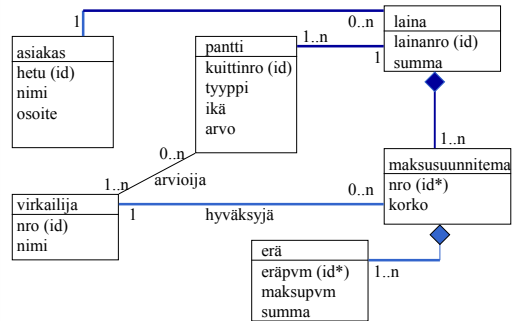
Muunnoksen perussäännöt: Yhden suhde moneen yhteydet

- **Vaihtoehto 1:**
- Lisää yhteyskumppaniin osoittava viiteavain sitä osapuolta vastaavaan tauluun, jolla on korkeintaan yksi kumppani
 - Jos kokoonpanoyhteydessä kokoonpanon osien tunnistamiseen tarvitaan tieto kokonaisuudesta, niin lisätty viiteavain kuuluu taulun avaimiin. Tällaisessa tilanteessa ei ole muita vaihtoehtoja.

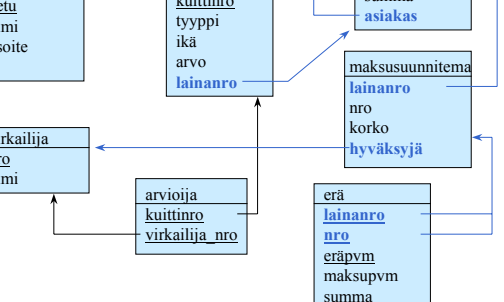
Yhden suhde moneen yhteydet

- **Vaihtoehto 2:**
- Toimi kuten monen suhde moneen yhteyksien kohdalla ja tee yhteyttä varten erillinen taulu
- Taulujen määrän minimoimiseksi vaihtoehtoa 1 tulisi suosia, ellei siitä seuraa epäedullisia viittaussyklejä taulujen välille.

Yhden suhde moneen yhteydet



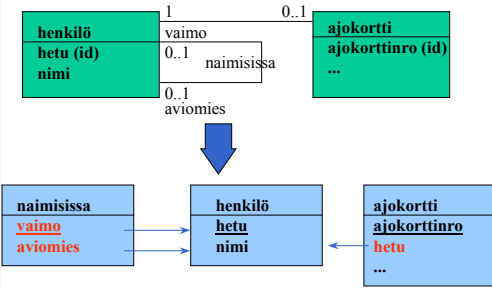
Yhden suhde yhteen yhteydet



Yhden suhde yhteen yhteydet

- **Vaihtoehtoja:**
- Lisää yhteyskumppaniin osoittava viiteavain yhteyden **toista** osapuolta vastaavaan tauluun
 - ensisijaisesti sille, jolle yhteys on **pakollinen**
 - jos yhteys on pakollinen molemmille, valitse kumpi tahansa
- Tee erillinen taulu, kuten monen suhde moneen tapauksessa.
 - jos yhteys on molemmille osapuolille valinnainen

Yhden suhde yhteen yhteydet

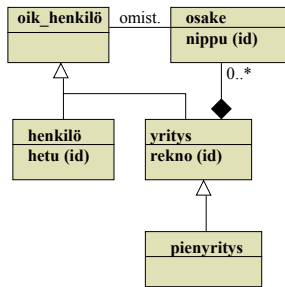


Yleistyshierarkia

➤ Vaihtoehto 1:

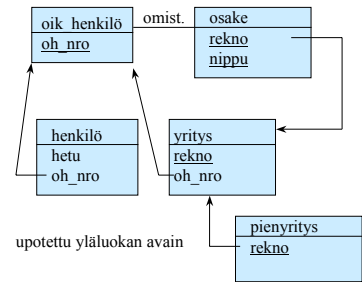
- Taulu sekä ala- että yläluokalle. Alaluokan tauluun sijoitetaan yläluokan tauluun osoittava viiteavain.
 - Jos yläluokalla on käyttäjän määrittelemiä tunnistavia attribuutteja otetaan viiteavain alaluokan taulun avaimeksi.
 - Jos yläluokan taululla ei ole käyttäjän määrittämää tunnistavaa attribuuttia, joudutaan sille luomaan keinotekoinen avain. Tätä ei oteta alaluokan taulun avaimeksi, jos alaluokalle on määritelty tunnistavia attribuutteja.

Yleistyshierarkia



Yleistyshierarkia

käytetty keinotekoisista avainta



Yleistyshierarkia

- **Vaihtoehto 2:** Alaluokalle ei tehdä omaa taulua, vaan alaluokan tauluun loogisti kuuluvat sarakkeet upotetaan yläluokan tauluun.

