

### Tietokantasuunnittelusta

- Tietokantasuunnittelun pääperiaatteena on tiedon toiston välttäminen.
- Tiedon toistumiseen liittyy monenlaisia ongelmia
  - toistuva tieto vie 'turhaa' tilaa
  - ylläpito muodostuu hankalaksi kaikki 'kopiot' päivitettävä
  - ylläpito-operaatioilla voi olla odottamattomia sivuvaikutuksia.
  - Toisaalta toistosta on etuakin – tiedon haku saattaa nopeutua

1

### Tietokantasuunnittelusta

- Esimerkki taulusta, joka ei käyttydy 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 t:n kohdalla

Jos Hallinto muuttaa Espooseen om muutettava useita rivejä

2

### Ei tarpeetonta toistoa

E_no	E_name	E_bdate	D_no
1	M.Seppä	1.3.59	3
2	D.Leivo	4.10.40	3
3	K.Koivu	30.1.66	4
4	B.Oja	2.5.65	4
5	O.Itä	10.2.55	6

D_no	D_name	D_location
3	Myynti	Helsinki
4	Hallinto	Lahti
6	Tuotanto	Helsinki

Parempi ratkaisu

3

### Tietokantasuunnittelusta

- Tietokannan suunnittelun vaiheita ovat
  - tietosisällön kartoitus
    - käsiteanalyysi
    - kartoittamalla attribuutit ja niiden väliset riippuvuudet
  - loogisten rakenteiden suunnittelu ja
  - teknisten rakenteiden suunnittelu.
- Loogisen rakenteen suunnittelun tavoite on sijoittaa yhteenkuuluvat tiedot samaan taulu

4

### Muunnos käsitekaaviosta relaatiokaavioon

- Käsiteanalyysi tuottaa tietosisällön määrittelyn käsitekaavion muodossa
- Käsitekaavio kuvaa tällöin järjestelmän pysyväisluonteisia (persistent) tietoja - tietojen on säilyttävä ohjelman suorituskertojen välillä

5

### Muunnos käsitekaaviosta relaatiokaavioon

- § Lähtökohtana normalisoitu käsitekaavio = tiettyjä sääntöjä noudattava kaavio
  - Yhteydet näkyviä, ei attributeiksi piilotettuja
  - Kukin asia esitetään vain kertaalleen
  - Ei johdettavissa / pääteltävissä olevaa tietoja
  - Yhteydet määriteltä oikeiden osapuolten välillä

6

### Yhteydet piilotettu

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

Kytkeä on piilotettuna kurssin attribuuttiin opettajan\_nimi

7

### Yhteyden piilotus

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

8

### Asiat esitetään vain kertaalleen

- Samaa asiaa ei pidä esittää sekä yhteytenä että ominaisuutena

9

### 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

10

### Ei esitetä johdettavissa olevaa tietoa

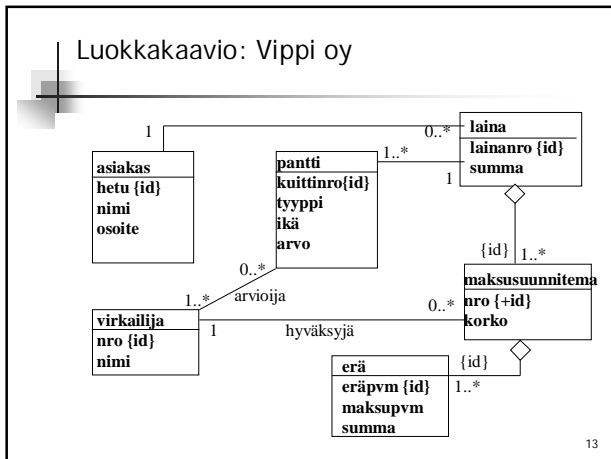
Kurssilla-yhteys on pääteltävissä, jos jokaisen kurssin osanottajan on oltava jossain ryhmässä

11

### Muunnoksen perussäännöt:

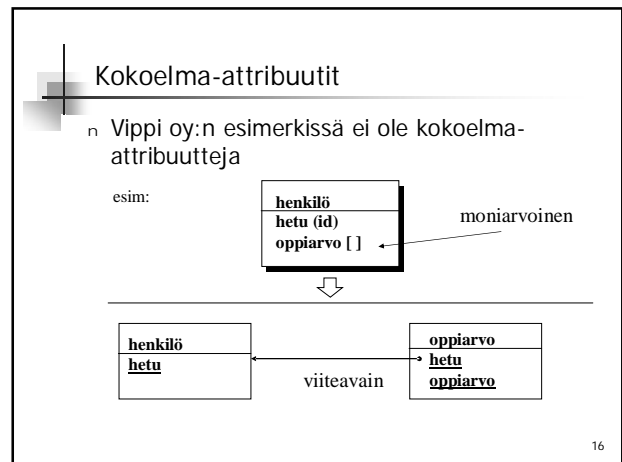
- Kutakin tietokohdetta (tyyppiä) vastaa samanniminen taulu
- Kullakin kohteen yksiarvoisella attribuutilla on samanniminen vastinsarake kohdetta vastaavassa taulussa
- Kohteen tunnistavia attribuutteja vastaavat sarakkeet kuuluvat kohdetta vastaavan taulun avaimiin

12



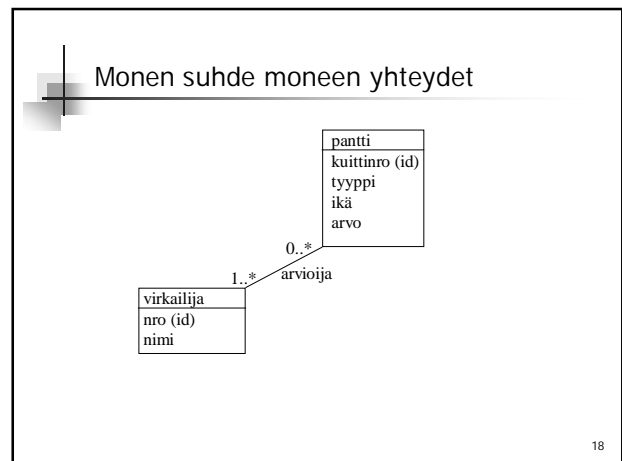
Kokoelma-attribuutit

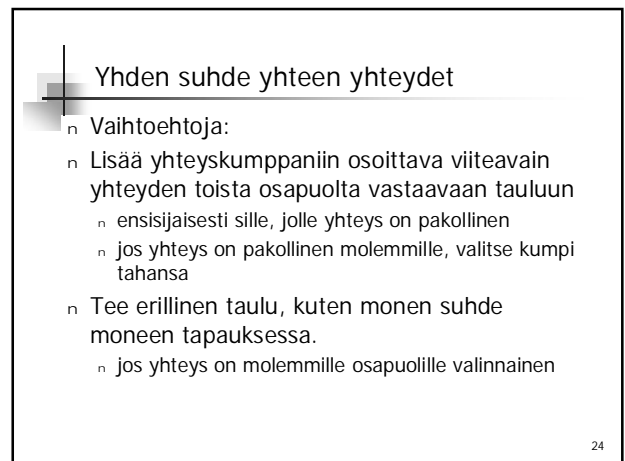
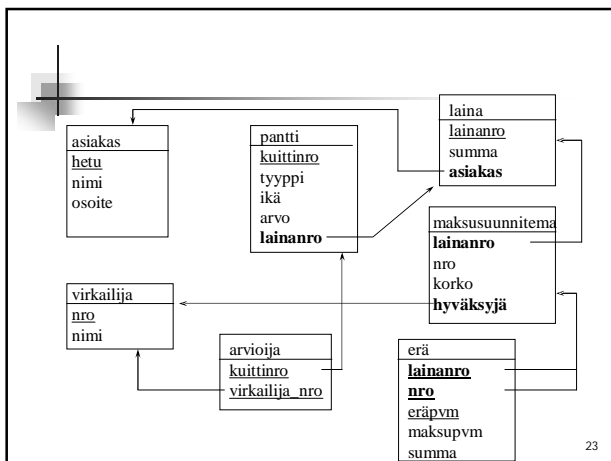
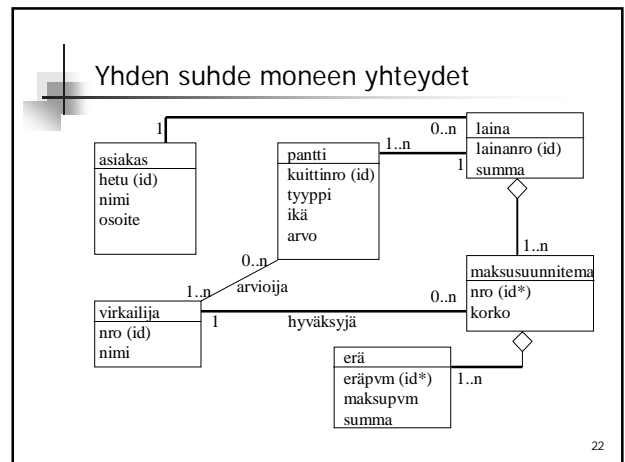
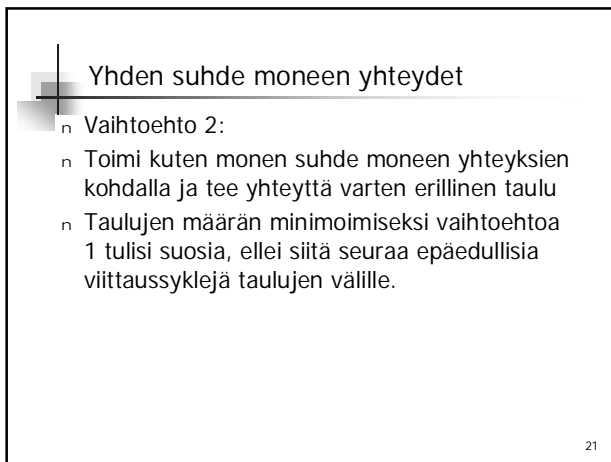
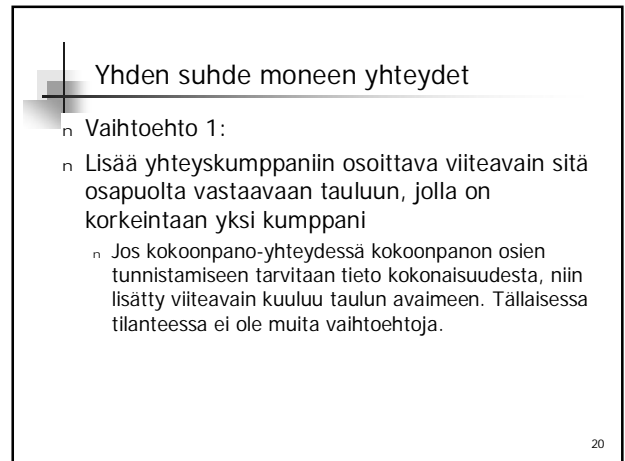
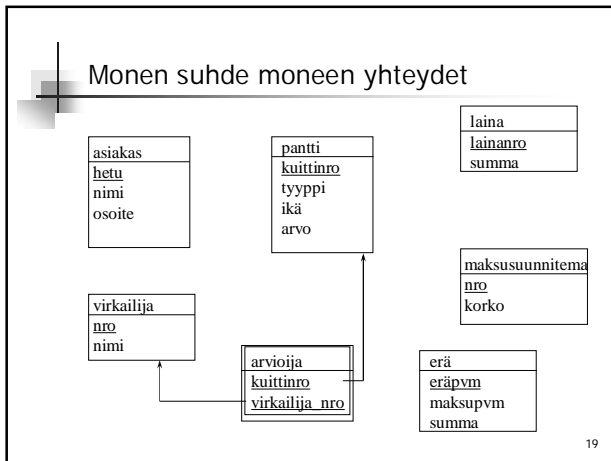
- Kutakin kokoelma-attribuuttia (moniarvoista) vastaa taulu, jonka sarakkeina ovat
  - viiteavain kokoelmatyyppisen attribuutin sisältävää kohdetta vastaavaan tauluun,
  - sarake attribuutin arvoa varten.
- Taulun kaikki sarakkeet kuuluvat avaimen

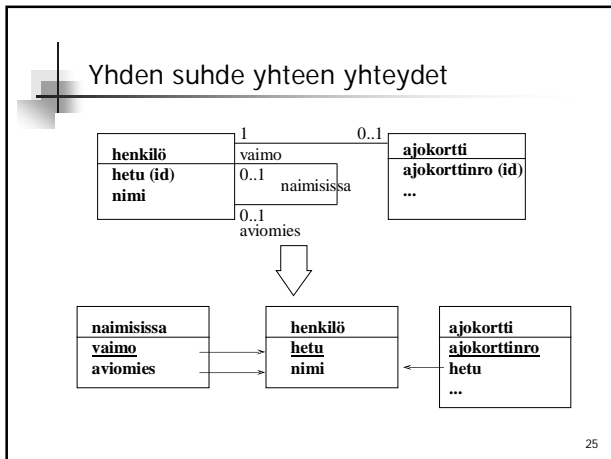


Monen suhde moneen yhteydet

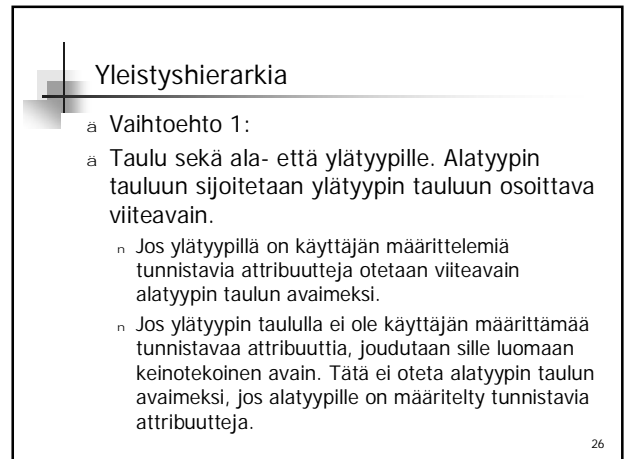
- Kutakin monen suhde moneen yhteyttä vastaa taulu
  - Taulun nimi = yhteyden nimi
  - Taulun sarakkeina ovat yhteyden osapuoliin osoittavat viiteavaimet.
  - Taulun kaikki sarakkeet kuuluvat taulun pääavaimen
  - monen suhde moneen yhteys on yhteys, jossa kummankin osapuolen maksimiosallistumisrajoite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



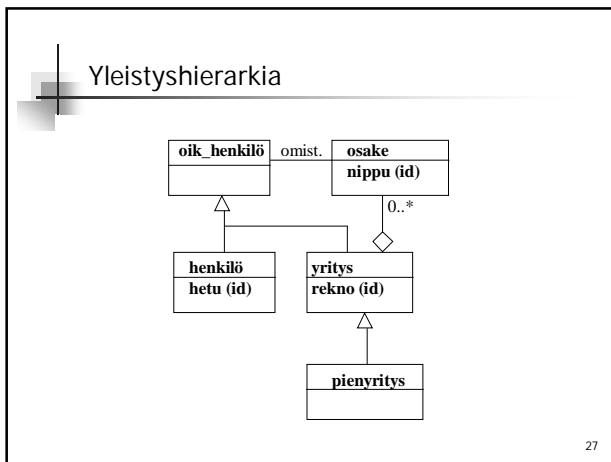




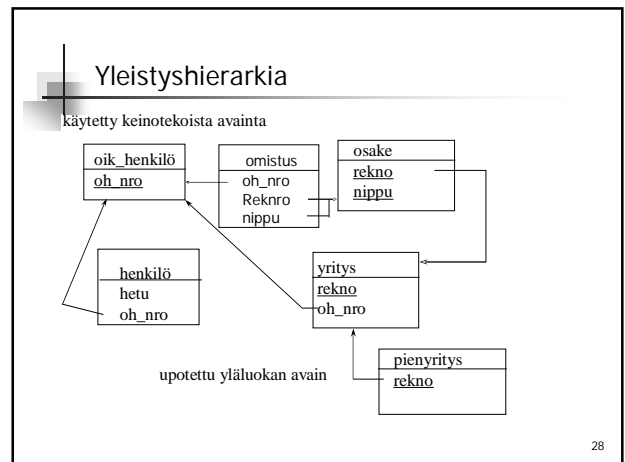
25



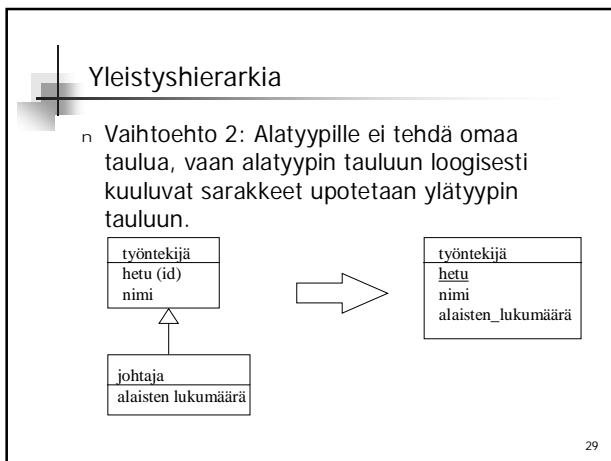
26



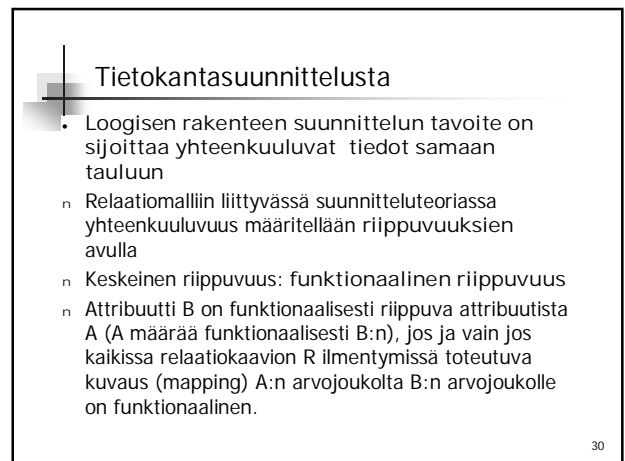
27



28



29



30

### Tietokantasuunnittelusta

- n Kuvaus  $f:v(A) \rightarrow v(B)$  on funktionaalinen, jos kaikissa relaatiokaavion R ilmentymissä jokainen A:n arvo kuvautuu enintään yhdelle B:n arvolla eli,
  - n jos riveillä r ja s attribuutilla A on sama arvo ( $r.A=s.A$ ), niin näillä riveillä täytyy myös B-attribuuteilla olla keskenään sama arvo ( $r.B=s.B$ ).
  - n Funktionaalinen riippuvuus tarkoittaa sitä, että attribuutin B arvo on yksikäsitteisesti selvittävissä, kun tiedetään attribuutin A arvo. Selvittäminen voisi tapahtua kyselyllä
  - n `select distinct B from R where A=a;` (tuloksena olisi enintään yksi rivi)

31

### Tietokantasuunnittelusta

- n Funktionaalista riippuvuutta, jossa A määrää B:n merkitään  $A \twoheadrightarrow B$ . Attribuuttia A kutsutaan määrääjäksi.
- n Yksittäisen attribuutin A tilalla voi olla myös attribuuttiyhdistelmä. Tavoitteena on kuitenkin löytää yhdistelmät, joissa on minimaalinen määrä attribuutteja.
- n Jos  $A \twoheadrightarrow B$ , voidaan määrääjään lisätä mikä tahansa attribuutti x ja pätee  $Ax \twoheadrightarrow B$ .
- n Avain määrää funktionaalisesti kaikki relaatiokaavan attribuutit (= kun tietoa haetaan avaimen arviolla saadaan vastaukseksi enintään yksi rivi)
- n Riippuvuudet ilmaisevat jonkin kuvattavaan todellisuuteen liittyvän säännön

32

### Tietokantasuunnittelusta

- n Tarkastellaan relaatiokaaviota
- n Kurssilainen(
  - n Kurssikoodi,
  - n Hetu,
  - n OpiskelijaNimi,
  - n KurssiNimi,
  - n TehtavaLkm).
- n Oletetaan, että relaatio sisältää tietoja useista opiskelijoista ja useista kursseista ja kuvaa opiskelijoiden osallistumista kursseille.

33

### Tietokantasuunnittelusta

- n Hetu  $\twoheadrightarrow$  OpiskelijaNimi
  - Yhtä henkilötunnusta kohden on vain yksi Opiskelijanimi
  - Henkilötunnuksen perusteella on selvittävissä opiskelijan nimi
- n Kurssikoodi  $\twoheadrightarrow$  KurssiNimi
  - Yhtä kurssikoodia kohti on vain yksi kurssinimi
- n Hetu, Kurssikoodi  $\twoheadrightarrow$  TehtäväLkm
  - Yhtä henkilötunnus & kurssikoodi yhdistelmää kohti on vain yksi tehtävälukumäärä = Henkilön tekemien tehtävien lukumäärä on kurssikohtainen
  - n Yhtä henkilötunnusta kohti voi olla useita tehtävämääriä (opiskelija on monella kursilla)
  - n Yhtä kurssikoodia kohti voi olla monta tehtävämääriä (kursilla on monta opiskelijaa)

34

### Tietokantasuunnittelusta

- n Mitkä sarakkeet samaan relaatiokaavioon?
- n Erilaisia kriteerejä
- n Yhteenkuuluvuussääntönä, ns. Boyce-Codd normaalimuodon sääntö:
  - relaatiokaavion R attribuutit kuuluvat yhteen, jos ja vain jos relaatiokaavioon R ei liity yhtään sellaista funktionaalista riippuvuutta, jossa määrääjä ei sisältäisi relaation avainta.
- n Kurssilainen relaation avain on pari Kurssikoodi, Hetu
- n => Kaikki attribuutit eivät BC-säännön mukaan kuulu yhteen koska
  - n Hetu  $\twoheadrightarrow$  OpiskelijaNimi ja
  - n Kurssikoodi  $\twoheadrightarrow$  KurssiNimi rikkovat BC-sääntöä

35

### Tietokantasuunnittelusta

- n Attribuutit on uudelleenjärjesteltävä relaatiokaavioiksi, joissa yhteenkuuluvuussäännöt ovat voimassa.
- n päädytään relaatiokaavioihin
  - n Opiskelija(Hetu, OpiskelijaNimi)
  - n Kurssi(Kurssikoodi, KurssiNimi)
  - n Osallistuminen(Hetu, Kurssikoodi, TehtäväLkm).

36

### Jako Boyce-Codd normaalimuotoon

- n Miten järjestellään:
  - n Karsi riippuvuusjoukko minimaaliseksi – poista johdettavissa olevat riippuvuudet (materiaalissa sääntöjä)
  - n Ryhmitä riippuvuudet yhteisen määrääjän (vasen puoli) perusteella.
  - n Muodosta jokaista ryhmää kohti oma relaatio
  - n Jos alkuperäisen kaavion yksikään avain ei sisälly muodostuneisiin relaatiokaavioihin tee sille oma kaavio
  - n Anna relaatiokaavioille nimet

37

### Taulurakenteen laadunvalvontaa

- n Määrittele taulussa voimassaolevia funktionaalisia riippuvuuksia
- n Määrää taulun avain
  - n avaimeen sisältyvät ainakin kaikki sarakkeet, jotka eivät esiinny missään funktionaalisessa riippuvuudessa vasemmalla puolella (= eivät riipu mistään)
- n Jos löytyy yksikin riippuvuus, jossa määrääjä ei sisällä taulun avainta relaatio ei ole BCNF-muodossa ja on mietittävä sen jakamista

38

### Tietokantasuunnittelusta

- n Tilauslomaketta analysoitaessa löydettiin seuraavat attribuutit:
  - n lomakenumero,
  - n tilaajan tunnus,
  - n tilaajan nimi,
  - n tilaajan osoite,
  - n tilaajan puhelinnumero,
  - n toimitusosoite,
  - n rivinumero,
  - n tavarankoodi,
  - n tavaranimi,
  - n tilattu määrä, ja
  - n tilauspäivä.

39

### Tietokantasuunnittelusta

- lomakenumero → tilaajan tunnus (lomakkeella voidaan ilmoittaa vain yksi tilaaja)
- tilaajan tunnus → tilaajan nimi (tunnus identifioi tilaajan, joten sen perusteella saamme selville kaikki tilaajaan liittyvät tiedot)
- tilaajan tunnus → tilaajan osoite,
- tilaajan tunnus → tilaajan puhelinnumero,
- lomakenumero → toimitusosoite (lomakkeella voidaan tilata tavaroita vain yhteen paikkaan)
- lomakenumero → tilauspäivä (lomakkeeseen liittyy yksikäsitteinen tilauspäivä)

40

### Tietokantasuunnittelusta

- tavarankoodi → tavaranimi (tavarankoodi on tavarantunniste, jonka kautta päästään kaikkiin tavarantietoihin)
- lomakenumero, rivinumero → tavarankoodi (lomakkeen rivillä voi ilmoittaa yhden tilattavan tavarankoodin)
- lomakenumero, rivinumero → tilattu määrä (lomakkeen rivillä voi ilmoittaa yhden tilattavan tavarankoodin määrän)

41

### Tietokantasuunnittelusta

- n Jako relaatioihin yhteisen määrääjän perusteella (jos riippuvuuksissa sama määrääjä, niin niissä esiintyvät attribuutit samaan relaatioon)
  - n X(tilaajan\_tunnus, tilaajan nimi, tilaajan osoite, tilaajan puhelinnumero)
  - n Y(tavarankoodi, tavaranimi)
  - n Z(lomakenumero, tilauspäivä, tilaajan\_tunnus, toimitusosoite)
  - n T(lomakenumero, rivinumero, tavarankoodi, tilattu määrä)

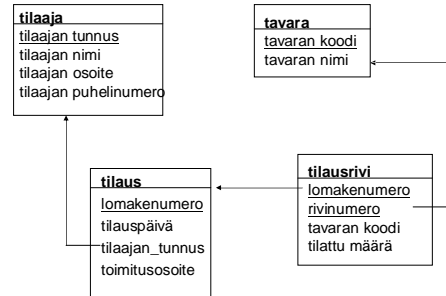
42

### Tietokantasuunnittelusta

- Jos kaavioille löytyy kuvaava nimi, jako on onnistunut:
  - tilaaja(tilaajan tunnus, tilaajan nimi, tilaajan osoite, tilaajan puhelinnumero)
  - tavara(tavaran koodi, tavarannimi)
  - tilaus(lomakenumero, tilauspäivä, tilaajan\_tunnus, toimitusosoite)
  - tilausrivi(lomakenumero\_rivinumero, tavarannimi, tilattu määrä)

43

### Tietokantasuunnittelusta



44