

Tietokantojen suunnittelu - Luento 20

- [Tietokannan suunnittelu](#)
 - [Taulut](#)
 - [Kentät \(Ominaisuudet\)](#)
 - [Taulujen väliset suhteet \(viite-eheydet\)](#)
- [Normalisointi](#)
 - [Ei normaalimuotoinen tietokanta](#)
- [Tietokannan suunnittelu käytännössä](#)

Tällä luennolla käydään läpi tietokantojen suunnittelua ja normalisointia. Lisämateriaalina voi lukea seuraavat dokumentit:

[Tietokannan suunnittelun perusteet](#)

[Tietokantasuunnittelua maalaisjärjellä](#)

Tietokannan suunnittelu

Yleensä tietokantoja suunnitellaan [ER-kaavioiden](#) avulla mutta aikapulan takia emme käsittele niitä tällä kurssilla.

Tietokantaa suunniteltaessa täytyy miettiä millaisia tauluja ja niiden välisiä riippuvuuksia tarvitaan.

Taulut

- Taulu (taulukko) esittää jotakin kohdetta
- Kohde on *ominaisuuskokoelma*
- Kohde ei ole ominaisuus tai yksittäinen tieto!
- **Tauluun liitetään vain tauluun välittömästi liittyviä tietoja (ominaisuuksia).**
- **Kunkin tiedon päivitys tapahtuu vain yhteen paikkaan**

- Kohde on tunnistettavissa oleva asia tai tapahtuma.
- Taulua kuvaa usein substantiivi.
- Helposti havaittavia kohteita ovat käyttäjien puheissa esiintyvät henkilöt, esineet, tilat ja tuotteet. Hankalampia ovat käsitteelliset kohteet kuten tilaus tai sopimus.
- Kohteet ovat sellaisia, joista halutaan tallentaa tietoja pysyvästi tietokantaan. Raportit ja tulosteet eivät ole kohteita vaan tietokannan tiedoista johdettuja tulostietoja

Kentät (Ominaisuudet)

- Ominaisuus vastaa taulussa sarake**otsikoita**.
- Jokaisella samantyyppisellä kohteella on tiettyjä yhteisiä ominaisuuksia
 - Henkilöillä on kaikilla nimi, hetu, osoite yms.
- **Ominaisuus ei ole tallennettava tieto (arvo)! Henkilökohteen ominaisuus voi olla Nimi mutta ei Tommi.**

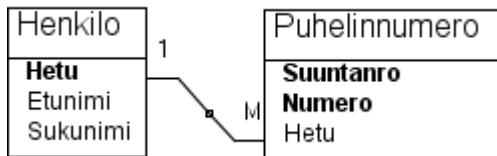
Vinkki: Ajattele taulu lomakkeena ja taulun kenttiä lomakkeen kenttinä joihin pitäisi syöttää tietoja.

- Ominaisuuksien joukosta valitaan **yksilöivä perusavain**
 - Avaimen pitää olla uniikki eli jokaisella rivillä eri.
 - Avain voi olla myös yhdistelmä useammasta ominaisuudesta. Esim. puhelinnumero: etuosa + loppuosa
 - Hyvin usein perusavain on muuttumaton esim. hetu
 - Avain voi olla myös keksitty juokseva numero omana ominaisuutena.
- Ominaisuus voi olla yksi- tai moniarvoinen. **Jos ominaisuus on moniarvoinen niin siitä pitää tehdä oma kohde (taulu).** Esim. Jos henkilöllä voi olla yksi puhelinnumero niin kyseessä on henkilön ominaisuus (henkilö-taulun sarake) mutta jos henkilölle sallitaan useampia puhelinnumeroita niin puhelinnumerosta täytyy tehdä oma taulu.
- Ominaisuuden arvojen pitää olla jakamattomia eli osat pitää erottaa aina omiksi ominaisuuksikseen. Esim. etunimi ja sukunimi.
- Muista ominaisuuksista **johdettavia tietoja** ei kannata laittaa ominaisuuksiksi
 - Tilausten kokonaislukumäärä lasketaan yksittäistilausten kappalemääristä.
 - Ikä lasketaan syntymäajasta.

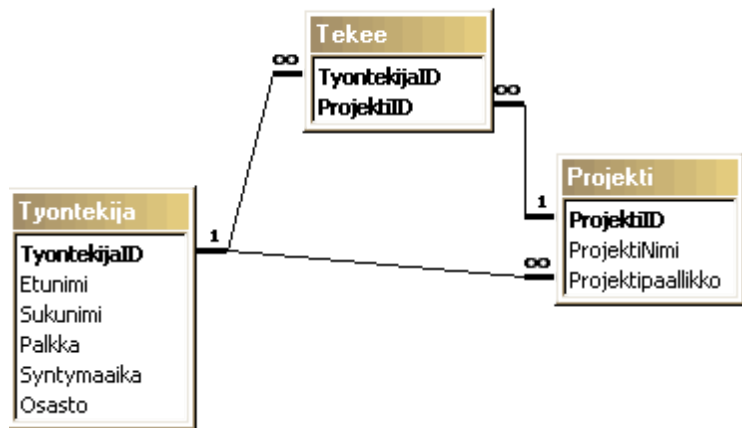
- Ominaisuus voi saada arvonsa tietyistä arvojoukosta (esim. kurssin arvosana voi olla 1,2,3,4,5 tai 0 eli hylätty).
- Sallitaanko ominaisuuksille tyhjät arvot. (esim. koditon)
- Sallitaanko tuntemattomat arvot. Vältä aina kuin mahdollista.

Taulujen väliset suhteet (viite-eheydet)

- Suhde on vähintään kahden taulun (kohteen) välinen riippuvuus eli viite-eheys
- Suhdetta kuvaa usein verbi.
- Suhde voi merkitä olemassaoloa, toiminnallista suhdetta tai tapahtumaa
- Keksi, jos mahdollista, sellainen kuvaava sana, joka toimii molempiin suuntiin.
- Suhde voi olla yhden suhde yhteen (**one-to-one, 1-to-1**) eli sama asia kuin ominaisuus.
- Suhde voi olla yhden suhde moneen (**one-to-many, 1-to-M**) (monen suhde yhteen (**many-to-one, M-to-1**)) eli kahden taulun välille syntyy viite-eheys:



- Suhde voi olla monen suhde moneen (**many-to-many, M-to-M**) jolloin tarvitaan uusi taulu kuvaamaan suhdetta.



Työntekijä voi tehdä montaa projektia ja jokaisessa projektissa voi olla monta työntekijää (m-to-m-suhde).

Työntekijä voi johtaa montaa projektia mutta projektissa voi olla vain yksi johtaja (1-to-m-suhde).

Normalisointi

Tietokannan rakenteen suunnittelun jälkeen on rakenne vielä tarkistettava ja varmistettava sen järkevyys. Tätä kutsutaan normalisoinniksi.

- Pyritään vähentämään tietojen ylimäärää ja tästä aiheutuvia ongelmia tietojen lisäämisessä, poistamisessa ja päivityksessä
- Lisätään rakenteiden selkeyttä, yhtenäisyyttä ja laajennettavuutta
- Normalisointi suosii ympäristöjä, joissa tietoja päivitetään usein
 - Hakeminen vaatii enemmän tehoa kuin normalisoimattomassa ratkaisussa koska tiedot joudutaan hakemaan useammasta taulusta
- Normalisointi tapahtuu suhteellisen automaattisesti jos noudatetaan sääntöjä:
 - **Kohteen yhteyteen tallennetaan vain kohteeseen välittömästi liittyviä tietoja**
 - **Kunkin tiedon päivitys tapahtuu vain yhteen paikkaan**

Ei normaalimuotoinen tietokanta

Myynti		
Piiri	Myyja	Tuote
P1 Länsi	M1 Pirkko Puoti	T1 23500, T2 17200, T4 12300
P1 Länsi	M2 Heikki Helppo	T1 26800, T4 19300
P2 Itä	M3 Kaija Kauppa	T1 16200, T3 39200
P2 Itä	M4 Rauno Ratsu	T3 16500, T4 17200, T5 19100

Virheet

- Kenttien arvot eivät ole jakamattomia

Myynti									
PiiriID	Piiri	MyyjaID	Myyja	Tuote1ID	Tuote1myynti	Tuote2ID	Tuote2myynti	Tuote3ID	Tuote3myynti
P1	Länsi	M1	Pirkko Puoti	T1	23500	T2	17200	T4	12300
P1	Länsi	M2	Heikki Helppo	T1	26800	T4	19300		
P2	Itä	M3	Kaija Kauppa	T1	16200	T3	39200		

P2	Itä	M4	Rauno Ratsu	T3	16500	T4	17200	T5	19100
----	-----	----	-------------	----	-------	----	-------	----	-------

Virheet

- Riveillä on eri määrä tietoa. Ei voida tietää etukäteen montaako tuotetta kukin myyjä tulee myymään.

Myynti					
PiiriID	Piiri	MyyjaID	Myyja	TuoteID	Myynti
P1	Länsi	M1	Pirkko Puoti	T1	23500
P1	Länsi	M1	Pirkko Puoti	T2	17200
P1	Länsi	M1	Pirkko Puoti	T4	12300
P1	Länsi	M2	Heikki Helppo	T1	26800
P1	Länsi	M2	Heikki Helppo	T4	19300
P2	Itä	M3	Kaija Kauppa	T1	16200
P2	Itä	M3	Kaija Kauppa	T3	39200
P2	Itä	M4	Rauno Ratsu	T3	16500

P2	Itä	M4	Rauno Ratsu	T4	17200
P2	Itä	M4	Rauno Ratsu	T5	19100

Virheet

- Taulun perusavaimeksi muodostuu yhdistetty avain PiiriID, MyyjaID ja TuoteID -kentistä. Kaikkien muiden kenttien pitäisi riippua taulun avaimesta. Piiri-kenttä riippuu kuitenkin vain PiiriID-kentästä eikä koko perusavaimesta. Myyja-kenttä riippuu vain MyyjaID-kentästä eikä koko perusavaimesta.
- Taulussa on myös havaittavissa aivan turhaa toistoa josta pitää päästä eroon.
- Mitä jos tulee uusi myyjä, joka ei ole vielä myynyt mitään? Miten hänen tietonsa tallennetaan järjestelmään? (insertion anomaly)
- Mitä jos Pirkko Puoti ja Heikki Helppo saavat potkut? Poistamalla heidän tietonsa järjestelmästä menetetään myös Länsi-piiri. (deletion anomaly)
- Mitä jos Pirkko Puoti siirretään uuteen piiriin? Täytyy muistaa päivittää useiden rivien tiedot samalla kertaa. (modification anomaly)

Myynti							
MyyjaID		Myyja		TuoteID		Myynti	
M1		Pirkko Puoti		T1		23500	
M1		Pirkko Puoti		T2		17200	

M1		Pirkko Puoti		T4		1230 0	
M2		Heikko Helppo		T2		2680 0	
M2		Heikko Helppo		T4		1930 0	
M3		Kaija Kaupp a		T1		1620 0	
M3		Kaija Kaupp a		T3		3920 0	
M4		Rauno Ratsu		T3		1650 0	
M4		Rauno Ratsu		T4		1720 0	
M4		Rauno Ratsu		T5		1910 0	

Piiri						
PiiriID	Piiri		Myyjä ID		Myyjä	
P1	Länsi		M1		Pirkko Puoti	
P1	Länsi		M2		Heikki Helppo	
P2	Itä		M3		Kaija Kauppa	
P2	I		M4		Rauno	

	t ä				Ratsu	
--	--------	--	--	--	-------	--

Virheet

- Edelleen jää epäselväksi, että minne pitäisi uuden myyjän tiedot lisätä. Edelleen ongelmia myös päivittämisessä ja poistamisessa.
- Edelleen turhaa toistoa

Myynti				
MyyjaID		Tuot eID		Myynti
M1		T1		23500
M1		T2		17200
M1		T4		12300
M2		T2		26800
M2		T4		19300
M3		T1		16200
M3		T3		39200

M4		T3		16500
M4		T4		17200
M4		T5		19100

Piiri		
PiiriID	P i i r i	
P1	L ä n s i	
P2	I t ä	

Myyja		
MyyjaID	Myyja	PiiriID
M1	Pirkko Puoti	P1
M2	Heikki Helppo	P1
M3	Kaija Kauppa	P2
M4	Rauno Ratsu	P2

- Nyt jokaisen asian päivittäminen, poistaminen tai lisääminen tapahtuu vain yhteen paikkaan.
- Jokaisessa taulussa (kohteessa) on vain siihen liittyviä tietoja.

Mitä jos jokaisesta tuotteesta olisi tallennettuna myös tuotteen nimi?

Myynti			
MyyjaID	TuoteID	TuoteNimi	Myynti
M1	T1	Pölynimuri	23500
M1	T2	Linkkuveitsi	17200
M1	T4	Tietokone	12300

M2	T2	Linkkuveitsi	26800
M2	T4	Tietokone	19300
M3	T1	Pölynimuri	16200
M3	T3	Kirjahylly	39200
M4	T3	Kirjahylly	16500
M4	T4	Tietokone	17200
M4	T5	Mopo	19100

Virheet

- Taas on havaittavissa toistoa.
- Mitä jos tuotetta ei ole myyty ollenkaan? Missä sen tiedot ovat silloin? Vastaavanlaiset ongelmat ilmenevät myös poistamisen ja päivittämisen kanssa.

Myynti				
MyyjaID		TuoteID	My yn ti	

M1		T1	23 50 0	
M1		T2	17 20 0	
M1		T4	12 30 0	
M2		T2	26 80 0	
M2		T4	19 30 0	
M3		T1	16 20 0	
M3		T3	39	

			200	
M4		T3	16500	
M4		T4	17200	
M4		T5	19100	
Tuote				
TuoteID	Tuotteen Nimi			

	i		
T1	P ö l y n i m u r i		
T2	L i n k k u v e i t s i		

T3	K i r j a h y l l y		
T4	T i e t o k o n e		
T5	M o p o		

Tietokannan suunnittelu käytännössä

Suunnitellaan tietokanta 24 tunnin viestihiihdon tuloslaskentaa varten. Tietokannan yksinkertainen vaatimusmäärittely on seuraava:

- 24 tunnin viestihiihdossa on joukkueita joissa on yksi tai useampia hiihtäjiä, jotka hiihtävät joukkueen itse määräämässä järjestyksessä yhden tai useampia hiihtovuoroja. Yhdellä hiihtovuorolla täytyy hiihtää vähintään yksi kierros.
- Tuloslaskennassa halutaan laskea montako kierrosta kukin joukkue on hiihtänyt, kauanko jokaisen hiihtäjän kuhunkin kierrokseen on mennyt aikaa ja mihin kellonaikaan hiihtäjä on saanut kierroksensa loppuun, montako kierrosta ja kilometriä kukin hiihtäjä on hiihtänyt, montako kilometriä kukin joukkue on hiihtänyt
- Viesti hiihdetään kiertäen jotain ennaltasovittua tietynmittaista reittiä
- Viestin vaihdot tehdään aina samassa pisteessä eli tulosseurannan edessä. Hiihtäjää ei siis voi vaihtaa kesken kierrosta vaan aina täytyy kierros hiihtää loppuun
- Jokaisella hiihtäjällä on oma numerolappu josta hiihtäjä tunnistetaan.