

Lopputyö – Tietokannat ja tiedonhallinta

Jaanu Sami

Koskipahta Tuomas

Tuominen Ilari

SWD1TN003-3003

Tietokannat ja tiedonhallinta

Sisällys

1 Johdanto 1

2 Sovellusalueen kuvaus 1

3 Käsitteellinen mallinnus 2

3.1 Luokkakaavio 2

3.2 Tietohakemistokuvaus 3

4 Looginen mallinnus 5

4.1 Normalisointi & tarkistettu Relaatiokaavio 5

4.2 Eheysmäärittelyt 5

5 Toteutus 6

5.1 SQL-luontilauseet 6

6 Testaus 8

6.1 Testidatan syöttö 8

6.2 Testilauseet ja testitulokset 13

7 Loppuarvio ja työajan käyttö 16

# Johdanto

Olemme Haaga-Helian tietojenkäsittely koulutusohjelman ensimmäisen vuoden opiskelijoita. Tietokannat ja tietojenhallinta kurssilla saimme tehtäväksi luoda tietokantaohjelman Kurun kulttuuriyhdistykselle, joka korvaa nykyisen lipunmyyntijärjestelmän. Tehtävänämme on kehittää järjestelmä, jonka avulla Kurun kulttuurinystävät yhdistys, pystyy käsittelemään tehokkaammin lippujen varausta

# Sovellusalueen kuvaus

Kurun kulttuurinystävät on yhdistys, joka järjestää konsertteja sekä tanssi- ja teatteriesityksiä. Lipunmyynti on toistaiseksi hoidettu käsipelillä. Nyt heidän tavoitteenansa on rakentaa tietokantapohjainen palvelu, jossa asiakas voi etsiä kiinnostavan tilaisuuden Kurun Kulttuurinystävien Internet-sivuilta, tehdä varauksen ja saada varausnumeron www-lomakkeella. Lippujen maksaminen ja lunastus hoidettaisiin vastedeskin käteisellä Kurun kirjastossa 3 päivän sisällä varauksesta.

Järjestelmän tietotarpeisiin kuuluvat

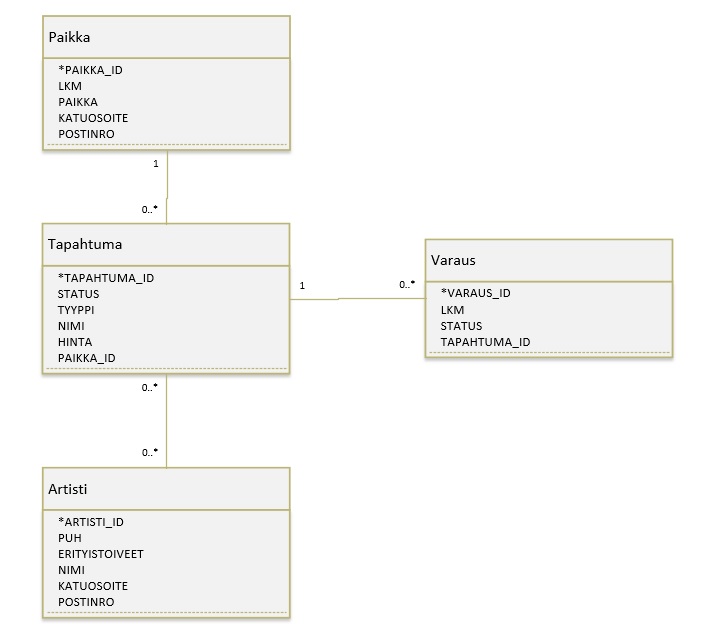
1. Lippuvarauksen tekeminen
2. Lippuvarauksen peruuttaminen
3. Lippuvarauksen lippumäärän muuttaminen
4. Lunastettujen lippujen kirjaaminen myydyiksi
5. Lunastamattomien varausten poistaminen tilaisuuksista, jotka ovat jo toteutuneet.
6. Tilaisuuden peruuttaminen (Lippujen jo lunastaneiden täytyy saada rahansa takaisin)

Tietojärjestelmän vaatimuksina ovat, mitä esityksiä ja esiintyjiä on tarjolla, ja kuinka paljon paikkoja on jäljellä tapahtumiin, sekä missä ja milloin esitykset ja esiintyjät ovat.

Tietojärjestelmä toteutetaan SQL-kielellä, jossa määritellään taulut ja niiden väliset yhteydet. Tavoitteena on, että järjestelmän käyttäjä pystyy seuraamaan lippujen määrää ja myyntiä, tekemään tilauksen tiettyyn tapahtumaan ja myös peruuttamaan tilauksensa.  Myös artistien yhteystiedot olisivat saatavilla helposti.

# Käsitteellinen mallinnus

## Luokkakaavio



## Tietohakemistokuvaus

|  |  |
| --- | --- |
| Käsitteen nimi: | Tapahtuma |
| Nimilyhenne: |  |
| Määritelmä: | Kulttuurinen tapahtuma johon asiakkaat voivat varata lippuja. |
| Ominaisuudet: | TAPAHTUMA\_ID NIMI STATUS TYYPPI HINTA PAIKKA\_ID |
| Pääavain: | TAPAHTUMA\_ID |

|  |  |
| --- | --- |
| Käsitteen nimi: | Paikka |
| Nimilyhenne: |  |
| Määritelmä: | Paikka tai alue, jossa tapahtuma järjestetään |
| Ominaisuudet: | PAIKKA\_ID  KATUOSOITE PAIKKA LKM POSTINRO |
| Pääavain: | PAIKKA\_ID |

|  |  |
| --- | --- |
| Käsitteen nimi: | Artisti |
| Nimilyhenne: |  |
| Määritelmä: | Tapahtumassa esiintyvä henkilö/yhtye. |
| Ominaisuudet: | ARTISTI\_ID PUH ERITYISTOIVEET NIMI KATUOSOITE POSTINRO |
| Pääavain: | ARTISTI\_ID |

|  |  |
| --- | --- |
| Käsitteen nimi: | Varaus |
| Nimilyhenne: |  |
| Määritelmä: | Asiakkaan tekemä varaus lipusta. |
| Ominaisuudet: | VARAUS\_ID STATUS LKM TAPAHTUMA |
| Pääavain: | varaus\_id |

# Looginen mallinnus

## Normalisointi & tarkistettu Relaatiokaavio

## Eheysmäärittelyt

**VARAUS** (varaus\_id(PK)(int), status(*nimetty ‘tilanne’ tietokannassa*)(char 10), lkm(integer 100), tapahtuma\_ID(FK)(varchar 100)

**TAPAHTUMA** (tapahtuma\_id(PK)(varchar 100), status (*nimetty ‘tilanne’ tietokannassa*) (char 10), tyyppi(char 50), nimi(char 50), hinta(integer 10), paikka\_id(FK)(varchar 10))

**PAIKKA** (paikka\_id(PK)(varchar 10), katuosoite(varchar 50), postinro(FK)(integer 5), paikka(char 50), lkm(integer 100))

**ESIINTYMINEN** (tapahtuma\_id(PK)(varchar 100), artisti\_id(PK)(varchar 100))

**ARTISTI** (artisti\_id(PK)(varchar 100), nimi(char 50), katuosoite(varchar 50), postinro(FK)(integer 5), erityistoiveet(char 100), puh(integer 15))

**POSTITMP**(postinro(PK)(integer 5), postitmp(char 25))

# Toteutus

## SQL-luontilauseet

/\*Tässä luodaan taulut tietokantaan\*/

*CREATE TABLE VARAUS*

*(varaus\_id int,*

*tilanne char(10),*

*lkm int,*

*tapahtuma\_id varchar(100),*

*primary key (varaus\_id));*

*CREATE TABLE ARTISTI*

*(artisti\_id varchar(100),*

*nimi char(50),*

*katuosoite varchar(50),*

*postinro varchar(5),*

*erityistoiveet char(100),*

*puh int,*

*primary key (artisti\_id));*

*CREATE TABLE PAIKKA*

*(paikka\_id varchar(10),*

*katuosoite varchar(50),*

*paikka char(50),*

*postinro varchar(5),*

*lkm int,*

*primary key (paikka\_id));*

*CREATE TABLE POSTITMP*

*(postinro varchar(5),*

*postitmp char(25)*

*primary key (postinro));*

*CREATE TABLE TAPAHTUMA*

*(tapahtuma\_id varchar(100),tilanne char(10), tyyppi char(50), nimi char(50),*

*hinta decimal(5,2), paikka\_id varchar(10), pvm date*

*primary key (tapahtuma\_id));*

*CREATE TABLE ESIINTYMINEN*

*(tapahtuma\_id varchar (100), artisti\_id varchar(100)*

*primary key (tapahtuma\_id, artisti\_id));*

*/\*Tässä luodaan viiteavaimet tauluihin\*/*

*ALTER TABLE VARAUS*

*ADD FOREIGN KEY (tapahtuma\_id) REFERENCES TAPAHTUMA(tapahtuma\_id);*

*ALTER TABLE TAPAHTUMA*

*ADD FOREIGN KEY (paikka\_id) REFERENCES PAIKKA(paikka\_id);*

*ALTER TABLE PAIKKA*

*ADD FOREIGN KEY (postinro) REFERENCES POSTITMP(postinro);*

*ALTER TABLE ARTISTI*

*ADD FOREIGN KEY (postinro) REFERENCES POSTITMP(postinro);*

*/\*Tässä luodaan sequence joka tuottaa automaattisesti varaus\_Id:n\*/*

*CREATE SEQUENCE S\_NUM*

*START WITH 1*

*INCREMENT BY 1;*

# Testaus

## Testidatan syöttö

*/\*Tässä syötetään data tauluihin\*/*

*INSERT INTO POSTITMP*

*(postinro, postitmp)*

*values ('01700', 'Vantaa');*

*INSERT INTO POSTITMP*

*(postinro, postitmp)*

*values ('04200', 'Kerava');*

*INSERT INTO POSTITMP*

*(postinro, postitmp)*

*values ('01800', 'Nurmijärvi');*

*INSERT INTO POSTITMP*

*(postinro, postitmp)*

*values ('34300', 'Kuru');*

*INSERT INTO POSTITMP*

*(postinro, postitmp)*

*values ('03100', 'Lohja');*

*INSERT INTO POSTITMP*

*(postinro, postitmp)*

*values ('04150', 'Sipoo');*

*INSERT INTO POSTITMP*

*(postinro, postitmp)*

*values ('04400', 'Järvenpää');*

*INSERT INTO ARTISTI*

*(artisti\_id, nimi, katuosoite, postinro, erityistoiveet, puh)*

*values ('anske', 'Anssi Kela', 'Ryhmätie 5', '03100', 'kalaa ja kaljaa', '05033344');*

*INSERT INTO ARTISTI*

*(artisti\_id, nimi, katuosoite, postinro, erityistoiveet, puh)*

*values ('anttu', 'Antti Tuisku', 'Kariniementie 4', '04150', 'banaaneja', '0501233211');*

*INSERT INTO ARTISTI*

*(artisti\_id, nimi, katuosoite, postinro, erityistoiveet, puh)*

*values ('sirk', 'Sirkus Finlandia', 'Pöytäalhontie 72', '04400', 'juomia', '0401233211');*

*INSERT INTO ARTISTI*

*(artisti\_id, nimi, katuosoite, postinro, erityistoiveet, puh)*

*values ('arko', 'Arja Koriseva', 'Lipputie 10', '01700', 'Ruusuja', '050132447');*

*INSERT INTO ARTISTI*

*(artisti\_id, nimi, katuosoite, postinro, erityistoiveet, puh)*

*values ('arjra', 'Arja Raatikainen', 'Koskitie 8', '01800', 'Pyyhkeitä', '045167852');*

*INSERT INTO ARTISTI*

*(artisti\_id, nimi, katuosoite, postinro, erityistoiveet, puh)*

*values ('shado', 'Shadowplay', 'Ritvatie 5', '04200', 'Juotavaa ja syötävää', '010167852');*

*INSERT INTO PAIKKA*

*(paikka\_id, katuosoite, postinro, paikka, lkm)*

*values ('1', 'Kurunkatu 8', '34300', 'pieni', 50);*

*INSERT INTO PAIKKA*

*(paikka\_id, katuosoite, postinro, paikka, lkm)*

*values ('2', 'Kurunkuja 2', '34300', 'suuri', 500);*

*INSERT INTO TAPAHTUMA*

*(tapahtuma\_id, tilanne, tyyppi, nimi,*

*hinta, paikka\_id, pvm)*

*VALUES('Mus101','Tulossa', 'Musiikkiesitys', 'Umpikela', 29.90, '1', '2019-5-18');*

*INSERT INTO TAPAHTUMA*

*(tapahtuma\_id, tilanne, tyyppi, nimi,*

*hinta, paikka\_id, pvm)*

*VALUES('Mus102','Tulossa', 'Musiikkiesitys', 'Tuiske', 49.90, '2', '2019-8-5');*

*INSERT INTO TAPAHTUMA*

*(tapahtuma\_id, tilanne, tyyppi, nimi,*

*hinta, paikka\_id, pvm)*

*VALUES('Tah101','Pidetty', 'Viihde-esitys', 'Finlandia', 69.90, '2', '2019-4-15');*

*INSERT INTO TAPAHTUMA*

*(tapahtuma\_id, tilanne, tyyppi, nimi,*

*hinta, paikka\_id, pvm)*

*VALUES('Tah102','Tulossa', 'Tanssi-esitys', 'Finlandia', 59.90, '2', '2019-5-15');*

*INSERT INTO TAPAHTUMA*

*(tapahtuma\_id, tilanne, tyyppi, nimi,*

*hinta, paikka\_id, pvm)*

*VALUES('Tah103', 'Pidetty', 'Tanssi-esitys', 'Ego, 39.90, '1', '2010-10-10');*

*INSERT INTO TAPAHTUMA*

*(tapahtuma\_id, tilanne, tyyppi, nimi,*

*hinta, paikka\_id, pvm)*

*VALUES('Mus103', 'Tulossa', 'Musiikkiesitys', 'Shadowplay', 79.90, '2', '2019-10-10');*

*INSERT INTO ESIINTYMINEN(tapahtuma\_id, artisti\_id)*

*VALUES('Mus101', 'anske');*

*INSERT INTO ESIINTYMINEN(tapahtuma\_id, artisti\_id)*

*VALUES('Mus102', 'anttu');*

*INSERT INTO ESIINTYMINEN(tapahtuma\_id, artisti\_id)*

*VALUES('Tah101', 'sirk');*

*INSERT INTO ESIINTYMINEN(tapahtuma\_id, artisti\_id)*

*VALUES('Tah102', 'sirk');*

*INSERT INTO ESIINTYMINEN(tapahtuma\_id, artisti\_id)*

*VALUES('Tah103', 'arjra');*

*INSERT INTO ESIINTYMINEN(tapahtuma\_id, artisti\_id)*

*VALUES('Mus103', 'shado');*

*INSERT INTO VARAUS*

*(varaus\_id, tilanne, lkm, tapahtuma\_id)*

*VALUES(NEXT VALUE FOR S\_NUM, 'Myyty', 50, 'Mus101');*

*INSERT INTO VARAUS*

*(varaus\_id, tilanne, lkm, tapahtuma\_id)*

*VALUES(NEXT VALUE FOR S\_NUM, 'Myyty', 500, 'Tah101');*

*INSERT INTO VARAUS*

*(varaus\_id, tilanne, lkm, tapahtuma\_id)*

*VALUES(NEXT VALUE FOR S\_NUM, 'Myyty', 23, 'Mus103');*

*INSERT INTO VARAUS*

*(varaus\_id, tilanne, lkm, tapahtuma\_id)*

*VALUES(NEXT VALUE FOR S\_NUM, 'Myyty', 23, 'Tah103');*

*INSERT INTO VARAUS*

*(varaus\_id, tilanne, lkm, tapahtuma\_id)*

*values (NEXT VALUE FOR S\_NUM, 'Varattu', 2, 'Mus102');*

*INSERT INTO VARAUS*

*(varaus\_id, tilanne, lkm, tapahtuma\_id)*

*values (NEXT VALUE FOR S\_NUM, 'Varattu', 10, 'Mus103');*

*INSERT INTO VARAUS*

*(varaus\_id, tilanne, lkm, tapahtuma\_id)*

*values (NEXT VALUE FOR S\_NUM, 'Myyty', 20, 'Mus103');*

*INSERT INTO VARAUS*

*(varaus\_id, tilanne, lkm, tapahtuma\_id)*

*values (NEXT VALUE FOR S\_NUM, 'Myyty', 50, 'Mus103');*

*INSERT INTO VARAUS*

*(varaus\_id, tilanne, lkm, tapahtuma\_id)*

*values (NEXT VALUE FOR S\_NUM, 'Varaus', 40, 'Mus103');*

*INSERT INTO VARAUS*

*(varaus\_id, tilanne, lkm, tapahtuma\_id)*

*values (NEXT VALUE FOR S\_NUM, 'myyty', 3, 'Tah103');*

## Testilauseet ja testitulokset

*/\*@test Mitä esityksiä on tarjolla tämän kuukauden aikana? \*/*

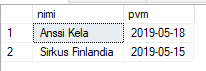
*SELECT a.nimi, pvm*

*FROM ARTISTI a*

*JOIN ESIINTYMINEN e on e.artisti\_id = a.artisti\_id*

*JOIN TAPAHTUMA t on t.tapahtuma\_id = e.tapahtuma\_id*

*Where month(pvm) = month(getDate());*

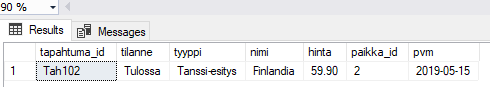


*/\*@test Mitä tanssiesityksiä on tarjolla tämän kuukauden aikana? \*/*

*SELECT \* FROM TAPAHTUMA*

*WHERE tyyppi ='Tanssi-esitys' AND*

*month(pvm) = month(getDate());*

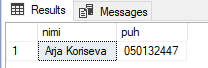


*/\*@test Mikä on Arja Korisevan puhelinnumero ?\*/*

*SELECT nimi, puh*

*FROM artisti*

*Where nimi = 'Arja Koriseva';*



*/\*@test Montako lippua on lunastettu Arja Raatikaisen tanssiesitykseen Ego 10.10.2010 ?\*/*

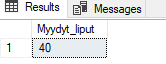
*Select sum(v.lkm) AS Myydyt\_liput*

*From VARAUS v*

*JOIN TAPAHTUMA t ON v.tapahtuma\_id = t.tapahtuma\_id*

*WHERE v.tilanne = 'Myyty'*

*AND t.nimi = 'Ego';*



*/\*@test Onko Shadowplayn konserttiin 20.10.2019 vielä paikkoja? \*/*

*/\*@test Montako paikkaa em. konserttiin on jäljellä? \*/*

*SELECT (p.lkm - sum(v.lkm)) as jäljellä*

*FROM VARAUS v*

*JOIN TAPAHTUMA t ON v.tapahtuma\_id = t.tapahtuma\_id*

*JOIN PAIKKA p ON p.paikka\_id = t.paikka\_id*

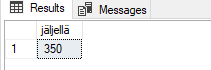
*JOIN ESIINTYMINEN e ON e.tapahtuma\_id = t.tapahtuma\_id*

*JOIN ARTISTI a ON a.artisti\_id = e.artisti\_id*

*WHERE a.nimi = 'Shadowplay'*

*AND pvm = '2019-10-10'*

*GROUP BY p.lkm;*

*\*Muokattiin Shadowplay-konsertin pvm vuoteen 2019*

*/\*@test Mikä on lipputulojen määrä kuluvana vuonna? \*/*

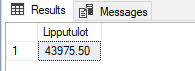
*SELECT sum (v.lkm \* hinta) AS Lipputulot*

*FROM tapahtuma t*

*JOIN varaus v on t.tapahtuma\_id = v.tapahtuma\_id*

*WHERE v.tilanne = 'Myyty'*

*AND year(pvm) = '2019';*



# Loppuarvio ja työajan käyttö

**Miten projekti eteni? Miten projektityöskentely sujui?**

Lopputyö eteni ryhmän kesken hyvin. Työ oli tarpeeksi haastava ja oli mukavaa soveltaa opittua tietoa käytännössä. Projekti oli opettavaista juurikin tästä syystä.

Pystyimme ryhmän kesken jakamaan eri alueita osa-alueita projektissa sekä vaihtamaan rooleja, jotta jokainen pääsi tekemään ns. kaikkea. Ryhmässä oli helppo sopia aikataulutuksesta ja työtehtävistä, jolloin työskentelystä ei todellakaan tullut ”pakkopullaa”. Jokainen projektin jäsen oli motivoitunut, jonka huomasi, kun työskentelimme yhdessä.

Olemme tyytyväisiä työn tulokseen. Saimme ryhmänä tehtyä toimivan SQL-tietokannan ja siihen liittyvät käsitteelliset sekä loogiset mallinnukset.

**Minkälaisia ongelmia tuli vastaan ja miten niistä selvittiin?**

Suurimpia ongelmia tuli alkuun sovellusalueen kokonaisuuden ymmärtämisessä. Lähdimme työskentelemään luokka- ja relaatiokaavioita tiedoilla, jotka olivat tarpeettomia tai triviaaleja. Opettajan ohjaus ja opastus auttoivat meitä hahmottamaan kokonaisuuden paremmin, jolloin sovellusalue hahmottui paremmin.

Ongelmia ilmeni SQL-lauseita tehdessä, kun eheysmäärittelyssä ilmoitetut tyypit eivät toimineetkaan SQL-serverissä oikein. Lauseita tehdessä ei muuten suurempia ongelmia ilmennyt, vain pieniä pysähdyksiä jotka pystyttiin ratkomaan lähes välittömästi.

**Käytetty aika**

Teimme projektia aina yhdessä jokaisen keskittyvän tiettyyn projektin kohdealueeseen. Tehdyt työtunnit jakautuivat näin tasan ja jokaiselle kertyi tehtyjä tunteja n. 7h. Yhteensä työaikaa kirjautui näin ollen n. 21h.