

DATA.DB.100-2024-2025-1 Tietokantojen peruste...

33.3% Completed12 / 36

Contents

tables

Teema 2. Oppimateriaalit suomeksi

Theme 2. Study materials in English

T2_Q1_

T2_Q2

T2_Q3

T2_Q4

T2_Q5

T2_Q6

T2_Q7

T2_Q8

T2_Q9

T2_Q10

T2_Q11

T2_Q12

Dashboard / My courses / DATA.DB.100-2024-2025-1-TAU / 2. Useita tauluja - Multiple tables

DATA.DB.100-2024-2025-1 Tietokantojen perusteet (Syksyn toteutus (FI/EN))

CourseParticipantsGrades

2. Useita tauluja - Multiple tables

Oppimateriaalit - Study materials

Teema 2. Oppimateriaalit suomeksi

Theme 2. Study materials in English

Teeman 2 harjoitustehtävät T2 - Theme 2 exercise tasks T2

Suosittelutakaraja: 9.9.2024 klo 23:59 / Lopullinentakaraja: 16.9.2024 klo 23:59

Recommended deadline: 9.9.2024 at 23:59 / Final deadline: 16.9.2024 at 23:59

T2-harjoitustehtävien suositeltu tehtäväpisteiden minimimäärä: 8 pistettä

The recommended minimum number of exercise points for T2 exercise tasks: 8 points

Tehtävissä T2_Q1-T2_Q11 tarvitaan kuvitteellisen ja yksinkertaistetun perunatietokannan tauluja

Taulujen luontilauseet ja tietojen lisäyslauseet on annettu tiedostossa potato.sql. Kannattaa luoda potato.db-tietokanta ja taulut SQLitella ja testata kyselyjä sillä.

Tietokannan tila on esitetty tiedostossa potato_database_state.pdf.

Tehtävät T2_Q1 ja T2_Q11 ovat monivalintakysymyksiä.

Tehtävässä T2_Q2 piirretään perunatietokannan kaavion graafinen esitys. SQL-tietokannan kaavion graafinen esitys on selitetty Viiteavain, viite-ehys -kalvopakettissa. Katso myös piirto-ohje sivulta Kurssin työkalujen ohjeita.

Tehtävät T2_Q3-T2_Q10 ovat kyselytehtäviä. Kiinnitä kyselyissä huomiota seuraaviin asioihin:

- liitosehto
- liitosoperaatio ja liitosehto
- taulun nimen käyttö sarakkeen nimen tarkenteena niin SELECT-, WHERE- kuin ORDER BY -osassa, jos kahdessa tai useammassa kyselyn FROM-osassa esiintyvässä taulussa on samanniminen sarakke
- Huom. CodeGrade käyttää tällä hetkellä SQLiten vanhempaa versiota, jossa on toteutettu vain LEFT OUTER JOIN, ei RIGHT and FULL OUTER JOIN. Käytä siis LEFT OUTER JOINia niissä tehtävissä, joissa tarvitaan ulkoliitosta. Selvitämme, voimmeko saada myöhemmin kurssilla CodeGradessa käyttöön SQLiten myöhemmän version, jossa on toteutettu myös muut ulkoliitokset. (4.9.)

Tehtävässä T2_Q12 kirjoitetaan taulujen luonti- ja tietojen lisäyslauseita.

In the tasks T2_Q1-T2_Q11 you need tables of the fictitious and simplified potato database.

Create table statements and data insertion statements are given in the file potato.sql. It is good to create the potato.db database and the tables by SQLite and test the queries with it.

The state of the database is presented in the file potato_database_state.pdf.

Tasks T2_Q1 ja T2_Q11 are multiple choice questions.

In the task T2_Q2 a graphical representation of the potato database schema is drawn. The graphical representation of an SQL database schema is explained in the slide set Foreign key, referential integrity slide set. See also drawing instructions on the page Instructions for course tools.

Tasks T2_Q3-T2_Q10 are query tasks. In the queries, pay attention to the following:

- join condition
- join operation and join condition
- using table names as qualifier for column name in SELECT, WHERE and ORDER BY clauses, if more than one table given in the FROM clause have a column with the same name.
- N.B. CodeGrade currently uses an older version of SQLite with only LEFT OUTER JOIN implemented, not RIGHT and FULL OUTER JOIN. Therefore, use LEFT OUTER JOIN in those tasks where an outer join is required. We will find out if we can use a later version of SQL in CodeGrade later in the course, where other outer joins have also been implemented. (4.9.)

In the task T2_Q12 table creation and data insertion statements are written.

T2_Q1_

To dot: Receive a grade

To dot: Receive a passing grade

1 p

Tämä on perunatietokannan taulujen rajoitteisiin ja tilaan (katso tiedostot potato.sql ja potato_database_state.pdf yllä olevassa ohjeistuksessa) sekä tauluihin tehtäviin kyselyihin liittyvä monivalintakysymys. Tehtävässä on 6 alkukysymystä.

This is a multiple choice question related to the constraints and the state of the potato database tables (please see the potato.sql and potato_database_state.pdf in the instructions above) as well as queries made to the tables. The task has 6 sub-questions.

T2_Q2

To dot: Receive a grade

To dot: Receive a passing grade

2 p

Tarkastellaan yllä annettuja perunatietokannan taulujen luontilauseita. Piirrä perunatietokannan kaavion graafinen esitys.

Let's look at the create table statements of the potato database given above. Draw a graphical presentation of the potato database schema.

Ratkaisudokumentin nimi - Solution document name: t2_q2.*

Ratkaisudokumentin tyytit - Solution document types: *.jpeg, *.jpg, *.png, *.ppt, *.pptx, *.pdf

Diagrams.net:

Vie (File - Export as) kaaviot *.png, *.jpeg tai *.jpg -tyyppisiksi.

Export (File - Export as) diagrams as *.png, *.jpeg or *.jpg.

T2_Q3

To dot: Receive a grade

To dot: Receive a passing grade

1 p

Hae kaikille perunankasvattajille (breeder) seuraavat tiedot: tunnus, nimi ja agentin nimi. Lajittele tulostaulun rivit kasvattajan tunnuksen perusteella nousevaan järjestykseen.

Retrieve the following information for all potato breeders: id, name, and agent's name. Sort the result rows in ascending order by breeder's id.

Tulosarakeet - Result columns: breeder_id | breeder_name | agent_name

Ratkaisudokumentti - Solution document: t2_q3.sql

Huom. Palautathan ratkaisudokumentissasi vain tekemäsi SQL-lauseen eli SELECT-lauseen, ei SQLiten .open -komentoa, joka avaa tietokannan. Tämä ohje pätee myös muihin tehtäviin, joissa pitää kirjoittaa SQL-lauseita.

N.B. In your solution document, please submit only the SQL statement you made, i.e. the SELECT statement, not the SQLite .open command that opens the database.

Completion Progress

NOW

Mouse over or touch bar for info.

Completion Progress

NOW

Mouse over or touch bar for info.

This instructions also applies to other tasks where you need to write SQL statements.



T2_Q4

To do: Receive a grade

To do: Receive a passing grade

1 p

Hae seuraavat tiedot perunankasvattajille, joiden maa (country) on Finland: tunnus, nimi ja agentin nimi. Lajittele tulostaulun rivit kasvattajan tunnuksen perusteella nousevaan järjestykseen.

Retrieve the following information for potato breeders, whose country is Finland: id, name, and agent's name. Sort the result rows in ascending order by breeder's id.

Tulossarakkeet - Result columns:
breeder_id | breeder_name | agent_name

Ratkaisudokumentti - Solution document: [t2_q4.sql](#)



T2_Q5

To do: Receive a grade

To do: Receive a passing grade

2 p

Hae seuraavat tiedot kaikille agenteille: agentin tunnus ja nimi sekä agentin edustaman perunankasvattajan tunnus ja nimi. Tulostaulussa tulee olla mukana myös ne agentit, joilla ei ole perunankasvattajia tietokannassa. Lajittele tulostaulun rivit agentin tunnuksen ja perunankasvattajan tunnuksen perusteella nousevaan järjestykseen.

Retrieve the following information for all agents: agent's ID and name as well as IDs and names of the potato breeders represented by the agent. The result table must also include those agents who do not have any potato breeders in the database. Sort the result rows in ascending order by agent ID and breeder ID.

Tulossarakkeet - Result columns:
agent_id | agent_name | breeder_id | breeder_name

Ratkaisudokumentti - Solution document: [t2_q5.sql](#)



T2_Q6

To do: Receive a grade

To do: Receive a passing grade

2 p

Hae perunalajikkeiden nimet ja niitä kasvattavien kasvattajien nimet. Lajittele tulostaulun rivit perunalajikkeen nimen ja kasvattajan nimen perusteella nousevaan järjestykseen.

Retrieve the names of potato varieties and the names of the breeders breeding them. Sort the result rows in ascending order by potato variety name and breeder name.

Tulossarakkeet - Result columns:
potato_name | breeder_name

Ratkaisudokumentti - Solution document: [t2_q6.sql](#)



T2_Q7

To do: Receive a grade

To do: Receive a passing grade

2 p

Hae kaikkien perunalajikkeiden nimet ja niitä kasvattavien kasvattajien nimet. Tulostaulussa tulee olla mukana myös ne perunalajikkeet, joita mikään kasvattaja ei kasvata. Lajittele tulostaulun rivit perunalajikkeen nimen ja kasvattajan nimen perusteella nousevaan järjestykseen.

Retrieve the names of all potato varieties and the names of the breeders breeding them. The result table must also include the potato varieties that are not bred by any breeder. Sort the result rows in ascending order by potato variety name and breeder name.

Tulossarakkeet - Result columns:
potato_name | breeder_name

Ratkaisudokumentti - Solution document: [t2_q7.sql](#)



T2_Q8

To do: Receive a grade

To do: Receive a passing grade

2 p

Hae niiden perunankasvattajien nimet, jotka kasvattavat Dunbar Rover -nimistä perunalajiketta. Lajittele tulostaulun rivit perunankasvattajan nimen perusteella nousevaan järjestykseen.

Retrieve the names for those potato breeders who breed the potato variety named Dunbar Rover. Sort the result rows in ascending order by the breeder name.

Tulossarakkeet - Result columns:
breeder_name

Ratkaisudokumentti - Solution document: [t2_q8.sql](#)



T2_Q9

To do: Receive a grade

To do: Receive a passing grade

2 p

Hae tunnuksot ja nimet niille perunalajikkeille, joita ainakin yksi perunankasvattaja kasvattaa. Kunkin lajikkeen tunnuksen ja nimen tulee esiintyä tulostaulussa vain kerran. Lajittele tulostaulun rivit perunalajikkeen tunnuksen perusteella nousevaan järjestykseen.

Retrieve IDs and names for potato varieties that are bred at least by one breeder. The ID and name of each variety should appear only once in the result table. Sort the result rows in ascending order by potato variety ID.

Tulossarakkeet - Result columns:
potato_id | potato_name

Ratkaisudokumentti - Solution document: [t2_q9.sql](#)

Vinkki - Tip



T2_Q10

To do: Receive a grade

To do: Receive a passing grade

2 p

Hae tunnuksot ja nimet niille perunalajikkeille, joita mikään perunankasvattaja ei kasvata. Lajittele tulostaulun rivit perunalajikkeen tunnuksen perusteella nousevaan järjestykseen.

Retrieve IDs and names for potato varieties that are not bred by any breeder. Sort the result rows in ascending order by potato variety ID.

Tulossarakkeet - Result columns:
potato_id | potato_name

Ratkaisudokumentti - Solution document: [t2_q10.sql](#)

Vinkki - Tip



T2_Q11

To do: Receive a grade

To do: Receive a passing grade

1 p

Tämä on INSERT INTO-, UPDATE- ja DELETE-lauseisiin ja perunatietokannan taulujen rajoitteisiin ja tilaan liittyvä monivalintakysymys, jossa on 9 alkusymystä.

This is a multiple choice question related to the INSERT INTO, UPDATE and DELETE statements and the constraints and state of the tables of the potato database, with 9 sub-questions.



T2_Q12

To do: Receive a grade

To do: Receive a passing grade

2 p

Edellisen viikon harjoituksissa käytettiin bike-nimistä taulua.

Avaa ensin SQLite:ssä tietokanta, joka sisältää bike-tilin, tai luo bike-tili ja lisää siihen tietoja.

Laista seuraavaksi SQLite:ssä viite-eheyden valvonta päälle antamalla komento PRAGMA foreign_keys = ON;

Tee luontilause bike_shop-nimiselle taululle, johon voidaan tallentaa tietoja pyöräkaupoista. Taulussa on seuraavat sarakkeet:

- Taulun pääavain on id-sarake, jossa ei sallita puuttuvia arvoja (NULL-arvoja).
- name-sarake on yksilöivä vaihtuvamittainen merkkijono ja siinä ei sallita puuttuvia arvoja.
- address on vaihtuvamittainen merkkijono

Luo taulu SQLite:llä. Lisää tauluun muutaman pyöräkaupan tiedot: keksi pyöräkauppojen tunnus-, nimi- ja osoitetiedot itse.

Tee luontilause sells-tilille, joka yhdistää pyörät ja pyöräkaupat toisiinsa: yhdessä pyöräkaupassa voidaan myydä useaa eri pyörää ja samaa pyörää voidaan myydä useassa eri pyöräkaupassa. Määrittele sells-tiliin viiteavainsarakkeet, jotka viittaavat bike-tiliin ja bike_shop-tiliin sekä pääavain. Sisällytä viiteavainsarakkeiden nimiin niiden taulujen nimet, joihin viiteavainsarakkeet viittaavat.

Luo taulu SQLite:llä ja lisää siihen muutama rivi tietoja.

Huom. Määrittele pääavaimet, avaimet (yksilöivä, unique) ja viiteavaimet luontilauseen lopussa oppimateriaalien esimerkkien tapaan automaattisen tarkastuksen vuoksi.

Palauta ratkaisudokumentissa [t2_q12.sql](#) bike_shop- ja sells-tilien luontilauseet ja tietojen lisäyslauseet.

In the previous week's exercises we used a table called bike.

In SQLite, first open the database containing the bike table or create the bike table and add data to it.

Next, in SQLite, turn the referential integrity control on by entering the command PRAGMA foreign_keys = ON;

Make a creation statement for a table called bike_shop, where information about bike shops can be stored. The table has the following columns:

- The primary key of the table is the id column, which does not allow missing values (NULL values).
- The name column is a unique variable-length string and does not allow missing values.
- address is a variable-length character string

Create the table with SQLite. Insert data of a few bike shops to the table; you can make up the id, name and address data of the bike shops.

Make a creation statement for a table sells that connects bikes and bike shops: Several different bikes can be sold in one bike shop and the same bike can be sold in several different bike shops. Define foreign key columns for the sells table that refer to the bike table and bike_shop table, as well as the primary key. In the foreign key column names, include the names of the tables that the foreign key columns refer to.

Create the table with SQLite and add a few rows of data to it.

N.B. Specify the primary key, keys (unique) and foreign keys at the end of the creation statement, as in the examples in study materials, for automatic evaluation.

In the solution document [t2_q12.sql](#), submit the creation statements of the bike_shop and sells tables and the data insertion statements.

You are logged in as [Shamsur Raza Chowdhury](#) (Log out)

[Policies](#)

[Dashboard](#)