

## T6 Sisäkkäisiä kyselyjä - Nested queries

```
INSERT INTO breeds VALUES (4,4, 30);
```

### T6\_Q0

1 p

Tämä on tehtävä, jossa hyödynnetään ER-kaavion ja SQL-tietokannan kaavion hyödyntämiseen kyselyä kirjoitettaessa.

Kirjoita kysely, joka hakee seuraavat tiedot kaikille rypälelajikkeille (grape), joiden väri on sininen (blue): Rypälelajikkeen id ja nimi ja rypälelajiketta sisältävän viinin (wine) id, nimi ja viinitilan (winery) nimi.

This is a task where ER diagram and SQL database diagram are utilized when writing a query.

Write a query to retrieve the following information for all grape varieties (grape) of blue color: ID and name of the grape variety and id, name and winery name of the wine containing the grape variety.

```
SELECT grape.id, grape.name, wine.id, wine.name, winery.name
FROM grape
  LEFT OUTER JOIN contains ON grape.id = contains.grape_id
  LEFT OUTER JOIN wine ON contains.wine_id = wine.id
  LEFT OUTER JOIN winery ON wine.winery_id = winery.id
WHERE color = 'blue ';
```

Huomaa ER-kaaviossa grape-entiteettityypin osittainen osallistuminen contains-suhdetyyppiin.

In the ER diagram, notice the partial participation of the grape entity type in the contains relationship type

### T6\_Q1

1 p

Hae tunnukset ja nimet agenteille, jotka edustavat yhtä tai useampaa perunankasvattajaa. Käytä IN-operaattoria.

Lajittele tulosrivit nousevaan järjestykseen agentin tunnuksen perusteella.

Retrieve IDs and names for agents that represent one or more potato breeders. Use the IN operator. Sort the result rows in ascending order by agent ID.

Tulossarakkeet – Result columns:

agent\_id | agent\_name

```
SELECT agent_id, agent_name
FROM agent
WHERE agent_id IN (SELECT agent_id
FROM breeder)
ORDER BY agent_id;
```

agent_id	agent_name
1	Potatis Institutet
2	Agrico
3	Perunatukku

## T6\_Q2

1 p

Hae tunnuksset ja nimet agenteille, jotka eivät edusta mitään perunankasvattajaa. Käytä NOT IN -operaattoria.

Lajittele tulokset nousevaan järjestykseen agentin tunnuksen perusteella.

Retrieve IDs and names for agents that do not represent any potato breeders. Use the NOT IN operator.

Sort the result rows in ascending order by agent ID.

Tulossarakkeet – Result columns:

agent\_id | agent\_name

```
SELECT agent_id, agent_name
FROM agent
WHERE agent_id NOT IN (SELECT agent_id
FROM breeder)
ORDER BY agent_id;
```

agent_id	agent_name
4	Kartoffelzentrum

## T6\_Q3

1 p

Hae tunnuksset ja nimet agenteille, jotka edustavat yhtä tai useampaa perunankasvattajaa. Käytä EXISTS-operaattoria.

Lajittele tulokset nousevaan järjestykseen agentin tunnuksen perusteella.

Retrieve IDs and names for agents that represent one or more potato breeders. Use the EXISTS operator.

Sort the result rows in ascending order by agent ID.

Tulossarakkeet – Result columns:

agent\_id | agent\_name

```
SELECT agent_id, agent_name
FROM agent
WHERE EXISTS (SELECT *
              FROM breeder
              WHERE agent.agent_id = breeder.agent_id)
ORDER BY agent_id;
```

agent_id	agent_name
1	Potatis Institutet
2	Agrico
3	Perunatukku

#### T6\_Q4

1 p

Hae tunnukset ja nimet agenteille, jotka eivät edusta mitään perunankasvattajaa. Käytä NOT EXISTS -operaattoria.

Lajittele tulosrivit nousevaan järjestykseen agentin tunnuksen perusteella.

Retrieve IDs and names for agents that do not represent any potato breeders. Use the NOT EXISTS operator.

Sort the result rows in ascending order by agent ID.

Tulossarakkeet – Result columns:

agent\_id | agent\_name

```
SELECT agent_id, agent_name
FROM agent
WHERE NOT EXISTS (SELECT *
                  FROM breeder
                  WHERE agent.agent_id = breeder.agent_id)
ORDER BY agent_id;
```

agent_id	agent_name
4	Kartoffelzentrum

#### T6\_Q5

2 p

Hae tunnukset ja nimet niille perunankasvattajille, jotka kasvattavat sekä Dunbar Rover että Cara -nimisiä perunalajikkeita.

Lajittele tulosrivit kasvattajan tunnuksen perusteella nousevaan järjestykseen.

Retrieve IDs and names for those potato breeders who breed both potato varieties named Dunbar Rover and Cara. Sort the result rows in ascending order by breeder ID.

Tulossarakkeet – Result columns:

breeder\_id | breeder\_name

```
SELECT breeder_id, breeder_name
FROM breeder
WHERE breeder_id IN
    (SELECT breeder_id
     FROM breeds, potato
     WHERE breeds.potato_id = potato.potato_id AND
     potato_name = 'Dunbar Rover')
AND breeder_id IN
    (SELECT breeder_id
     FROM breeds, potato
     WHERE breeds.potato_id = potato.potato_id AND
     potato_name = 'Cara')
ORDER BY breeder_id;
```

breeder_id	breeder_name
1	Chips

#### T6\_Q6

1 p

Hae tunnukset ja nimet niille perunankasvattajille, jotka kasvattavat Dunbar Rover -nimistä perunalajiketta mutta eivät kasvata Cara -nimistä perunalajiketta.

Lajittele tulossrivit kasvattajan tunnuksen perusteella nousevaan järjestykseen.

Retrieve IDs and names for those potato breeders who breed the potato variety named Dunbar Rover but do not breed the potato variety named Cara.

Sort the result rows in ascending order by breeder ID.

Tulossarakkeet – Result columns:

breeder\_id | breeder\_name

```
SELECT breeder_id, breeder_name
FROM breeder
WHERE breeder_id IN
    (SELECT breeder_id
     FROM breeds, potato
     WHERE breeds.potato_id = potato.potato_id AND
     potato_name = 'Dunbar Rover')
AND breeder_id NOT IN
    (SELECT breeder_id
     FROM breeds, potato
     WHERE breeds.potato_id = potato.potato_id AND
     potato_name = 'Cara')
ORDER BY breeder_id;
```

breeder_id	breeder_name
4	Penan pottu

### T6\_Q7

2 p

Hae nimi ja esittelyvuosi perunalajikkeelle, jolla on myöhäisin esittelyvuosi.

Retrieve the name and introduction year of the potato variety having the latest introduction year.

Tulossarakkeet – Result columns:

potato\_name | introduction\_year

```
SELECT potato_name, introduction_year
FROM potato
WHERE introduction_year = (SELECT MAX(introduction_year)
                           FROM potato);
```

potato_name	introduction_year
Blue Annelise	2007

### T6\_Q8

1 p

Hae nimi, esittelyvuosi ja kasvattajan nimi perunalajikkeelle, jolla on myöhäisin esittelyvuosi.

Retrieve the name, introduction year and name of the breeder for the potato variety with the latest introduction year.

Tulossarakkeet – Result columns:

potato\_name | introduction\_year | breeder\_name

```
SELECT potato_name, introduction_year, breeder_name
FROM potato, breeds, breeder
WHERE potato.potato_id = breeds.potato_id AND
breeds.breeder_id = breeder.breeder_id AND
introduction_year = (SELECT MAX(introduction_year)
                     FROM potato);
```

potato_name	introduction_year	breeder_name
Blue Annelise	2007	Sipsix

### T6\_Q9

2 p

Hae nimet perunankasvattajille, joiden perunalajikkeiden vuosittaisten kasvatusmäärien (tonnes\_per\_year)

keskiarvo on suurempi kuin kaikkien vuosittaisten kasvatusmäärien keskiarvo.

Lajittele tulosrivit kasvattajan nimen perusteella nousevaan järjestykseen.

Retrieve the names for potato breeders having the average of annual breeding quantities (tonnes\_per\_year) of potato varieties higher than the average of all annual breeding quantities. Sort the result rows in ascending order by breeder name.

Tulossarakkeet – Result columns:  
breeder\_name

```
SELECT breeder_name
FROM breeder, breeds
WHERE breeder.breeder_id = breeds.breeder_id
GROUP BY breeder_name
HAVING AVG(tonnes_per_year) > (SELECT AVG(tonnes_per_year)
                                FROM breeds);
```

```
breeder_name
-----
Chips
Kartoffelmeister
```

Kyselyn osien tuloksia:

```
SELECT AVG(tonnes_per_year)
FROM breeds;
```

```
AVG(tonnes_per_year)
-----
61.5
```

```
SELECT breeder_name, AVG(tonnes_per_year)
FROM breeder, breeds
WHERE breeder.breeder_id = breeds.breeder_id
GROUP BY breeder_name;
```

```
breeder_name      AVG(tonnes_per_year)
-----
Chips              75.33333333333333
Kartoffelmeister  134.0
Penan pottu        26.0
Sipsix             40.0
```

## T6\_Q10

2 p

Hae kullekin agentille tunnus, nimi, perunankasvattajien lukumäärä ja kasvattajien kasvattamien eri perunalajikkeiden lukumäärä.

Lajittele tulosrivit nousevaan järjestykseen agentin tunnuksen perusteella.

Retrieve for each agent ID, name, number of potato breeders and number of different potato varieties bred by breeders.

Sort the result rows in ascending order based on the agent ID.

Tulossarakkeet – Result columns:

agent\_id | agent\_name | breeders | potatoes

```
SELECT agent.agent_id,
       agent_name,
       COUNT(DISTINCT breeder.breeder_id) AS breeders,
       COUNT(DISTINCT potato_id) AS potatoes
FROM agent
     LEFT OUTER JOIN breeder ON agent.agent_id = breeder.agent_id
     LEFT OUTER JOIN breeds ON breeder.breeder_id = breeds.breeder_id
GROUP BY agent.agent_id, agent_name
ORDER BY agent.agent_id;
```

agent_id	agent_name	breeders	potatoes
1	Potatis Institutet	1	3
2	Agrico	2	2
3	Perunatukku	1	2
4	Kartoffelzentrum	0	0

## T6\_Q11

2 p

Hae kullekin perunankasvattajalle seuraavat tiedot:

tunnus,

nimi,

agentin nimi ja

agentin edustamien perunankasvattajien lukumäärä.

Lajittele tulost rivit perunankasvattajan tunnuksen perusteella nousevaan järjestykseen.

Retrieve the following information for each potato breeder:

ID,

name,

agent name, and

number of potato breeders represented by the agent.

Sort the result rows by the potato breeder id in ascending order.

Tulostaulu - Result table:

breeder_id	breeder_name	agent_name	number_of_breeders
1	Chips	Potatis Institutet	1
2	Sipsix	Perunatukku	1
3	Kartoffelmeister	Agrico	2
4	Penan pottu	Agrico	2

```

SELECT breeder_id, breeder_name, agent_name, number_of_breeders
FROM breeder, agent,
    (SELECT agent_id, COUNT(*) AS number_of_breeders
     FROM breeder
     GROUP BY agent_id) AS stats
WHERE breeder.agent_id = agent.agent_id AND
      agent.agent_id = stats.agent_id
ORDER BY breeder_id;

```

breeder_id	breeder_name	agent_name	number_of_breeders
1	Chips	Potatis Institutet	1
2	Sipsix	Perunatukku	1
3	Kartoffelmeister	Agrico	2
4	Penan pottu	Agrico	2

## T6\_Q12

1 p

Hae kaikki perunalajikkeiden nimet, jotka sisältävät merkkijonon "li".

Lajittele tulokset perunalajikkeen nimen mukaan nousevaan järjestykseen.

Retrieve all potato variety names that contain the string "li".

Sort the result rows in ascending order by potato variety name.

Tulossarakkeet – Result columns:

potato\_name

```

SELECT potato_name
FROM potato
WHERE potato_name LIKE '%li%'
ORDER BY potato_name;

```

potato_name
Blue Annelise
Moonlight

## T6\_Q13

1 p

Hae nimet niille perunalajikkeille, joiden esittelyvuosi on 2000 ja 2010 välillä. Hyödynnä BETWEEN-operaattoria.

Lajittele tulokset perunalajikkeen nimen mukaan nousevaan järjestykseen.

Retrieve the names for potato varieties with introduction year between 2000 and 2010. Use the BETWEEN operator.

Sort the result rows in ascending order by potato variety name.

Tulossarakkeet – Result columns:

potato\_name



```
SELECT potato_name
FROM potato
WHERE introduction_year BETWEEN 2000 AND 2010
ORDER BY potato_name;
```

```
potato_name
-----
Bambino
Blue Annelise
Moonlight
```

#### T6\_Q14

1 p

Hae Chips-nimiselle perunankasvattajalle kunkin kasvatettavan perunalajikkeen nimi ja vuosittainen kasvatusmäärä 10 % korotettuna.

Lajittele tulosrivit perunalajikkeen nimen mukaan nousevaan järjestykseen.

For the potato breeder named Chips, retrieve the name of each potato variety bred and its yearly breeding quantity increased by 10%.

Sort the result rows in ascending order by potato variety name.

```
SELECT potato_name, 1.1 * tonnes_per_year AS increased_tonnes
FROM potato, breeds, breeder
WHERE potato.potato_id = breeds.potato_id AND
      breeds.breeder_id = breeder.breeder_id AND
      breeder_name = 'Chips'
ORDER BY potato_name;
```

```
potato_name    increased_tonnes
-----
Cara           83.6
Duke of York   88.0
Dunbar Rover   77.0
```