

Testi 1

Tietokantaan on luotu seuraava taulu, joka on aluksi tyhjä:

```
CREATE TABLE Testi (x INTEGER);
```

Tee ohjelma, joka toistaa 5000 kertaa seuraavan operaation:

1. Hae `SELECT`-kyselyllä taulun `Testi` suurin arvo `x`
2. Lisää tauluun `Testi` uusi rivi, jonka arvona on `x+1`
3. Tulosta lisätyn rivin arvo `x`

Testaa ensin, että ohjelma toimii: kun ohjelma suoritetaan ja taulu `Testi` on aluksi tyhjä, tauluun ilmestyy 5000 riviä, joilla on arvot 1, 2, 3, ..., 5000.

Tee sitten testi, jossa taulu on taas alussa tyhjä ja käynnistät samaan aikaan kaksi ohjelmaa, jotka molemmat lisäävät rinnakkain 5000 riviä tauluun.

Mitä havaitset? Montako riviä taulussa on lopuksi ja mikä on suurin arvo `x` rivillä?

- The table execution took long time from start till end

real 0m56.013s

user 0m0.015s

sys 0m0.000s

- There are 5000 rows in the table

- The largest value of `x` row is 4999

Testi 2

Toista testi 1 niin, että taulu onkin määriteltä näin:

```
CREATE TABLE Testi (x INTEGER UNIQUE);
```

Toteuta rivin lisääminen niin, että lisäämistä yritetään uudestaan (tarvittaessa useita kertoja), jos lisääminen epäonnistuu `UNIQUE`-ehdon takia.

Mitä havaitset? Montako riviä taulussa on lopuksi ja mikä on suurin arvo `x` rivillä?

- The table execution took less time than the Test 1

real 0m29.581s

user 0m0.000s

sys 0m0.015s

- When executing 2 programs at the same time, `UNIQUE` constraints failed. Therefore, the Test 2 will run with only one program

- There are 5000 rows in the table

- The largest value of `x` row is 4999

Testi 3

Toista testi 1 niin, että komennot `SELECT` ja `INSERT` suoritetaan transaktion sisällä (eli alussa suoritetaan `BEGIN` ja lopussa `COMMIT`).

Toteuta ohjelma niin, että transaktiota yritetään uudestaan (tarvittaessa useita kertoja), jos komento epäonnistuu. Suorita välissä komento `ROLLBACK`, jotta transaktio ei jää jumiin.

Mitä havaitset? Montako riviä taulussa on lopuksi ja mikä on suurin arvo x rivillä?

- The execution of the test 3 is the fastest, compared to Test 1 and Test 2

real 0m0.250s

user 0m0.000s

sys 0m0.015s

- There are 5000 rows in the table

- The largest value of x row is 4999

CODE

```
#TEHTÄVÄ 1:

import sqlite3
import random
from random import randint
import os
os.system("rm tehtava4.db")
db = sqlite3.connect("tehtava4.db")
db.isolation_level = None

# Testi 1
db.execute('CREATE TABLE Testi1 (x integer)')

#Ohjelma 1
n=5000

for i in range(n):
    suurin = db.execute('SELECT MAX(x) FROM Testi1').fetchone()
    suurin = suurin[0]
    if suurin==None:
        suurin = 0

    db.execute('INSERT INTO Testi1 (x) VALUES (?)', [suurin+1])
    print('Suurin arvo x rivillä', suurin)

#Ohjelma 2
db.execute('DELETE FROM Testi1')
for i in range(n):
    db.execute('INSERT INTO Testi1 (x) VALUES (?)', [random.randint(1,
5000)])

rivit = db.execute('Select count(*) from Testi1').fetchone()
print('Rivien taulussa on lopuksi', rivit[0])

-----
-----
```

#TEHTÄVÄ 2:

```
import sqlite3
import random
from random import randint
import os
os.system("rm tehtava4.db")
db = sqlite3.connect("tehtava4.db")
db.isolation_level = None

db.execute('CREATE TABLE Testi2 (x integer UNIQUE)')

n=5000

for i in range(n):
    suurin = db.execute('SELECT MAX(x) FROM Testi2').fetchone()
    suurin = suurin[0]
    if suurin==None:
        suurin = 0

    db.execute('INSERT INTO Testi2 (x) VALUES (?)', [suurin+1])
    print('Surrin arvo x rivilla', suurin)

rivit = db.execute('Select count(*) from Testi2').fetchone()
print('Rivia taulussa on lopuksi', rivit[0])
```


#TEHTÄVÄ 3:

```
import sqlite3
import random
from random import randint
import os
os.system("rm tehtava4.db")
db = sqlite3.connect("tehtava4.db")
db.isolation_level = None

db.execute('CREATE TABLE Testi3 (x integer UNIQUE)')

n=5000
db.execute('BEGIN')
for i in range(n):
    suurin = db.execute('SELECT MAX(x) FROM Testi3').fetchone()
    suurin = suurin[0]
    if suurin==None:
        suurin = 0

    db.execute('INSERT INTO Testi3 (x) VALUES (?)', [suurin+1])
    print('Surrin arvo x rivilla', suurin)
db.execute('COMMIT')
```

```
rivit = db.execute('Select count(*) from Testi3').fetchone()
print('Rivia taulussa on lopuksi',rivit[0])
```