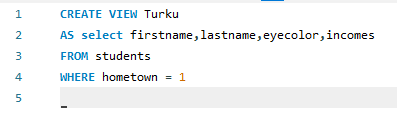
# HARJOITUS 7 0/15p

## TEHT 1 2/2p

A . Tee **näkymä** Turku, joka hakee students-taulusta kaikki turkulaiset opiskelijat. Sarakkeista tulostetaan mukaan järjestyksessä seuraavat: firstname, lastname, eyecolor ja incomes. Testaa näkymän toiminta kokeilemalla sitä.



Execute:

> SELECT \* FROM Turku

+ -------------- + ------------- + ------------- + ------------ +

| firstname | lastname | eyecolor | incomes |

+ -------------- + ------------- + ------------- + ------------ +

| Ken | Guru | Ruskea | 12010.12 |

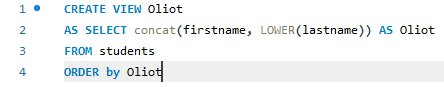
| Tino | Saurus | Ruskea | 14010.22 |

| Sini | Tiainen | Sininen | 16010.32 |

+ -------------- + ------------- + ------------- + ------------ +

3 rows

B. Tee **näkymä** Oliot, joka tulostaa kaikki students-taulun opiskelijoiden nimet yhdessä sarakkeessa muodossa Etunimisukunimi. Esim. Kenguru, Tinosaurus, jne. Vain ensimmäinen merkki voi tulostua isolla (suuraakkonen, versaali). Sarakkeen otsikko tulee olla myös **Oliot**. Näkymää käytettäessä oliot tulee tulostaa aakkosjärjestyksessä. Testaa näkymän toiminta kokeilemalla sitä. Vastauksessa tulee olla näkyvillä näkymän luonti ja sen testaaminen.



Execute:

> SELECT \* FROM Oliot

+ ---------- +

| Oliot |

+ ---------- +

| Elovainio |

| Kenguru |

| Kimalainen |

| Muurahainen |

| Ruutana |

| Sinitiainen |

| Tinosaurus |

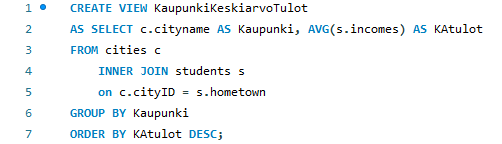
| Viljavainio |

+ ---------- +

8 rows

## Tehtävä 2 4/4P

A. **[2p]** Tee **näkymä** KaupunkiKeskiarvoTulot joka tulostaa opiskelijoiden keskiarvotulot kaupungeittain. Tulostuksessa näytetään kunkin kaupungin nimi ja kaupunkikohtaiset keskiarvotulot ja sarakkeiden otsikot tulee olla Kaupunki ja KAtulot. Kotikaupungittomia ei oteta mukaan tähän näkymään. Näkymää käytettäessä kaupungit tulee lajitella keskiarvotulon perusteella laskevassa järjestyksessä. Testaa näkymän toiminta kokeilemalla sitä. Vastauksessa tulee olla näkyvillä näkymän luonti ja sen testaaminen.



Execute:

> SELECT \* FROM KaupunkiKeskiarvoTulot

+ ------------- + ------------ +

| Kaupunki | KAtulot |

+ ------------- + ------------ +

| Tampere | 15510.120000 |

| Turku | 14010.220000 |

| Lahti | 0.000000 |

+ ------------- + ------------ +

3 rows

B. **[2p]** Tee **näkymä** Stipendiehdokkaat jonka avulla voidaan tulostaa 4 parasta opintosuoritusta. Riveittäin näkyy opiskelijan sukunimi, etunimi, kurssi ja kurssiarvosana, kuten alla mallissa. Arvosanan lisäksi lajitteluehdot ovat järjestyksessä kurssin nimi (ASC), sukunimi (ASC) ja etunimi(ASC). Testaa näkymän toiminta kokeilemalla sitä. Vastauksessa tulee olla näkyvillä näkymän luonti ja sen testaaminen.

**CREATE VIEW Stipendiehdokkaat**

**AS SELECT s.lastname, s.firstname, c.coursename, g.grade**

**FROM students s**

**INNER JOIN studentgrades g**

**ON s.studentID = g.studentID**

**INNER JOIN courses c**

**ON g.courseID = c.courseID**

**ORDER BY grade DESC, coursename, lastname, firstname LIMIT 4;**

**Execute:**

**> SELECT \* FROM Stipendiehdokkaat**

**+ ------------- + -------------- + --------------- + ---------- +**

**| lastname | firstname | coursename | grade |**

**+ ------------- + -------------- + --------------- + ---------- +**

**| Guru | Ken | Ohjelmointi | 5 |**

**| Guru | Ken | Ruotsi | 5 |**

**| Ana | Ruut | Tietokannat | 5 |**

**| Guru | Ken | Tietokannat | 5 |**

**+ ------------- + -------------- + --------------- + ---------- +**

**4 rows**

## TEHTÄVÄ 3 3/3P

1. Hae ne opiskelijat (suku- ja etunimellä), joilla on sama kotikunta kuin Ken Gurulla. Ken Gurun kotikuntaa ei tiedetä kyselyä kirjoitettaessa (se voi olla mikä tahansa). Käytä alikyselyä.

Execute:

> SELECT firstname, lastname

FROM students

WHERE hometown =

(SELECT hometown FROM students WHERE concat(firstname, lastname)='KenGuru')

+ -------------- + ------------- +

| firstname | lastname |

+ -------------- + ------------- +

| Ken | Guru |

| Tino | Saurus |

| Sini | Tiainen |

+ -------------- + ------------- +

3 rows

1. Hae pienituloisimmat opiskelijat. Pienintä palkkaa ei tiedetä kyselyä kirjoitettaessa. Tulossarakkeina suku- ja etunimi sekä tulot. Käytä alikyselyä.

Execute:

> SELECT lastname, firstname, incomes

FROM students

WHERE incomes =

(SELECT MIN(incomes) FROM students)

+ ------------- + -------------- + ------------ +

| lastname | firstname | incomes |

+ ------------- + -------------- + ------------ +

| Vainio | Vilja | 0.00 |

| Vainio | Elo | 0.00 |

+ ------------- + -------------- + ------------ +

2 rows

1. Hae opiskelijan 2007 (= studentID) suoritettujen opintojaksojen nimet. Käytä alikyselyä.

SELECT coursename

FROM courses

WHERE courseID IN

(SELECT courseID FROM studentgrades WHERE studentID='2007')

Execute:

> SELECT coursename

FROM courses

WHERE courseID IN

(SELECT courseID FROM studentgrades WHERE studentID='2007')

+ --------------- +

| coursename |

+ --------------- +

| Ohjelmointi |

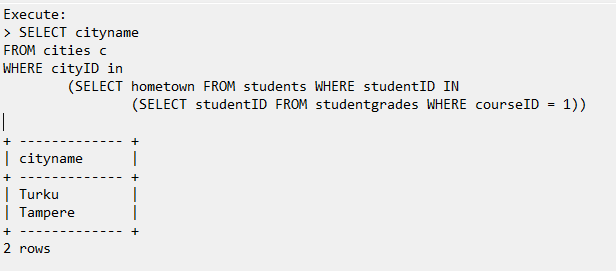
| Tietokannat |

+ --------------- +

2 rows

## TEHTÄVÄ 4 4/4P

1. **[2p]** Hae niiden kaupunkien nimet, joiden opiskelijoista ainakin yksi on suorittanut **Ohjelmointi**-kurssin.



1. **[2p]** Hae opiskelijan 2007 (= studentID) suoritettujen opintojaksojen nimet käyttäen EXISTS-alikyselyä.

Execute:

> SELECT coursename

FROM courses c

WHERE EXISTS

(SELECT courseID FROM studentgrades

WHERE courseID = c.courseID AND studentID = '2007')

+ --------------- +

| coursename |

+ --------------- +

| Ohjelmointi |

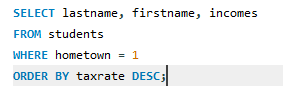
| Tietokannat |

+ --------------- +

2 rows

## TEHTÄVÄ 5 2/2P

Luo nk. 3 tähden paksu indeksi SQL-kyselylle:



Vastaus :

CREATE INDEX i\_turkulaiset\_tuloineen

ON students(hometown,taxrate, lastname, firstname, incomes);