

Εξέταση 102 – Μάθημα 3

105.3 Διαχείριση δεδομένων SQL



Σχεσιακά Συστήματα Διαχείρισης Βάσεων Δεδομένων (RDBMS) - SQL

- Χρησιμοποιώντας Βάσεις Δεδομένων έχουμε την δυνατότητα για πιο δομημένη αποθήκευση των δεδομένων
- Για το λόγο αυτό χρησιμοποιούνται τα Σχεσιακά Συστήματα Βάσεων Διαχείρισης Δεδομένων (Relational Database Management Systems – RDBMS)
- Η πιο δημοφιλής γλώσσα για RDBMS είναι η SQL
- Βάσεις Δεδομένων που πληρούν τις πιο πάνω προϋποθέσεις σε Linux είναι η MySQL, η PostgreSQL, η SQLite κ.α.



Εγκατάσταση MySQL

- Η **MySQL** είναι το πιο δημοφιλές σύστημα βάσης δεδομένων σε Linux, λόγω της ευχρηστίας και ταχύτητας της.
- Η **MySQL** αποτελεί μέρος του ευρύτερου συστήματος LAMP (Linux Apache MySQL PHP)
- **# apt-get -y install mysql-server #** σε Debian (αφήστε κενό όταν σας ζητήσει κωδικό)
- **# yum -y install mysql-server #** σε RedHat



Διαχείριση και Σύνδεση με MySQL

- `# /etc/init.d/mysql start|stop|restart|reload|status`
διαχείριση «δαίμονα» `mysqld` σε Debian
 - `# /etc/rc.d/init.d/mysqld start|stop|restart|reload|status`
διαχείριση «δαίμονα» `mysqld` σε RedHat
 - Αρχεία ρυθμίσεων:
`/etc/my.cnf` # σε RedHat και άλλες διανομές
`/etc/mysql/my.cnf` # σε Debian
 - `# mysql` # σύνδεση σε `mysqld` σαν `root` χωρίς χρήση κωδικού (δεν συστήνεται)
- 

Διαχείριση και Σύνδεση με MySQL

- `# mysql dbname #` επιλογή βάσης `dbname` με την σύνδεση σε `mysqld` χωρίς να χρειαστεί να εκτελεστεί το ερώτημα `USE dbname`
- `$ mysql -uroot -p -hlocalhost #` πιο γενική μορφή της εντολής `mysql` με καθορισμό χρήστη δικτυακό όνομα και διαδραστικό κωδικό
- `$ mysql dbname -uduser -pdpass -h192.168.3.6 #` κωδικός = `dpass`

Επιλογές:

- u**: καθορισμός χρήστη βάσης. Αν δεν καθοριστεί χρησιμοποιεί το όνομα χρήστη του συστήματος που καλεί την εντολή
- h**: καθορισμός δικτυακής διεύθυνσης ή ονόματος υπολογιστή. Αν δεν οριστεί επιλέγεται ο τοπικός υπολογιστής (`localhost`)
- p**: καθορισμός κωδικού. Ο κωδικός μπορεί να καθοριστεί στην γραμμή εντολών (δεν συστήνεται) ή αν δεν καθοριστεί θα ζητηθεί διαδραστικά.

Σήμα ετοιμότητας (prompt) MySQL

- Μετά την σύνδεση με `mysql` θα δείτε το prompt:
`mysql>`
- Η MySQL έχει το δικό της κέλυφος με το πιο πάνω σήμα ετοιμότητας όπου μπορείτε να τρέξετε εντολές της γλώσσας **SQL**
- Για έξοδο από το κέλυφος `mysql>` μπορείτε να εκτελέσετε την εντολή `\q` ή `quit`




Τύποι δεδομένων MySQL

Τύποι	Περιγραφή
INTEGER	Ακέραιοι αριθμοί
FLOAT	Αριθμοί κινητής υποδιαστολής (δεκαδικοί)
BOOLEAN	Λογικές τιμές. Το "0" συμβολίζει ψευδής τιμές (false) και οτιδήποτε διαφορετικό του "0" συμβολίζει αληθής τιμές (true)
CHAR	Συμβολοσειρές με σταθερό μήκος μέχρι 255 χαρακτήρες
VARCHAR	Συμβολοσειρές με μεταβλητό μήκος
DATE	Ημερομηνίες της μορφής "XXXX-MM-HH" (YYYY-MM-DD)
DATETIME	Συνδιασμός ημερομηνίας και ώρας
BLOB	Δυαδικά δεδομένα (binary)
TEXT	Κείμενο μέχρι 65535 χαρακτήρες



Δημιουργία καινούργιας βάσης δεδομένων με *CREATE DATABASE*

- Οι εντολές που εκτελούνται στο κέλυφος `mysql` ονομάζονται και ερωτήματα (*queries*)
 - `# mysql #` εισαγωγή στο κέλυφος `mysql`. Το κέλυφος `mysql` έχει ιστορικό εντολών και δουλεύει με παρόμοιο τρόπο όπως και το `bash`
 - `mysql> CREATE DATABASE school; #` Το `”;` είναι υποχρεωτικό για να τερματιστούν τα ερωτήματα
 - `mysql> SHOW DATABASES; #` προβολή λίστας με τις βάσεις δεδομένων
 - `mysql> USE school; #` επιλογή βάσης δεδομένων με το όνομα `school`, για χρήση
- 

Δημιουργία πινάκων με *CREATE TABLE*

- **mysql> CREATE TABLE students (# δημιουργία πίνακα**
 - **id INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,**
 - **name VARCHAR(100),**
 - **surname VARCHAR(100),**
 - **birth_date DATE,**
 - **math_score FLOAT,**
 - **modern_greek_score FLOAT,**
 - **suspensions INTEGER UNSIGNED,**
 - **conduct VARCHAR(100),**
 - **other_info TEXT,**
 - **PRIMARY KEY (id) # ρυθμίζει το id ώστε να είναι μοναδικό**
 - **);**
- **mysql> SHOW TABLES; # προβολή όλων των πινάκων στην school**

Προβολή πληροφοριών πίνακα με *DESCRIBE*

- `mysql> DESCRIBE students;`

Field	Type	Null	Key	Default	Extra
id	int(10) unsigned	NO	PRI	NULL	auto_increment
name	varchar(100)	YES		NULL	
surname	varchar(100)	YES		NULL	
father	varchar(100)	YES		NULL	
mother	varchar(100)	YES		NULL	
birth_date	date	YES		NULL	
math_score	float	YES		NULL	
modern_greek_score	float	YES		NULL	
suspensions	int(10) unsigned	YES		NULL	
conduct	varchar(100)	YES		NULL	
other_info	text	YES		NULL	

11 rows in set (0.04 sec)



Εισαγωγή δεδομένων με *INSERT*

- `mysql> INSERT INTO students # Εισαγωγή νέας εγγραφής`
 - `(name, surname, birth_date, math_score,`
 - `modern_greek_score, suspensions, conduct, other_info)`
 - `VALUES`
 - `("Rikos", "Mapouros", "1989-06-23", "18.5",`
 - `"16.0", "2", "Perfect", "Football Player");`
- `mysql> SELECT * FROM students; # προβολή όλων των`
δεδομένων. Η πρώτη γραμμή καθορίζει τα πεδία (fields) και οι υπόλοιπες τις εγγραφές (records)
- `mysql> SELECT id, name, surname, birth_date FROM students;`

```
+-----+-----+-----+-----+
| id | name | surname | birth_date |
+-----+-----+-----+-----+
| 1 | Rikos | Mapouros | 1989-06-23 |
+-----+-----+-----+-----+
1 rows in set (0.00 sec)
```

Εισαγωγή δεδομένων με *INSERT*

- `mysql> INSERT INTO students SET # άλλος τρόπος με SET`
 - `name = "Takis" , surname = "Svistos",`
 - `birth_date = "1990-03-02", math_score = "15.5",`
 - `modern_greek_score = "16.7", suspensions = "4",`
 - `conduct = "Good", other_info = "Member of the school`
 - `orchestra";`
- `mysql> SELECT id, name, surname, birth_date FROM students;`

id	name	surname	birth_date
1	Rikos	Mapouros	1989-06-23
2	Takis	Svistos	1990-03-02

2 rows in set (0.00 sec)



Τροποποίηση δεδομένων με *UPDATE*

- `mysql> UPDATE students SET # τροποποίηση εγγραφής
birth_date = "1989-06-22"
WHERE id = "1"; # αν δεν ορίσετε το WHERE θα αλλάξουν
όλες οι τιμές του πεδίου birth_date!`
- `mysql> SELECT id, name, surname, birth_date FROM students;`

```
+-----+-----+-----+-----+
| id | name | surname | birth_date |
+-----+-----+-----+-----+
| 1 | Rikos | Mapouros | 1989-06-22 |
| 2 | Takis | Svistos | 1990-03-02 |
+-----+-----+-----+-----+
2 row in set (0.00 sec)
```



Διαγραφή εγγραφών με *DELETE*

- `mysql> DELETE FROM students`
 - `WHERE name = "Takis" AND surname = "Svistos";` # **Αν**
δεν μπει το WHERE θα διαγραφούν όλες οι εγγραφές του πίνακα!
- `mysql> SELECT id, name, surname, birth_date FROM students;`

id	name	surname	birth_date
1	Rikos	Mapouros	1989-06-22

1 row in set (0.00 sec)



Τροποποίηση πίνακα με *ALTER*

- `mysql> ALTER TABLE students # τροποποίηση πίνακα students`
 - `ADD COLUMN father VARCHAR(100) AFTER surname,`
 - `ADD COLUMN mother VARCHAR(100) AFTER father;`
- `mysql> INSERT INTO students (name, surname, father, mother,`
 - `birth_date, math_score, modern_greek_score, suspensions,`
 - `conduct, other_info) VALUES ("Maricou", "Lefkara",`
 - `"Kotsios", "Pinou", "1991-12-06", "17.6", "19.0", "1",`
 - `"Perfect", "President of Class");`
- `mysql> SELECT id, name, surname, father, mother FROM`
`students;`


```
+-----+-----+-----+-----+-----+
| id | name   | surname | father | mother |
+-----+-----+-----+-----+-----+
|  1 | Rikos  | Mapouros | NULL   | NULL   |
|  3 | Maricou | Lefkara  | Kotsios | Pinou  |
+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

Τροποποίηση πίνακα με *ALTER*

- `mysql> UPDATE students SET
→ father = "Yiannis", mother = "Tallou"
→ WHERE id = "1";`
- `mysql> SELECT id, name, surname, father, mother FROM
students;`

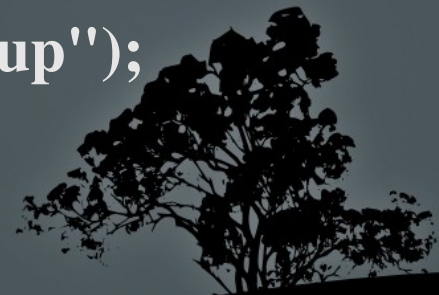
```
+-----+-----+-----+-----+
| id | name   | surname | father | mother |
+-----+-----+-----+-----+
|  1 | Rikos  | Mapouros | Yiannis | Tallou |
|  3 | Maricou | Lefkara | Kotsios | Pinou  |
+-----+-----+-----+-----+
2 rows in set (0.01 sec)
```

Σημείωση: όταν διαγραφεί μια εγγραφή το **ΠΡΩΤΕΥΩΝ ΚΛΕΙΔΙ** (**PRIMARY KEY**) της δεν χρησιμοποιείται ξανά γι' αυτό και η επόμενη εγγραφή ξεκινά με **id = "3"**



Προσθήκη επιπρόσθετων εγγραφών

- `mysql> INSERT INTO students SET name = "Takis", surname
→ = "Svistos", father = "Pampos", mother = "Ellou",
→ birth_date = "1990-03-22", math_score = "15.5",
→ modern_greek_score = "16.7", suspensions = "4", conduct
→ = "Good", other_info = "Member of the school
→ Orchestra";`
- `mysql> INSERT INTO students (name, surname, father, mother,
→ birth_date, math_score, modern_greek_score, suspensions,
→ conduct, other_info) VALUES ("Aglaia", "Pitsiou",
→ "Mitsos", "Euterpi", "1989-07-11", "18.7", "16.6", "0",
→ "Perfect", "Dancer in the school's group");`



Προβολή Εγγραφών με *SELECT* και φίλτράρισμα με *WHERE*

- mysql> **SELECT** id, name, surname, suspensions, conduct
FROM students;

```
+-----+-----+-----+-----+-----+
| id | name      | surname | suspensions | conduct |
+-----+-----+-----+-----+-----+
|  1 | Rikos    | Mapouros |           2 | Perfect |
|  3 | Maricou  | Lefkara  |           1 | Perfect |
|  4 | Takis    | Svistos  |           4 | Good    |
|  5 | Aglaia   | Pitsiou  |           0 | Perfect |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

- mysql> **SELECT** id, name, surname, suspensions, conduct
FROM students **WHERE** conduct = "Perfect";

```
+-----+-----+-----+-----+-----+
| id | name      | surname | suspensions | conduct |
+-----+-----+-----+-----+-----+
|  1 | Rikos    | Mapouros |           2 | Perfect |
|  3 | Maricou  | Lefkara  |           1 | Perfect |
|  5 | Aglaia   | Pitsiou  |           0 | Perfect |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```


Ομαδοποίηση με *GROUP BY*

- `mysql> SELECT count(conduct), conduct
FROM students GROUP BY conduct;`

count (conduct)	conduct
1	Good
3	Perfect

2 rows in set (0.00 sec)



Δημιουργία καινούργιου πίνακα με *CREATE TABLE*

- `mysql> CREATE TABLE books (
→ id INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
→ title VARCHAR(200) NOT NULL,
→ author VARCHAR(100),
→ student_id INTEGER UNSIGNED,
→ PRIMARY KEY (id));`
- `mysql> SHOW TABLES;`

```
+-----+  
| Tables_in_school |  
+-----+  
| books            |  
| students         |  
+-----+  
2 rows in set (0.00 sec)
```



Προβολή πληροφοριών καινούργιου πίνακα με *DESCRIBE*

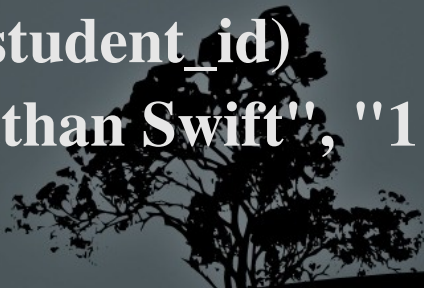
▪ `mysql> DESCRIBE books;`

Field	Type	Null	Key	Default	Extra
id	int(10) unsigned	NO	PRI	NULL	auto_increment
title	varchar(200)	NO		NULL	
author	varchar(100)	YES		NULL	
student_id	int(10) unsigned	YES		NULL	

4 rows in set (0.00 sec)



Εισαγωγή εγγραφών στο καινούργιο πίνακα με *INSERT*

- `mysql> INSERT INTO books (title, author, student_id)`
→ `VALUES ("Little Brother", "Cory Doctorow", "4");`
4 = Takis
 - `mysql> INSERT INTO books (title, author, student_id)`
→ `VALUES ("Alice's Adventures in Wonderland",`
→ `"Lewis Carroll", "5");` # 5 = Aglaia
 - `mysql> INSERT INTO books (title, author, student_id)`
→ `VALUES ("The Little Prince",`
→ `"Antoine de Saint-Exupéry ", "5");` # 5 = Aglaia
 - `mysql> INSERT INTO books (title, author, student_id)`
→ `VALUES ("Gulliver's Travels", "Jonathan Swift", "1");`
1 = Rikos
- 

Ταξινόμηση αποτελεσμάτων με *ORDER BY*

▪ `mysql> SELECT * FROM books;`

id	title	author	student_id
1	Little Brother	Cory Doctorow	4
2	Alice's Adventures in Wonderland	Lewis Carroll	5
3	The Little Prince	Antoine de Saint-Exupéry	5
4	Gulliver's Travels	Jonathan Swift	1

4 rows in set (0.00 sec)

▪ `mysql> SELECT * FROM books ORDER BY title;`

id	title	author	student_id
2	Alice's Adventures in Wonderland	Lewis Carroll	5
4	Gulliver's Travels	Jonathan Swift	1
1	Little Brother	Cory Doctorow	4
3	The Little Prince	Antoine de Saint-Exupéry	5

4 rows in set (0.00 sec)



Ερωτήματα σε πολλαπλούς πίνακες με *SELECT*

- `mysql> SELECT students.id, students.name, students.surname,
→ books.title, books.author FROM students, books WHERE
→ students.id = books.student_id;`

id	name	surname	title	author
1	Rikos	Mapouros	Gulliver's Travels	Jonathan Swift
4	Takis	Svistos	Little Brother	Cory Doctorow
5	Aglaia	Pitsiou	Alice's Adventures in Wonderland	Lewis Carroll
5	Aglaia	Pitsiou	The Little Prince	Antoine de Saint-Exupéry

4 rows in set (0.00 sec)

- `mysql> SELECT s.id, s.name, s.surname, b.title, b.author
→ FROM students s, books b WHERE s.id = b.student_id;`
Η πιο πάνω εντολή πετυχαίνει το ίδιο αποτέλεσμα με την πρώτη αλλά με λιγότερους χαρακτήρες. Το `students` αντικαθίσταται με "s" και το `books` με "b"

Ερωτήματα σε πολλαπλούς πίνακες με (*INNER*) JOIN

- `mysql> SELECT students.id, students.name, students.surname,
→ books.title, books.author FROM students INNER JOIN
→ books ON students.id = books.student_id; # Το αποτέλεσμα
αυτής της εντολής είναι το ίδιο με την προηγούμενη
διαφάνεια αλλά το JOIN θεωρείται καλύτερη μέθοδος για
συγχωνεύσεις αποτελεσμάτων. Το κλειδί INNER είναι
προαιρετικό`

id	name	surname	title	author
1	Rikos	Mapouros	Gulliver's Travels	Jonathan Swift
4	Takis	Svistos	Little Brother	Cory Doctorow
5	Aglaia	Pitsiou	Alice's Adventures in Wonderland	Lewis Carroll
5	Aglaia	Pitsiou	The Little Prince	Antoine de Saint-Exupéry

4 rows in set (0.00 sec)



Ερωτήματα σε πολλαπλούς πίνακες με *LEFT JOIN*

- `mysql> SELECT students.id, students.name, students.surname,
→ books.title, books.author FROM students LEFT JOIN
→ books ON students.id = books.student_id; # Η διαφορά του
LEFT JOIN από το INNER JOIN είναι ότι το πρώτο,
προβάλλει και τα αποτελέσματα που δεν έχουν αντίστοιχο
στο πίνακα books`

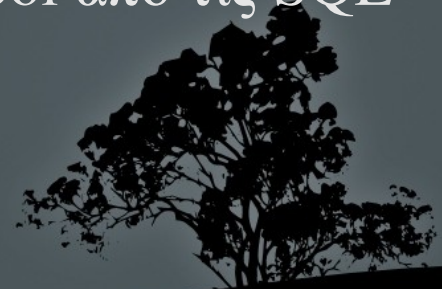
id	name	surname	title	author
1	Rikos	Mapouros	Gulliver's Travels	Jonathan Swift
3	Maricou	Lefkara	NULL	NULL
4	Takis	Svistos	Little Brother	Cory Doctorow
5	Aglaia	Pitsiou	Alice's Adventures in Wonderland	Lewis Carroll
5	Aglaia	Pitsiou	The Little Prince	Antoine de Saint-Exupéry

5 rows in set (0.13 sec)



Άλλες χρήσιμες πληροφορίες

- `$ echo "SELECT * FROM students" | mysql school -uroot -p`
Μπορούμε να περνούμε τα ερωτήματα μέσω αγωγών από το κέλυφος στη `mysql`
- `$ mysqldump school -uroot -p > school-bak.sql` # φύλαξη των δεδομένων και της δομής της βάσης `school`, με την μορφή ερωτημάτων SQL, στο αρχείο κειμένου `school-bak.sql`.
Συστήνεται να δημιουργείτε αντίγραφα ασφαλείας (backups) των βάσεων πριν αρχίσετε να δουλεύετε με αυτές
- `$ mysql school -uroot -p < school-bak.sql` # επαναφορά των δεδομένων και της δομής της βάσης `school` από τις SQL εντολές στο αρχείο `school-bak.sql`



Εργαστήριο 3

Ξεκινήστε και τις δύο εικονικές μηχανές και συνδεθείτε σαν "root"

- `# apt-get install mysql-server #` σε Debian. Μη βάλετε κωδικό όταν ερωτηθείτε σχετικά
- `# yum install mysql-server #` σε RedHat
- `# /etc/rc.d/init.d/mysqld start #` στα RedHat τα daemons δεν ξεκινούν μετά την εγκατάσταση
- `# mysql`
- `mysql> SHOW DATABASES;`
- `mysql> \q`
- `mysql> CREATE DATABASE school;`
- `mysql> SHOW DATABASES;`
- `mysql> USE school;`
- `mysql> CREATE TABLE students (`
 - `id INTEGER UNSIGNED`
 - `NOT NULL`
 - `AUTO_INCREMENT,`
 - `name VARCHAR(100),`
 - `surname VARCHAR(100),`
 - `birth_date DATE,`
 - `math_score FLOAT,`
 - `modern_greek_score FLOAT,`
 - `suspensions INTEGER`
 - `UNSIGNED,`
 - `conduct VARCHAR(100),`
 - `other_info TEXT,`
 - `PRIMARY KEY (id)`
 - `);`
- `mysql> SHOW TABLES;`

Εργαστήριο 3

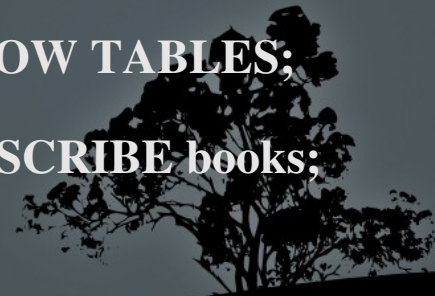
- `mysql> DESCRIBE students;`
- `mysql> quit`
- `# mysql school`
- `mysql> SHOW TABLES;`
- `mysql> INSERT INTO students`
 - `(name, surname, birth_date,`
 - `math_score,`
 - `modern_greek_score,`
 - `suspensions, conduct,`
 - `other_info)`
 - `VALUES ("Rikos",`
 - `"Mapouros", "1989-06-23",`
 - `"18.5", "16.0", "2", "Perfect",`
 - `"Football Player");`
- `mysql> SELECT * FROM students;`
- `mysql> SELECT id, name, surname,`
 - `birth_date FROM students;`
- `mysql> INSERT INTO students SET`
 - `name = "Takis" , "Svistos",`
 - `birth_date = "1990-03-02",`
 - `math_score = "15.5",`
 - `modern_greek_score = "16.7",`
 - `suspensions = "4", conduct =`
 - `"Good", other_info =`
 - `"Member of school orchestra";`
- `mysql> SELECT id, name, surname,`
 - `birth_date FROM students;`
- `mysql> UPDATE students SET`
 - `birth_date = "1989-06-22"`
 - `WHERE id = "1";`

Εργαστήριο 3

- **mysql> SELECT id, name, surname,
→ birth_date FROM students;**
- **mysql> DELETE FROM students
→ WHERE name = "Takis" AND
→ surname = "Svistos";**
- **mysql> SELECT id, name, surname,
→ birth_date FROM students;**
- **mysql> ALTER TABLE students
→ ADD COLUMN father
→ VARCHAR(100) AFTER
→ surname, ADD COLUMN
→ mother VARCHAR(100)
→ AFTER father;**
- **mysql> SELECT id, name, surname,
→ father, mother FROM students;**
- **mysql> UPDATE students SET
→ father = "Yiannis", mother =
→ "Tallou"
→ WHERE id = "1";**
- **mysql> SELECT id, name, surname,
→ father, mother FROM students;**
- **mysql> INSERT INTO students SET
→ name = "Takis", surname
→ = "Svistos", father = "Pampos",
→ mother = "Ellou", birth_date =
→ "1990-03-22", math_score =
→ "15.5", modern_greek_score =
→ "16.7", suspensions = "4",
→ conduct = "Good", other_info =
→ "Member of the school
→ Orchestra";**

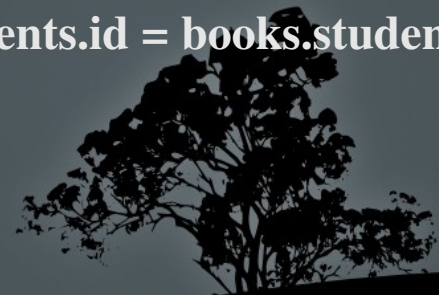
Εργαστήριο 3

- **mysql> INSERT INTO students (name, surname, father, mother, birth_date, math_score, modern_greek_score, suspensions, conduct, other_info) VALUES ("Aglaia", "Pitsiou", "Mitsos", "Euterpi", "1989-07-11", "18.7", "16.6", "0", "Perfect", "Dancer in the school's group");**
- **mysql> SELECT id, name, surname, suspensions, conduct FROM students;**
- **mysql> SELECT id, name, surname, suspensions, conduct FROM students WHERE conduct = "Perfect";**
- **mysql> SELECT count(conduct), conduct FROM students GROUP BY conduct;**
- **mysql> CREATE TABLE books (id INTEGER UNSIGNED NOT NULL AUTO_INCREMENT, title VARCHAR(200) NOT NULL, author VARCHAR(100), student_id INTEGER UNSIGNED, PRIMARY KEY (id));**
- **mysql> SHOW TABLES;**
- **mysql> DESCRIBE books;**



Εργαστήριο 3

- `mysql> INSERT INTO books (title,
→ author, student_id) VALUES
→ ("Little Brother",
→ "Cory Doctorow", "4");`
- `mysql> INSERT INTO books (title,
→ author, student_id) VALUES
→ ("Alice's Adventures in
→ Wonderland",
→ "Lewis Carroll", "5");`
- `mysql> INSERT INTO books (title,
→ author, student_id) VALUES
→ ("The Little Prince",
→ "Antoine de Saint-Exupéry ",
→ "5");`
- `mysql> INSERT INTO books (title,
→ author, student_id)
→ VALUES ("Gulliver's Travels",
→ "Jonathan Swift", "1");`
- `mysql> SELECT * FROM books;`
- `mysql> SELECT * FROM books
→ ORDER BY title;`
- `mysql> SELECT students.id,
→ students.name,
→ students.surname, books.title,
→ books.author FROM students,
→ books WHERE
→ students.id = books.student_id;`



Εργαστήριο 3

- **mysql> SELECT s.id, s.name,
→ s.surname, b.title, b.author
→ FROM students s, books b
→ WHERE s.id = b.student_id;**
- **mysql> SELECT students.id,
→ students.name,
→ students.surname,
→ books.title, books.author FROM
→ students JOIN books ON
→ students.id = books.student_id;**
- **mysql> SELECT students.id,
→ students.name,
→ students.surname,
→ books.title, books.author FROM
→ students INNER JOIN books
→ ON students.id =
→ books.student_id;**
- **mysql> SELECT students.id,
→ students.name,
→ students.surname,
→ books.title, books.author FROM
→ students LEFT JOIN books
→ ON students.id =
→ books.student_id;**
- **mysql> \q**
- **\$ echo "SELECT * FROM students" | \
mysql school;**
- **\$ mysqldump school | less**
- **\$ mysqldump school > school-bak.sql**
- **mysql> CREATE DATABASE
school_old;**
- **mysql> \q**

Εργαστήριο 3

- `# mysql school_old -uroot -p < \school-bak.sql`
 - `# mysql`
 - `mysql> SHOW DATABASES;`
 - `mysql> USE school_old;`
 - `mysql> SHOW TABLES;`
 - `mysql> SELECT * FROM students;`
 - `mysql> SELECT * FROM books;`
 - `mysql> quit`
 - `# su – user`
 - `$ mysql`
 - `mysql> CREATE DATABASE testing;`
 - `mysql> SELECT user();` # οι κανονικοί χρήστες μπορούν να ενωθούν σαν root αν δεν μπει κωδικός!
 - `mysql> \q`
 - `$ exit`
 - `# mysql`
 - `mysql> GRANT ALL PRIVILEGES
→ ON *.* TO 'root'@'localhost'
→ IDENTIFIED BY 'SecretPass'
→ WITH GRANT OPTION;`
 - `# su – user`
 - `$ mysql`
 - `$ mysql -p`
 - `$ mysql -uroot -p`
- 