

# MYSQL 1

---

Διδάσκοντες: Π. Αγγελάτος, Δ. Ζήνδρος  
Επιμέλεια διαφανειών: Δ. Ζήνδρος

Σχολή Ηλεκτρολόγων Μηχανικών  
και Μηχανικών Υπολογιστών



# Στόχος της ώρας

- Εισαγωγή στη MySQL και τις βάσεις δεδομένων
  - Τι είναι βάση δεδομένων;
  - Πίνακες, στήλες, πεδία
  - Σχήματα βάσεων
  - phpMyAdmin
  - Εισαγωγή στη γλώσσα SQL
  - SELECT
  - WHERE, λογικοί τελεστές
  - INSERT INTO, DELETE, UPDATE
  - Συνδυασμός PHP/MySQL



# Σύστημα βάσης δεδομένων

- Ένα **σύστημα** που αποθηκεύει δεδομένα
- Θα μπορούσαμε να κάνουμε την ίδια δουλειά μόνοι μας
  - π.χ. σε αρχεία
- Πλεονεκτήματα:
  - **Εύκολος** κώδικας
  - Πολύ **γρήγορη** ταχύτητα εκτέλεσης
  - Έτοιμες λύσεις σε γνωστά προβλήματα
    - Αλγόριθμοι αναζήτησης/ενημέρωσης
    - Συγχρονισμός
    - Αντίγραφα ασφαλείας
    - **Ταυτόχρονη χρήση**

# Σχεσιακή βάση δεδομένων

- Τα δεδομένα αποθηκεύονται σε **πίνακες**
- Ένας πίνακας αντιπροσωπεύει **μία σχέση**
- Η σχέση αποτελείται από **M στοιχεία**
  - **M-μελής**
    - Κάθε ένα από τα στοιχεία ανήκει σε κάποιο **σύνολο**
- **Εγγραφή:** Κάθε **M-άδα** για την οποία η σχέση αληθεύει
- **Στήλη ή πεδίο:** Κάθε ένα από τα στοιχεία της M-άδας
- **Τιμή πεδίου:** Η τιμή ενός πεδίου για μία συγκεκριμένη εγγραφή

# Γίγαντες της λογικής

Τίτλος πίνακα

Όνομα	Επώνυμο	Γέννηση	Θάνατος	Επίτευγμα
Georg	Cantor	1845	1918	Θεωρία συνόλων
Gottlob	Frege	1848	1925	Σύγχρονη λογική
Bertrand	Russell	1872	1970	Παράδοξο Russell
David	Hilbert	1862	1943	Προβλήματα Hilbert
Kurt	Gödel	1906	1978	Μη πληρότητα
Alan	Turing	1912	1954	Μηχανή Turing

Πίνακας

# Γίγαντες της λογικής

Τίτλος στήλης

Όνομα	Επώνυμο	Γέννηση	Θάνατος	Επίτευγμα
Georg	Cantor	1845	1918	Θεωρία συνόλων
Gottlob	Frege	1848	1925	Σύγχρονη λογική
Bertrand	Russell	1872	1970	Παράδοξο Russell
David	Hilbert	1862	1943	Προβλήματα Hilbert
Kurt	Gödel	1906	1978	Μη πληρότητα
Alan	Turing	1912	1954	Μηχανή Turing

Στήλη ή πεδίο

# Γίγαντες της λογικής

Όνομα	Επώνυμο	Γέννηση	Θάνατος	Επίτευγμα
Georg	Cantor	1845	1918	Θεωρία συνόλων
Gottlob	Frege	1848	1925	Σύγχρονη λογική
Bertrand	Russell	1872	1970	Παράδοξο Russell
David	Hilbert	1862	1943	Προβλήματα Hilbert
Kurt	Gödel	1906	1978	Μη πληρότητα
Alan	Turing	1912	1954	Μηχανή Turing

Εγγραφή





# Γίγαντες της λογικής

Όνομα	Επώνυμο	Γέννηση	Θάνατος	Επίτευγμα
Georg	Cantor	1845	1918	Θεωρία συνόλων
Gottlob	Frege	1848	1925	Σύγχρονη λογική
Bertrand	Russell	1872	1970	Παράδοξο Russell
David	Hilbert	1862	1943	Προβλήματα Hilbert
Kurt	Gödel	1906	1978	Μη πληρότητα
Alan	Turing	1912	1954	Μηχανή Turing

Τιμή πεδίου



# Ονοματολογία

- Επιτρεπόμενα **ονόματα** στην MySQL:
  - Αρχίζουν από γράμμα
  - Περιέχουν λατινικά γράμματα, αριθμούς, \_
  - Είναι ευαίσθητα σε πεζά/κεφαλαία
- Αφορά ονόματα πινάκων, στηλών, και βάσεων

# Σχήμα

- **Σχήμα**: Η **δομή** πινάκων και στηλών μίας βάσης
- Δημιουργία σχήματος. Σκέφτομαι:
  - Τι είδους **αντικείμενα** θα αποθηκεύσει η εφαρμογή μου; → Πίνακες
  - Τι **ιδιότητες** έχει κάθε τέτοιο αντικείμενο; → Στήλες

# Δημιουργία σχήματος

- Παράδειγμα: Ιστοσελίδα συγγραμάτων Εύδοξος
- Τι είδους αντικείμενα χρειαζόμαστε;
  - Φοιτητές
  - Μαθήματα
  - Βιβλία
  - **Εκδότες**
- Τι ιδιότητες χρειαζόμαστε για κάθε αντικείμενο;

# Δημιουργία σχήματος

- publishers
  - name
  - street
  - addressnumber
  - postcode
  - phone

# Δημιουργία σχήματος

publishers:

name	street	addressnumber	postcode	phone
Παπασωτηρίου	Στουρνάρη	35	10442	2103323300
Συμμετρία	Θεολόγου	80	15773	2107707114
Εκδότες Θεσσαλονίκης	Πεσμαζόγλου	5	10564	2103211097
Τζιόλα	Πεσμαζόγλου	5	10564	2103211097

# Μοναδικό αναγνωριστικό

- Σε κάθε πίνακα έχουμε μία **στήλη-αναγνωριστικό**
- Συχνά την ονομάζουμε “**id**”
- Ή όνομα πίνακα + “**id**”
  - Πίνακας students: studentid
- Είναι **θετικοί φυσικοί αριθμοί**
- Ξεκινούν από το 1
- Κάθε νέα εγγραφή έχει τον επόμενο αριθμό
- Ρυθμίζεται αυτόματα από το σύστημα βάσης δεδομένων
- Η στήλη αναφέρεται και ως **AUTO\_INCREMENT**

# Μοναδικό αναγνωριστικό

- Επιτρέπει να **αναφερθούμε** σύντομα σε μία γραμμή
  - Ο μαθητής με id 5
  - Ο εκδότης με id 109
  - Το βιβλίο id 49,999
- **Ξεχωρίζει** δύο εγγραφές με **ίδια στοιχεία**

Μαθητής **Αλέξης Παπαντωνίου** του **Αναστάση**, id 5

≠

Μαθητής **Αλέξης Παπαντωνίου** του **Αναστάση**, id 105

- Αυτή η στήλη ονομάζεται **πρωτεύον κλειδί**



# Δημιουργία σχήματος

- publishers
  - **id**
  - name
  - street
  - addressnumber
  - postcode
  - phone

# Δημιουργία σχήματος

publishers:

id	name	street	addressnumber	postcode	phone
1	Παπασωτηρίου	Στουρνάρη	35	10442	2103323300
2	Συμμετρία	Θεολόγου	80	15773	2107707114
3	Εκδότες Θεσσαλονίκης	Πεσμαζόγλου	5	10564	2103211097
4	Τζιόλα	Πεσμαζόγλου	5	10564	2103211097

↑  
Στήλη **AUTO\_INCREMENT**

# Δημιουργία σχήματος

- **students**

- studentid
- name
- surname
- schoolname
- schoolfounded
- schoollocation
- semester

- **courses**

- courseid
- name
- description
- schoolname
- schoolfounded
- schoollocation
- semester

- **books**

- bookid
- title
- description
- schoolname
- schoolfounded
- schoollocation
- semester

# Δημιουργία σχήματος

students:

studentid	name	surname	schoolname	schoolfunded	schoollocation	semester
1	Πέτρος	Αγγελάτος	ΗΜΜΥ ΕΜΠ	1836	Ζωγράφου, Αθήνα	5
2	Γιώργος	Πάνου	ΗΜΜΥ ΕΜΠ	1836	Ζωγράφου, Αθήνα	5
3	Παύλος	Τσιμής	ΗΜΜΥ ΕΜΠ	1836	Ζωγράφου, Αθήνα	7
4	Αρετή	Τσαμάκη	ΗΜΜΥ ΑΠΘ	1926	Θεσσαλονίκη	3

Ανεπιθύμητη επανάληψη

# Δημιουργία σχήματος

- Αποφυγή επανάληψης:
  - Ας φτιάξουμε έναν ξεχωριστό πίνακα για τις σχολές
- schools
  - schoolid
  - name
  - founded
  - location

# Δημιουργία σχήματος

schools:

schoolid	name	founded	location
1	ΗΜΜΥ ΕΜΠ	1836	Ζωγράφου, Αθήνα
2	ΗΜΜΥ ΑΠΘ	1926	Θεσσαλονίκη

students:

αναφέρεται σε

studentid	name	surname	schoolid	semester
1	Πέτρος	Αγγελάτος	1	5
2	Γιώργος	Πάνου	1	5
3	Παύλος	Τσιμής	1	7
4	Αρετή	Τσαμάκη	2	3

# Δημιουργία σχήματος

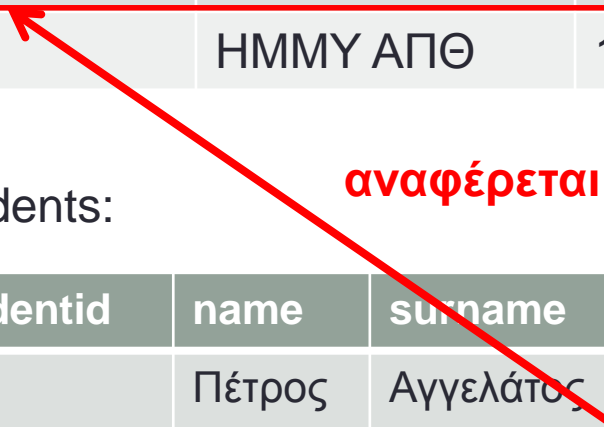
schools:

schoolid	name	funded	location
1	ΗΜΜΥ ΕΜΠ	1836	Ζωγράφου, Αθήνα
2	ΗΜΜΥ ΑΠΘ	1926	Θεσσαλονίκη

students:

studentid	name	surname	schoolid	semester
1	Πέτρος	Αγγελάτος	1	5
2	Γιώργος	Πάνου	1	5
3	Παύλος	Τσιμής	1	7
4	Αρετή	Τσαμάκη	2	3

αναφέρεται σε



# Δημιουργία σχήματος

schools:

schoolid	name	funded	location
1	ΗΜΜΥ ΕΜΠ	1836	Ζωγράφου, Αθήνα
2	ΗΜΜΥ ΑΠΘ	1926	Θεσσαλονίκη

students:

**αναφέρεται σε**

studentid	name	surname	schoolid	semester
1	Πέτρος	Αγγελάτος	1	5
2	Γιώργος	Πάνου	1	5
3	Παύλος	Τσιμής	1	7
4	Αρετή	Τσαμάκη	2	3



# Βάση δεδομένων

- Βάση δεδομένων ονομάζεται **το σύνολο όλων των πινάκων** και των εγγραφών τους.
- Βάση δεδομένων = Σχήμα + Δεδομένα
- Κάθε βάση δεδομένων έχει ένα **όνομα**

eudoxus

students

publishers

books

Γιώργος

Μαρία

Νίκος

HTMLDog

ppk

PHP and MySQL  
web development

Παπασωτηρίου

Συμμετρία

name Γιώργος

AM 08149

street Πεσμαζόγλου

street Στουρνάρη

# Τύποι

- Κάθε στήλη αποθηκεύει συγκεκριμένο **τύπο** δεδομένων
- Ο τύπος ορίζεται στο σχήμα
  - **INT**: Φυσικός αριθμός
  - **FLOAT**: Δεκαδικός αριθμός
  - **TEXT**: Αλφαριθμητικό
  - **CHAR ( N )**: Αλφαριθμητικό σταθερού μεγέθους N
  - **VARCHAR ( N )**: Αλφαριθμητικό μέγιστου μεγέθους N
  - **DATE**: Ημερομηνία
  - **DATETIME**: Ημερομηνία και ώρα
- ...[και άλλοι](#), δείτε τους όταν τους χρειαστείτε

# Δημιουργία σχήματος

- Χρησιμοποιούμε το πρόγραμμα **phpMyAdmin**
  - Ή κάποιο άλλο MySQL client π.χ. CLI client
- Με αυτό δημιουργούμε το σχήμα
- **Το σχήμα δεν αλλάζει κατά τον χρόνο εκτέλεσης**
- Μπορεί να αλλάξει από εμάς αργότερα
  - π.χ. όταν προσθέτουμε λειτουργίες στην εφαρμογή

# phpMyAdmin

- Ένα πρόγραμμα γραμμένο σε PHP
- Μας επιτρέπει να δημιουργούμε:
  - Βάσεις δεδομένων
  - Πίνακες
  - Στήλες
  - Εγγραφές
- ...και να τα αλλάζουμε.
  
- Θα αναρτηθούν αναλυτικές οδηγίες στο site για την εγκατάστασή του.

# Ο εξυπηρετητής MySQL

- Η MySQL είναι ένα **πρόγραμμα εξυπηρετητή**
- Συχνά τρέχει στον ίδιο υπολογιστή με τον Apache
  - Στα μεγάλα sites τρέχει αλλού
- Η PHP συνδέεται στην MySQL
  - Στέλνει **ερωτήματα**
  - Δέχεται **απαντήσεις**
- Η PHP είναι ένας **MySQL client**
  - Η PHP συνεχίζει να τρέχει στον HTTP server (Apache)
- Το πρόγραμμα της MySQL είναι ο **MySQL server**



Δώσε μου τη σελίδα X



Φτιάξε μου τη σελίδα X



Εκτέλεση κώδικα PHP  
Παραγωγή HTML

Ορίστε η σελίδα X



Ορίστε η σελίδα X



Εμφάνιση σελίδας



HTTP



MySQL ερώτημα



Χώρος αποθήκευσης



MySQL απάντηση



HTTP



MySQL Client



MySQL Server

Φέρε μου τις πληροφορίες των μαθητών

Ορίστε οι πληροφορίες των μαθητών

Πρόσθεσε τον μαθητή με τις τάδε πληροφορίες

Χώρος αποθήκευσης



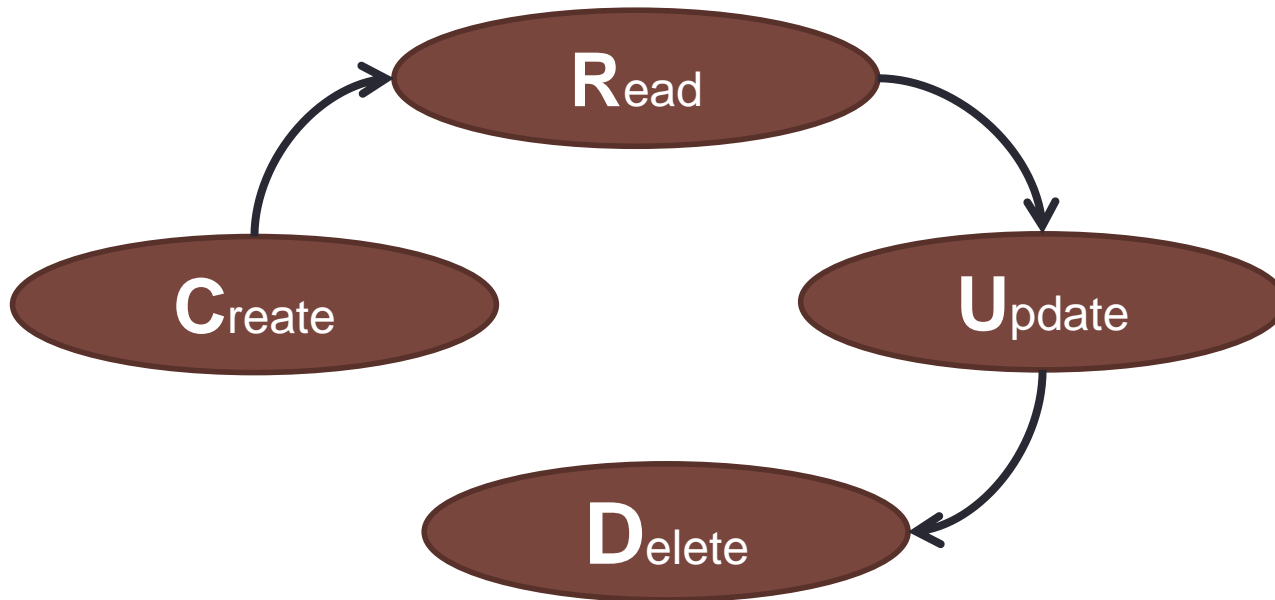
OK

Διέγραψε τους μαθητές του 2008

OK



# Κύκλος ζωής μιας εγγραφής: CRUD



# Ερώτημα

- Μία **εντολή** προς το σύστημα βάσεων δεδομένων
- Ο τρόπος που επικοινωνούμε με το σύστημα
- Ζητάμε...
  - Την **δημιουργία** εγγραφών (**C**reate)
  - Την **ανάγνωση** εγγραφών (**R**ead)
  - Την **ενημέρωση** εγγραφών (**U**ppdate)
  - Την **διαγραφή** εγγραφών (**D**elete)

# Ερωτήματα

- Αντίθετα από την δομή που φτιάχνουμε με phpMyAdmin:
- Τα ερωτήματα τρέχουν σε χρόνο εκτέλεσης
- π.χ. όταν τρέξει ένα PHP script
  
- π.χ.
- Ο χρήστης πατάει το κουμπί δημιουργίας λογαριασμού
- → Καλείται το PHP script register.php με HTTP POST
- → Τρέχει το ερώτημα προσθήκης μέλους

# Η γλώσσα SQL

- Χρησιμοποιείται για να γράψουμε **ερωτήματα**
- Περιγράφει **τι** θέλουμε να συμβεί
- **Όχι πώς** θέλουμε να συμβεί
- Κάθε ερώτημα τελειώνει με **;**
- Τα κενά δεν παίζουν ρόλο

# Απαντήσεις

Το αποτέλεσμα ενός ερωτήματος

Ερώτημα	Απάντηση
Δημιουργία	<ul style="list-style-type: none"><li>• <b>Εντάξει</b> ή <b>Όχι εντάξει</b></li><li>• Τιμή πρωτεύοντος κλειδιού</li></ul>
Ανάγνωση	<ul style="list-style-type: none"><li>• <b>Δεδομένα</b></li></ul>
Επεξεργασία	<ul style="list-style-type: none"><li>• <b>Εντάξει</b> ή <b>Όχι εντάξει</b></li><li>• Πλήθος εγγραφών</li></ul>
Διαγραφή	<ul style="list-style-type: none"><li>• <b>Εντάξει</b> ή <b>Όχι εντάξει</b></li><li>• Πλήθος εγγραφών</li></ul>

# Σταθερές



- Αριθμοί:
  - 2, 3, 5, 7, 11, 13, 16.9999
- Αλφαριθμητικά:
  - "Hello, world", "Natsos", "GINOSAJI"
  - Εμπεριέχονται σε 'μονά' ή "διπλά" εισαγωγικά
- Ημερομηνίες
  - "1987-30-11"
  - Στη μορφή "YYYY-MM-DD"
  - Έτος-μήνας-μέρα
  - Σε 'μονά' ή "διπλά" εισαγωγικά
- Ημερομηνίες και ώρες
  - "1987-30-11 10:24:09"
  - Στη μορφή "YYYY-MM-DD HH:MM:SS"
  - Έτος-μήνας-μέρα ώρα:λεπτά:δευτερόλεπτα
  - Σε 'μονά' ή "διπλά" εισαγωγικά

# Δημιουργία

- Ερώτημα **INSERT INTO**
- Μορφή:

**INSERT INTO**

πίνακας

**SET**

πεδίο1 = τιμή1,

πεδίο2 = τιμή2,

...

πεδίοN = τιμήN;



# Δημιουργία

- Προσθήκη ενός μαθητή:

```
INSERT INTO
    students
SET
    name = "Πέτρος",
    surname = "Αγγελάτος",
    am = "03108666";
```

# Δημιουργία

- Παραλείπεται η τιμή για το πεδίο **AUTO\_INCREMENT**
- Όσα πεδία παραληφθούν παίρνουν:
  - Την προεπιλεγμένη τιμή που ορίζουμε στο σχήμα μας.
  - Την προεπιλεγμένη τιμή σύμφωνα με τον τύπο τους αν δεν έχουμε ορίσει προεπιλεγμένη τιμή στο σχήμα.

# Δημιουργία

- Εναλλακτική μορφή:

```
INSERT INTO
    πίνακας
    ( πεδίο1, πεδίο2, ..., πεδίοN )
VALUES
    ( τιμή1, τιμή2, ..., τιμήN );
```

# Ανάγνωση

- Ερώτημα **SELECT**
- Μορφή:

**SELECT**

πεδίο1, πεδίο2, ..., πεδίοN

**FROM**

πίνακας;

- Επιστρέφει συγκεκριμένα πεδία από όλες τις εγγραφές ενός πίνακα

# Ανάγνωση

- Ανάγνωση όλων των μικρών ονομάτων και εξαμήνων όλων των μαθητών:

```
SELECT
```

```
    surname, semester
```

```
FROM
```

```
    students;
```

# Τελεστές

Τελεστές	Λειτουργία
<code>+, -, *, /</code>	Αριθμητικές πράξεις
<code>OR, AND, NOT</code>	Λογικές πράξεις: ή, και, όχι
<code>=, !=, &lt;, &gt;, &lt;=, &gt;=</code>	Σύγκριση

# Λογικές προτάσεις

- Παίρνουν τιμή **αλήθεια (1)** ή **ψεύδος (0)**
- Χρησιμοποιούνται για να **φιλτράρουμε** εγγραφές
- Μπορούν να περιέχουν **ονόματα πεδίων**
  - studentid
  - name
  - addressnumber
- ...ή **σταθερές**
  - 5
  - “hello”
  - “1987-11-30 10:24:09”

# Λογικές προτάσεις

- Αληθεύουν πιθανώς για κάποιες εγγραφές
- Δεν αληθεύουν πιθανώς για άλλες εγγραφές
- **name = "Πέτρος"**
  - Αληθεύει για την εγγραφή του μαθητή Πέτρου
  - Δεν αληθεύει για την εγγραφή του μαθητή Διονύση
- **surname != "Αγγελάτος"**
  - Δεν αληθεύει για την εγγραφή του μαθητή Πέτρου
  - Αληθεύει για την εγγραφή του μαθητή Διονύση



# Φιλτράρισμα εγγραφών

- Όρος **WHERE**
- Μορφή:
  - **WHERE** λογική πρόταση
- Π.χ.
  - **WHERE** name = 'Πέτρος'
  - **WHERE** surname != 'Αγγελάτος'
  - **WHERE** semester < 5
  - **WHERE** name = 'Πέτρος' AND surname != 'Αγγελάτος'

# Ανάγνωση

- Μορφή:

**SELECT**

πεδίο1, πεδίο2, ..., πεδίοN

**FROM**

πίνακας

**WHERE**

λογική πρόταση;

# Ανάγνωση

- Ανάγνωση όλων των μικρών ονομάτων και εξαμήνων όλων των μαθητών 1<sup>ου</sup> και 2<sup>ου</sup> έτους:

```
SELECT
    surname, semester
FROM
    students
WHERE
    semester < 5;
```

# Ανάγνωση

- Ο πίνακας που είναι αποθηκευμένος και ο πίνακας που επιστρέφει το ερώτημα μπορεί να είναι διαφορετικοί
- Διαφορετικά πεδία
  - Η απάντηση έχει συνήθως μόνο κάποια πεδία
- Διαφορετικές εγγραφές
  - Η απάντηση συνήθως έχει μόνο κάποιες εγγραφές
- Ο προσωρινός πίνακας που επιστρέφεται ονομάζεται **πίνακας αποτελέσματος**.

# Ποιο είναι το ερώτημα;

- Αποθηκευμένος πίνακας

studentid	name	surname	schoolid	semester
1	Πέτρος	Αγγελάτος	1	5
2	Γιώργος	Πάνου	1	5
3	Παύλος	Τσιμής	1	7
4	Αρετή	Τσαμάκη	2	3

- Πίνακας αποτελέσματος

name	surname
Πέτρος	Αγγελάτος
Γιώργος	Πάνου
Παύλος	Τσιμής

```
SELECT  
    name, surname  
FROM  
    students  
WHERE  
    schoolid = 1;
```

# Ενημέρωση εγγραφών

- Αλλάζει τις **τιμές πεδίων** σε εγγραφές που υπάρχουν ήδη
- Ερώτημα **UPDATE**
- **Μορφή :**

**UPDATE**

πίνακας

**SET**

πεδίο1 = τιμή1,

πεδίο2 = τιμή2,

...

πεδίοN = τιμήN;

- Αλλάζει **όλες** τις εγγραφές στον πίνακα
- Τα πεδία που **δεν** εμφανίζονται **δεν** αλλάζουν

# Ενημέρωση εγγραφών

- Φιλτράρουμε με **WHERE**
- Μορφή:

**UPDATE**

πίνακας

**SET**

πεδίο1 = τιμή1,

πεδίο2 = τιμή2,

...

πεδίοN = τιμήN

**WHERE**

λογική πρόταση;

# Ενημέρωση εγγραφών

```
UPDATE
```

```
    publishers
```

```
SET
```

```
    postcode = 10564
```

```
WHERE
```

```
    street = 'Πεσμαζόγλου' ;
```



# Διαγραφή εγγραφών

- Ερώτημα **DELETE**
- Μορφή :

**DELETE FROM**  
πίνακας;

- Διαγράφει **όλες** τις εγγραφές στον πίνακα

# Διαγραφή εγγραφών

- Φιλτράρουμε με **WHERE**
- Μορφή:

**DELETE FROM**

πίνακας

**WHERE**

λογική πρόταση;

# Το επίμαχο ερώτημα

```
DELETE FROM  
    students  
WHERE  
    semester > 100;
```

# Συνδυασμός PHP και MySQL

- Απαιτείται η επέκταση MySQL της PHP
- Απαιτείται εγκατάσταση του προγράμματος MySQL
- Αναλυτικές οδηγίες θα υπάρχουν στο site
- Τα SQL ερωτήματα για την PHP είναι **αλφαριθμητικά**
- Η βιβλιοθήκη MySQL για PHP περιέχει **συναρτήσεις** που τις καλούμε για να πετύχουμε επικοινωνία PHP και MySQL μέσω των ερωτημάτων.
- Τα ερωτήματα περνούν ως **παράμετροι**.

# Σύνδεση στη βάση

```
mysql_connect(  
    διεύθυνση,  
    χρήστης,  
    κωδικός  
);
```

- Συνδέεται στον εξυπηρετητή MySQL
- Το τρέχουμε συνήθως στην αρχή του κώδικα που περιέχει MySQL εντολές

- Διεύθυνση:
  - Σε ποιον υπολογιστή τρέχει ο εξυπηρετητής MySQL;
  - Συνήθως `localhost`
- Χρήστης και κωδικός:
  - Απαιτούνται για την πιστοποίηση πρόσβασης
  - Ρυθμίζονται κατά την εγκατάσταση της MySQL ή από phpMyAdmin
- Η σύνδεση κλείνει αυτόματα στο τέλος του script

# Σύνδεση στη βάση

```
mysql_select_db( όνομα_βάσης );
```

- Επιλέγει σε ποια βάση θα τρέξουν τα ερωτήματα
  - Ο εξυπηρετητής μπορεί να αποθηκεύει πολλές βάσεις
  - (συνήθως μία για κάθε web εφαρμογή)

# Εκτέλεση ερωτήματος

- `απάντηση = mysql_query( ερώτημα );`
- Εκτελεί το ερώτημα και επιστρέφει την απάντηση



# Παράδειγμα

```
mysql_connect("localhost", "dionyziz", "30/11/1987");  
mysql_select_db( "eudoxus" );  
mysql_query(  
    "UPDATE  
        students  
    SET  
        semester = 1  
    WHERE  
        surname = 'Αγγελάτος'  
        AND name = 'Πέτρος' ;"  
);
```

# Προσθήκη εγγραφής

- `mysql_insert_id()`;
- Επιστρέφει το **AUTO\_INCREMENT** της εγγραφής που δημιουργήθηκε
- Αναφέρεται στο ερώτημα που έτρεξε πιο πρόσφατα

# Παράδειγμα

**database.php:**

```
mysql_connect(  
    "localhost",  
    "dionyziz",  
    "30/11/1987"  
);  
mysql_select_db( "eudoxus" );
```

# Παράδειγμα

```
include "database.php";
mysql_query(
    "INSERT INTO
        students
    SET
        name = 'Μάκης',
        surname = 'Δελιγιάννης',
        semester = 1;"
);
$id = mysql_insert_id();
echo "Ο φοιτητής ". $id . " καταχωρήθηκε ";
```

# Ενημέρωση/διαγραφή

- `mysql_affected_rows()` ;
- Επιστρέφει το πλήθος των γραμμών που άλλαξαν
- Αναφέρεται στο πιο πρόσφατο ερώτημα

# Παράδειγμα

```
include "database.php";
mysql_query(
    "DELETE FROM
        students
    WHERE
        name = 'Μάκης'
        AND surname = 'Δελιγιάννης';"
);
if ( mysql_affected_rows() == 1 ) {
    echo "Ο φοιτητής <em>διαγράφηκε</em>.";
}
else {
    echo "Χιούστον, έχουμε <em>πρόβλημα</em>!";
}
```

# Τιμή απάντησης

- Το `mysql_query()` επιστρέφει:
  - Σε περίπτωση ερωτημάτων Δημιουργίας, Ενημέρωσης, Διαγραφής:
    - `true` αν το ερώτημα έτρεξε επιτυχώς
    - `false` αν το ερώτημα απέτυχε
    - `$success = mysql_query( ερώτημα );`
  - Σε περίπτωση ερωτήματος Ανάγνωσης:
    - ένα `resource` που μπορεί να αξιοποιηθεί αργότερα
    - `$res = mysql_query( ερώτημα );`

# Ανάγνωση δεδομένων

- `mysql_num_rows( $res );`
- Παίρνει ως παράμετρο το **resource** που επέστρεψε το `mysql_query()`
- Επιστρέφει το πλήθος των εγγραφών του αποτελέσματος



# Παράδειγμα

```
include "database.php";  
$res = mysql_query(  
    "SELECT  
        name  
    FROM  
        students  
    WHERE  
        semester = 3;"  
);  
echo "Έχουμε " . mysql_num_rows( $res );  
> φοιτητές στο 3ο εξάμηνο.
```



Έχουμε 492 φοιτητές στο 3<sup>ο</sup> εξάμηνο.

# Ανάγνωση δεδομένων

- `mysql_fetch_array( $res );`
- Παίρνει ως παράμετρο το **resource** που επέστρεψε το `mysql_query()`
- Αν ο πίνακας αποτελέσματος έχει **N** εγγραφές, επιστρέφει:
  - Την **πρώτη** φορά που θα κληθεί, ένα λεξικό
    - Αντιπροσωπεύει την **πρώτη** εγγραφή του πίνακα αποτελέσματος
  - Την **δεύτερη** φορά που θα κληθεί, ένα λεξικό
    - Αντιπροσωπεύει την **δεύτερη** εγγραφή του πίνακα αποτελέσματος
  - ...
  - Την **N-οστή** φορά που θα κληθεί, ένα λεξικό
    - Αντιπροσωπεύει την τελευταία εγγραφή του πίνακα αποτελέσματος
  - Την **(N + 1)-οστή** φορά που θα κληθεί, `false`

# Ανάγνωση δεδομένων

- Η **πρώτη** κλήση της `mysql_fetch_array( $res )` επιστρέφει ένα λεξικό
- Αντιπροσωπεύει την **πρώτη** εγγραφή του αποτελέσματος
- Κλειδιά: τα **ονόματα των στηλών**
- Τιμές: οι **τιμές** των συγκεκριμένων πεδίων για την συγκεκριμένη εγγραφή
- κ.ό.κ. για τις πρώτες N κλήσεις
- έως να εξαντληθεί ο πίνακας αποτελέσματος

name	surname
Πέτρος	Αγγελάτος
Γιώργος	Πάνου
Παύλος	Τσιμής

```
$sql = "SELECT
        name, surname
    FROM
        students
    WHERE
        schoolid = 1;";
```

```
include 'database.php';
$res = mysql_query( $sql );
$first  = mysql_fetch_array( $res );
$second = mysql_fetch_array( $res );
$third  = mysql_fetch_array( $res );
$end    = mysql_fetch_array( $res );

echo $first[ 'name' ] . "<br />";
echo $second[ 'name' ] . " ";
echo $second[ 'surname' ] . "<br />";
echo $third[ 'surname' ] . "<br />";
```

```
echo $first[ 'name' ] . "<br />";  
echo $second[ 'name' ] . " ";  
echo $second[ 'surname' ] . "<br />";  
echo $third[ 'surname' ] . "<br />";
```



**Πέτρος  
Γιώργος Πάνου  
Τσιμής**

name	surname
Πέτρος	Αγγελάτος
Γιώργος	Πάνου
Παύλος	Τσιμής

# Ανάγνωση όλων των εγγραφών

```
include 'database.php'; $sql = '...';
$res = mysql_query( $sql );
$count = mysql_num_rows( $res );
for ( $i = 0; $i < $count; ++$i ) {
    $row = mysql_fetch_array( $res );
    echo "Row $i:";
    $name = $row[ 'name' ];
    $surname = $row[ 'surname' ];
    echo "The name is $surname, $name.";
    ?><br /><?php
}
```



name	surname
Πέτρος	Αγγελάτος
Γιώργος	Πάνου
Παύλος	Τσιμής

Row 0: The name is Αγγελάτος, Πέτρος.

Row 1: The name is Πάνου, Γιώργος.

Row 2: The name is Τσιμής, Παύλος.



# Ανάγνωση όλων των εγγραφών

Συνηθέστερη μορφή:

```
include 'database.php'; $sql = '...';  
$res = mysql_query( $sql );  
while ( $row = mysql_fetch_array( $res ) ){  
    $name = $row[ 'name' ];  
    $surname = $row[ 'surname' ];  
    echo "The name is $surname, $name."  
?><br /><?php  
}
```

# Μάθαμε

- Βάση δεδομένων
- SQL
- Πίνακες, στήλες, πεδία
- Σχήματα, phpMyAdmin
- SELECT
- INSERT INTO
- DELETE
- UPDATE
- Συνδυασμός PHP/MySQL

# Συγχαρητήρια!

- Μάθατε MySQL.
- Μπορείτε να **αποθηκεύετε δεδομένα** αποδοτικά!



# Την επόμενη φορά...

- Εμβάθυνση στην MySQL
  - Γρήγορη ταξινόμηση
  - Περισσότεροι λογικοί τελεστές: IN, BETWEEN
  - Ενώσεις πινάκων
  - Όρια
  - Ομαδοποίηση
  - Συνοπτικές συναρτήσεις
  - ...και άλλα λαχταριστά θέματα!