

## Ερωτήματα με την εντολή SELECT

Μέχρις αυτό το σημείο έχουμε δει εντολές που αφορούν τη δομή της βάση καθώς και την εισαγωγή και τροποποίηση των αποθηκευμένων εγγραφών. Μια όμως από τις βασικότερες εντολές της **SQL** είναι η εντολή **SELECT**, με την οποία υποβάλλουμε στη βάση ερωτήματα για την άντληση πληροφοριών.

Αυτή η εντολή μπορεί να γίνει πολύ σύνθετη και να υλοποιήσει πολύ εξειδικευμένες αναζητήσεις σε μια βάση, αλλά στην γενικότερη και απλούστερη μορφή της έχει ως εξής;

**SELECT** <fields list> **FROM** <table list> **WHERE** <condition>

όπου <fields list> είναι μια λίστα τα ονόματα των πεδίων που θέλουμε να ανακτήσουμε, όπου <table list> μια λίστα με τα ονόματα των πινάκων που θα χρησιμοποιηθούνε στη δημιουργία του ερωτήματος, και όπου <condition>, είναι τα κριτήρια αναζήτησης, που θα πρέπει να ικανοποιούνται από τα δεδομένα των πινάκων, προκειμένου να συμπεριληφθούν στα τελικά αποτελέσματα.

Πριν παρουσιάσουμε τα διάφορα παραδείγματα της εντολής SELECT ας υποθέσουμε ότι οι πίνακες της βάσης έχουν κάποια δεδομένα. Με βάση αυτά τα δεδομένα θα παρουσιάζουμε τα αποτελέσματα των διάφορων ερωτημάτων.

Author		Publication	
ID_auth	name	ID_publication	name
1	Διδώ Σωτηρίου	1	Άγρα
2	Νίκος Καββαδίας	2	Κέδρος
3	Μ. Καραγάτσης	3	Πεδίο
4	Αλέξανδρος Παπαδιαμάντης	4	Εστία

### Member

ID_member	name
11	Νικόλας Παπαδόπουλος
12	Άννα Γεωργίου
13	Κώστας Καρανίκος
14	Παναγιώτα Παπαγεωργίου

## Books

ID	ISBN	title	author	publication	price	status	date_added
101	11-222-33	Χίμαιρα	3	4	15.42	1	1995/5/15
102	11-223-15	Το Δέκα	3	4	18.50	1	1996/3/28
103	11-223-28	Ηλέκτρα	1	2	22.00	1	1997/4/2
104	11-226-45	Εντολή	1	2	7.78	1	1999/10/25
105	11-223-36	Μαραμπού	2	1	11.40	1	2001/11/2
106	11-245-35	Όνειρο στο κύμα	4	2	21.60	0	2008/6/15
107	11-225-38	Η φόνισσα	4	3	15.80	1	2009/7/30
108	11-224-98	Η γυφτοπούλα	4	4	9.90	1	2012/1/28
109	11-229-78	Η μετανάστρια	4	3	23.10	1	2014/2/16
110	11-222-85	Θα σας τηλεφωνώ	2	4	17.30	0	2015/3/29

## Λίστα πεδίων

Ένα πολύ απλό παράδειγμα της εντολής **SELECT** είναι η παρακάτω, στην οποία το τμήμα **WHERE** απουσιάζει και αντί για μια λίστα πεδίων χρησιμοποιούμε τον χαρακτήρα **\***. Το ερώτημα αυτό επιστρέφει **όλα τα πεδία** από **όλες τις εγγραφές** του πίνακα books.

```
SELECT * FROM books
```

+ Options		ID	ISBN	title	author	publication	price	status	date_added
<input type="checkbox"/>	Edit Copy Delete	101	11-222-33	Χίμαιρα	3	4	15.42	1	1995-05-15
<input type="checkbox"/>	Edit Copy Delete	102	11-223-15	Το Δέκα	3	4	18.50	1	1996-03-28
<input type="checkbox"/>	Edit Copy Delete	103	11-223-28	Ηλέκτρα	1	2	22.00	1	1997-04-02
<input type="checkbox"/>	Edit Copy Delete	104	11-226-45	Εντολή	1	2	7.78	1	1999-10-25
<input type="checkbox"/>	Edit Copy Delete	105	11-223-36	Μαραμπού	2	1	11.40	1	2001-11-02
<input type="checkbox"/>	Edit Copy Delete	106	11-245-35	Όνειρο στο κύμα	4	2	21.60	0	2008-06-15
<input type="checkbox"/>	Edit Copy Delete	107	11-225-38	Η φόνισσα	4	3	15.80	1	2009-07-30
<input type="checkbox"/>	Edit Copy Delete	108	11-224-98	Η γυφτοπούλα	4	4	9.90	1	2012-01-28
<input type="checkbox"/>	Edit Copy Delete	109	11-229-78	Η μετανάστρια	4	3	23.10	1	2014-02-16
<input type="checkbox"/>	Edit Copy Delete	110	11-222-85	Θα σας τηλεφωνώ	2	4	17.30	0	2015-03-29

Αν δεν επιθυμούμε να επιστραφούν όλα τα πεδία του πίνακα, τότε αντί για το **\*** μπορούμε να βάλουμε μια λίστα από τα πεδία που επιθυμούμε. Για παράδειγμα η

παρακάτω εντολή που επιστρέφει τα πεδία **ISBN** και **title** για όλες τις εγγραφές του πίνακα **books**.

```
SELECT ISBN, title FROM books
```

+ Options

ISBN	title
11-222-33	Χίμαιρα
11-223-15	Το Δέκα
11-223-28	Ηλέκτρα
11-226-45	Εντολή
11-223-36	Μαραμπού
11-245-35	Όνειρο στο κύμα
11-225-38	Η φόνισσα
11-224-98	Η γυφτοπούλα
11-229-78	Η μετανάστρια
11-222-85	Θα σας τηλεφωνώ

Αν θέλουμε κάποιο πεδίο να επιστραφεί στα αποτελέσματα με διαφορετικό όνομα, τότε μπορούμε να χρησιμοποιήσουμε τη δήλωση «**as new\_name**» μετά από την αναφορά του πεδίου. Για παράδειγμα

```
SELECT ISBN, title, publication as ekdotiki FROM books
```

Αυτό βέβαια δεν έχει ιδιαίτερο νόημα σε απλά πεδία, γι' αυτό το χρησιμοποιούμε συνήθως σε πεδία που έχουν προκύψει από κάποια έκφραση. Για τη δημιουργία εκφράσεων στη λίστα των πεδίων μπορούμε να χρησιμοποιήσουμε **αριθμητικούς, συγκριτικούς και λογικούς τελεστές (+, -, \*, /, div, mod, <, >, <=, >=, =, <>, and, or, not)** ή ακόμα κάποια από τις συναρτήσεις της **MySQL** που θα δούμε παρακάτω.

Ακολουθούν κάποια παραδείγματα με τη χρήση σύνθετων πεδίων που προκύπτουν από κάποια έκφραση. Πρέπει να τονίσουμε ότι οι υπολογιζόμενες εκφράσεις των παρακάτω παραδειγμάτων δεν αποθηκεύονται στη βάση και γενικά δεν προκαλούν αλλαγές στα περιεχόμενά της.

- 1) **SELECT ISBN, title, price>0 as hasPrice FROM books**
- 2) **SELECT ISBN, title, round(price) as newPrice FROM books**
- 3) **SELECT ISBN, title, price-price\*10/100 as specialPrice FROM books**
- 4) **SELECT CONCAT(title, ' (', ISBN, ')') as fullTitle FROM books**
- 5) **SELECT UPPER(title) as newTitle FROM books**

Ακολουθεί η ερμηνεία και τα αποτελέσματα του κάθε ερωτήματος

- 1) Αντί για το ίδιο το πεδίο **price**, επιστρέφουμε μια λογική τιμή που μας λέει αν το βιβλίο έχει ή όχι τιμή. Η λογική τιμή υπολογίζεται από την έκφραση **price>0** και επιστρέφει 0 ή 1.

ISBN	title	hasPrice
11-222-33	Χίμαιρα	1
11-223-15	Το Δέκα	1
11-223-28	Ηλέκτρα	1
11-226-45	Εντολή	1
11-223-36	Μαραμπού	1
11-245-35	Όνειρο στο κύμα	1
11-225-38	Η φόνισσα	1
11-224-98	Η γυφτοπούλα	1
11-229-78	Η μετανάστρια	1
11-222-85	Θα σας τηλεφωνώ	1

- 2) Με τη χρήση της συνάρτησης **ROUND** της **MySQL** στρογγυλοποιούμε την τιμή και το επιστρεφόμενο αποτέλεσμα αποκαλείται πλέον **newPrice**. Αντί για την **ROUND** θα μπορούσαμε να χρησιμοποιήσουμε κάποια από τις υπόλοιπες **μαθηματικές συναρτήσεις** της **SQL**, όπως τις **ABS**, **SIN**, **COS**, **TAN**, **RAND**, **SQRT** (απόλυτη τιμή, ημίτονο, συνημίτονο, εφαπτομένη, τυχαίος αριθμός, τετραγωνική ρίζα). Μια πλήρη λίστα των διαθέσιμων συναρτήσεων μπορείτε να βρείτε εδώ <http://dev.mysql.com/doc/refman/5.7/en/mathematical-functions.html>.

ISBN	title	newPrice
11-222-33	Χίμαιρα	15
11-223-15	Το Δέκα	19
11-223-28	Ηλέκτρα	22
11-226-45	Εντολή	8
11-223-36	Μαραμπού	11
11-245-35	Όνειρο στο κύμα	22
11-225-38	Η φόνισσα	16
11-224-98	Η γυφτοπούλα	10
11-229-78	Η μετανάστρια	23
11-222-85	Θα σας τηλεφωνώ	17

- 3) Υπολογίζεται και παρουσιάζεται στα αποτελέσματα μια νέα τιμή από τον υπολογισμό μιας κλασσικής αριθμητικής έκφρασης. Το συγκεκριμένο παράδειγμα θα μπορούσε να είναι ο υπολογισμός μιας νέας τιμής με έκπτωση 10%.

ISBN	title	specialPrice
11-222-33	Χίμαιρα	13.878000
11-223-15	Το Δέκα	16.650000
11-223-28	Ηλέκτρα	19.800000
11-226-45	Εντολή	7.002000
11-223-36	Μαραμπού	10.260000
11-245-35	Όνειρο στο κύμα	19.440000
11-225-38	Η φόνισσα	14.220000
11-224-98	Η γυφτοπούλα	8.910000
11-229-78	Η μετανάστρια	20.790000
11-222-85	Θα σας τηλεφωνώ	15.570000

- 4) και 5) εδώ χρησιμοποιούνται δυο συναρτήσεις κειμένου (strings). Με τη συνάρτηση **CONCAT** μπορούμε να συνενώσουμε κείμενα δημιουργώντας ένα νέο κείμενο, ενώ με την **UPPER** μετατρέπουμε ένα κείμενο σε κεφαλαία. Μια πλήρη λίστα και περιγραφή των συναρτήσεων κειμένου μπορούμε να δούμε εδώ <http://dev.mysql.com/doc/refman/5.7/en/string-functions.html>

fullTitle	newTitle
Χίμαιρα (11-222-33)	ΧΙΜΑΙΡΑ
Το Δέκα (11-223-15)	ΤΟ ΔΈΚΑ
Ηλέκτρα (11-223-28)	ΗΛΈΚΤΡΑ
Εντολή (11-226-45)	ΕΝΤΟΛΉ
Μαραμπού (11-223-36)	ΜΑΡΑΜΠΟΥ
Όνειρο στο κύμα (11-245-35)	ΌΝΕΙΡΟ ΣΤΟ ΚΎΜΑ
Η φόνισσα (11-225-38)	Η ΦΟΝΙΣΣΑ
Η γυφτοπούλα (11-224-98)	Η ΓΥΦΤΟΠΟΥΛΑ
Η μετανάστρια (11-229-78)	Η ΜΕΤΑΝΑΣΤΡΙΑ
Θα σας τηλεφωνώ (11-222-85)	ΘΑ ΣΑΣ ΤΗΛΕΦΩΝΩ

## Συναρτήσεις συνόλων (aggregate functions)

Εκτός από τις μαθηματικές συναρτήσεις και τις συναρτήσεις κειμένου, υπάρχει και μια ακόμη οικογένεια συναρτήσεων που ονομάζονται **συναρτήσεις συνόλων**. Οι συναρτήσεις αυτές επιστρέφουν αποτελέσματα κάνοντας υπολογισμούς πάνω στο σύνολο των εγγραφών που επιλέγει το ερώτημα. Έτσι οι επιστρεφόμενες τιμές δεν είναι οι εγγραφές αλλά το αποτέλεσμα της συνάρτησης που εφαρμόζεται σε αυτές.

Οι βασικότερες από αυτές τις συναρτήσεις είναι η **AVG** (μέσος όρος), η **COUNT** (πλήθος), η **MAX** (μέγιστη τιμή), η **MIN** (ελάχιστη τιμή) και η **SUM** (άθροισμα). Πλήρης λίστα και περιγραφή μπορεί να βρεθεί εδώ <http://dev.mysql.com/doc/refman/5.7/en/group-by-functions.html>.

Στο παρακάτω παράδειγμα βλέπουμε πως χρησιμοποιούνται οι συναρτήσεις συνόλων.

```
SELECT COUNT(*) as total, AVG(price) as mesi_timi, MAX(date_added) as neotero, SUM(price) as synoliko_kostos FROM books
```

Σε αυτό το ερώτημα βρίσκουμε το σύνολο των εγγραφών, τη μέση τιμή, την ημερομηνία του νεότερου βιβλίου και το συνολικό κόστος όλων των βιβλίων. Το σύνολο στο οποίο εφαρμόζονται οι συναρτήσεις είναι όλα τα βιβλία του πίνακα (10 βιβλία). Παρ' όλα αυτά το ερώτημα **επιστέφει μόνο μια σειρά δεδομένων**. Αυτό είναι και το νόημα των συναρτήσεων συνόλων. Να εξάγουν δηλαδή συγκεντρωτικά αποτελέσματα για μια ομάδα εγγραφών και όχι να επιστρέφουν τις ίδιες τις εγγραφές.

total	mesi_timi	neotero	synoliko_kostos
10	16.280000	2015-03-29	162.80

Στο προηγούμενο παράδειγμα το σύνολο στο οποίο εφαρμόστηκαν οι συναρτήσεις ήταν όλα τα βιβλία του πίνακα **books**. Ακόμα και αν υπήρχε μια δήλωση **WHERE** που θα περιόριζε τις εγγραφές, αυτές πάλι θα αποτελούσαν μια μόνο ομάδα, απλά πιο μικρή. Σε περίπτωση που εμείς επιθυμούμε να ομαδοποιήσουμε τις εγγραφές ως προς κάποια ιδιότητα, μπορούμε να χρησιμοποιήσουμε τη δήλωση **GROUP BY**. Με τη δήλωση αυτή θέτουμε ένα κριτήριο βάση του οποίου οι εγγραφές ομαδοποιούνται, και οι συναρτήσεις εφαρμόζονται σε κάθε σύνολο ξεχωριστά.

Ας υποθέσουμε ότι **θέλουμε να βρούμε τη μέση τιμή των βιβλίων του κάθε συγγραφέα**. Προφανώς θέλουμε να χρησιμοποιήσουμε την συνάρτηση **AVG**, αλλά χωριστά για τα βιβλία του κάθε συγγραφέα. Αυτό θα μπορούσε να γίνει με το ερώτημα

```
SELECT author, AVG(price) as mesi_timi FROM books GROUP BY author
```

Στα αποτελέσματα του ερωτήματος, που φαίνονται παρακάτω, γίνεται φανερό ότι υπολογίστηκαν 4 μέσοι όροι, ένας για κάθε ομάδα βιβλίων του κάθε συγγραφέα.

author	mesi_timi
1	14.890000
2	14.350000
3	16.960000
4	17.600000

Ενδιαφέρον παρουσιάζει επίσης η περίπτωση όπου κάποια από τις υποψήφιες ομάδες δεν μας ενδιαφέρει να συμπεριληφθεί στα αποτελέσματα. Έστω για παράδειγμα ότι **θέλουμε να μάθουμε τη μέγιστη τιμή από τα βιβλία του κάθε συγγραφέα, αλλά δεν μας ενδιαφέρουν οι συγγραφείς όπου η μέση τιμή είναι κάτω από 15€**. Το σχετικό ερώτημα έχει μετά την **GROUP BY**, μια δήλωση **HAVING** με την οποία ορίζεται ένα κριτήριο για τις ομάδες που θα συμπεριληφθούν.

```
SELECT author, MAX(price) as megisti_timi FROM books
GROUP BY author HAVING AVG(price)>=15
```

Στη περίπτωση αυτή μόνο δυο ομάδες πληρούν το κριτήριο του μέσου όρου

author	megisti_timi
3	18.50
4	23.10

Τέλος μπορούμε να χρησιμοποιήσουμε στην **GROUP BY** περισσότερα από ένα πεδία για να ορίσουμε τις ομάδες. Αν για παράδειγμα θέλαμε οι ομάδες να ορίζονται για κάθε πιθανό ζευγάρι συγγραφέα-εκδοτικού οίκου, τότε το ερώτημα θα γινόταν

```
SELECT author, publication, COUNT(*) as synolo, MAX(price) as megisti_timi FROM
books GROUP BY author, publication
```

Και τα αποτελέσματα θα ήταν 7 ομάδες βιβλίων.

author	publication	synolo	megisti_timi
1	2	2	22.00
2	1	1	11.40
2	4	1	17.30
3	4	2	18.50
4	2	1	21.60
4	3	2	23.10
4	4	1	9.90

## Δήλωση WHERE

Η δήλωση **WHERE** σε ένα ερώτημα **SELECT** χρησιμεύει στο να περιορίσουμε το σύνολο των επιστρεφόμενων εγγραφών, βάση κάποιων κριτηρίων. Η δήλωση **WHERE** συνοδεύεται από μια λογική παράσταση. Από το σύνολο των πληροφοριών που ζητά το ερώτημα, επιστρέφονται μόνο εκείνες που πληρούν το κριτήριο της **WHERE**.

Ακολουθεί ένα απλό παράδειγμα όπου φαίνεται η χρήση της **WHERE**. Σε αυτό ζητάμε **να βρούμε το τίτλο, το ISBN και τη τιμή, μόνο για εκείνα τα βιβλία που έχουν τιμή πάνω 12€.**

```
SELECT title, ISBN, price FROM books WHERE price > 12
```

title	ISBN	price
Χίμαιρα	11-222-33	15.42
Το Δέκα	11-223-15	18.50
Ηλέκτρα	11-223-28	22.00
Όνειρο στο κύμα	11-245-35	21.60
Η φόνισσα	11-225-38	15.80
Η μετανάστρια	11-229-78	23.10
Θα σας τηλεφωνώ	11-222-85	17.30

Όπως και στη λίστα των πεδίων (το πρώτο τμήμα της **SELECT**) έτσι και στην **WHERE** μπορούμε να χρησιμοποιήσουμε συναρτήσεις (μαθηματικές, κειμένου, ημερομηνίας κ.α.), όπως φαίνεται στα παρακάτω παραδείγματα.

```
SELECT title, ISBN, price FROM books WHERE ROUND(price) > 16
```

Σε αυτό το παράδειγμα χρησιμοποιούμε τη συνάρτηση **ROUND** στη συνθήκη της **WHERE**. Άρα αυτό που ζητάμε είναι τα βιβλία που η τιμή τους είναι, μετά από στρογγυλοποίηση, μεγαλύτερη του 16. Ακολουθούν τα αποτελέσματα του ερωτήματος.

title	ISBN	price
Το Δέκα	11-223-15	18.50
Ηλέκτρα	11-223-28	22.00
Όνειρο στο κύμα	11-245-35	21.60
Η μετανάστρια	11-229-78	23.10
Θα σας τηλεφωνώ	11-222-85	17.30

```
SELECT title, ISBN, price, date_added FROM books
WHERE date_added > (NOW() - INTERVAL 10 YEAR)
```

Σε αυτό το παράδειγμα χρησιμοποιείται η συνάρτηση **NOW**, που ανήκει στις συναρτήσεις ημερομηνίας και ώρας (περισσότερα θα βρείτε εδώ



<http://dev.mysql.com/doc/refman/5.7/en/date-and-time-functions.html>). Στο ερώτημα βλέπουμε τη συνθήκη «**date\_added > (NOW() – INTERVAL 10 YEAR)**». Αυτή η συνθήκη ελέγχει αν η ημερομηνία προσθήκης του βιβλίου είναι μεγαλύτερη από την τωρινή ημερομηνία πλην 10 χρόνια. Άρα τα επιστρεφόμενα αποτελέσματα, που φαίνονται παρακάτω, είναι τα βιβλία που προστέθηκαν στη βάση τα τελευταία 10 χρόνια.

title	ISBN	price	date_added
Όνειρο στο κύμα	11-245-35	21.60	2008-06-15
Η φόνισσα	11-225-38	15.80	2009-07-30
Η γυφτοπούλα	11-224-98	9.90	2012-01-28
Η μετανάστρια	11-229-78	23.10	2014-02-16
Θα σας τηλεφωνώ	11-222-85	17.30	2015-03-29

Ιδιαίτερο ενδιαφέρον παρουσιάζουν οι συγκρίσεις αλφαριθμητικών. Μπορούμε φυσικά να χρησιμοποιήσουμε τους συγκριτικούς τελεστές αλλά δεν είναι και πολύ χρηστικοί στις αναζητήσεις βάση κειμένου. Ας δούμε δυο παραδείγματα όπου χρησιμοποιούνται συγκριτικοί τελεστές.

```
SELECT title, ISBN, price, date_added FROM books WHERE title = 'Η φόνισσα'
```

Σε αυτό το παράδειγμα ψάχνουμε να βρούμε αν υπάρχει βιβλίο με ακριβώς το τίτλο 'Η φόνισσα'. Φυσικά αν το βιβλίο υπάρχει θα επιστραφεί ένα αποτέλεσμα.

```
SELECT title, ISBN, price, date_added FROM books WHERE title >= 'Θ'
```

Η παραπάνω αναζήτηση μπορεί να φαίνεται λίγο πιο μπερδεμένη και όχι τόσο χρήσιμη. **Αναζητούμε όλα τα βιβλία των οποίων ο τίτλος αρχίζει από 'Θ' ή οποιοδήποτε επόμενο γράμμα του αλφάβητου.** Τα αποτελέσματα από την εκτέλεση είναι τα εξής:

title	ISBN	price	date_added
Χίμαιρα	11-222-33	15.42	1995-05-15
Το Δέκα	11-223-15	18.50	1996-03-28
Μαραμπού	11-223-36	11.40	2001-11-02
Όνειρο στο κύμα	11-245-35	21.60	2008-06-15
Θα σας τηλεφωνώ	11-222-85	17.30	2015-03-29

Η πιο ενδιαφέρουσα επιλογή μας όταν θέλουμε να κάνουμε αναζητήσεις βάση κειμένου είναι ο τελεστής **LIKE**. Η **LIKE** μας βοηθά να εντοπίσουμε κείμενα σε πεδία τα οποία ακολουθούν ένα μοτίβο. Το μοτίβο που μας ενδιαφέρει δημιουργείται με **τη χρήση δυο χαρακτήρων μπαλαντέρ, το '\_' και το '%'.**

Ο χαρακτήρας % χρησιμοποιείται για να υποδηλώσει μια ακολουθία αδιάφορων χαρακτήρων, αδιευκρίνιστου πλήθους.

Ο χαρακτήρας `_` χρησιμοποιείται για να υποδηλώσει ακριβώς έναν αδιάφορο χαρακτήρα.

Ας δούμε κάποια παραδείγματα ερωτημάτων με **LIKE** και διάφορα μοτίβα κειμένου. Πρώτα θα διατυπώσουμε το ζητούμενο και μετά θα δούμε το σχετικό ερώτημα και τέλος τα αντίστοιχα αποτελέσματα από τη βάση.

Να βρεθούν όλα τα βιβλία που ο τίτλος τους:

- Αρχίζει από 'Η'
- Περιέχει το string 'αι' (οπουδήποτε)
- Ο τρίτος χαρακτήρας του είναι το 'μ'
- Τελειώνει σε 'α'

```
SELECT * FROM books WHERE title LIKE 'Η%'
```

```
SELECT * FROM books WHERE title LIKE '%αι%'
```

```
SELECT * FROM books WHERE title LIKE '__μ%'
```

```
SELECT * FROM books WHERE title LIKE '%α'
```

	ID	ISBN	title	author	publication	price	status	date_added
<input type="checkbox"/> Edit Copy Delete	103	11-223-28	Ηλέκτρα	1	2	22.00	1	1997-04-02
<input type="checkbox"/> Edit Copy Delete	107	11-225-38	Η φόνισσα	4	3	15.80	1	2009-07-30
<input type="checkbox"/> Edit Copy Delete	108	11-224-98	Η γυφτοπούλα	4	4	9.90	1	2012-01-28
<input type="checkbox"/> Edit Copy Delete	109	11-229-78	Η μετανάστρια	4	3	23.10	1	2014-02-16

	ID	ISBN	title	author	publication	price	status	date_added
<input type="checkbox"/> Edit Copy Delete	101	11-222-33	Χίμαιρα	3	4	15.42	1	1995-05-15

	ID	ISBN	title	author	publication	price	status	date_added
<input type="checkbox"/> Edit Copy Delete	101	11-222-33	Χίμαιρα	3	4	15.42	1	1995-05-15
<input type="checkbox"/> Edit Copy Delete	109	11-229-78	Η μετανάστρια	4	3	23.10	1	2014-02-16

	ID	ISBN	title	author	publication	price	status	date_added
<input type="checkbox"/> Edit Copy Delete	101	11-222-33	Χίμαιρα	3	4	15.42	1	1995-05-15
<input type="checkbox"/> Edit Copy Delete	102	11-223-15	Το Δέκα	3	4	18.50	1	1996-03-28
<input type="checkbox"/> Edit Copy Delete	103	11-223-28	Ηλέκτρα	1	2	22.00	1	1997-04-02
<input type="checkbox"/> Edit Copy Delete	106	11-245-35	Όνειρο στο κύμα	4	2	21.60	0	2008-06-15
<input type="checkbox"/> Edit Copy Delete	107	11-225-38	Η φόνισσα	4	3	15.80	1	2009-07-30
<input type="checkbox"/> Edit Copy Delete	108	11-224-98	Η γυφτοπούλα	4	4	9.90	1	2012-01-28
<input type="checkbox"/> Edit Copy Delete	109	11-229-78	Η μετανάστρια	4	3	23.10	1	2014-02-16

Στη συνθήκη της WHERE μπορούμε να βάλουμε και σύνθετες λογικές παρατάξεις χρησιμοποιώντας τους λογικούς τελεστές **and**, **or** και **not**. Οι τελεστές αυτοί εφαρμόζονται επάνω σε λογικές ποσότητες (που μπορεί και οι ίδιες να είναι ολόκληρες εκφράσεις) και τις συνδυάζουν για να δώσουν μια λογική τιμή σαν αποτέλεσμα.

**AND:** Χρησιμοποιείται όταν θέλουμε δυο ή περισσότερα γεγονότα να συμβαίνουν ταυτόχρονα. Δίνει σαν αποτέλεσμα 1 (αληθές) όταν όλα τα ορίσματα είναι 1 (αληθή).

**OR:** Χρησιμοποιείται όταν θέλουμε να συμβαίνει τουλάχιστον ένα από τα συνδυαζόμενα γεγονότα. Δίνει σαν αποτέλεσμα 1 (αληθές) όταν ένα τουλάχιστον από τα ορίσματα είναι 1 (αληθή).

**NOT:** Εφαρμόζεται σε ένα μόνο γεγονός αντιστρέφοντας τη σημασία του. Έτσι αν μια συνθήκη βγει 1, θα γίνει 0 με την εφαρμογή της NOT και αντίστροφα.

Ας δούμε κάποια σχετικά παραδείγματα και τα αποτελέσματά τους. Να βρεθούν όλα τα βιβλία:

- Του συγγραφέα 4 που κοστίζουν πάνω από 20€
- Των εκδοτικών οίκων 1 και 3
- Που ο τίτλος τους είναι μονολεκτικός (δεν έχει κενό)
- Του συγγραφέα 4 και των εκδοτικών οίκων 3 ή 4

**SELECT \* FROM books WHERE author=4 AND price > 20**

**SELECT \* FROM books WHERE publication = 1 OR publication = 3**

**SELECT \* FROM books WHERE title NOT LIKE '% %'**

**SELECT \* FROM books WHERE author=4 AND (publication = 3 OR publication = 4)**

ID	ISBN	title	author	publication	price	status	date_added
106	11-245-35	Όνειρο στο κύμα	4	2	21.60	0	2008-06-15
109	11-229-78	Η μετανάστρια	4	3	23.10	1	2014-02-16

ID	ISBN	title	author	publication	price	status	date_added
105	11-223-36	Μαραμπού	2	1	11.40	1	2001-11-02
107	11-225-38	Η φόνισσα	4	3	15.80	1	2009-07-30
109	11-229-78	Η μετανάστρια	4	3	23.10	1	2014-02-16

ID	ISBN	title	author	publication	price	status	date_added
101	11-222-33	Χίμαιρα	3	4	15.42	1	1995-05-15
103	11-223-28	Ηλέκτρα	1	2	22.00	1	1997-04-02
104	11-226-45	Εντολή	1	2	7.78	1	1999-10-25
105	11-223-36	Μαραμπού	2	1	11.40	1	2001-11-02

ID	ISBN	title	author	publication	price	status	date_added
107	11-225-38	Η φόνισσα	4	3	15.80	1	2009-07-30
108	11-224-98	Η γυφτοπούλα	4	4	9.90	1	2012-01-28
109	11-229-78	Η μετανάστρια	4	3	23.10	1	2014-02-16

Στο 2<sup>ο</sup> από τα προηγούμενα παραδείγματα βλέπουμε τη συνθήκη «publication = 1 OR publication=3». Όταν οι πιθανές τιμές που θέλω να ελέγξω για ένα πεδίο είναι πολλές, τότε δε βολεύει να χρησιμοποιήσουμε αυτή τη γραφή, διότι η παράσταση μπορεί να γίνει

πολύ μεγάλη. Αν για παράδειγμα θέλουμε να δούμε αν ο εκδοτικός οίκος είναι ο 1 ή 2 ή 3, αυτό θα μπορούσε να γραφεί με δυο τρόπου

```
SELECT * FROM books WHERE publication = 1 OR publication = 2 OR publication = 3
```

```
SELECT * FROM books WHERE publication IN (1,2,3)
```

Με το τελεστή **IN** μπορούμε απευθείας να ρωτήσουμε αν ένα πεδίο ανήκει σε ένα σύνολο τιμών. Αυτός ο τελεστής είναι πολύ χρήσιμος σε περιπτώσεις όπου το σύνολο δεν είναι συγκεκριμένο, αλλά προκύπτει από δευτερεύον ερώτημα. Τέτοια παραδείγματα θα δούμε στις **εμφωλευμένες SELECT**

## Δήλωση ORDER BY

Τα αποτελέσματα ενός ερωτήματος **SELECT** συνήθως είναι ταξινομημένα ως προς το πρωτεύον κλειδί σε αύξουσα σειρά. Δηλαδή από το παλαιότερο προς το νεότερο. Πολλές φορές όμως θέλουμε μια διαφορετική διάταξη των αποτελεσμάτων. Τη διάταξη αυτή μπορούμε να την ορίσουμε με τη δήλωση **ORDER BY**, που τοποθετείται μετά τη δήλωση **WHERE**. Στην **ORDER BY** Μπορούμε να βάλουμε ένα ή περισσότερα πεδία βάση των οποίων θα γίνει η ταξινόμηση και μετά να δηλώσουμε αν θέλουμε να είναι **αύξουσα (ASCENDING)** ή **φθίνουσα (DESCENDING)**.

Ας δούμε μερικά παραδείγματα. Να βρεθούν όλα τα βιβλία ταξινομημένα ως προς:

- Τη τιμή σε αύξουσα τάξη μεγέθους
- Την ημερομηνία εισαγωγής από το νεότερο στο παλαιότερο
- Ως προς τον συγγραφέα και τον εκδοτικό οίκο (αύξουσα)
- Ως προς τον συγγραφέα (φθίνουσα) και τον τίτλο (αύξουσα)

```
SELECT * FROM books ORDER BY price ASC
```

```
SELECT * FROM books ORDER BY date_added DESC
```

```
SELECT * FROM books ORDER BY author, publication ASC
```

```
SELECT * FROM books ORDER BY author DESC, title ASC
```

	ID	ISBN	title	author	publication	price	status	date_added
<input type="checkbox"/> Edit Copy Delete	104	11-226-45	Εντολή	1	2	7.78	1	1999-10-25
<input type="checkbox"/> Edit Copy Delete	108	11-224-98	Η γυφτοπούλα	4	4	9.90	1	2012-01-28
<input type="checkbox"/> Edit Copy Delete	105	11-223-36	Μαραμπού	2	1	11.40	1	2001-11-02
<input type="checkbox"/> Edit Copy Delete	101	11-222-33	Χίμαιρα	3	4	15.42	1	1995-05-15
<input type="checkbox"/> Edit Copy Delete	107	11-225-38	Η φόνισσα	4	3	15.80	1	2009-07-30
<input type="checkbox"/> Edit Copy Delete	110	11-222-85	Θα σας τηλεφωνώ	2	4	17.30	0	2015-03-29
<input type="checkbox"/> Edit Copy Delete	102	11-223-15	Το Δέκα	3	4	18.50	1	1996-03-28
<input type="checkbox"/> Edit Copy Delete	106	11-245-35	Όνειρο στο κύμα	4	2	21.60	0	2008-06-15
<input type="checkbox"/> Edit Copy Delete	103	11-223-28	Ηλέκτρα	1	2	22.00	1	1997-04-02
<input type="checkbox"/> Edit Copy Delete	109	11-229-78	Η μετανάστρια	4	3	23.10	1	2014-02-16

	ID	ISBN	title	author	publication	price	status	date_added
<input type="checkbox"/> Edit Copy Delete	110	11-222-85	Θα σας τηλεφωνώ	2	4	17.30	0	2015-03-29
<input type="checkbox"/> Edit Copy Delete	109	11-229-78	Η μετανάστρια	4	3	23.10	1	2014-02-16
<input type="checkbox"/> Edit Copy Delete	108	11-224-98	Η γυφτοπούλα	4	4	9.90	1	2012-01-28
<input type="checkbox"/> Edit Copy Delete	107	11-225-38	Η φόνισσα	4	3	15.80	1	2009-07-30
<input type="checkbox"/> Edit Copy Delete	106	11-245-35	Όνειρο στο κύμα	4	2	21.60	0	2008-06-15
<input type="checkbox"/> Edit Copy Delete	105	11-223-36	Μαραμπού	2	1	11.40	1	2001-11-02
<input type="checkbox"/> Edit Copy Delete	104	11-226-45	Εντολή	1	2	7.78	1	1999-10-25
<input type="checkbox"/> Edit Copy Delete	103	11-223-28	Ηλέκτρα	1	2	22.00	1	1997-04-02
<input type="checkbox"/> Edit Copy Delete	102	11-223-15	Το Δέκα	3	4	18.50	1	1996-03-28
<input type="checkbox"/> Edit Copy Delete	101	11-222-33	Χίμαιρα	3	4	15.42	1	1995-05-15

	ID	ISBN	title	author	publication	price	status	date_added
<input type="checkbox"/> Edit Copy Delete	103	11-223-28	Ηλέκτρα	1	2	22.00	1	1997-04-02
<input type="checkbox"/> Edit Copy Delete	104	11-226-45	Εντολή	1	2	7.78	1	1999-10-25
<input type="checkbox"/> Edit Copy Delete	105	11-223-36	Μαραμπού	2	1	11.40	1	2001-11-02
<input type="checkbox"/> Edit Copy Delete	110	11-222-85	Θα σας τηλεφωνώ	2	4	17.30	0	2015-03-29
<input type="checkbox"/> Edit Copy Delete	101	11-222-33	Χίμαιρα	3	4	15.42	1	1995-05-15
<input type="checkbox"/> Edit Copy Delete	102	11-223-15	Το Δέκα	3	4	18.50	1	1996-03-28
<input type="checkbox"/> Edit Copy Delete	106	11-245-35	Όνειρο στο κύμα	4	2	21.60	0	2008-06-15
<input type="checkbox"/> Edit Copy Delete	107	11-225-38	Η φόνισσα	4	3	15.80	1	2009-07-30
<input type="checkbox"/> Edit Copy Delete	109	11-229-78	Η μετανάστρια	4	3	23.10	1	2014-02-16
<input type="checkbox"/> Edit Copy Delete	108	11-224-98	Η γυφτοπούλα	4	4	9.90	1	2012-01-28

	ID	ISBN	title	author	publication	price	status	date_added
<input type="checkbox"/> Edit Copy Delete	108	11-224-98	Η γυφτοπούλα	4	4	9.90	1	2012-01-28
<input type="checkbox"/> Edit Copy Delete	109	11-229-78	Η μετανάστρια	4	3	23.10	1	2014-02-16
<input type="checkbox"/> Edit Copy Delete	107	11-225-38	Η φόνισσα	4	3	15.80	1	2009-07-30
<input type="checkbox"/> Edit Copy Delete	106	11-245-35	Όνειρο στο κύμα	4	2	21.60	0	2008-06-15
<input type="checkbox"/> Edit Copy Delete	102	11-223-15	Το Δέκα	3	4	18.50	1	1996-03-28
<input type="checkbox"/> Edit Copy Delete	101	11-222-33	Χίμαιρα	3	4	15.42	1	1995-05-15
<input type="checkbox"/> Edit Copy Delete	110	11-222-85	Θα σας τηλεφωνώ	2	4	17.30	0	2015-03-29
<input type="checkbox"/> Edit Copy Delete	105	11-223-36	Μαραμπού	2	1	11.40	1	2001-11-02
<input type="checkbox"/> Edit Copy Delete	104	11-226-45	Εντολή	1	2	7.78	1	1999-10-25
<input type="checkbox"/> Edit Copy Delete	103	11-223-28	Ηλέκτρα	1	2	22.00	1	1997-04-02

Στην δήλωση **ORDER BY** μπορούν να μπούνε, εκτός από τα πεδία των εμπλεκόμενων πινάκων, πεδία που μπορεί να έχουμε δημιουργήσει εμείς με τη χρήση εκφράσεων ή κάποιες εκφράσεις απευθείας. Για παράδειγμα στη παρακάτω εντολή έχουμε μια έκφραση στην **ORDER BY** όπου υπολογίζεται το δεκαδικό μέρος της κάθε τιμής και η ταξινόμηση γίνεται βάση αυτής. Το ίδιο το αποτέλεσμα της έκφρασης δε θα παρουσιαστεί στα αποτελέσματα αν δε το βάλουμε στη λίστα των πεδίων, παρά μόνο θα επηρεάσει τη ταξινόμηση. (Η **FLOOR** είναι μια μαθηματική συνάρτηση που επιστρέφει τον το ακέραιο μέρος μιας δεκαδικής τιμής).

```
SELECT * FROM books ORDER BY price-FLOOR(price) ASC
```

Ή ισοδύναμα

```
SELECT *, (price-FLOOR(price)) AS new_price FROM books ORDER BY new_price ASC
```