

Βάσεις Δεδομένων



Κεφάλαιο 6

SQL – Select(συνεχεια)

SQL:παράδειγμα εντολής

- ΙΣ (κωδι, όνομα, επίπεδο) -> ΙΣΤΙΟΠΛΟΟΙ
- ΣΚ (κωδσ, όνομα, χρώμα) -> ΣΚΑΦΗ
- ΚΡ (κωδι, κωδσ, ημερομηνία) -> ΚΡΑΤΗΣΕΙΣ

Ερώτημα

Τα ονόματα των ιστ/ων που έχουν κάνει κράτηση σε κόκκινο ή πράσινο σκάφος;

SELECT

FROM

WHERE

SQL: παράδειγμα εντολής

- ΙΣ (κωδι, όνομα, επίπεδο) -> ΙΣΤΙΟΠΛΟΟΙ
- ΣΚ (κωδσ, όνομα, χρώμα) -> ΣΚΑΦΗ
- ΚΡ (κωδι, κωδσ, ημερομηνία) -> ΚΡΑΤΗΣΕΙΣ

-
- Ερώτημα

Τα ονόματα των ιστ/ων που έχουν κάνει κράτηση σε κόκκινο ή πράσινο σκάφος;

SELECT ι.όνομα

FROM ΙΣ ι, ΚΡ κ, ΣΚ σ

WHERE ι.κωδι=κ.κωδι and κ.κωδσ=σ.κωδσ and (σ.χρώμα="κόκκινο" or σ.χρώμα="πράσινο")

SQL:παράδειγμα εντολής

- ΙΣ (κωδι, όνομα, επίπεδο) -> ΙΣΤΙΟΠΛΟΟΙ
- ΣΚ (κωδσ, όνομα, χρώμα) -> ΣΚΑΦΗ
- ΚΡ (κωδι, κωδσ, ημερομηνία) -> ΚΡΑΤΗΣΕΙΣ

Ερώτημα

Τα ονόματα των ιστ/ων που έχουν κάνει κράτηση σε κάποιο κόκκινο και σε κάποιο πράσινο σκάφος

- **SELECT** ι.όνομα
- **FROM** ΙΣ ι, ΚΡ κ1, ΣΚ σ1, ΚΡ κ2, ΣΚ σ2
- **WHERE** ι.κωδι=κ1.κωδι and κ1.κωδσ=σ1.κωδσ and σ1.χρώμα="κόκκινο" and ι.κωδι=κ2.κωδι and κ2.κωδσ=σ2.κωδσ and σ2.χρώμα="πράσινο"

SQL:παράδειγμα εντολής

- ΙΣ (κωδι, όνομα, επίπεδο) -> ΙΣΤΙΟΠΛΟΟΙ
 - ΣΚ (κωδσ, όνομα, χρώμα) -> ΣΚΑΦΗ
 - ΚΡ (κωδι, κωδσ, ημερομηνία) -> ΚΡΑΤΗΣΕΙΣ
-

Ερώτημα

Τα ονόματα των ιστιοπλόων που έχουν κάνει κράτηση δύο διαφορετικά σκάφη την ίδια μέρα;

- **SELECT DISTINCT** ι.όνομα
- **FROM** ΙΣ ι, ΚΡ κ1, ΚΡ κ2
- **WHERE** ι.κωδι=κ1.κωδι and ι.κωδι=κ2.κωδι and κ1.ημερομηνία=κ2.ημερομηνία and κ1.κωδσ <> κ2.κωδσ

SQL: παράδειγμα εντολής

- ΙΣ (κωδι, όνομα, επίπεδο) -> ΙΣΤΙΟΠΛΟΟΙ
 - ΣΚ (κωδσ, όνομα, χρώμα) -> ΣΚΑΦΗ
 - ΚΡ (κωδι, κωδσ, ημερομηνία) -> ΚΡΑΤΗΣΕΙΣ
-

Ερώτημα

Τα ονόματα των ιστιοπλόων που έχουν κάνει κράτηση δύο διαφορετικά σκάφη την ίδια μέρα; Δώσε μου τους με bonus στο επίπεδό τους +2

SELECT DISTINCT ι.όνομα, νέο-επίπεδο=ι.επίπεδο+2

FROM ΙΣ ι, ΚΡ κ1, ΚΡ κ2

WHERE ι.κωδι=κ1.κωδι and ι.κωδι=κ2.κωδι and κ1.ημερομηνία=κ2.ημερομηνία and κ1.κωδσ <>κ2.κωδσ

Αριθμητικές εκφράσεις στη συνθήκη (WHERE)

- Ζεύγη ονομάτων ιστοπλόων όπου το διπλάσιο επίπεδο του ενός ισούται με το επίπεδο του άλλου συν 1
- **SELECT** όνομα1 =ι1.όνομα, όνομα2 =ι2.όνομα
- **FROM** ΙΣ ι1, ΙΣ ι2
- **WHERE** 2 * ι1.επίπεδο = ι2.επίπεδο +1

- Οι ατομικές συνθήκες μπορεί να είναι της μορφής
 - <αριθμητική έκφραση1> τ <αριθμητική έκφραση2>
 - Όπου τ ανήκει {= < > <= > >=}

Αριθμητικές εκφράσεις στο SELECT

Στη λίστα αποτελεσμάτων (SELECT), μπορούμε να βάζουμε στοιχεία της μορφής:

<όνομα πεδίου> = <αριθμητική έκφραση>

ή

<αριθμητική έκφραση> **AS**<όνομα πεδίου>

ώστε στο αποτέλεσμα να έχουμε (αν χρειάζεται) νέα ονόματα πεδίων ή/και πεδία που προκύπτουν από πολύπλοκους συνδυασμούς.

Εμφάνιση στήλης με άλλο όνομα

- **SELECT** π. Κωδικός_Προϊόντος AS Κωδικός,
κ.Ονομασία AS Όνομα_Είδους
- **FROM** ΠΡΟΪΟΝΤΑ π, ΚΑΤΗΓΟΡΙΕΣ κ

Εγγραφή	<u>Κωδικός Προϊόντος</u>	Περιγραφή	Τιμή Αγοράς	Κωδικός Προμηθευτή	Κατηγορία
π1	1001	Στυλό	0.75	Π200	1
π2	1002	Μολύβι	1	Π250	1
π3	1003	Μαρκαδόρος	1.3	Π130	5
π4	1004	Δισκέτα 3,5"	0.8	Π120	2

Προϊόντα

Εγγραφή	Κωδικός Κατηγορίας	Ονομασία
κ1	1	Είδη γραφείου
κ2	2	Είδη υπολογιστή
κ3	3	Υπολογιστής

Κατηγορίες

Οι μεταβλητές πλειάδας δεν είναι υποχρεωτικές

- Όταν έχουμε ασαφή ονόματα ή ίδια ονόματα, για να το αποφύγουμε, προσδιορίζουμε **μεταβλητές πλειάδας** (ή ψευδώνυμο –alias), δηλαδή το όνομα κάθε χαρακτηριστικού με το όνομα της αντίστοιχης σχέσης.
- Όμως, οι μεταβλητές πλειάδας μπορούν να μην αναφέρονται αν υπονοούνται από το πεδίο που περιέχουν.

Οι μεταβλητές πλειάδας δεν είναι υποχρεωτικές

- Το όνομα ενός πίνακα μπορεί να χρησιμεύσει κατευθείαν ως μεταβλητή πλειάδας αν είναι η μοναδική
 - Π.χ. Ονόματα ηλικίες και ονόματα τμημάτων υπαλλήλων που είναι πάνω από 40 και δουλεύουν στον 1ο όροφο:
 - **SELECT** ΥΠ.όνομα, ηλικία, ΤΜ.όνομα
 - **FROM** ΥΠ, ΤΜ
 - **WHERE** ΥΠ.κωδτ=ΤΜ.κωδτ and ηλικία >40 and όροφος =1
- Δεν ορίζουμε μετ. πλειαδος για το πεδιο ηλικια
- Υπονοειται ότι
ΥΠ ΥΠ
ΤΜ ΤΜ

Εμφάνιση εγγραφών σε συγκεκριμένη σειρά

- SQL: ORDER BY
- Π.χ. για να εμφανιστούν όλα τα δεδομένα του πίνακα σε *αύξουσα* σειρά ως προς την τιμή:
- Εμφάνιση εγγραφών σε συγκεκριμένη σειρά

```
ΠΡΟΪΟΝΤΑ  
SELECT*  
FROM ΠΡΟΪΟΝΤΑ  
ORDER BY Τιμή  
Αγοράς ASC
```

Εγγραφή	<u>Κωδικός Προϊόντος</u>	Περιγραφή	Τιμή Αγοράς	Κωδικός Προμηθευτή	Κατηγορία
π1	1001	Στυλό	0.75	Π200	1
π2	1002	Μολύβι	1	Π250	1
π3	1003	Μαρκαδόρος	1.3	Π130	5
π4	1004	Δισκέτα 3,5"	0.8	Π120	2

SQL: ORDER BY

Π.χ. για να εμφανιστούν όλα τα δεδομένα του πίνακα σε *φθίνουσα* σειρά ως προς την ποσότητα:

SELECT *

FROM ΠΡΟΪΟΝΤΑ

ORDER BY Ποσότητα DESC



Όνομα προϊόντος	Τιμή μονάδας	Ποσότητα
ΠΕΠΟΝΙ	880 ΔΡΧ	300
ΦΡΑΟΥΛΑ	150ΔΡΧ	1100
ΜΗΛΟ	120ΔΡΧ	1700

SQL: ORDER BY

- Π.χ. *αύξουσα* σειρά ως προς την ποσότητα κι *φθίνουσα* ως προς την τιμή:
- SQL: ORDER BY
SELECT *
- **FROM** ΠΡΟΪΟΝΤΑ
- **ORDER BY** Ποσότητα ASC, Τιμή_μονάδας DESC

Όνομα προϊόντος	Τιμή μονάδας	Ποσότητα
ΑΧΛΑΔΙ	110ΔΡΧ	1100
ΦΡΑΟΥΛΑ	150ΔΡΧ	1100
ΠΕΠΟΝΙ	880 ΔΡΧ	300
ΜΗΛΟ	120ΔΡΧ	1700

SQL: ORDER BY

- Π.χ. *αύξουσα* σειρά ως προς την ποσότητα κι *φθίνουσα* ως προς την τιμή:
- SQL: ORDERBY
- **SELECT ***
- **FROM ΠΡΟΪΟΝΤΑ**
- **ORDER BY Ποσότητα ASC, Τιμή_μονάδαςDESC**

Όνομα προϊόντος	Τιμή μονάδας	Ποσότητα
ΠΕΠΟΝΙ	880 ΔΡΧ	300
ΦΡΑΟΥΛΑ	150ΔΡΧ	1100
ΑΧΛΑΔΙ	110ΔΡΧ	1100
ΜΗΛΟ	120ΔΡΧ	1700

LIKE %

- Εάν δεν είμαστε σίγουροι για αυτό που αναζητούμε, μπορούμε να βάλουμε %
- **SELECT ***
- **FROM ΦΡΟΥΤΑ**
- **WHERE όνομα_προϊόντος LIKE '%ΟΝΙ';**

Όνομα προϊόντος	Τιμή μονάδας
ΠΕΠΟΝΙ	880 ΔΡΧ
ΦΡΑΟΥΛΑ	150ΔΡΧ
ΜΗΛΟ	120ΔΡΧ
ΛΕΜΟΝΙ	200ΔΡΧ

Όνομα προϊόντος	Τιμή μονάδας
ΠΕΠΟΝΙ	880 ΔΡΧ
ΛΕΜΟΝΙ	200ΔΡΧ

Συνολοθεωρητικοί τελεστές

- Σύνολα στην SQL (συνολοθεωρητικοί τελεστές):
 - ένωση, τομή, διαφορά
 - union, intersect, except
- Πράξεις σε ομοειδή σύνολα, δηλαδή συμβατά τμήματα SELECT στα ερωτήματα

Συνολοθεωρητικοί τελεστές

ένωση, τομή, διαφορά
union, intersect, except

- σύνταξη:
- **(SELECT... FROM...**
- **WHERE...)**
- UNION | INTERSECT | EXCEPT
- **(SELECT... FROM...**
- **WHERE...)**



Συνολοθεωρητικοί τελεστές - EXCEPT

- Π.χ. κωδικοί υπαλλήλων που βγάζουν πάνω από 600 ευρώ και δε δουλεύουν στα «τυριά»
- (**SELECT** κωδυ **FROM** ΥΠ **WHERE** μισθός >600)
- **EXCEPT**
- (**SELECT** κωδυ **FROM** ΥΠ, ΤΜ **WHERE** ΥΠ.κωδτ=ΤΜ.κωδτ and ΤΜ.όνομα = «τυριά»)

Συνολοθεωρητικοί τελεστές: UNION

- Π.χ. όλοι οι κωδικοί έργων στα οποία εμφανίζονται εργαζόμενοι με επίθετο Smith, είτε είναι διευθυντές είτε απασχολούμενοι στο έργο

(**SELECT** DISTINCT κωδ_έργου **FROM** ΕΡΓΟ, ΤΜΗΜΑ, ΕΡΓΑΖΟΜΕΝΟΣ **WHERE** κ_τμήμα=κωδ_τμήμα AND διευθυντής=ΑΔΤ AND Επίθετο="Smith") UNION

(**SELECT** DISTINCT κωδ_έργου **FROM** ΕΡΓΟ, ΑΠΑΣΧΟΛΗΣΗ, ΕΡΓΑΖΟΜΕΝΟΣ **WHERE** κ_έργο=κωδ_έργου AND ε_ΑΔΤ=ΑΔΤ AND Επίθετο="Smith")

Εμφωλιασμένα ερωτήματα

Ορισμένες φορές είναι απαραίτητο το αποτέλεσμα μιας αρχικής αναζήτησης να χρησιμοποιηθεί από μια 2^η αναζήτηση.

Εμφωλιασμένο ή εμφωλευμένο ερώτημα
(nested query)

ή

υποερώτημα(sub-query)

Εμφωλιασμένα ερωτήματα

- Σε κάθε σημείο ενός ερωτήματος όπου αναμένεται ένας πίνακας, μπορεί να γραφτεί μια εντολή SQL **εμφωλιασμένη** στην αρχική εντολή και να χρησιμοποιηθεί ως πίνακας το αποτέλεσμα της.
- Μπορούν να εμφανισθούν σε όλα τα τμήματα μιας SELECT, αλλά εμείς θα δώσουμε βάρος μόνο στο WHERE (κι όχι στα SELECT, FROM).

Εμφωλιασμένα ερωτήματα

- Σύνταξη (δομή):
- (**SELECT... FROM... WHERE...**)
- <τελεστής>
- (**SELECT... FROM... WHERE...**)
- Το εσωτερικό (εμφωλιασμένο) υπο-ερώτημα υπολογίζεται για κάθε γραμμή (πλειάδα) του εξωτερικού ερωτήματος

Εμφωλιασμένα ερωτήματα

- Άρα μιλάμε για πλήρεις προτάσεις `SELECT ... FROM... WHERE` μέσα σε μια πρόταση `WHERE` κάποιας άλλης ερώτησης.

εξωτερικό ερώτημα

εσωτερικό ερώτημα

- Θα εισάγουμε καινούργιο συντακτικό, συγκεκριμένα τον τελεστή (συνδετικό) **IN**, που ελέγχει την παρουσία στην εμφωλιασμένη εντολή.

Εμφωλιασμένα ερωτήματα:IN

- **r.AIN** <εντολή SELECT>
- Αποτιμάται σε:
 - *αληθές(true)* αν η τιμή r. Αυπάρχει στο σύνολο που προκύπτει από την εκτέλεση της εντολής SELECT
 - *Αλλιώς σε ψευδές(false)*
- π.χ.
- 5 IN (0, 5, 6) επιστρέφει true
- 5 IN (0, 6, 9) επιστρέφει false

Εμφωλιασμένα ερωτήματα: IN – Παράδειγμα

- Ονόματα υπαλλήλων που δουλεύουν στον 1^ο όροφο;

κωδ	όνομα	μισθός	κωδτ
44	Ελένη	...	106
82	Πάρις		80
7	Μενέλαος		42
100	Αχιλλέας		123
11	Ηλέκτρα		17
30	Διομήδης		42

κωδτ	όνοματ	όροφος
17	...	3
80		1
42		2
106		2
123		1

Εμφωλιασμένα ερωτήματα: IN – Παράδειγμα

- Ονόματα υπαλλήλων που δουλεύουν στον 1^ο όροφο;

κωδυ	όνομα	μισθός	κωδτ
44	Ελένη	...	106
82	Πάρις		80
7	Μενέλαος		42
100	Αχιλλέας		123
11	Ηλέκτρα		17
30	Διομήδης		42

κωδτ	όνοματ	όροφος
17	...	3
80		1
42		2
106		2
123		1

```
SELECT υ.όνομα FROM ΥΠυ,ΤΜ τ WHERE  
τ.όροφος=1 AND υ.κωδτ=τ.κωδτ
```

Εμφωλιασμένα ερωτήματα: IN – Παράδειγμα

- Ονόματα υπαλλήλων που δουλεύουν στον 1^ο όροφο;

κωδυ	όνομα	μισθός	κωδτ
44	Ελένη	...	106
82	Πάρις		80
7	Μενέλαος		42
100	Αχιλλέας		123
11	Ηλέκτρα		17
30	Διομήδης		42

κωδτ	όνοματ	όροφος
17	...	3
80		1
42		2
106		2
123		1

```
SELECT υ.όνομα FROM ΥΠ υ WHERE  
υ.κωδτ IN  
(SELECT τ.κωδτ FROM ΤΜ τ WHERE  
τ.όροφος=1)
```

Επίπεδο ερώτημα & ισοδύναμο εμφωλιασμένο

- **SELECT** υ.όνομα
- **FROM** ΥΠυ,ΤΜ τ
- **WHERE** τ.όροφος=1 **AND**
υ.κωδτ=τ.κωδτ
- **SELECT** υ.όνομα
- **FROM** ΥΠ υ
- **WHERE** υ.κωδτ **IN**
- (**SELECT** τ.κωδτ
- **FROM** ΤΜτ
- **WHERE** τ.όροφος=1)

Σημασιολογία εμφωλιασμού

- Για κάθε εμφωλιασμένο ερώτημα δημιουργείται **μια νέα στήλη στο καρτεσιανό γινόμενο FROM** όπου καταγράφεται το αποτέλεσμα του εμφωλιασμένου ερωτήματος για κάθε εγγραφή του καρτεσιανού γινομένου.
- Η συνθήκη (και όποια άλλη είναι δυνατή) αποτιμάται ανάλογα με τις τιμές όλων των στηλών, μαζί και των νέων.

Σημασιολογία εμφωλιασμού

- Φτιάχνουμε το καρτεσιανό γινόμενο του FROM.
- Μετά πάμε στον εμφωλιασμό (εσωτερικό ερώτημα) και βρίσκουμε το αποτέλεσμα.
- Φτιάχνουμε μια **νέα στήλη**. Εκεί μέσα θα βάλουμε την αποτίμηση του εσωτερικού ερωτήματος.
- Το αποτέλεσμα, ως **σύνολο**, το βάζουμε ως τιμή στην καινούργια στήλη..

Εμφωλιασμένα ερωτήματα:IN –Παράδειγμα 1

- Ονόματα υπαλλήλων που δουλεύουν στον 1^ο όροφο:

SELECT όνομα **FROM** ΥΠ **WHERE** Εκωδτ **IN**

(SELECT κωδτ **FROM** ΤΜ **WHERE** όροφος=1)

Αποτέλεσμα

όνομα
Πάρις
Αχιλλέας

Εμφωλιασμένα ερωτήματα: IN – Παράδειγμα 2

Όλες τις πληροφορίες για τα προϊόντα των οποίων ο όγκος πωλήσεων είναι μεγαλύτερος ή ίσος του 1000:

```
SELECT * FROM ΠΡΟΪΟΝΤΑ
```

```
WHERE Κωδ_προϊόντος IN
```

```
(SELECT Κωδ_προϊόντος FROM ΠΩΛΗΣΕΙΣ WHERE Ποσότητα >= 1000);
```

Κωδ_προϊόντος	Όνομα προϊόντος	Τιμή μονάδας
101	ΠΕΠΟΝΙ	880
102	ΦΡΑΟΥΛΑ	150
103	ΜΗΛΟ	120

Κωδ_καρτέλας	Κωδ_προϊόντος	Ποσότητα
1101	101	1100
1102	102	300
1103	103	1700



Εμφωλιασμένα ερωτήματα: IN – Παράδειγμα 2

Όλες τις πληροφορίες για τα προϊόντα των οποίων ο όγκος πωλήσεων είναι μεγαλύτερος ή ίσος του 1000:

SELECT * FROM ΠΡΟΪΟΝΤΑ

WHERE Κωδ_προϊόντος IN

(SELECT Κωδ_προϊόντος FROM ΠΩΛΗΣΕΙΣ WHERE Ποσότητα >= 1000);

Κωδ_προϊόντος	Όνομα προϊόντος	Τιμή μονάδας	(SELECT κωδ_π...
101	ΠΕΠΟΝΙ	880	{101,103}
102	ΦΡΑΟΥΛΑ	150	{101,103}
103	ΜΗΛΟ	120	{101,103}

IN

Κωδ_καρτέλας	Κωδ_προϊόντος	Ποσότητα
1101	101	1100
1102	102	300
1103	103	1700

Εμφωλιασμένα ερωτήματα: IN – Παράδειγμα 2

Όλες τις πληροφορίες για τα προϊόντα των οποίων ο όγκος πωλήσεων είναι μεγαλύτερος ή ίσος του 1000:

```
SELECT * FROM ΠΡΟΪΟΝΤΑ
```

```
WHERE Κωδ_προϊόντος IN
```

```
(SELECT Κωδ_προϊόντος FROM ΠΩΛΗΣΕΙΣ WHERE Ποσότητα >= 1000);
```

Αποτέλεσμα

προϊόντος	Όνομα προϊόντος	Τιμή μονάδας
101	ΠΕΠΟΝΙ	880
103	ΜΗΛΟ	120

Εμφωλιασμένα ερωτήματα: IN – Παράδειγμα 3

- Τα ονόματα των πελατών που έκαναν τη συγκεκριμένη παραγγελία #4:
- **SELECT** π.Όνομα_πελάτη **FROM** ΠΕΛΑΤΕΣ π
- **WHERE** π.Αρ_λογαριασμού **IN** (
- **SELECT** ρ.Αρ_λογαριασμού **FROM** ΠΑΡΑΓΓΕΛΙΕΣ ρ **WHERE** ρ.Κωδ_παραγγελίας= 4);

Όνομα_πελάτη	Αρ_λογαριασμού
Κίτσος	12345
Λάκης	55555
Ελένη	67676

Κωδ_παραγγελίας	Αρ_λογαριασμού
3	12345
4	55555
5	67676

Εμφωλιασμένα ερωτήματα: IN – Παράδειγμα 3

- Το ίδιο χωρίς εμφωλιασμό:
- **SELECT** π.όνομα_πελάτη
- **FROM** ΠΕΛΑΤΕΣ π, ΠΑΡΑΓΓΕΛΙΕΣ ρ
- **WHERE** π.Αρ_λογαριασμού=ρ.Αρ_λογαριασμού **AND**
ρ.Κωδ_παραγγελίας= 4;

Όνομα_πελάτη	Αρ_λογαριασμού
Κίτσος	12345
Λάκης	55555
Ελένη	67676

Κωδ_παραγγελίας	Αρ_λογαριασμού
3	12345
4	55555
5	67676

Εμφωλιασμένα ερωτήματα: NOT IN, ANY, ALL

Οι σύνδεσμοι(τελεστές) με εμφωλιασμένα:

IN, NOT IN, τ ANY (SOME), τ ALL
όπου τ ανήκει {= <> < <= > >=}

IN \equiv =ANY, NOT IN \equiv <>ALL

- Π.χ.
- $5 < ALL (0, 5, 6)$ επιστρέφει false
- $5 < ALL (6, 9)$ επιστρέφει true
- $5 < ANY (0, 5, 6)$ επιστρέφει true
- $5 < ANY (6, 9)$ επιστρέφει true

To 5 είναι μικροτερο από ολα

To 5 είναι μικροτερο από ένα

Εμφωλιασμένα ερωτήματα: NOT IN, ANY, ALL

- Τα **NOT IN**, **ANY**, **ALL** (ενίστε και το **IN**), ανάλογα και με το περιεχόμενο, δεν μπορούν να εκφραστούν με επίπεδο τρόπο.
- Άρα είναι σημαντικά εργαλεία για να αυξήσουμε την εκφραστικότητα την οποία έχουμε στην SQL.

Εμφωλιασμένα ερωτήματα: NOT IN, ANY, ALL

-π.χ.

- Όνομα υπαλλήλου με μισθό μεγαλύτερο απ' όλους τους υπαλλήλους (από οποιονδήποτε) στον 1^ο όροφο:
- **SELECT** υ.όνομα **FROM** ΥΠ υ
- **WHERE** υ.μισθός > **ALL**
- (**SELECT** υ.μισθός **FROM** ΥΠ υ, ΤΜ τ **WHERE** υ.τμ=τ.κωδτ and τ.όροφος=1)

Εμφωλιασμένα ερωτήματα: εμβέλεια

- Υπάρχουν περιπτώσεις όπου το αποτέλεσμα του εσωτερικού ερωτήματος είναι διαφορετικό για κάθε πλειάδα.
- Αυτό μπορεί να γίνει όταν το εσωτερικό ερώτημα χρησιμοποιεί μέσα του πεδία και μεταβλητές πλειάδος από το εξωτερικό ερώτημα.
- Με άλλα λόγια, επιτρέπεται η χρήση πεδίων από εξωτερικό ερώτημα σε εσωτερικό. Η **εμβέλεια** μιας μεταβλητής είναι μέχρι το επίπεδο που ξαναορίζεται.

Εμφωλιασμένα ερωτήματα: εμβέλεια - παράδειγμα

- ΦΟΙΤ (κωδφ, όνομα, σύμβουλος)
- ΜΑΘ (κωδμ, διδάσκων)
- ΔΗΛΩΣΕΙΣ (κφοιτ, κμαθ)

- Ονόματα φοιτητών που παίρνουν μάθημα από το σύμβουλό τους:

- **SELECT** όνομα **FROM** ΦΟΙΤ **WHERE** κωδφ **IN**
- **(SELECT** κφοιτ **FROM** ΔΗΛΩΣΕΙΣ **WHERE** κμαθ **IN**
- **(SELECT** κωδμ **FROM** ΜΑΘ **WHERE** διδάσκων=σύμβουλος))

Εμφωλιασμένα ερωτήματα: εμβέλεια - παράδειγμα

- ΦΟΙΤ (κωδφ, όνομα, σύμβουλος)
- ΜΑΘ (κωδμ, διδάσκων)
- ΔΗΛΩΣΕΙΣ (κφοιτ, κμαθ)

- Ονόματα φοιτητών που παίρνουν μάθημα από το σύμβουλό τους:
- **SELECT** όνομα **FROM** ΦΟΙΤ **WHERE** κωδφ **IN**
- (**SELECT** κφοιτ **FROM** ΔΗΛΩΣΕΙΣ **WHERE** κμαθ **IN**
- (**SELECT** κωδμ **FROM** ΜΑΘ **WHERE** διδάσκων=σύμβουλος))

Εμφωλιασμένα ερωτήματα: εμβέλεια - παράδειγμα

- ΦΟΙΤ (κωδφ, όνομα, σύμβουλος)
- ΜΑΘ (κωδμ, διδάσκων)
- ΔΗΛΩΣΕΙΣ (κφοιτ, κμαθ)

- Ονόματα φοιτητών που παίρνουν μάθημα από το σύμβουλό τους:

- **SELECT** όνομα **FROM** ΦΟΙΤ **WHERE** κωδφ **IN**
- (**SELECT** κφοιτ **FROM** ΔΗΛΩΣΕΙΣ **WHERE** κμαθ **IN**
- (**SELECT** κωδμ **FROM** ΜΑΘ **WHERE** διδάσκων=σύμβουλος))

Εμφωλιασμένα ερωτήματα: εμβέλεια - παράδειγμα

- ΦΟΙΤ (κωδφ, όνομα, σύμβουλος)
- ΜΑΘ (κωδμ, διδάσκων)
- ΔΗΛΩΣΕΙΣ (κφοιτ, κμαθ)

- Ονόματα φοιτητών που παίρνουν μάθημα από το σύμβουλό τους:

- **SELECT** όνομα **FROM** ΦΟΙΤ **WHERE** κωδφ **IN**
- (**SELECT** κφοιτ **FROM** ΔΗΛΩΣΕΙΣ **WHERE** κμαθ **IN**
- (**SELECT** κωδμ **FROM** ΜΑΘ **WHERE** διδάσκων=σύμβουλος))

Εμφωλιασμένα ερωτήματα: εμβέλεια - παράδειγμα

- ΦΟΙΤ (κωδφ, όνομα, σύμβουλος)
- ΜΑΘ (κωδμ, διδάσκων)
- ΔΗΛΩΣΕΙΣ (κφοιτ, κμαθ)

- Ονόματα φοιτητών που παίρνουν μάθημα από το σύμβουλό τους:

- **SELECT** όνομα **FROM** ΦΟΙΤ **WHERE** κωδφ **IN**
- (**SELECT** κφοιτ **FROM** ΔΗΛΩΣΕΙΣ **WHERE** κμαθ **IN**
- (**SELECT** κωδμ **FROM** ΜΑΘ **WHERE** διδάσκων=σύμβουλος))

Κεφάλαιο
7.1.7:Συναθρ.
Συν. & 7.1.8:
Ομαδοποίηση

SQL
Συναθροιστικές
Συναρτήσεις

Συναθροιστικές συναρτήσεις (Aggregate functions)

- Είσοδος: ένα σύνολο τιμών ή πλειάδων
- Έξοδος: μία τιμή

• Στην SQL οι βασικές συναρτήσεις είναι:

- **COUNT**([DISTINCT] x)
 - Ή COUNT(*) ή COUNT (DISTINCT x)
- **SUM**([DISTINCT] x)
- **AVG**([DISTINCT] x)
- **MIN**(x)
- **MAX**(x)

αριθμός των [μοναδικών] τιμών

άθροισμα των " "

μέσος όρος των " "

ελάχιστη τιμή

μέγιστη τιμή

Συναθροιστικές συναρτήσεις: COUNT(*)- παράδειγμα

- Αριθμός των φοιτητών:
- **SELECT**COUNT(*) **FROM** ΦΟΙΤΗΤΕΣ;
 - Το αποτέλεσμα θα είναι ένας πίνακας:

Count(*)
52

- Εφαρμόζεται ό,τι ξέρουμε ως τώρα για το FROM... WHERE(δηλ. η υπόλοιπη εκφραστικότητα της SQL παραμένει σε ισχύ) και μετά εφαρμόζουμε τις σ.σ. στις πλειάδες που μένουν, μετά από κάποιες πιθανές προβολές.

Συναθροιστικές συναρτήσεις: SUM() - παράδειγμα

- Η συνάρτηση SUM() επιστρέφει το άθροισμα (αριθμητικό) από μια στήλη (πεδίο):
- **SELECT SUM(Τιμή) FROM ΠΑΡΑΓΓΕΛΙΕΣ;**

κωδπ	Ημερομηνία	Τιμή	Όνομα
1	2010/11/12	1000	Ελένη
2	2010/10/23	1600	Πάρις
3	2010/09/02	700	Μενέλαος
4	2010/09/03	300	Αχιλλέας
5	2010/08/30	2000	Ηλέκτρα
6	2010/10/04	100	Πάτροκλος



SUM(Τιμή)
5700

Συναθροιστικές συναρτήσεις: AVG() - παράδειγμα

- Η συνάρτηση AVG() επιστρέφει το μέσο όρο (αριθμητικό) από μια στήλη (πεδίο):
- **SELECT** AVG(Τιμή) **FROM** ΠΑΡΑΓΓΕΛΙΕΣ;

κωδπ	Ημερομηνία	Τιμή	Όνομα
1	2010/11/12	1000	Ελένη
2	2010/10/23	1600	Πάρις
3	2010/09/02	700	Μενέλαος
4	2010/09/03	300	Αχιλλέας
5	2010/08/30	2000	Ηλέκτρα
6	2010/10/04	100	Πάτροκλος



SUM(Τιμή)
950

Συναθροιστικές συναρτήσεις: MIN() - παράδειγμα

- Η συνάρτηση MIN() επιστρέφει τη μικρότερη τιμή (αριθμητική) από μια στήλη (πεδίο):
- **SELECT** MIN(Τιμή) **FROM** ΠΑΡΑΓΓΕΛΙΕΣ;

κωδπ	Ημερομηνία	Τιμή	Όνομα
1	2010/11/12	1000	Ελένη
2	2010/10/23	1600	Πάρις
3	2010/09/02	700	Μενέλαος
4	2010/09/03	300	Αχιλλέας
5	2010/08/30	2000	Ηλέκτρα
6	2010/10/04	100	Πάτροκλος



SUM(Τιμή)
100

Συναθροιστικές συναρτήσεις: MAX() - παράδειγμα

- Η συνάρτηση MAX() επιστρέφει τη μεγαλύτερη τιμή (αριθμητική) από μια στήλη (πεδίο):
- **SELECT MAX(Τιμή) FROM ΠΑΡΑΓΓΕΛΙΕΣ;**

κωδπ	Ημερομηνία	Τιμή	Όνομα
1	2010/11/12	1000	Ελένη
2	2010/10/23	1600	Πάρις
3	2010/09/02	700	Μενέλαος
4	2010/09/03	300	Αχιλλέας
5	2010/08/30	2000	Ηλέκτρα
6	2010/10/04	100	Πάτροκλος



SUM(Τιμή)
2000

Συναθροιστικές συναρτήσεις: παράδειγμα

- Στο τμήμα `SELECT` μπορούν να υπάρχουν πολλές συναθροιστικές συναθροίσεις (όσες χρειάζονται).
- Π.χ. Το άθροισμα των τιμών των παραγγελιών όλων των υπαλλήλων καθώς κι η μέγιστη, η ελάχιστη και η μέση τιμή παραγγελίας:
- Συναθροιστικές συναρτήσεις: παράδειγμα
- **`SELECT SUM(Τιμή), MAX(Τιμή), MIN(Τιμή), AVG(Τιμή) FROM ΠΑΡΑΓΓΕΛΙΕΣ;`**

<code>SUM(Τιμή)</code>	<code>MAX(Τιμή)</code>	<code>MIN(Τιμή)</code>	<code>AVG(Τιμή)</code>
5700	2000	100	950

Συναθροιστικές συναρτήσεις ομάδων

- Συναθροιστικές συναρτήσεις ομάδων: ομαδοποιείται το αποτέλεσμα του WHERE (καρτεσιανό γινόμενο) ως προς τις τιμές σε ένα ή περισσότερα πεδία του (κάποιο σύνολο πεδίων) και υπολογισμός της συναθροιστικής συνάρτησης για κάθε ομάδα.
- Τα πεδία ομαδοποίησης καθορίζονται στο **GROUP BY**

Ευχαριστώ!

- <https://eclass.uoa.gr/courses/DIND136/> Έγγραφα > Διαλέξεις
- Βιβλία:
 - Κεφάλαιο 6
 - SQL

