

# SQL: Ερωτήματα ομαδοποίησης και συνάθροισης

## GROUP BY, HAVING, COUNT, MIN, MAX, SUM, AVG, ROLLUP

Αθανάσιος Σταυρακούδης

<http://stavrakoudis.econ.uoi.gr>

Άνοιξη 2014



# Περιεχόμενα

- 1 Εισαγωγή, γενικές παρατηρήσεις
- 2 Συναρτήσεις συνάθροισης
- 3 Τιμές NULL και μοναδικές τιμές
- 4 Ομαδοποίηση εγγραφών
- 5 Περιορισμός μετά την ομαδοποίηση
- 6 Μερικά και ολικά αθροίσματα



# Σκοπός του μαθήματος

Αν κατανοήσετε αυτό το μάθημα, θα μπορείτε να:

- Βρίσκετε πλήθος εγγραφών ή ακραίες τιμές (μέγιστα, ελάχιστα) πεδίων.
- Βρίσκετε αθροίσματα και μέσους όρους αριθμητικών πεδίων.
- Ομαδοποιείτε εγγραφές με βάση ταυτιζόμενες τιμές σε κάποια πεδία με τη φράση **GROUP BY**.
- Εφαρμόζετε τις συναρτήσεις συνάθροισης στις εγγραφές που προκύπτουν από την ομαδοποίηση/συνάθροιση.
- Εφαρμόζετε την επέκταση της ομαδοποίησης για να παίρνετε μερικά αθροίσματα με την επιλογή **WITH ROLLUP**.
- Εφαρμόζετε περιορισμό στις εγγραφές που προκύπτουν μετά την ομαδοποίηση με τη φράση **HAVING**.



# Συναρτήσεις συνάθροισης

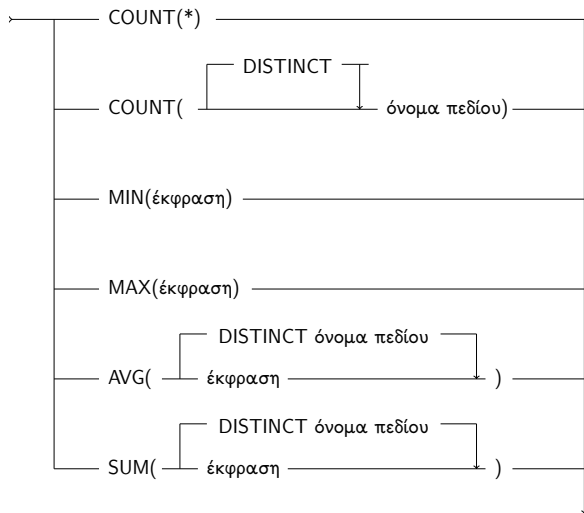
Υπάρχουν πέντε βασικές συναρτήσεις συνάθροισης, που υποστηρίζονται από όλα τα συστήματα SQL

- **COUNT** για την καταμέτρηση πλήθους
- **SUM** για το άθροισμα
- **AVG** για το μέσο όρο
- **MIN** για την ελάχιστη τιμή
- **MAX** για την μέγιστη τιμή

Υπάρχουν αρκετές άλλες συναρτήσεις (πχ **STDEV**, **VAR**), αλλά η υποστήριξή τους ποικίλει από σύστημα σε σύστημα.



# Διάγραμμα σύνταξης των συναρτήσεων συνάθροισης



## Ο πίνακας *employees* από τη βάση *company*

Έστω η σχέση *employees* με σχήμα:

*employees*(*empid*, *firstname*, *lastname*, *depid*, *salary*, *hiredate*)

<b>empid</b>	<b>firstname</b>	<b>lastname</b>	<b>depid</b>	<b>salary</b>	<b>hiredate</b>
102	Νικηφόρος	Διαμαντίδης	6	1212.50	2003-06-02
109	Μαρία	Αθανασίου	1	2787.69	2000-01-26
153	Μαρία	Αλεβιζάτου	2	1321.92	2001-05-15
172	Χρήστος	Βλάσσης	3	1101.70	2000-07-04
189	Θεόδωρος	Αγγελίνας	6	1908.28	2000-06-19
...	...	...	...	...	...

Δείγμα από τα δεδομένα του πίνακα.

Δείτε τα πλήρη περιεχόμενα εδώ:

<http://stavrakoudis.econ.uoi.gr/stavrakoudis/?iid=400>



# Περιεχόμενα

- 1 Εισαγωγή, γενικές παρατηρήσεις
- 2 **Συναρτήσεις συνάθροισης**
- 3 Τιμές NULL και μοναδικές τιμές
- 4 Ομαδοποίηση εγγραφών
- 5 Περιορισμός μετά την ομαδοποίηση
- 6 Μερικά και ολικά αθροίσματα



# Παράδειγμα αθροίσματος στήλης

Να βρεθεί το άθροισμα των μισθών όλων των υπαλλήλων

$$\mathcal{G}_{sum(salary)}(employees)$$




# Παράδειγμα αθροίσματος στήλης

Να βρεθεί το άθροισμα των μισθών όλων των υπαλλήλων

$$\mathcal{G}_{sum(salary)}(employees)$$

```
1  SELECT SUM(salary)
2      FROM employees;
```

```
3
4  | SUM(salary) |
5  |-----|
6  |    38232.03 |
```



# Παράδειγμα αθροίσματος στήλης

Να βρεθεί το άθροισμα των μισθών όλων των υπαλλήλων

$$\mathcal{G}_{sum(salary)}(employees)$$

```
1  SELECT SUM(salary)
2      FROM employees;
```

```
3
4  | SUM(salary) |
5  |-----|
6  |    38232.03 |
```



# Παράδειγμα αθροίσματος υπό συνθήκη

Να βρεθεί το άθροισμα των μισθών όλων των υπαλλήλων του τμήματος 4

$$\mathcal{G}_{sum(salary)}(\sigma_{deptid=4}(employees))$$



# Παράδειγμα αθροίσματος υπό συνθήκη

Να βρεθεί το άθροισμα των μισθών όλων των υπαλλήλων του τμήματος 4

$$\mathcal{G}_{sum(salary)}(\sigma_{depid=4}(employees))$$

```
1  SELECT SUM(salary)
2      FROM employees
3      WHERE depid = 4;
```

```
5  | SUM(salary) |
6  |-----|
7  |      3542.80 |
```



# Παράδειγμα αθροίσματος γενικευμένης προβολής

Να βρεθεί το επιπλέον ποσό που θα δοθεί σε μισθούς αν οι υπάλληλοι του τμήματος 3 πάρουν αύξηση 3%

$$\mathcal{G}_{sum(salary \times 0.03) \rightarrow sumsal3}(\sigma_{deptid=3}(employees))$$



# Παράδειγμα αθροίσματος γενικευμένης προβολής

Να βρεθεί το επιπλέον ποσό που θα δοθεί σε μισθούς αν οι υπάλληλοι του τμήματος 3 πάρουν αύξηση 3%

$$\mathcal{G}_{sum(salary \times 0.03) \rightarrow sumsal3}(\sigma_{depid=3}(employees))$$

```
1 SELECT SUM(salary*0.03) AS sumsal3
2   FROM employees
3  WHERE depid = 3;
```



# Παράδειγμα αθροίσματος γενικευμένης προβολής

Να βρεθεί το επιπλέον ποσό που θα δοθεί σε μισθούς αν οι υπάλληλοι του τμήματος 3 πάρουν αύξηση 3%

$$\mathcal{G}_{sum(salary \times 0.03) \rightarrow sumsal3}(\sigma_{depid=3}(employees))$$

```
1 SELECT SUM(salary*0.03) AS sumsal3
2   FROM employees
3  WHERE depid = 3;
```

- Η μετονομασία του πεδίου είναι χρήσιμη αλλά προαιρετική.
- Κάντε την πράξη με λογιστικό φύλλο και αναλύστε πλεονεκτήματα/μειονεκτήματα.
- Προσέξτε πως το μόνο που δηλώνει ο χρήστης στην **SQL** είναι ονόματα πεδίων και πινάκων.



# Παράδειγμα υπολογισμού μέσου όρου

Να βρεθεί ο μέσος μισθός όλων των υπαλλήλων

$$\mathcal{G}_{avg(salary)}(employees)$$




# Παράδειγμα υπολογισμού μέσου όρου

Να βρεθεί ο μέσος μισθός όλων των υπαλλήλων

$$\mathcal{G}_{avg(salary)}(employees)$$

```
1  SELECT AVG(salary)
2      FROM employees;
```

```
3
4  | AVG(salary) |
5  -----
6  | 1416.001111 |
```



# Μέσος όρος με περιορισμό εγγραφών

Να βρεθεί ο μέσος μισθός των υπαλλήλων του τμήματος 3

$$\mathcal{G}_{avg}(salary)(\sigma_{deptid=3}(employees))$$



# Μέσος όρος με περιορισμό εγγραφών

Να βρεθεί ο μέσος μισθός των υπαλλήλων του τμήματος 3

$$\mathcal{G}_{avg(salary)}(\sigma_{depid=3}(employees))$$

```
1  SELECT AVG(salary)
2      FROM employees
3      WHERE depid = 3;
```

```
4
5  | AVG(salary) |
6  -----
7  | 1311.651250 |
```



# Ελάχιστη τιμή

Να βρεθεί ο μικρότερος μισθός των υπαλλήλων του τμήματος 3

$$\mathcal{G}_{\min(\text{salary})\sigma_{\text{depid}=3}(\text{employees})}$$



# Ελάχιστη τιμή

Να βρεθεί ο μικρότερος μισθός των υπαλλήλων του τμήματος 3

$$\mathcal{G}_{\min(\text{salary})\sigma_{\text{depid}=3}(\text{employees})}$$

```
1  SELECT MIN(salary)
2      FROM employees
3      WHERE depid = 3;
```

```
4
5  | MIN(salary) |
6  -----
7  |      1050.96 |
```



# Μέγιστη τιμή

Να βρεθεί η ημερομηνία της πιο πρόσφατης πρόσληψης

$\mathcal{G}_{\max(\text{hiredate})}(\text{employees})$

```
1  SELECT MAX(hiredate)
2      FROM employees;
```

```
3
4  | MAX(hiredate) |
```

```
5  -----
```

```
6  | 2004-10-05    |
```



# Καταμέτρηση πλήθους

Να βρεθεί πόσοι υπάλληλοι εργάζονται στο τμήμα 4

$$\mathcal{G}_{count(empid)}(\sigma_{deptid=4}(employees))$$



# Καταμέτρηση πλήθους

Να βρεθεί πόσοι υπάλληλοι εργάζονται στο τμήμα 4

$$\mathcal{G}_{count(empid)}(\sigma_{depid=4}(employees))$$

```
1  SELECT COUNT(empid)
2      FROM employees
3      WHERE depid = 4;
```

```
4
5  | COUNT(empid) |
6  -----
7  |              5 |
```





# COUNT(\*)

Να βρεθεί πόσοι υπάλληλοι εργάζονται στο τμήμα 4

```
1  SELECT COUNT(*)
2      FROM employees
3      WHERE depid = 4;
```

```
5  | COUNT(*)      |
6  -----
7  |                5 |
```

- ❶ **COUNT(empid)** : Καταμέτρηση εγγγραφών με μη NULL τιμές στο πεδίο *empid*.
- ❷ **COUNT(\*)** : Καταμέτρηση εγγγραφών



# Δύο συναρτήσεις στο ίδιο ερώτημα

Να βρεθεί το εύρος το μισθών του τμήματος 4

$G_{max(salary)-min(salary)}(employees)$



# Δύο συναρτήσεις στο ίδιο ερώτημα

Να βρεθεί το εύρος το μισθών του τμήματος 4

$G_{\max(\text{salary})-\min(\text{salary})}(\text{employees})$

```
1  SELECT MAX(salary)-MIN(salary)
2      FROM employees
3      WHERE depid = 4;
```

```
4
5  | MAX(salary)-MIN(salary) |
6  -----
7  |                      331.34 |
```



# Περιεχόμενα

- 1 Εισαγωγή, γενικές παρατηρήσεις
- 2 Συναρτήσεις συνάθροισης
- 3 Τιμές NULL και μοναδικές τιμές
- 4 Ομαδοποίηση εγγραφών
- 5 Περιορισμός μετά την ομαδοποίηση
- 6 Μερικά και ολικά αθροίσματα



# Πλήθος μισθοδοτούμενων υπαλλήλων

Να βρεθεί το πλήθος  
καταχωρημένων μισθών

```
1  SELECT COUNT(salary)
2      FROM employees;
3
4  | count(salary) |
5  -----
6  |              27 |
```

Να βρεθεί το πλήθος  
των κωδικών των  
υπαλλήλων

```
1  SELECT COUNT(empid)
2      FROM employees;
3
4  | count(empid) |
5  -----
6  |             30 |
```

- Υπάρχουν 30 υπάλληλοι αλλά 27 μισθοί
- **COUNT(salary)** δεν μετράει τις τιμές **NULL**
- Το πεδίο **empid** δεν παίρνει ποτέ τιμές **NULL** γιατί είναι πρωτεύον κλειδί



# Τιμές NULL

Να βρεθεί το πλήθος των υπαλλήλων χωρίς μισθό

```
1 SELECT COUNT(salary)
2   FROM employees
3  WHERE salary IS NULL;
4
5 | COUNT(salary) |
6 -----
7 |                0 |
```

Να βρεθεί το πλήθος των υπαλλήλων χωρίς μισθό

```
1 SELECT COUNT(empid)
2   FROM employees
3  WHERE salary IS NULL;
4
5 | count(empid) |
6 -----
7 |                3 |
```

- Οι τιμές **NULL** δεν απαριθμούνται.
- Το πλήθος των εγγραφών με τιμή **NULL** στο πεδίο *salary* είναι 3.



# Καταμέτρηση μοναδικών τιμών (Λάθος)

Να βρεθεί το πλήθος των τμημάτων των υπαλλήλων

```
1 SELECT COUNT(depid)
2   FROM employees;
```

```
3
4 | COUNT(depid) |
```

```
5 -----
```

```
6 |                30 |
```



# Καταμέτρηση μοναδικών τιμών (Σωστό)

Να βρεθεί το πλήθος των τμημάτων των υπαλλήλων

```
1 SELECT COUNT(DISTINCT depid)
2   FROM employees;
```

```
3
4 | COUNT(DISTINCT depid) |
5 -----
6 |                      6 |
```





# Περιεχόμενα

- 1 Εισαγωγή, γενικές παρατηρήσεις
- 2 Συναρτήσεις συνάθροισης
- 3 Τιμές NULL και μοναδικές τιμές
- 4 Ομαδοποίηση εγγραφών
- 5 Περιορισμός μετά την ομαδοποίηση
- 6 Μερικά και ολικά αθροίσματα



# Ομαδοποίηση

- Μέχρι τώρα είδαμε απλά ερωτήματα, η απάντηση των οποίων ήταν μία τιμή.
- Πολλές φορές το ζητούμενο είναι μια λίστα τιμών, μία τιμή **ανά** κατηγορία.
- Πχ ο μέσος μισθός **ανά** τμήμα, ή ο αρχαιότερος υπάλληλος **ανά** τμήμα
- Για αυτές τις περιπτώσεις θα χρειαστούμε μια νέα φράση: **GROUP BY**



# Πλήθος υπαλλήλων ανά τμήμα

## Πλήθος υπαλλήλων ανά τμήμα

$depid \mathcal{G}_{count(*)}(employees)$



# Πλήθος υπαλλήλων ανά τμήμα

## Πλήθος υπαλλήλων ανά τμήμα

$depid \mathcal{G}_{count(*)}(employees)$

```
1 SELECT depid, COUNT(*)  
2 FROM employees  
3 GROUP BY depid;
```

depid	COUNT(*)
1	3
2	4
3	9
4	5
5	2
6	7



## Πλήθος υπαλλήλων ανά τμήμα

## Πλήθος υπαλλήλων ανά τμήμα

$$depid\mathcal{G}_{count(*)}(employees)$$

```
SELECT depid, COUNT(*)
FROM employees
GROUP BY depid;
```

depid	COUNT(*)
1	3
2	4
3	9
4	5
5	2
6	7

Η **SQL** έχει πλεονεκτήματα:

- Δεν χρειάζεται να γνωρίζουμε το μέγεθος του πίνακα.
- Το ίδιο ακριβώς ερώτημα **SQL** θα χρησιμοποιηθεί έστω και αν αλλάξουν τα δεδομένα του πίνακα μετά από προσθήκη ή αφαίρεση εγγραφών.



Ο μικρότερος μισθός ανά τμήμα υπαλλήλων

$deptid \mathcal{G}_{min(salary)}(employees)$



# Μικρότερο ανά ...

Ο μικρότερος μισθός ανά τμήμα υπαλλήλων

$depid \mathcal{G}_{min(salary)}(employees)$

```
1  SELECT depid, MIN(salary)
2      FROM employees
3  GROUP BY depid;
```

depid	MIN(salary)
1	1754.67
2	1105.04
3	1050.96
4	1054.71
5	1051.92
6	1100.13



# Περιορισμός και ομαδοποίηση

Να βρεθεί το άθροισμα των μισθών των υπαλλήλων ανά τμήμα για τους υπαλλήλους που προσλήφθηκαν μέσα στο 2004

$$depid \mathcal{G}_{sum}(salary)$$
  
$$(\sigma_{hiredate \geq '2004-01-01' \wedge hiredate \leq '2004-12-31'}(employees))$$





# Περιορισμός και ομαδοποίηση

Να βρεθεί το άθροισμα των μισθών των υπαλλήλων ανά τμήμα για τους υπαλλήλους που προσλήφθηκαν μέσα στο 2004

$$depid \mathcal{G}_{sum(salary)} (\sigma_{hiredate \geq '2004-01-01' \wedge hiredate \leq '2004-12-31'}(employees))$$

```
1 SELECT depid, SUM(salary)
2   FROM employees
3   WHERE hiredate BETWEEN '2004-01-01'
4                                     AND '2004-12-31'
5 GROUP BY depid;
```



# Ομαδοποίηση και ταξινόμηση

Να βρεθεί το άθροισμα των μισθών των υπαλλήλων ανά τμήμα με φθίνουσα ταξινόμηση ως προς το άθροισμα των μισθών



# Ομαδοποίηση και ταξινόμηση

Να βρεθεί το άθροισμα των μισθών των υπαλλήλων ανά τμήμα με φθίνουσα ταξινόμηση ως προς το άθροισμα των μισθών

```
1  SELECT depid, SUM(salary)
2      FROM employees
3  GROUP BY depid
4  ORDER BY SUM(salary) DESC;
```

depid	SUM(salary)
3	10493.21
6	9853.82
1	6395.35
2	5360.28
4	3542.80
5	2586.57



# Περιεχόμενα

- 1 Εισαγωγή, γενικές παρατηρήσεις
- 2 Συναρτήσεις συνάθροισης
- 3 Τιμές NULL και μοναδικές τιμές
- 4 Ομαδοποίηση εγγραφών
- 5 Περιορισμός μετά την ομαδοποίηση
- 6 Μερικά και ολικά αθροίσματα



# Περιορισμός μετά την ομαδοποίηση

Τμήματα με περισσότερους από 4 υπαλλήλους

$$\sigma_{count(depid)>4} (depid \mathcal{G}_{count(empid)}(employees))$$



# Περιορισμός μετά την ομαδοποίηση

Τμήματα με περισσότερους από 4 υπαλλήλους

$\sigma_{count(depid) > 4} (depid \mathcal{G}_{count(empid)}(employees))$

```
1  SELECT depid, COUNT(depid)
2      FROM employees
3  GROUP BY depid
4  HAVING COUNT(depid) > 4;
```



## Περιορισμός μετά την ομαδοποίηση

Τμήματα με περισσότερους από 4 υπαλλήλους

$$\sigma_{count(depId) > 4} (depId \mathcal{G}_{count(empId)}(employees))$$

```
SELECT depid, COUNT(depid)
FROM employees
GROUP BY depid
HAVING COUNT(depid) > 4;
```

depid	COUNT(*)
1	3
2	4
3	9
4	5
5	2
6	7

- Ο όρος **HAVING** τοποθετείται μετά τον όρο **GROUP BY**.
- Η σύνταξη είναι παρόμοια με αυτή του όρου **WHERE**.
- Ο όρος **HAVING** περιορίζει το αποτέλεσμα του ερωτήματος με βάση πεδία που παράγονται από τον όρο **GROUP BY**.



# Περιορισμός πριν και μετά την ομαδοποίηση

Να βρεθούν τα τμήματα με περισσότερους από 3 υπαλλήλους με μισθό μεγαλύτερο από 1200 €

$$\sigma_{count(depId)>3} \left( depId \mathcal{G}_{count(empId)} \left( \sigma_{salary>1200}(employees) \right) \right)$$




# Περιορισμός πριν και μετά την ομαδοποίηση

Να βρεθούν τα τμήματα με περισσότερους από 3 υπαλλήλους με μισθό μεγαλύτερο από 1200 €

$$\sigma_{count(depId)>3} (depId \mathcal{G}_{count(empId)} (\sigma_{salary>1200}(employees)))$$

```
1  SELECT depid
2      FROM employees
3      WHERE salary > 1200
4  GROUP BY depid
5      HAVING COUNT(depid) > 3;
```

depid
3
6



# Περιεχόμενα

- 1 Εισαγωγή, γενικές παρατηρήσεις
- 2 Συναρτήσεις συνάθροισης
- 3 Τιμές NULL και μοναδικές τιμές
- 4 Ομαδοποίηση εγγραφών
- 5 Περιορισμός μετά την ομαδοποίηση
- 6 Μερικά και ολικά αθροίσματα



# ROLLUP (μερικά αθροίσματα)

Πλήθος των υπαλλήλων ανά τμήμα και το συνολικό πλήθος των υπαλλήλων της εταιρείας



# ROLLUP (μερικά αθροίσματα)

Πλήθος των υπαλλήλων ανά τμήμα και το συνολικό πλήθος των υπαλλήλων της εταιρείας

```
1  SELECT depid, COUNT(*)  
2      FROM employees  
3  GROUP BY depid WITH ROLLUP;
```



# ROLLUP (μερικά αθροίσματα)

Πλήθος των υπαλλήλων ανά τμήμα και το συνολικό πλήθος των υπαλλήλων της εταιρείας

```
1  SELECT depid, COUNT(*)  
2      FROM employees  
3  GROUP BY depid WITH ROLLUP;
```

depid	COUNT(*)
1	3
2	4
3	9
4	5
5	2
6	7
NULL	30

- **WITH ROLLUP** : μετά το πεδίο που ακολουθεί τον όρο **GROUP BY**.
- Προσοχή την τιμή **NULL** στο τέλος που αντιστοιχεί στο σύνολο των εγγραφών.



# Μερικά αθροίσματα με διπλή ομαδοποίηση

Να βρεθεί ο μέσος μισθός των υπαλλήλων ανά τμήμα και έτος πρόσληψης για τμήματα με κωδικό 2 και 6

```
1  SELECT depid, YEAR(hiredate), AVG(salary)
2  FROM employees
3  WHERE depid IN (2,6)
4  GROUP BY depid, YEAR(hiredate) WITH ROLLUP;
```

depid	YEAR(hiredate)	AVG(salary)
2	1999	1609.520000
2	2000	1323.800000
2	2001	1213.480000
2	NULL	1340.070000
6	2000	1771.620000
6	2001	1336.985000
6	2002	1323.980000
6	2003	1156.315000
6	NULL	1407.688571
NULL	NULL	1383.100000



# Αναστροφή της λίστας πεδίων

Ο μέσος μισθός των υπαλλήλων ανά έτος πρόσληψης και τμήμα για τμήματα με κωδικό 2 και 6

```
1  SELECT YEAR(hiredate), depid, AVG(salary)
2      FROM employees
3      WHERE depid IN (2,6)
4  GROUP BY YEAR(hiredate), depid WITH ROLLUP;
```

YEAR(hiredate)	depid	AVG(salary)
1999	2	1609.520000
1999	NULL	1609.520000
2000	2	1323.800000
2000	6	1771.620000
.....		
NULL	NULL	1383.100000



Σας ευχαριστώ  
για την προσοχή σας

Είμαι στη διάθεσή σας για σχόλια, απορίες και ερωτήσεις

