

# Περισσότερα για τα απλά ερωτήματα SQL DISTINCT, LIKE, NULL, AND, OR, BETWEEN

Αθανάσιος Σταυρακούδης

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

Άνοιξη 2016



# Σκοπός του μαθήματος

- Αποκλείσετε διπλοεγγραφές από αποτελέσματα ερωτημάτων
- Χειρίζεστε άγνωστες ή ελλιπείς τιμές (**NULL**)
- Εφαρμόζετε αναζήτηση με βάση ταίριασμα κειμένου
- Εκτελείτε ερωτήματα πολύπλοκων περιορισμών εφαρμόζοντας λογικούς τελεστές
- Εκτελείτε ερωτήματα με τη χρήση του τελεστή συνόλου και του τελεστή περιοχής



# Περιεχόμενα

- 1 Απαλοιφή διπλοεγγραφών με DISTINCT
- 2 Έλεγχος για τιμές NULL
- 3 Ταίριασμα κειμένου
- 4 Λογική σύζευξη και διάζευξη
- 5 Τελεστής συνόλου IN
- 6 Τελεστής περιοχής τιμών BETWEEN ... AND ...



# Απαλοιφή διπλοεγγραφών

Να βρεθεί σε ποια τμήματα απασχολούνται οι υπάλληλοι

Χωρίς απαλοιφή  
διπλοεγγραφών

```
1 SELECT depid
2 FROM employees;
```

| depid |
|-------|
| 1     |
| 1     |
| 1     |
| 2     |
| 2     |
| ...   |

Με απαλοιφή  
διπλοεγγραφών

```
1 SELECT DISTINCT depid
2 FROM employees;
```

| depid |
|-------|
| 1     |
| 2     |
| 3     |
| 4     |
| 5     |
| 6     |



## Να δοθεί το τμήμα και ο μισθός των υπαλλήλων

```
1 SELECT DISTINCT depid, salary
2 FROM employees;
```

| depid | salary  |
|-------|---------|
| 6     | 1212.50 |
| 1     | 2787.69 |
| 2     | 1321.92 |
| 3     | 1101.70 |
| 6     | 1908.28 |
| ...   | ...     |

- Εδώ ελέγχεται ο συνδυασμός των τιμών στα πεδία **depid** και **salary**
- Έτσι, μια τιμή στο πεδίο **depid** μπορεί να εμφανιστεί δεύτερη (ή πολλαπλή) επειδή αντιστοιχίζεται με διαφορετική τιμή στο πεδίο **salary**
- Αν το ζεύγος τιμών τμήμα-μισθός είναι ίδιο για δύο διαφορετικές εγγραφές τότε θα εμφανιστεί στο αποτέλεσμα μόνο μία φορά (λόγω **DISTINCT**)



# DISTINCT και προβολή στη σχεσιακή άλγεβρα

Η σχεσιακή πράξη της προβολής, πχ:

$$\Pi_{depid}(employees)$$

εκφράζεται στην SQL πάντα με DISTINCT:

```
1 SELECT DISTINCT depid
2 FROM employees;
```

και όχι χωρίς αυτό, πχ:

```
1 SELECT depid
2 FROM employees;
```

Ο λόγος είναι απλός: η σχεσιακή άλγεβρα αφορά πράξεις συνόλων, οπότε δε μπορεί να υπάρχουν διπλοεγγραφές, τα στοιχεία ενός συνόλου είναι μοναδικά.



# Και κάτι για εξάσκηση

Να βρεθούν τα τμήματα στα οποία ο μισθός των υπαλλήλων θα ανέβει πάνω από 80 € αν πάρουν αύξηση 3.5%

$$\Pi_{depid}(\sigma_{salary*0.035>80}(employees))$$

```
1 SELECT DISTINCT depid
2   FROM employees
3   WHERE salary*0.035 > 80;
```

| depid |
|-------|
| 1     |
| 3     |

Μόνο στα τμήματα 1 και 3 υπάρχουν υπάλληλοι που αν ο μισθός τους αυξηθεί κατά 3.5% η αύξηση θα είναι μεγαλύτερη από 80 €. Στα τμήματα αυτά, πιθανά να υπάρχουν περισσότεροι από ένας υπάλληλος με τέτοια αύξηση. Αλλά αυτό που ενδιαφέρει είναι σε **ποιο τμήμα**, όχι σε ποιους και πόσους υπαλλήλους.



# Περιεχόμενα

- 1 Απαλοιφή διπλοεγγραφών με DISTINCT
- 2 Έλεγχος για τιμές NULL
- 3 Ταίριασμα κειμένου
- 4 Λογική σύζευξη και διάζευξη
- 5 Τελεστής συνόλου IN
- 6 Τελεστής περιοχής τιμών BETWEEN ... AND ...





# Χειρισμός τιμών NULL



# Χειρισμός τιμών NULL

- 1 Δε μπορούμε να χειριστούμε τιμές **NULL** με τελεστές σύγκρισης



# Χειρισμός τιμών NULL

- 1 Δε μπορούμε να χειριστούμε τιμές **NULL** με τελεστές σύγκρισης
- 2 Μια τιμή είναι (**IS**) ή δεν είναι (**IS NOT**) **NULL**



# Χειρισμός τιμών NULL

- 1 Δε μπορούμε να χειριστούμε τιμές **NULL** με τελεστές σύγκρισης
- 2 Μια τιμή είναι (**IS**) ή δεν είναι (**IS NOT**) **NULL**
- 3 Ο έλεγχος για τιμές **NULL** αποδίδει πάντοτε **TRUE** ή **FALSE**



# Χειρισμός τιμών NULL

- 1 Δε μπορούμε να χειριστούμε τιμές **NULL** με τελεστές σύγκρισης
- 2 Μια τιμή είναι (**IS**) ή δεν είναι (**IS NOT**) **NULL**
- 3 Ο έλεγχος για τιμές **NULL** αποδίδει πάντοτε **TRUE** ή **FALSE**
- 4 Δύο τιμές **NULL** δεν είναι ίσες μεταξύ τους, η μεταξύ τους σύγκριση δεν έχει νόημα



# Παραδείγματα με NULL

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων χωρίς καταχωρημένο μισθό

```
1 SELECT *  
2   FROM employees  
3   WHERE salary IS NULL;
```



# Παραδείγματα με NULL

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων χωρίς καταχωρημένο μισθό

```
1 SELECT *  
2   FROM employees  
3   WHERE salary IS NULL;
```

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων με καταχωρημένο μισθό

```
1 SELECT *  
2   FROM employees  
3   WHERE salary IS NOT NULL;
```



# NULL και DISTINCT

## Χωρίς DISTINCT

```
1 SELECT salary
2   FROM employees
3  WHERE salary IS NULL;
```

```
4 | salary |
```

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

```
6 |  NULL  |
```

```
7 |  NULL  |
```

```
8 |  NULL  |
```





# NULL και DISTINCT

## Χωρίς DISTINCT

```
1 SELECT salary
2   FROM employees
3  WHERE salary IS NULL;
4 | salary |
5 -----
6 |  NULL  |
7 |  NULL  |
8 |  NULL  |
```

## Με DISTINCT

```
1 SELECT DISTINCT salary
2   FROM employees
3  WHERE salary IS NULL;
4 | salary |
5 -----
6 |  NULL  |
```



# Λάθη στον έλεγχο για NULL

Με τον τελεστή =

```
1 SELECT *  
2   FROM employees  
3   WHERE salary = NULL;
```

```
4  
5 Empty set (0.00 sec)
```



# Λάθη στον έλεγχο για NULL

Με τον τελεστή =

```
1 SELECT *  
2   FROM employees  
3   WHERE salary = NULL;
```

```
4  
5 Empty set (0.00 sec)
```

Ως κείμενο

```
1 SELECT *  
2   FROM employees  
3   WHERE salary = 'NULL';
```

```
4  
5 Empty set, 1 warning (0.00 sec)
```



# NULL και αλγεβρικές πράξεις

Πρόσθεση 1000 € στο μισθό των υπαλλήλων χωρίς μισθό

```
1 SELECT empid, salary, salary+1000
2 FROM employees
3 WHERE salary IS NULL;
```

```
4
5 | empid | salary | salary+1000 |
6 -----+-----+-----+
7 | 230 | NULL | NULL |
8 | 483 | NULL | NULL |
9 | 593 | NULL | NULL |
10 -----+-----+-----+
```



# Περιεχόμενα

- 1 Απαλοιφή διπλοεγγραφών με DISTINCT
- 2 Έλεγχος για τιμές NULL
- 3 Ταίριασμα κειμένου**
- 4 Λογική σύζευξη και διάζευξη
- 5 Τελεστής συνόλου IN
- 6 Τελεστής περιοχής τιμών BETWEEN ... AND ...



# Ταίριασμα κειμένου

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων με μικρό όνομα Νίκη

```
1 SELECT *  
2   FROM employees  
3   WHERE firstname = 'Νίκη';
```

- 1 Μπορούμε να κάνουμε συγκρίσεις αλφαριθμητικών με τελεστές σύγκρισης όπως και με αριθμούς.
- 2 Ωστόσο, το αλφαριθμητικό τοποθετείται πάντα μέσα σε εισαγωγικά.
- 3 Τα εισαγωγικά μπορεί να είναι μονά (') ή διπλά (""), **ποτέ όμως δεν είναι αυτάκια!**



# Τελεστές ανισότητας και σύγκριση κειμένου

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων με μικρό μετά τη Νίκη

```
1 SELECT *  
2   FROM employees  
3   WHERE firstname > 'Νίκη';
```



# Τελεστές ανισότητας και σύγκριση κειμένου

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων με μικρό μετά τη Νίκη

```
1 SELECT *  
2   FROM employees  
3   WHERE firstname > 'Νίκη';
```

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων με μικρό όνομα μετά το Ν

```
1 SELECT *  
2   FROM employees  
3   WHERE firstname > 'N';
```





# Ταίριασμα κειμένου με τον τελεστή LIKE

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων με μικρό όνομα που αρχίζει από N

```
1 SELECT *  
2   FROM employees  
3   WHERE firstname LIKE 'N%';
```



# Ταίριασμα κειμένου με τον τελεστή LIKE

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων με μικρό όνομα που αρχίζει από N

```
1 SELECT *  
2   FROM employees  
3   WHERE firstname LIKE 'N%';
```

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων με μικρό όνομα που αρχίζει από A και τελειώνει σε A

```
1 SELECT *  
2   FROM employees  
3   WHERE firstname LIKE 'A%A';
```



# LIKE: χαρακτήρες υποκατάστασης

- % (ή \*) : υποκαθιστά από κανένα ως πολλούς χαρακτήρες
- \_ (ή ?) : υποκαθιστά ακριβώς ένα χαρακτήρα
- Οι χαρακτήρες υποκατάστασης μπορούν αν συνδυαστούν μεταξύ τους



# LIKE: χαρακτήρες υποκατάστασης

- % (ή \*) : υποκαθιστά από κανένα ως πολλούς χαρακτήρες
- \_ (ή ?) : υποκαθιστά ακριβώς ένα χαρακτήρα
- Οι χαρακτήρες υποκατάστασης μπορούν αν συνδυαστούν μεταξύ τους

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων με μικρό όνομα που έχει 5 χαρακτήρες και τελιώνει σε A

```
1 SELECT *  
2     FROM employees  
3     WHERE firstname LIKE '____A';
```



# Ασκήσεις με ταίριασμα κειμένου

- 1 Να βρεθούν οι υπάλληλοι που το όνομά τους αρχίζει από Α.
- 2 Να βρεθούν οι υπάλληλοι που το όνομά τους τελειώνει σε ς.
- 3 Να βρεθούν οι υπάλληλοι που το όνομά τους έχει δεύτερο χαρακτήρα το α.
- 4 Να βρεθούν οι υπάλληλοι που το όνομά τους έχει δεύτερο χαρακτήρα το α και τελειώνει σε ς.



# Περιεχόμενα

- 1 Απαλοιφή διπλοεγγραφών με DISTINCT
- 2 Έλεγχος για τιμές NULL
- 3 Ταίριασμα κειμένου
- 4 Λογική σύζευξη και διάζευξη**
- 5 Τελεστής συνόλου IN
- 6 Τελεστής περιοχής τιμών BETWEEN ... AND ...



# Ο τελεστής AND



# Ο τελεστής AND

- 1 Δύο παραστάσεις στον όρο **WHERE** μπορούν να συνδυαστούν μεταξύ τους με το λογικό τελεστή **AND**.





# Ο τελεστής AND

- 1 Δύο παραστάσεις στον όρο **WHERE** μπορούν να συνδυαστούν μεταξύ τους με το λογικό τελεστή **AND**.
- 2 Χρησιμοποιούμε τον τελεστή **AND** όταν θέλουμε να ελέγξουμε την τιμή αληθείας και των δύο παραστάσεων ταυτόχρονα.



# Ο τελεστής AND

- 1 Δύο παραστάσεις στον όρο **WHERE** μπορούν να συνδυαστούν μεταξύ τους με το λογικό τελεστή **AND**.
- 2 Χρησιμοποιούμε τον τελεστή **AND** όταν θέλουμε να ελέγξουμε την τιμή αληθείας και των δύο παραστάσεων ταυτόχρονα.
- 3 Οι βάσεις δεδομένων ακολουθούν την τριαδική λογική: **TRUE, FALSE, UNK**.



# Ο τελεστής AND

- 1 Δύο παραστάσεις στον όρο **WHERE** μπορούν να συνδυαστούν μεταξύ τους με το λογικό τελεστή **AND**.
- 2 Χρησιμοποιούμε τον τελεστή **AND** όταν θέλουμε να ελέγξουμε την τιμή αληθείας και των δύο παραστάσεων ταυτόχρονα.
- 3 Οι βάσεις δεδομένων ακολουθούν την τριαδική λογική: **TRUE, FALSE, UNK**.
- 4 Στο αποτέλεσμα του ερωτήματος εισέρχονται μόνο οι εγγραφές που αποδίδουν **TRUE**.



# Πίνακας αληθείας AND

| <b>AND</b>   | <b>TRUE</b> | <b>FALSE</b> | <b>UNK</b> |
|--------------|-------------|--------------|------------|
| <b>TRUE</b>  | TRUE        | FALSE        | UNK        |
| <b>FALSE</b> | FALSE       | FALSE        | FALSE      |
| <b>UNK</b>   | UNK         | FALSE        | UNK        |

Ο τελεστής **AND** αποδίδει **TRUE**  
μόνο όταν και οι δύο παραστάσεις είναι **TRUE**



# Ένα παράδειγμα με τον τελεστή AND

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων που εργάζονται στο τμήμα 4 και ο μισθός τους είναι μεγαλύτερος από 1100 €

```
1 SELECT *  
2   FROM employees  
3   WHERE depid = 4  
4         AND salary > 1100;
```

| empid | firstname | lastname      | depid | salary  | hiredate   |
|-------|-----------|---------------|-------|---------|------------|
| 206   | Νίκος     | Βλάχος        | 4     | 1102.04 | 2002-12-03 |
| 311   | Νίκος     | Στεργιόπουλος | 4     | 1386.05 | 2002-02-01 |



# Ο τελεστής OR



# Ο τελεστής OR

- 1 Δύο παραστάσεις στον όρο **WHERE** μπορούν να συνδυαστούν μεταξύ τους με το λογικό τελεστή **OR**.



# Ο τελεστής OR

- 1 Δύο παραστάσεις στον όρο **WHERE** μπορούν να συνδυαστούν μεταξύ τους με το λογικό τελεστή **OR**.
- 2 Χρησιμοποιούμε τον τελεστή **OR** όταν θέλουμε να ελέγξουμε την τιμή αληθείας τουλάχιστον μιας από τις δύο παραστάσεις.





# Ο τελεστής OR

- 1 Δύο παραστάσεις στον όρο **WHERE** μπορούν να συνδυαστούν μεταξύ τους με το λογικό τελεστή **OR**.
- 2 Χρησιμοποιούμε τον τελεστή **OR** όταν θέλουμε να ελέγξουμε την τιμή αληθείας τουλάχιστον μιας από τις δύο παραστάσεις.
- 3 Οι βάσεις δεδομένων ακολουθούν την τριαδική λογική: **TRUE, FALSE, UNK**.



# Ο τελεστής OR

- 1 Δύο παραστάσεις στον όρο **WHERE** μπορούν να συνδυαστούν μεταξύ τους με το λογικό τελεστή **OR**.
- 2 Χρησιμοποιούμε τον τελεστή **OR** όταν θέλουμε να ελέγξουμε την τιμή αληθείας τουλάχιστον μιας από τις δύο παραστάσεις.
- 3 Οι βάσεις δεδομένων ακολουθούν την τριαδική λογική: **TRUE, FALSE, UNK**.
- 4 Στο αποτέλεσμα του ερωτήματος εισέρχονται μόνο οι εγγραφές που αποδίδουν **TRUE**.



# Πίνακας αληθείας OR

| OR    | TRUE | FALSE | UNK  |
|-------|------|-------|------|
| TRUE  | TRUE | TRUE  | TRUE |
| FALSE | TRUE | FALSE | UNK  |
| UNK   | TRUE | UNK   | UNK  |

Ο τελεστής **OR** αποδίδει **TRUE** όταν μία τουλάχιστον παράσταση είναι **TRUE**.



# Ένα παράδειγμα με τον τελεστή OR

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων που εργάζονται στο τμήμα 4 ή ο μισθός τους είναι μεγαλύτερος από 1100 €

```
1 SELECT *  
2   FROM employees  
3   WHERE depid = 4  
4         OR salary >= 1100;
```

| empid | firstname | lastname      | depid | salary  | hiredate   |
|-------|-----------|---------------|-------|---------|------------|
| 206   | Νίκος     | Βλάχος        | 4     | 1102.04 | 2002-12-03 |
| 230   | Βαγγέλης  | Χριστόπουλος  | 4     | NULL    | 2002-12-03 |
| 234   | Αδαμαντία | Θεοτοκάτου    | 5     | 1534.65 | 1999-10-01 |
| 243   | Δέσποινα  | Παπαδοπούλου  | 2     | 1609.52 | 1999-03-05 |
| 311   | Νίκος     | Στεργιόπουλος | 4     | 1386.05 | 2002-02-01 |



# Μακριά από παγίδες

Να βρεθούν τα ονοματεπώνυμα των υπαλλήλων που εργάζονται στα τμήματα 1 και 3



# Μακριά από παγίδες

Να βρεθούν τα ονοματεπώνυμα των υπαλλήλων που εργάζονται στα τμήματα 1 και 3

## Λάθος απάντηση

```
1 SELECT firstname, lastname
2   FROM employees
3   WHERE depid = 1
4         AND depid = 3;
```



# Μακριά από παγίδες

Να βρεθούν τα ονοματεπώνυμα των υπαλλήλων που εργάζονται στα τμήματα 1 και 3

## Λάθος απάντηση

```
1 SELECT firstname, lastname
2   FROM employees
3   WHERE depid = 1
4         AND depid = 3;
```

## Σωστή απάντηση

```
1 SELECT firstname, lastname
2   FROM employees
3   WHERE depid = 1
4         OR depid = 3;
```



# Σύνθετες παραστάσεις και παρενθέσεις

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων που δεν προσλήφθηκαν το 2002 και εργάζονται στο τμήμα 3





# Σύνθετες παραστάσεις και παρενθέσεις

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων που δεν προσλήφθηκαν το 2002 και εργάζονται στο τμήμα 3

## Λάθος απάντηση

```
1 SELECT *  
2   FROM employees  
3   WHERE depid = 3  
4     AND hiredate < '2002-01-01'  
5     OR hiredate > '2002-12-31' ;
```



# Σύνθετες παραστάσεις και παρενθέσεις

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων που δεν προσλήφθηκαν το 2002 και εργάζονται στο τμήμα 3

## Λάθος απάντηση

```
1 SELECT *  
2   FROM employees  
3   WHERE depid = 3  
4     AND hiredate < '2002-01-01'  
5     OR hiredate > '2002-12-31' ;
```

## Σωστή απάντηση

```
1 SELECT *  
2   FROM employees  
3   WHERE depid = 3  
4     AND ( hiredate < '2002-01-01' OR  
5         hiredate > '2002-12-31' );
```



# Ασκήσεις με λογική σύζευξη και διάζευξη



# Ασκήσεις με λογική σύζευξη και διάζευξη

1  $\Pi_{empid,lastname,depid} (\sigma_{depid=1 \vee salary > 1500} (employees))$



# Ασκήσεις με λογική σύζευξη και διάζευξη

- 1  $\Pi_{empid,lastname,depid} (\sigma_{depid=1 \vee salary > 1500} (employees))$
- 2  $\Pi_{empid,lastname,depid} (\sigma_{depid=2 \wedge salary < 1200} (employees))$



# Ασκήσεις με λογική σύζευξη και διάζευξη

- 1  $\Pi_{empid,lastname,depid} (\sigma_{depid=1 \vee salary > 1500} (employees))$
- 2  $\Pi_{empid,lastname,depid} (\sigma_{depid=2 \wedge salary < 1200} (employees))$
- 3  $\Pi_{empid,lastname} (\sigma_{depid=4 \wedge salary * 0.05 > 60} (employees))$



# Ασκήσεις με λογική σύζευξη και διάζευξη

1  $\Pi_{empid,lastname,depid} (\sigma_{depid=1 \vee salary > 1500} (employees))$

2  $\Pi_{empid,lastname,depid} (\sigma_{depid=2 \wedge salary < 1200} (employees))$

3  $\Pi_{empid,lastname} (\sigma_{depid=4 \wedge salary * 0.05 > 60} (employees))$

4

$\Pi_{empid,lastname} (\sigma_{(depid=3 \vee depid=4) \wedge salary = NULL} (employees))$



# Περιεχόμενα

- 1 Απαλοιφή διπλοεγγραφών με DISTINCT
- 2 Έλεγχος για τιμές NULL
- 3 Ταίριασμα κειμένου
- 4 Λογική σύζευξη και διάζευξη
- 5 Τελεστής συνόλου IN**
- 6 Τελεστής περιοχής τιμών BETWEEN ... AND ...





# Ο τελεστής συνόλου IN

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων που εργάζονται στα τμήματα 2, 3 και 4

```
1 SELECT *  
2     FROM employees  
3     WHERE depid = 2  
4         OR depid = 3  
5         OR depid = 4;
```

Το ίδιο, αλλά πιο απλά

```
1 SELECT *  
2     FROM employees  
3     WHERE depid IN (2,3,4);
```



# Άρνηση του IN

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων που δεν εργάζονται στα τμήματα 2, 3 και 4

```
1 SELECT *  
2   FROM employees  
3   WHERE depid NOT IN (2,3,4);
```

Ή, ισοδύναμα με:

```
1 SELECT *  
2   FROM employees  
3   WHERE depid != 2  
4     AND depid != 3  
5     AND depid != 4;
```



# Περιεχόμενα

- 1 Απαλοιφή διπλοεγγραφών με DISTINCT
- 2 Έλεγχος για τιμές NULL
- 3 Ταίριασμα κειμένου
- 4 Λογική σύζευξη και διάζευξη
- 5 Τελεστής συνόλου IN
- 6 Τελεστής περιοχής τιμών BETWEEN ... AND ...



# Ο τελεστής περιοχής τιμών

Να βρεθούν όλες οι λεπτομέρειες των υπαλλήλων για εκείνους τους υπαλλήλους που παίρνουν μισθό από 1100 ως 1200 €

```
1 SELECT *  
2   FROM employees  
3   WHERE salary BETWEEN 1100 AND 1200;
```

| empid | firstname | lastname     | depid | salary  | hiredate   |
|-------|-----------|--------------|-------|---------|------------|
| 172   | Χρήστος   | Βλάσσης      | 3     | 1101.70 | 2000-07-04 |
| 206   | Νίκος     | Βλάχος       | 4     | 1102.04 | 2002-12-03 |
| 381   | Περικλής  | Κιτσάκης     | 6     | 1100.13 | 2003-02-14 |
| 431   | Κώστας    | Παπαδόπουλος | 3     | 1100.23 | 2002-09-16 |
| 503   | Μαριλένα  | Κρέσπα       | 2     | 1105.04 | 2001-03-07 |



# Ο τελεστής περιοχής τιμών για ημερομηνίες

Να βρεθεί ο κωδικός των τμημάτων στα οποία απασχολούνται υπάλληλοι που προσλήφθηκαν μέσα στο 2004

```
1 SELECT DISTINCT depid
2   FROM employees
3   WHERE hiredate BETWEEN '2004-01-01'
4                               AND '2004-12-31';
```

---

**depid**

---

4

3

---



Σας ευχαριστώ για την προσοχή σας

Είμαι στη διάθεσή σας για σχόλια, απορίες και ερωτήσεις

