

# Ερωτήματα SQL με σύζευξη πινάκων

Αθανάσιος Σταυρακούδης

<http://stavrakoudis.econ.uoi.gr>  
astavrak@uoi.gr  
@AStavrakoudis

Άνοιξη 2016



- 1 Γενικά για τη σύζευξη πινάκων
- 2 Καρτεσιανό γινόμενο και θ σύζευξη
- 3 Φυσική σύζευξη πινάκων στην SQL
- 4 Εσωτερική σύζευξη πινάκων στην SQL
- 5 Εξωτερική σύζευξη πινάκων στην SQL



# Σκοπός του μαθήματος

- 1 Εκτελείτε ερωτήματα ανάσυρσης δεδομένων από πολλούς πίνακες.
- 2 Εκτελείτε ερωτήματα που αντιστοιχούν στις σχεσιακές πράξεις καρτεσιανού γινομένου και σύζευξης.
- 3 Εφαρμόζετε κατάλληλες συνδέσεις (**JOIN**) πινάκων.
- 4 Αντιληφθείτε τις διαφορές και ομοιότητες ανάμεσα στους διαφορετικούς τύπους συζεύξεων.



# Το σχήμα της βάσης company

*departments*(depid, depname, manager)

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

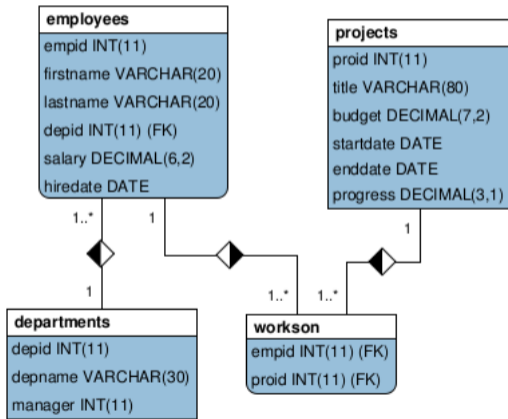
*projects*(proid, title, budget, startdate, enddate, progress)

*workson*(empid, proid)

- *departments*, τα τμήματα της εταιρείας
- *employees*, οι υπάλληλοι της εταιρείας
- *projects*, τα έργα που εκτελεί η εταιρεία
- *workson*, η απασχόληση των υπαλλήλων στα έργα



# Το σχήμα της βάσης company



- *departments*, τα τμήματα της εταιρείας
- *employees*, οι υπάλληλοι της εταιρείας
- *projects*, τα έργα που εκτελεί η εταιρεία
- *workson*, η απασχόληση των υπαλλήλων στα έργα



# Πρωτεύοντα και ξένα κλειδιά

- 1 Κάθε πίνακας έχει ένα **πρωτεύον κλειδί**.
- 2 Το **πρωτεύον κλειδί** μπορεί να είναι απλό (ένα πεδίο), ή σύνθετο (συνδυασμός πεδίων).
- 3 Κάθε εγγραφή ενός πίνακα μπορεί να προσδιοριστεί με τη χρήση του πρωτεύοντος κλειδιού.
- 4 Η σύνδεση δεδομένων από διαφορετικούς πίνακες **σύζευξη** γίνεται (συνήθως) με τη χρήση του **ξένου κλειδιού**,
- 5 Ένας πίνακας μπορεί να έχει πολλά ξένα κλειδιά ή να μην έχει κανένα.



# Συσχέτιση departments και employees N:1

- Ο πίνακας *departments* έχει **πρωτεύον κλειδί** το πεδίο *depid*.
- Ο πίνακας *employees* έχει **πρωτεύον κλειδί** το πεδίο *empid*.
- Ο πίνακας *employees* έχει **ξένο κλειδί** το πεδίο *depid*, το οποίο παίρνει τιμές που **αναφέρονται** στις τιμές του πεδίου *depid* του πίνακα *departments*:

*departments.depid* = *employees.depid*



# Συσχέτιση departments και employees N:1

- Ο πίνακας *departments* έχει **πρωτεύον κλειδί** το πεδίο *depid*.
- Ο πίνακας *employees* έχει **πρωτεύον κλειδί** το πεδίο *empid*.
- Ο πίνακας *employees* έχει **ξένο κλειδί** το πεδίο *depid*, το οποίο παίρνει τιμές που **αναφέρονται** στις τιμές του πεδίου *depid* του πίνακα *departments*:

*departments.depid = employees.depid*

- Το πεδίο *employees.depid* **δεν είναι πρωτεύον κλειδί** και δεν παίρνει μοναδικές τιμές: πολλοί υπάλληλοι εργάζονται στο ίδιο τμήμα.
- Οι πίνακες *departments* και *employees* συσχετίζονται μεταξύ τους με συσχέτιση **Πολλά προς Ένα**.





# Συσχέτιση departments - employees

employees - company1 - OpenOffice.org Base: Table Data

Αρχείο Επεξεργασία Προβολή Εισαγωγή Εργαλεία Παράθυρο Βοήθεια

empid	firstname	lastname	depid	salary	hiredate
102	Νικηφόρος	Διαμαντίδης	6	1212.50	2/6/2003
109	Μαρία	Αθανασίου	1	2787.69	26/1/2000
153	Μαρία	Αλεβιζάτου	2	1321.92	15/5/2001
172	Χρήστος	Βλάσης	3	1101.70	4/7/2000
189	Θεόδωρος	Αγγελίνας	6	1908.28	19/6/2000
205	Ηλίας	Μακρής	3	1050.96	23/2/2001
206	Νίκος	Βλάχος	4	1102.04	3/12/2002
230	Βαγγέλης	Χριστόπουλος	4		3/12/2002

Εγγραφή 2 από 30 (1)

depid	depname	manager
1	Διοίκησης/Επιβλεψης	109
2	Οικονομολόγων/Λογιστών	153
3	Επιστημόνων/Μηχανικών	431
4	Εξωτερικών συνεργατών	230
5	Γραμματείας	234
6	Μάνατζμεντ/Πωλήσεων	189

Εγγραφή 1 από 6 (1)



# Συσχέτιση departments και employees 1:1

- Ο πίνακας *departments* έχει **πρωτεύον κλειδί** το πεδίο *depid*.
- Ο πίνακας *employees* έχει **πρωτεύον κλειδί** το πεδίο *empid*.
- Ο πίνακας *departments* έχει **ξένο κλειδί** το πεδίο *manager*, το οποίο παίρνει τιμές που **αναφέρονται** στις τιμές του πεδίου *empid* του πίνακα *employees*:

*departments.manager* = *employees.empid*



# Συσχέτιση departments και employees 1:1

- Ο πίνακας *departments* έχει **πρωτεύον κλειδί** το πεδίο *depid*.
- Ο πίνακας *employees* έχει **πρωτεύον κλειδί** το πεδίο *empid*.
- Ο πίνακας *departments* έχει **ξένο κλειδί** το πεδίο *manager*, το οποίο παίρνει τιμές που **αναφέρονται** στις τιμές του πεδίου *empid* του πίνακα *employees*:

*departments.manager* = *employees.empid*

- Το πεδίο *departments.manager* **δεν είναι πρωτεύον κλειδί** αλλά παίρνει μοναδικές τιμές (**UNIQUE**).
- Οι πίνακες *departments* και *employees* συσχετίζονται μεταξύ τους με συσχέτιση **Ένα προς Ένα**.

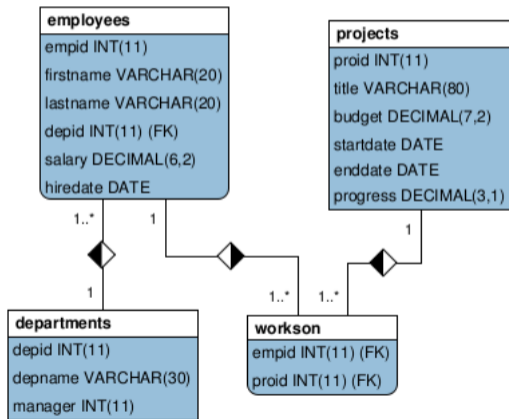


# Συσχέτιση employees και projects N:N

- 1 Ο πίνακας *projects* έχει **πρωτεύον κλειδί** το πεδίο *proid*.
- 2 Ο πίνακας *workson* έχει **πρωτεύον κλειδί** το συνδυασμό των πεδίων *empid* και *proid* (**σύνθετο κλειδί**).
- 3 Το πεδίο *empid* είναι **ξένο κλειδί** στον πίνακα *workson* και αναφέρεται στο πεδίο *employees.empid*.
- 4 Το πεδίο *proid* είναι **ξένο κλειδί** στον πίνακα *workson* και αναφέρεται στο πεδίο *projects.proid*.
- 5 Ένας υπάλληλος απασχολείται σε πολλά έργα, ένα έργο απασχολεί πολλούς υπαλλήλους, επομένως η συσχέτιση είναι **Πολλά προς Πολλά**.
- 6 Η σύζευξη των πινάκων *employees* και *projects* γίνεται μέσω του πίνακα *workson*.



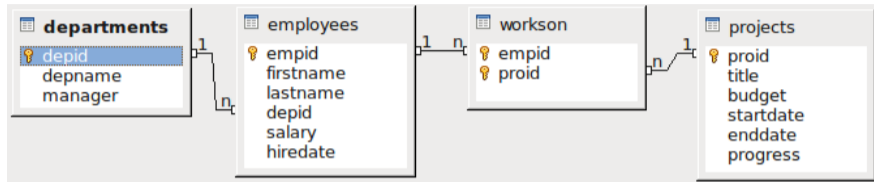
# Το σχήμα της βάσης company (MySQL)



- *departments*, τα τμήματα της εταιρείας
- *employees*, οι υπάλληλοι της εταιρείας
- *projects*, τα έργα που εκτελεί η εταιρεία
- *workson*, η απασχόληση των υπαλλήλων στα έργα



# Το σχήμα της βάσης company (.odp)



- *departments*, τα τμήματα της εταιρείας
- *employees*, οι υπάλληλοι της εταιρείας
- *projects*, τα έργα που εκτελεί η εταιρεία
- *workson*, η απασχόληση των υπαλλήλων στα έργα



# Περιεχόμενα

- 1 Γενικά για τη σύζευξη πινάκων
- 2 Καρτεσιανό γινόμενο και θ σύζευξη
- 3 Φυσική σύζευξη πινάκων στην SQL
- 4 Εσωτερική σύζευξη πινάκων στην SQL
- 5 Εξωτερική σύζευξη πινάκων στην SQL



# Καρτεσιανό γινόμενο

## Καρτεσιανό γινόμενο:

Ο συνδυασμός των πινάκων *departments* και *employees*  
 $departments \times employees$

λέγεται καρτεσιανό γινόμενο, και περιλαμβάνει όλους τους συνδυασμούς των εγγραφών των πινάκων *departments* και *employees*.





# Καρτεσιανό γινόμενο

## Καρτεσιανό γινόμενο:

Ο συνδυασμός των πινάκων *departments* και *employees*  
 $departments \times employees$

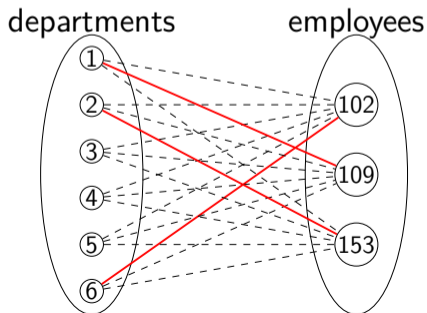
λέγεται καρτεσιανό γινόμενο, και περιλαμβάνει όλους τους συνδυασμούς των εγγραφών των πινάκων *departments* και *employees*.

- Στο αποτέλεσμα περιλαμβάνονται όλα τα πεδία των πινάκων *departments* και *employees*.
- Οι εγγραφές του αποτελέσματος περιλαμβάνει όλους τους συνδυασμούς των εγγραφών των πινάκων *departments* και *employees*.



# Καρτεσιανό γινόμενο *departments* $\times$ *employees*

- Όλες οι δυνατές συνδέσεις ανάμεσα στα τμήματα και σε δείγμα 3 υπαλλήλων.
- Μόνο οι **κόκκινες γραμμές** αντιστοιχούν στην πραγματικότητα: (1, 109), (6, 102), (2,153).
- Τα καρτεσιανό γινόμενο απεικονίζει όλα τα **πιθανά ενδεχόμενα**, όχι απαραίτητα τα **πραγματικά γεγονότα**.



## Καρτεσιανό γινόμενο τμημάτων και υπαλλήλων

*departments* × *employees*

```
1 SELECT *  
2 FROM departments, employees;
```

- Η σύνταξη στην **SQL** είναι απλή: γράφουμε τους πίνακες μετά τον όρο **FROM** και τους χωρίζουμε με κόμμα.
- Μπορούμε να γράψουμε περισσότερο από δύο πίνακες.
- Προσοχή! το αποτέλεσμα μπορεί να περιέχει μεγάλο όγκο εγγραφών, πχ δύο πίνακες με 6 και 30 εγγραφές αντίστοιχα δίνουν στο αποτέλεσμα  $6 \times 30 = 180$  εγγραφές.



# Καρτεσιανό γινόμενο στην SQL92, CROSS JOIN



# Καρτεσιανό γινόμενο στην SQL92, CROSS JOIN

## Με χρήση του όρου CROSS JOIN

```
1 SELECT *  
2 FROM departments CROSS JOIN employees;
```



# Καρτεσιανό γινόμενο στην SQL92, CROSS JOIN

## Με χρήση του όρου CROSS JOIN

```
1 SELECT *  
2 FROM departments CROSS JOIN employees;
```

## Η απλά JOIN

```
1 SELECT *  
2 FROM departments JOIN employees;
```



# Καρτεσιανό γινόμενο στην SQL92, CROSS JOIN

## Με χρήση του όρου CROSS JOIN

```
1 SELECT *  
2 FROM departments CROSS JOIN employees;
```

## Η απλά JOIN

```
1 SELECT *  
2 FROM departments JOIN employees;
```

- Οι δύο εκφράσεις είναι ισοδύναμες, θα επιστρέψουν το ίδιο αποτέλεσμα.
- Προτιμούμε τον πρώτο τρόπο **CROSS JOIN**, δηλώνει με πιο καθαρό τρόπο τη σύζευξη με βάση το καρτεσιανό γινόμενο.



# Αποτέλεσμα καρτεσιανού γινομένου

```
1 SELECT *
2 FROM departments, employees;
```

4	depid	depname	manager	empid	firstname	lastname	depid	salary	hiredate
5	-----								
6	1	Διοίκ./Επιβ.	109	102	Νικηφόρος	Διαμαντίδης	6	1212.50	2003-06-02
7	2	Οικον./Λογ.	153	102	Νικηφόρος	Διαμαντίδης	6	1212.50	2003-06-02
8	3	Επιστημ./Μηχ.	431	102	Νικηφόρος	Διαμαντίδης	6	1212.50	2003-06-02
9	4	Εξωτ. συνερ.	230	102	Νικηφόρος	Διαμαντίδης	6	1212.50	2003-06-02
10	5	Γραμματείας	234	102	Νικηφόρος	Διαμαντίδης	6	1212.50	2003-06-02
11	6	Μάνατζ./Πωλ.	189	102	Νικηφόρος	Διαμαντίδης	6	1212.50	2003-06-02
12	1	Διοίκ./Επιβ.	109	109	Μαρία	Αθανασίου	1	2787.69	2000-01-26
13	2	Οικον./Λογ.	153	109	Μαρία	Αθανασίου	1	2787.69	2000-01-26
14	3	Επιστημ./Μηχ.	431	109	Μαρία	Αθανασίου	1	2787.69	2000-01-26

```
15 .....
16 180 rows in set (0.00 sec)
```





# Επιλογή πεδίων από πίνακες

```
1 SELECT departments.depid, depname, empid, lastname
2 FROM departments, employees;
```

```
3
4 depid  depname          empid  lastname
5 -----
6      1  Διοίκ./Επιβ.      102    Διαμαντίδης
7      2  Οικον./Λογ.      153    Διαμαντίδης
8      3  Επιστημ./Μηχ.    431    Διαμαντίδης
9      4  Εξωτ. συνερ.    230    Διαμαντίδης
10     5  Γραμματείας    234    Διαμαντίδης
11     6  Μάνατζ./Πωλ.    189    Διαμαντίδης
12     1  Διοίκ./Επιβ.      109    Αθανασίου
13     2  Οικον./Λογ.      153    Αθανασίου
14     3  Επιστημ./Μηχ.    431    Αθανασίου
```

```
15 .....
16 180 rows in set (0.00 sec)
```



# Επιλογή πεδίων από πίνακες

```
1 SELECT departments.depid, depname, empid, lastname  
2 FROM departments, employees;
```

## Τι προσέχουμε:

- 1 Οποιοδήποτε όνομα πεδίου/στήλης υπάρχει στους πίνακες που ακολουθούν τον όρο **FROM** μπορούν να τοποθετηθούν μετά τον όρο **SELECT**.
- 2 Σε περίπτωση που το όνομα πεδίου είναι μοναδικό σε όλους τους πίνακες μπορούμε να το γράψουμε ως έχει, πχ *depname* ή *lastname*.
- 3 Αν το ίδιο όνομα πεδίου υπάρχει σε δύο διαφορετικούς πίνακες, τότε πρέπει να γραφεί με τη μορφή **πίνακας.πεδίο**.



## Τι προσέχουμε:

```
1 SELECT d.depid, d.depname, e.empid, e.lastname  
2 FROM departments d, employees e;
```

- 1 Η χρήση ψευδωνύμων είναι προαιρετική.
- 2 Βολεύει όταν το όνομα του πίνακα γράφεται πολλές φορές.



## Τι προσέχουμε:

```
1 SELECT d.depid, d.depname, e.empid, e.lastname  
2 FROM departments d, employees e;
```

- 1 Η χρήση ψευδωνύμων είναι προαιρετική.
- 2 Βολεύει όταν το όνομα του πίνακα γράφεται πολλές φορές.

## Ίδιοι κανόνες στην SQL92:

```
1 SELECT d.depid, d.depname, e.empid, e.lastname  
2 FROM departments d CROSS JOIN employees e;
```



# Καρτεσιανό γινόμενο υπαλλήλων και έργων

```
1 SELECT e.empid, p.proid
2 FROM employees e, projects p;
```

```
3
4 | empid | proid |
```

```
5 -----
6 | 102 | 5 |
```

```
7 | 102 | 12 |
```

```
8 | 102 | 14 |
```

```
9 | 102 | 21 |
```

```
10 | 102 | 38 |
```

```
11 | 102 | 43 |
```

```
12 | 109 | 5 |
```

```
13 | 109 | 12 |
```

```
14 .....
15
```

```
180 rows in set (0.00 sec)
```



# Καρτεσιανού γινόμενο υπαλλήλων και έργων

```
1 SELECT e.empid, p.proid
2 FROM employees e, projects p;
```



# Καρτεσιανού γινόμενο υπαλλήλων και έργων

```
1 SELECT e.empid, p.proid  
2 FROM employees e, projects p;
```

- 1 Το αποτέλεσμα συνδέει **όλους** τους υπαλλήλους με **όλα** τα έργα.
- 2 Είναι **πιθανό** ένας υπάλληλος να απασχολείται σε όλα τα έργα, αλλά αυτό δεν συμβαίνει **υποχρεωτικά** για όλους τους υπαλλήλους.
- 3 Είναι **πιθανό** ένα έργο να απασχολεί όλους τους υπαλλήλους, αλλά αυτό δεν συμβαίνει **υποχρεωτικά** για όλα τα έργα.
- 4 Το καρτεσιανό γινόμενο μας δίνει όλα τα **πιθανά ενδεχόμενα**, δεν μας αποκαλύπτει το «τι συμβαίνει».



# Σύζευξη θήτα στην SQL

## Σύζευξη τμημάτων και υπαλλήλων

*departments* ⋈<sub>departments.depid=employees.depid</sub> *employees*

```
1 SELECT *  
2 FROM departments, employees  
3 WHERE departments.depid = employees.depid;
```





# Σύζευξη θήτα στην SQL

## Σύζευξη τμημάτων και υπαλλήλων

*departments* ⋈<sub>departments.depid=employees.depid</sub> *employees*

```
1 SELECT *  
2 FROM departments, employees  
3 WHERE departments.depid = employees.depid;
```

- 1 Προσέξτε τη γραφή **πίνακας.πεδίο**.
- 2 Στη φράση **WHERE** μπορούν να προστεθούν επιπλέον περιορισμοί με λογική σύζευξη (**AND**).
- 3 Η θήτα σύζευξη είναι δυνατό να πραγματοποιηθεί και με πεδία που δεν έχουν το ίδιο όνομα.
- 4 Στο αποτέλεσμα εμφανίζονται μόνο οι εγγραφές για τις οποίες η συνθήκη είναι **TRUE**.



# θ σύζευξη, επιπλέον παράδειγμα

Τμήματα με υπαλλήλους με μισθό πάνω από 1500 €

```
1 SELECT DISTINCT d.depname
2   FROM departments d, employees e
3   WHERE d.depid = e.depid
4         AND e.salary > 1500;
```



# θ σύζευξη, επιπλέον παράδειγμα

## Τμήματα με υπαλλήλους με μισθό πάνω από 1500 €

```
1 SELECT DISTINCT d.depname
2   FROM departments d, employees e
3   WHERE d.depid = e.depid
4         AND e.salary > 1500;
```

- 1 Περιορισμός εγγραφών της θ σύζευξης με βάση μια παράσταση σύγκρισης.
- 2 Στο αποτέλεσμα εμφανίζονται μόνο οι εγγραφές για τις οποίες όλες οι είναι **TRUE**.
- 3 Η θ σύζευξη δεν έχει νόημα να ακολουθείται από λογική διάζευξη **OR**.



# θ σύζευξη στην SQL 1:1

Να βρεθούν τα ονοματεπώνυμα των υπαλλήλων που είναι διευθυντές τμημάτων

```
1 SELECT firstname, lastname  
2   FROM departments, employees  
3   WHERE departments.manager = employees.empid;
```



# θ σύζευξη στην SQL 1:1

Να βρεθούν τα ονοματεπώνυμα των υπαλλήλων που είναι διευθυντές τμημάτων

```
1 SELECT firstname, lastname
2     FROM departments, employees
3     WHERE departments.manager = employees.empid;
```

Να βρεθεί το ονοματεπώνυμο του υπαλλήλου που διευθύνει το τμήμα 2

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



# θ σύζευξη στην SQL N:N

Να βρεθεί ο τίτλος των έργων και τα ονοματεπώνυμα των υπαλλήλων που συμμετέχουν σε αυτά.

```
1 SELECT title, firstname, lastname  
2 FROM employees, workson, projects  
3 WHERE employees.empid = workson.empid  
4 AND workson.proid = projects.proid;
```



# Θ σύζευξη στην SQL N:N

Να βρεθεί ο τίτλος των έργων και τα ονοματεπώνυμα των υπαλλήλων που συμμετέχουν σε αυτά.

```
1 SELECT title, firstname, lastname
2   FROM employees, workson, projects
3   WHERE employees.empid = workson.empid
4         AND workson.proid = projects.proid;
```

Να βρεθεί ο τίτλος των έργων και τα ονοματεπώνυμα των υπαλλήλων που συμμετέχουν σε αυτά.

```
1 SELECT title, firstname, lastname
2   FROM employees, projects
3   WHERE employees.empid = workson.empid
4         AND workson.proid = projects.proid;
```



# θ ανισοσύζευξη στην SQL

Να βρεθούν τα ονοματεπώνυμα των υπαλλήλων που προσλήφθηκαν μετά την έναρξη του έργου με κωδικό 21.

```
1 SELECT firstname, lastname
2   FROM employees, workson, projects
3  WHERE employees.empid = workson.empid
4         AND workson.proid = projects.proid
5         AND employees.hiredate > projects.startdate
6         AND projects.proid = 21;
```





# Θ ανισοσύζευξη στην SQL

Να βρεθούν τα ονοματεπώνυμα των υπαλλήλων που προσλήφθηκαν μετά την έναρξη του έργου με κωδικό 21.

```
1 SELECT firstname, lastname
2   FROM employees, workson, projects
3   WHERE employees.empid = workson.empid
4         AND workson.proid = projects.proid
5         AND employees.hiredate > projects.startdate
6         AND projects.proid = 21;
```

Να βρεθούν τα ονοματεπώνυμα των υπαλλήλων που προσλήφθηκαν μετά την έναρξη του έργου με κωδικό 21.

```
1 SELECT firstname, lastname
2   FROM employees, projects
3   WHERE employees.hiredate > projects.startdate
4         AND projects.proid = 21;
```



# Μετονομασία πινάκων

## Με χρήση του AS

```
1 FROM employees AS e
```

## Χωρίς χρήση του AS

```
1 FROM employees e
```



# Μετονομασία πινάκων

## Με χρήση του AS

```
1 FROM employees AS e
```

## Χωρίς χρήση του AS

```
1 FROM employees e
```

## Στη σύζευξη

```
1 FROM departments d, employees e
```



# Μετονομασία πινάκων

## Με χρήση του AS

```
1 FROM employees AS e
```

## Χωρίς χρήση του AS

```
1 FROM employees e
```

## Στη σύζευξη

```
1 FROM departments d, employees e
```

## Στο ερώτημα

```
1 SELECT e.firstname, e.lastname  
2 FROM employees e, projects p  
3 WHERE e.hiredate > p.startdate  
4 AND p.projid = 21;
```



# Περιεχόμενα

- 1 Γενικά για τη σύζευξη πινάκων
- 2 Καρτεσιανό γινόμενο και θ σύζευξη
- 3 Φυσική σύζευξη πινάκων στην SQL**
- 4 Εσωτερική σύζευξη πινάκων στην SQL
- 5 Εξωτερική σύζευξη πινάκων στην SQL



## Σύζευξη τμημάτων και υπαλλήλων

*departments* ⋈ *employees*

```
1 SELECT *  
2 FROM departments NATURAL JOIN employees;
```



## Σύζευξη τμημάτων και υπαλλήλων

*departments* ⋈ *employees*

```
1 SELECT *  
2 FROM departments NATURAL JOIN employees;
```

- 1 Απαιτείται η ύπαρξη ενός τουλάχιστον κοινού πεδίου, εδώ το πεδίο **depid**.
- 2 Μόνο οι εγγραφές όπου οι τιμές του κοινού πεδίου ταυτίζονται υπάρχουν στο αποτέλεσμα του ερωτήματος.
- 3 Το κοινό γνώρισμα υπάρχει μόνο μία φορά στο αποτέλεσμα.



# Αποτέλεσμα φυσικής σύζευξης

```
1 SELECT *  
2 FROM departments NATURAL JOIN employees;  
3
```

4	depid	depname	manager	empid	firstname	lastname	salary	hiredate
5	-----							
6	1	Διοίκ./Επιβ.	109	109	Μαρία	Αθανασίου	2787.69	2000-01-26
7	1	Διοίκ./Επιβ.	109	502	Κρινιώ	Μαροπούλου	1754.67	2001-03-07
8	1	Διοίκ./Επιβ.	109	901	Κυριάκος	Ρούσσης	1852.99	2001-11-01
9	2	Οικον./Λογ.	153	153	Μαρία	Αλεβιζάτου	1321.92	2001-05-15
10	2	Οικον./Λογ.	153	243	Δέσποινα	Παπαδοπούλου	1609.52	1999-03-05

```
11 .....  
12 30 rows in set (0.00 sec)
```





# Προβολή και φυσική σύζευξη στην SQL

Να βρεθεί ο κωδικός και το επώνυμο των υπαλλήλων, καθώς και το όνομα του τμήματος στο οποίο απασχολούνται



# Προβολή και φυσική σύζευξη στην SQL

Να βρεθεί ο κωδικός και το επώνυμο των υπαλλήλων, καθώς και το όνομα του τμήματος στο οποίο απασχολούνται

$\Pi_{empid,lastname,depname}(departments \bowtie employees)$



# Προβολή και φυσική σύζευξη στην SQL

Να βρεθεί ο κωδικός και το επώνυμο των υπαλλήλων, καθώς και το όνομα του τμήματος στο οποίο απασχολούνται

$\Pi_{empid,lastname,depname}(departments \bowtie employees)$

```
1 SELECT empid, lastname, depname
2 FROM departments NATURAL JOIN employees;
```



# Φυσική σύζευξη τριών πινάκων στην SQL

Να βρεθεί το ονοματεπώνυμο των υπαλλήλων, ο κωδικός και ο τίτλος του έργου που απασχολούνται



# Φυσική σύζευξη τριών πινάκων στην SQL

Να βρεθεί το ονοματεπώνυμο των υπαλλήλων, ο κωδικός και ο τίτλος του έργου που απασχολούνται

$\Pi_{firstname,lastname,proid,title}(employees \bowtie workson \bowtie projects)$



# Φυσική σύζευξη τριών πινάκων στην SQL

Να βρεθεί το ονοματεπώνυμο των υπαλλήλων, ο κωδικός και ο τίτλος του έργου που απασχολούνται

$\Pi_{firstname,lastname,proid,title}(employees \bowtie workson \bowtie projects)$

```
1 SELECT firstname, lastname, proid, title
2   FROM employees NATURAL JOIN workson
3         NATURAL JOIN projects;
```



# Φυσική σύζευξη ή καρτεσιανό γινόμενο;

$departments \bowtie projects = departments \times projects$

$\Pi_{depid,proid}(employees \bowtie projects)$

```
1 SELECT depid, proid
2 FROM departments NATURAL JOIN projects;
```



# Φυσική σύζευξη ή καρτεσιανό γινόμενο;

$departments \bowtie projects = departments \times projects$

$\Pi_{depid,proid}(employees \bowtie projects)$

```
1 SELECT depid, proid
2 FROM departments NATURAL JOIN projects;
```

depid	proid
1	5
2	5
6	5
4	5
5	5
3	5
1	12
2	12
...	...

- 1 Το αποτέλεσμα είναι ίδιο με αυτό του φυσικού γινομένου **(ειδική περίπτωση)**.
- 2 Αν δεν υπάρχει κοινό πεδίο στους πίνακες της φυσικής σύζευξης τότε φυσική σύζευξη και καρτεσιανό γινόμενο θα δώσουν το ίδιο αποτέλεσμα.





# Περιορισμός εγγραφών στη φυσική σύζευξη

Να βρεθεί ο κωδικός και το επώνυμο των υπαλλήλων που απασχολούνται στο έργο με κωδικό 5

$$\Pi_{empid,lastname} (\sigma_{proid=5}(employees \bowtie workson))$$


# Περιορισμός εγγραφών στη φυσική σύζευξη

Να βρεθεί ο κωδικός και το επώνυμο των υπαλλήλων που απασχολούνται στο έργο με κωδικό 5

$\Pi_{empid,lastname} (\sigma_{proid=5}(employees \bowtie workson))$

```
1 SELECT empid, lastname
2 FROM employees NATURAL JOIN workson
3 WHERE proid = 5;
```



# Περιορισμός εγγραφών στη φυσική σύζευξη

Να βρεθεί ο κωδικός και το επώνυμο των υπαλλήλων που απασχολούνται στο έργο με κωδικό 5

$$\Pi_{empid,lastname} (\sigma_{proid=5}(employees \bowtie workson))$$

```
1 SELECT empid, lastname
2 FROM employees NATURAL JOIN workson
3 WHERE proid = 5;
```

Να βρεθεί ο κωδικός και το επώνυμο των υπαλλήλων που απασχολούνται στο έργο με κωδικό 5

$$\Pi_{empid,lastname} (\sigma_{proid=5}(employees))$$

```
1 SELECT empid, lastname
2 FROM employees
3 WHERE proid = 5;
```



# Περιεχόμενα

- 1 Γενικά για τη σύζευξη πινάκων
- 2 Καρτεσιανό γινόμενο και θ σύζευξη
- 3 Φυσική σύζευξη πινάκων στην SQL
- 4 Εσωτερική σύζευξη πινάκων στην SQL**
- 5 Εξωτερική σύζευξη πινάκων στην SQL



## Σύζευξη τμημάτων και υπαλλήλων

```
1 SELECT *  
2 FROM departments INNER JOIN employees  
3 ON departments.depid = employees.depid;
```



## Σύζευξη τμημάτων και υπαλλήλων

```
1 SELECT *  
2 FROM departments INNER JOIN employees  
3 ON departments.depid = employees.depid;
```



## Σύζευξη τμημάτων και υπαλλήλων

```
1 SELECT *  
2 FROM departments INNER JOIN employees  
3 ON departments.depid = employees.depid;
```

- 1 Οι δύο πίνακες ενώνονται με τον όρο **INNER JOIN**.



## Σύζευξη τμημάτων και υπαλλήλων

```
1 SELECT *  
2   FROM departments INNER JOIN employees  
3     ON departments.depid = employees.depid;
```

- 1 Οι δύο πίνακες ενώνονται με τον όρο **INNER JOIN**.
- 2 Αντί για **WHERE** υπάρχει (**υποχρεωτικά**) μετά το **INNER JOIN** η φράση **ON**.





## Σύζευξη τμημάτων και υπαλλήλων

```
1 SELECT *  
2 FROM departments INNER JOIN employees  
3 ON departments.depid = employees.depid;
```

- 1 Οι δύο πίνακες ενώνονται με τον όρο **INNER JOIN**.
- 2 Αντί για **WHERE** υπάρχει (**υποχρεωτικά**) μετά το **INNER JOIN** η φράση **ON**.
- 3 Το πεδίο σύζευξης (εδώ το *depid*) υπάρχει δύο φορές στο αποτέλεσμα.



## Σύζευξη τμημάτων και υπαλλήλων

```
1 SELECT *  
2 FROM departments INNER JOIN employees  
3 ON departments.depid = employees.depid;
```

- 1 Οι δύο πίνακες ενώνονται με τον όρο **INNER JOIN**.
- 2 Αντί για **WHERE** υπάρχει (**υποχρεωτικά**) μετά το **INNER JOIN** η φράση **ON**.
- 3 Το πεδίο σύζευξης (εδώ το *depid*) υπάρχει δύο φορές στο αποτέλεσμα.
- 4 Το πεδίο σύζευξης μπορεί να έχει διαφορετικό όνομα στους δύο πίνακες.



# Ισοδυναμία εσωτερικής και θ σύζευξης στην SQL

## Σύζευξη τμημάτων και υπαλλήλων με εσωτερική σύζευξη

```
1 SELECT *  
2   FROM departments INNER JOIN employees  
3   ON departments.depid = employees.depid;
```

## Σύζευξη τμημάτων και υπαλλήλων με θ σύζευξη

```
1 SELECT *  
2   FROM departments JOIN employees  
3   WHERE departments.depid = employees.depid;
```



# Όχι στις υπερβολές

Σύζευξη τμημάτων και υπαλλήλων με εσωτερική και θ σύζευξη

```
1 SELECT *  
2     FROM departments INNER JOIN employees  
3         ON departments.depid = employees.depid  
4 WHERE departments.depid = employees.depid;
```



## Σύζευξη τμημάτων και υπαλλήλων με εσωτερική και θ σύζευξη

```
1 SELECT *  
2 FROM departments INNER JOIN employees  
3 ON departments.depid = employees.depid  
4 WHERE departments.depid = employees.depid;
```

- 1 Περιττό.
- 2 Δεν είναι συντακτικά λάθος, είναι όμως εννοιολογικά μπερδεμένο.
- 3 Μία φορά αρκεί, η πολυλογία φέρνει λάθη.
- 4 Δυσκολία συντήρησης του κώδικα.



# Μετονομασία πινάκων

## Με χρήση του AS

```
1 FROM employees AS e
```

## Χωρίς χρήση του AS

```
1 FROM employees e
```



# Μετονομασία πινάκων

## Με χρήση του AS

```
1 FROM employees AS e
```

## Χωρίς χρήση του AS

```
1 FROM employees e
```

## Στη σύζευξη

```
1 FROM departments d INNER JOIN employees e
```



# Μετονομασία πινάκων

## Με χρήση του AS

```
1 FROM employees AS e
```

## Χωρίς χρήση του AS

```
1 FROM employees e
```

## Στη σύζευξη

```
1 FROM departments d INNER JOIN employees e
```

## Στο ερώτημα

```
1 SELECT *  
2 FROM departments d INNER JOIN employees e;
```





# Παράδειγμα INNER JOIN – 1

Να δοθεί το επώνυμο των υπαλλήλων και το όνομα του τμήματος στο οποίο εργάζονται

```
Π firstname,lastname,depname  
(departments ⋈departments.depid=employees.depid employees)
```



# Παράδειγμα INNER JOIN – 1

Να δοθεί το επώνυμο των υπαλλήλων και το όνομα του τμήματος στο οποίο εργάζονται

$\Pi_{firstname,lastname,depname}$   
 $(departments \bowtie_{departments.depid=employees.depid} employees)$

```
1 SELECT employees.lastname, departments.depname
2 FROM departments INNER JOIN employees
3 ON employees.depid = departments.depid;
```

- Η σύζευξη γίνεται με βάση το κοινό του πεδίο *depid*.
- Η σύζευξη με βάση **πρωτεύον** και **ξένο** κλειδί είναι η πλέον συνηθισμένη περίπτωση σύζευξης.



## Παράδειγμα INNER JOIN – 2

Να δοθεί το όνομα των υπαλλήλων, το όνομα του τμήματος στο οποίο εργάζονται, και ο μισθός τους

$$\Pi_{\text{firstname, lastname, depname, salary}} (\rho_d(\text{departments}) \bowtie_{d.\text{depid}=e.\text{depid}} \rho_e(\text{employees}))$$


# Παράδειγμα INNER JOIN – 2

Να δοθεί το όνομα των υπαλλήλων, το όνομα του τμήματος στο οποίο εργάζονται, και ο μισθός τους

$$\Pi_{\text{firstname, lastname, depname, salary}}(\sigma_d(\text{departments}) \bowtie_{d.\text{depid}=e.\text{depid}} \sigma_e(\text{employees}))$$

```
1 SELECT e.firstname, e.lastname, d.depname, e.salary
2 FROM employees e INNER JOIN departments d
3 ON e.depid = d.depid;
```

firstname	lastname	depname	salary
Μαρία	Αθανασίου	Διοίκησης/Επίβλεψης	2787.69
Κρινιώ	Μαροπούλου	Διοίκησης/Επίβλεψης	1754.67
Κυριάκος	Ρούσσης	Διοίκησης/Επίβλεψης	1852.99
Μαρία	Αλεβιζάτου	Οικονομολόγων/Λογιστών	1321.92
Δέσποινα	Παπαδοπούλου	Οικονομολόγων/Λογιστών	1609.52
Πέτρος	Αρβανιτάκης	Οικονομολόγων/Λογιστών	1323.80
...	...	...	...



## Παράδειγμα INNER JOIN – 3

Να δοθεί το όνομα των υπαλλήλων, το όνομα του τμήματος στο οποίο εργάζονται και ο μισθός τους, για υπαλλήλους με μισθό μεταξύ 1050 και 1300 €

$$\Pi_{firstname,lastname,depname,salary}(\sigma_{salary \geq 1050 \wedge salary \leq 1300}(\rho_d(departments) \bowtie_{d.depid=e.depid} \rho_e(employees)))$$


# Παράδειγμα INNER JOIN – 3

Να δοθεί το όνομα των υπαλλήλων, το όνομα του τμήματος στο οποίο εργάζονται και ο μισθός τους, για υπαλλήλους με μισθό μεταξύ 1050 και 1300 €

$$\Pi_{\text{firstname, lastname, depname, salary}}(\sigma_{\text{salary} \geq 1050 \wedge \text{salary} \leq 1300}(\rho_d(\text{departments}) \bowtie_{d.\text{depid} = e.\text{depid}} \rho_e(\text{employees})))$$

```
1 SELECT e.firstname, e.lastname, d.depname, e.salary
2 FROM employees e INNER JOIN departments d
3     ON e.depid = d.depid
4 WHERE e.salary BETWEEN 1050 AND 1300;
```

- Ο όρος **WHERE** μπορεί να χρησιμοποιηθεί για περιορισμό των εγγραφών.
- Η παράσταση συνθήκης μπορεί να αφορά οποιοδήποτε πεδίο από αυτά που υπάρχουν στους δύο πίνακες.



## Παράδειγμα INNER JOIN – 4

Κωδικός και όνομα όλων των υπαλλήλων που συμμετέχουν στο έργο με κωδικό 38, με αύξουσα ταξινόμηση ως προς το επώνυμο

$$\Pi_{empid,firstname,lastname}(\sigma_{proid=38}(\varrho_e(\text{employees}) \bowtie_{e.empid=w.empid} \varrho_w(\text{workson})))$$


# Παράδειγμα INNER JOIN – 4

Κωδικός και όνομα όλων των υπαλλήλων που συμμετέχουν στο έργο με κωδικό 38, με αύξουσα ταξινόμηση ως προς το επώνυμο

$\Pi_{empid,firstname,lastname}(\sigma_{proid=38}(\rho_e(\text{employees}) \bowtie_{e.empid=w.empid} \rho_w(\text{workson})))$

```
1 SELECT e.empid, e.firstname, e.lastname
2 FROM employees e INNER JOIN workson w
3 ON e.empid = w.empid
4 WHERE w.proid = 38
5 ORDER BY e.lastname ASC;
```

- Ο όρος **ORDER BY** πάντα στο τέλος.
- Προσέξτε πως χρειάζεται **σύζευξη** ακόμα και αν όλα τα πεδία που εμφανίζονται μετά τον όρο **SELECT** βρίσκονται σε ένα πίνακα.
- Ξαναγράψτε την απάντηση με υποερώτημα (παρακάτω μάθημα).





# Πολλά προς πολλά

Να βρεθούν τα ονόματα των υπαλλήλων και ο κωδικός και ο προϋπολογισμός των έργων στα οποία συμμετέχουν υπάλληλοι με μισθό μεγαλύτερο από €1700

$$\Pi_{firstname,lastname,proid,budget}(\sigma_{salary>1700}(\rho_e(\text{employees}) \bowtie_{d.depid=e.depid} \rho_w(\text{workson}) \bowtie_{w.proid=p.proid} \rho_p(\text{projects})))$$


# Πολλά προς πολλά

Να βρεθούν τα ονόματα των υπαλλήλων και ο κωδικός και ο προϋπολογισμός των έργων στα οποία συμμετέχουν υπάλληλοι με μισθό μεγαλύτερο από €1700

$$\Pi_{firstname, lastname, proid, budget} (\sigma_{salary > 1700} (\rho_e(\text{employees}) \bowtie_{d.depid=e.depid} \rho_w(\text{workson}) \bowtie_{w.proid=p.proid} \rho_p(\text{projects})))$$

```
1 SELECT e.firstname, e.lastname, p.proid, p.budget
2 FROM (employees e INNER JOIN workson w
3      ON e.empid = w.empid)
4      INNER JOIN projects p
5      ON p.proid = w.proid
6 WHERE e.salary > 1700;
```



# Πολλά προς πολλά – Εναλλακτικός τρόπος

Να βρεθούν τα ονόματα των υπαλλήλων και ο κωδικός και ο προϋπολογισμός των έργων στα οποία συμμετέχουν υπάλληλοι με μισθό μεγαλύτερο του 1700€

$\Pi_{\text{firstname, lastname, proid, budget}}(\sigma_{e.\text{empid}=w.\text{empid} \wedge p.\text{proid}=w.\text{proid} \wedge e.\text{salary} > 1700}(\varrho_e(\text{employees}) \times \varrho_w(\text{workson}) \times \varrho_p(\text{projects})))$

```
1 SELECT e.firstname, e.lastname, p.proid, p.budget
2 FROM employees e, workson w, projects p
3 WHERE e.empid = w.empid
4 AND p.proid = w.proid
5 AND e.salary > 1700;
```



# Ερώτημα με 4 πίνακες

Να βρεθεί το όνομα των υπαλλήλων και του τμήματος των υπαλλήλων για όλους τους υπαλλήλους που προσλήφθηκαν μετά από την 1/1/2002 και απασχολούνται σε έργα με βαθμό προόδου τουλάχιστον 20%

```

$$\Pi_{firstname, lastname, depname} (\sigma_{hiredate > '2002-01-01' \wedge budget > 100000}$$

$$(\rho_d(departments) \bowtie_{d.depid=e.depid} \rho_e(employees))$$

$$\bowtie_{e.empid=w.empid} \rho_w(workson) \bowtie_{w.proid=p.proid} \rho_p(projects))$$

```



# Ερώτημα με 4 πίνακες

Να βρεθεί το όνομα των υπαλλήλων και του τμήματος των υπαλλήλων για όλους τους υπαλλήλους που προσλήφθηκαν μετά από την 1/1/2002 και απασχολούνται σε έργα με βαθμό προόδου τουλάχιστον 20%

$\Pi_{firstname, lastname, depname} (\sigma_{hiredate > '2002-01-01' \wedge budget > 100000}$   
 $(\rho_d(departments) \bowtie_{d.depid=e.depid} \rho_e(employees))$   
 $\bowtie_{e.empid=w.empid} \rho_w(workson) \bowtie_{w.proid=p.proid} \rho_p(projects))$

```
1 SELECT DISTINCT e.firstname, e.lastname, d.depname
2 FROM ((departments d INNER JOIN employees e
3      ON d.depid = e.depid)
4      INNER JOIN workson w
5      ON e.empid = w.empid)
6      INNER JOIN projects p
7      ON p.proid = w.proid
8 WHERE e.hiredate > '2002-01-01'
9      AND p.progress > 20;
```



# Περιεχόμενα

- 1 Γενικά για τη σύζευξη πινάκων
- 2 Καρτεσιανό γινόμενο και θ σύζευξη
- 3 Φυσική σύζευξη πινάκων στην SQL
- 4 Εσωτερική σύζευξη πινάκων στην SQL
- 5 Εξωτερική σύζευξη πινάκων στην SQL



# Παράδειγμα εξωτερικής σύζευξης – το πρόβλημα

Να βρεθούν τα ονόματα των υπαλλήλων του τμήματος 4 και οι κωδικοί των έργων που συμμετέχουν

$$\Pi_{e.firstname, e.lastname, w.proid}(\sigma_{e.depid=4}(\rho_e(employees) \bowtie_{d.depid=e.depid} \rho_w(workson))))$$


# Παράδειγμα εξωτερικής σύζευξης – το πρόβλημα

Να βρεθούν τα ονόματα των υπαλλήλων του τμήματος 4 και οι κωδικοί των έργων που συμμετέχουν

$$\Pi_{e.firstname, e.lastname, w.proid}(\sigma_{e.depid=4}(\rho_e(employees) \bowtie_{d.depid=e.depid} \rho_w(workson))))$$

```
1 SELECT e.firstname, e.lastname, w.proid
2 FROM employees e INNER JOIN workson w
3 ON e.empid = w.empid
4 WHERE e.depid = 4;
```

firstname	lastname	proid
Νίκος	Βλάχος	12
Βαγγέλης	Χριστόπουλος	12
Βαγγέλης	Χριστόπουλος	14
Βαγγέλης	Χριστόπουλος	38
Παύλος	Περίδης	43

Τι γίνεται με τους υπαλλήλους του τμήματος 4 που δεν απασχολούνται σε κανένα έργο;





# Παράδειγμα εξωτερικής σύζευξης – Η λύση

Υπάλληλοι του τμήματος 4 και οι κωδικοί των έργων

$$\Pi_{e.firstname, e.lastname, w.proid} (\sigma_{e.depid=4} (\rho_e(employees) \bowtie_{e.empid=w.empid} \rho_w(workson))))$$


# Παράδειγμα εξωτερικής σύζευξης – Η λύση

Υπάλληλοι του τμήματος 4 και οι κωδικοί των έργων

$$\Pi_{e.firstname, e.lastname, w.proid} (\sigma_{e.depid=4} (\rho_e(\text{employees}) \bowtie_{e.empid=w.empid} \rho_w(\text{workson})))$$

```
1 SELECT e.firstname, e.lastname, w.proid
2 FROM employees e LEFT JOIN workson w
3 ON e.empid = w.empid
4 WHERE e.depid = 4;
```



# Παράδειγμα εξωτερικής σύζευξης – Η λύση

Υπάλληλοι του τμήματος 4 και οι κωδικοί των έργων

$$\Pi_{e.firstname, e.lastname, w.proid} (\sigma_{e.depid=4} (\rho_e(\text{employees}) \bowtie_{e.empid=w.empid} \rho_w(\text{workson})))$$

```
1 SELECT e.firstname, e.lastname, w.proid
2 FROM employees e LEFT JOIN workson w
3 ON e.empid = w.empid
4 WHERE e.depid = 4;
```

firstname	lastname	proid
Νίκος	Βλάχος	12
Βαγγέλης	Χριστόπουλος	12
Βαγγέλης	Χριστόπουλος	14
Βαγγέλης	Χριστόπουλος	38
Νίκος	Στεργιόπουλος	NULL
Πάυλος	Περίδης	43
Ευθαλεία	Μικράκη	NULL

Η στήλη *proid*  
συμπληρώνεται με **NULL**  
για τους υπαλλήλους που  
δεν απασχολούνται σε  
κανένα έργο.



# Αριστερή και δεξιά σύζευξη – Ισοδυναμία

## Αριστερή σύζευξη

$$\Pi_{e.firstname, e.lastname, w.proid}(\sigma_{e.depid=4}(\rho_e(\text{employees}) \bowtie_{e.empid=w.empid} \rho_w(\text{workson})))$$

```
1 SELECT e.firstname, e.lastname, w.proid
2   FROM employees e LEFT JOIN workson w
3     ON e.empid = w.empid
4  WHERE e.depid = 4;
```

## Δεξιά σύζευξη

$$\Pi_{e.firstname, e.lastname, w.proid}(\sigma_{e.depid=4}(\rho_w(\text{workson}) \bowtie_{w.empid=e.empid} \rho_e(\text{employees})))$$

```
1 SELECT e.firstname, e.lastname, w.proid
2   FROM workson w RIGHT JOIN employees e
3     ON w.empid = e.empid
4  WHERE e.depid = 4;
```



# Αριστερή και δεξιά σύζευξη

- 1 Οι δύο προτάσεις **SQL** είναι απολύτως **ισοδύναμες** και δίνουν το ίδιο αποτέλεσμα.
- 2 Κατά παράδοση, προτιμάται η **αριστερή** σύζευξη.
- 3 Οι εξωτερικές συζεύξεις χρησιμοποιούνται όταν θέλουμε στο αποτέλεσμα όλες τις εγγραφές ενός πίνακα, ανεξάρτητα αν αυτές έχουν σύνδεση με τον άλλο πίνακα που υπάρχει στη σύζευξη.
- 4 Κατά την εξωτερική σύζευξη, αν υπάρχουν μη συνδεδεμένες εγγραφές, δημιουργούνται τιμές **NULL**.
- 5 Ο έλεγχος (**WHERE**) για τιμές **NULL** είναι πολύ συχνός στις εξωτερικές συνδέσεις.



# Εξωτερική σύζευξη και έλεγχος NULL

Ποιοι υπάλληλοι δεν απασχολούνται σε κανένα έργο;

$$\Pi_{e.*}(\sigma_{w.empid \text{ IS NULL}} \\ (\rho_e(\text{employees}) \bowtie_{e.empid=w.empid} \rho_w(\text{workson})))$$


# Εξωτερική σύζευξη και έλεγχος NULL

Ποιοι υπάλληλοι δεν απασχολούνται σε κανένα έργο;

$$\Pi_{e.*}(\sigma_{w.empid \text{ IS NULL}}(\rho_e(\text{employees}) \bowtie_{e.empid=w.empid} \rho_w(\text{workson})))$$

```
1 SELECT e.*
2 FROM employees e LEFT JOIN workson w
3 ON e.empid = w.empid
4 WHERE w.empid IS NULL;
```

- 1 Το ερώτημα δε μπορεί να απαντηθεί με φυσική ή εσωτερική σύζευξη.
- 2 Οι τιμές των πεδίων *e.empid* και *w.empid* είτε ταυτίζονται, είτε κάποια τιμή του πεδίου *e.empid* δεν έχει αντίστοιχη τιμή στον πίνακα *workson*.



# Εξωτερική σύζευξη – επιπλέον ανάλυση

Ποιοι υπάλληλοι δεν απασχολούνται σε κανένα έργο;

```
1 SELECT e.empid, w.empid
2 FROM employees e LEFT JOIN workson w
3 ON e.empid = w.empid
4 WHERE w.empid IS NULL;
```





# Εξωτερική σύζευξη – επιπλέον ανάλυση

Ποιοι υπάλληλοι δεν απασχολούνται σε κανένα έργο;

```
1 SELECT e.empid, w.empid
2 FROM employees e LEFT JOIN workson w
3 ON e.empid = w.empid
4 WHERE w.empid IS NULL;
```

<i>e.empid</i>	<i>w.empid</i>
----------------	----------------

205	NULL
-----	------

311	NULL
-----	------

342	NULL
-----	------

...	...
-----	-----



# Εξωτερική σύζευξη – επιπλέον ανάλυση

Ποιοι υπάλληλοι δεν απασχολούνται σε κανένα έργο;

```
1 SELECT e.empid, w.empid
2 FROM employees e LEFT JOIN workson w
3 ON e.empid = w.empid
4 WHERE w.empid IS NULL;
```

<i>e.empid</i>	<i>w.empid</i>
----------------	----------------

205	NULL
-----	------

311	NULL
-----	------

342	NULL
-----	------

...	...
-----	-----

- 205: υπάρχει στον πίνακα *employees* αλλά δεν υπάρχει στον πίνακα *workson*.
- Η αριστερή σύζευξη επιτρέπει την εμφάνιση της τιμής 205 στο πεδίο *e.empid*.
- Το πεδίο *w.empid* θα συμπληρωθεί με **NULL**.



# Εξωτερική σύζευξη – σύνοψη



# Εξωτερική σύζευξη – σύνοψη

- 1 Κάποιες εγγραφές του πίνακα *employees* δεν έχουν ταιριαστές εγγραφές στον πίνακα *workson*.
- 2 Ο πίνακας *workson* δεν περιέχει καμία εγγραφή με **NULL** τιμές. Ο τρόπος με τον οποίο έγινε η σύζευξη των πινάκων παρήγαγε τις τιμές **NULL**.
- 3 Η αριστερή εξωτερική σύζευξη ορίζει πως στο αποτέλεσμα θα υπάρχουν όλες οι εγγραφές του αριστερού πίνακα και στα πεδία του δεξιού πίνακα θα τοποθετηθούν είτε τιμές που αντιστοιχούν στον κανόνα της σύζευξης είτε τιμές **NULL**, εκεί όπου δεν βρέθηκαν ταιριαστές εγγραφές.
- 4 Επομένως οι εγγραφές με **NULL** τιμές στα πεδία του πίνακα *workson*, δεν οφείλονται σε αποθηκευμένες τιμές αλλά σε παραγόμενες μετά από εξωτερική σύζευξη.



# Εξωτερική σύζευξη 1:1

Να βρεθεί ο κωδικός και το όνομα των υπαλλήλων που δεν είναι διευθυντές

```
1 SELECT e.empid, e.firstname, e.lastname
2     FROM employees e LEFT JOIN departments d
3         ON e.empid = d.manager
4     WHERE d.manager IS NULL;
```



# Εξωτερική σύζευξη 1:1

Να βρεθεί ο κωδικός και το όνομα των υπαλλήλων που δεν είναι διευθυντές

```
1 SELECT e.empid, e.firstname, e.lastname  
2 FROM employees e LEFT JOIN departments d  
3 ON e.empid = d.manager  
4 WHERE d.manager IS NULL;
```

Να βρεθεί ο κωδικός και το όνομα των υπαλλήλων που δεν είναι διευθυντές

```
1 SELECT e.empid, e.firstname, e.lastname  
2 FROM employees e INNER JOIN departments d  
3 ON e.empid = d.manager  
4 WHERE d.manager IS NULL;
```



# Εξωτερική ή εσωτερική σύζευξη;

Να βρεθεί ο κωδικός και το όνομα των υπαλλήλων που **είναι** διευθυντές

```
1 SELECT e.empid, e.firstname, e.lastname
2 FROM employees e LEFT JOIN departments d
3 ON e.empid = d.manager
4 WHERE d.manager IS NOT NULL;
```



# Εξωτερική ή εσωτερική σύζευξη;

Να βρεθεί ο κωδικός και το όνομα των υπαλλήλων που **είναι** διευθυντές

```
1 SELECT e.empid, e.firstname, e.lastname
2 FROM employees e LEFT JOIN departments d
3 ON e.empid = d.manager
4 WHERE d.manager IS NOT NULL;
```

Να βρεθεί ο κωδικός και το όνομα των υπαλλήλων που **είναι** διευθυντές

```
1 SELECT e.empid, e.firstname, e.lastname
2 FROM employees e INNER JOIN departments d
3 ON e.empid = d.manager;
```

**Σχολιάστε διαφορές και ομοιότητες στο αποτέλεσμα.**





Σας ευχαριστώ  
για την προσοχή σας

Είμαι στη διάθεσή σας για σχόλια, απορίες και ερωτήσεις

