



HY-360

Αρχεία και Βάσεις Δεδομένων

Διδάσκων: Δ. Πλεξουσάκης

Φροντιστήριο

SQL – Examples II

Ξένου Ρουμπίνη



SQL(DML) - Query Example 1

- Query:1

- Βρείτε τα ονόματα των έργων που δεν αφορούν το τμήμα “research” ή το τμήμα “design”.

PROJECT

<u>PNO</u>	PNAME	PLOCATION	DEPTNO
------------	-------	-----------	--------

{2, 7}

```
-SELECT pname FROM project WHERE deptno  
NOT IN( SELECT deptno FROM department  
WHERE deptname IN ('research', 'design')  
);
```

```
-WHERE deptname IN  
('design', 'research')
```

ή

```
-WHERE deptname='research'  
OR deptname ='design'
```

DEPARTMENT

<u>DEPTNO</u>	<u>DEPTNAME</u>	MGRSSN	MGRSTARDATE
1	Marketing	3244	2009/09/27
2	Research	1235	2002/04/14
3	Construction	745	2009/09/27
...
7	design	1256	2008/09/11



SQL(DML) - Query Example 1

- Query:1

- Βρείτε τα ονόματα των έργων που δεν αφορούν το τμήμα “research” ή το τμήμα “design”.

```
-SELECT pname FROM project p, department d  
WHERE p.deptno = d.deptno AND  
d.deptname <> 'research' AND d.deptname <> 'design';
```

ή

```
-SELECT pname FROM project p, department d  
WHERE d.deptno=p.deptno AND NOT (d.deptname='research' OR d.deptname  
='design');
```



SQL(DML) - Query Example 2

- Query:2

- Βρείτε τα ονοματεπώνυμα των υπαλλήλων που εργάζονται τουλάχιστον σε ένα project.

```
-SELECT name, fname FROM employee  
WHERE ssn = ANY( SELECT distinct ssn FROM works_on );
```

ή

```
-SELECT name, fname FROM employee e  
WHERE e.ssn IN ( SELECT distinct ssn FROM works_on );
```

WORKS_ON

<u>SSN</u>	<u>PNO</u>	HOURS
------------	------------	-------

EMPLOYEE

<u>SSN</u>	NAME	MID	FNAME	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DEPTNO
------------	------	-----	-------	-------	---------	-----	--------	----------	--------



SQL(DML) - Query Example 2

- Query:2

- Βρείτε τα ονοματεπώνυμα των υπαλλήλων που εργάζονται τουλάχιστον σε ένα project.

```
SELECT distinct name, fname FROM employee e, works_on w  
WHERE w.ssn = e.ssn;
```

WORKS_ON

<u>SSN</u>	<u>PNO</u>	HOURS
------------	------------	-------

EMPLOYEE

<u>SSN</u>	NAME	MID	FNAME	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DEPTNO
------------	------	-----	-------	-------	---------	-----	--------	----------	--------



SQL(DML) - Query Example 3

- Query:3

- Βρείτε τα ονοματεπώνυμα των υπαλλήλων που έχουν μικρότερο μισθό από οποιονδήποτε υπάλληλο που εργάζεται στο τμήμα 3.

```
-SELECT name, fname FROM employee  
WHERE salary < all (SELECT salary FROM employee  
                    WHERE deptno = 3);
```

EMPLOYEE

<u>SSN</u>	NAME	MID	FNAME	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DEPTNO
------------	------	-----	-------	-------	---------	-----	--------	----------	--------



SQL(DML) - Query Example 4

- Query:4

- Βρείτε τα ονοματεπώνυμα των υπαλλήλων που δεν εργάζονται σε project που ελέγχεται από το τμήμα 3.

EMPLOYEE

<u>SSN</u>	NAME	MID	FNAME	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DEPTNO
------------	------	-----	-------	-------	---------	-----	--------	----------	--------

```
- SELECT name, fname  
FROM employee  
WHERE ssn <> all (  
    SELECT ssn  
    FROM works_on  
    WHERE pno = any (  
        SELECT pno  
        FROM project  
        WHERE deptno = 3  
    )  
);
```

NOT IN

WORKS_ON

<u>SSN</u>	<u>PNO</u>	HOURS
------------	------------	-------

IN

PROJECT

<u>PNO</u>	PNAME	PLOCATION	DEPTNO
------------	-------	-----------	--------



SQL(DML) - Query Example 4

- Query:4

- Βρείτε τα ονοματεπώνυμα των υπαλλήλων που δεν εργάζονται σε project που ελέγχεται από το τμήμα 3.

EMPLOYEE

<u>SSN</u>	NAME	MID	FNAME	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DEPTNO
------------	------	-----	-------	-------	---------	-----	--------	----------	--------

```
- SELECT name, fname
FROM employee E
WHERE NOT EXISTS (
  SELECT *
  FROM works_on W, project P
  WHERE W.pno = P.pno
  AND E.ssn = W.ssn
  AND deptno = 3
```

WORKS_ON

<u>SSN</u>	<u>PNO</u>	HOURS
------------	------------	-------

- NOT EXISTS ελέγχει την ύπαρξη πλειάδων της υποερώτησης (subquery) κενό => αληθής.

PROJECT

<u>PNO</u>	PNAME	PLOCATION	DEPTNO
------------	-------	-----------	--------



SQL(DML) - Query Example 5

- Query:5

- Βρείτε τα ονοματεπώνυμα των υπαλλήλων που έχουν τουλάχιστον έναν εξαρτώμενο του ίδιου φύλου.

EMPLOYEE

SSN	NAME	MID	FNAME	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DEPTNO
-----	------	-----	-------	-------	---------	-----	--------	----------	--------

```
– SELECT name, fname
```

```
FROM employee E
```

```
WHERE EXISTS (
```

```
    SELECT *
```

```
    FROM dependent D
```

```
    WHERE ssn = essn
```

```
        AND E.sex = D.sex
```

```
);
```

- EXISTS ελέγχει την ύπαρξη πλειάδων της υποερώτησης (subquery) μη κενό => αληθής.

DEPENDENT

ESSN	DEPENDNAME	BDATE	SEX	RELATIONSHIP
------	------------	-------	-----	--------------



SQL(DML)

- Για τα επόμενα παραδείγματα θα χρησιμοποιηθεί το παρακάτω σχεσιακό σχήμα.

PRODUCT

<u>PID</u>	STOCK _x	SUPPLIER
------------	--------------------	----------

ORDER

<u>ORDERNO</u>	DATE	QUANTITY	PID	CID
----------------	------	----------	-----	-----

CLIENT

<u>CID</u>	NAME	ADDRESS	CITY
------------	------	---------	------

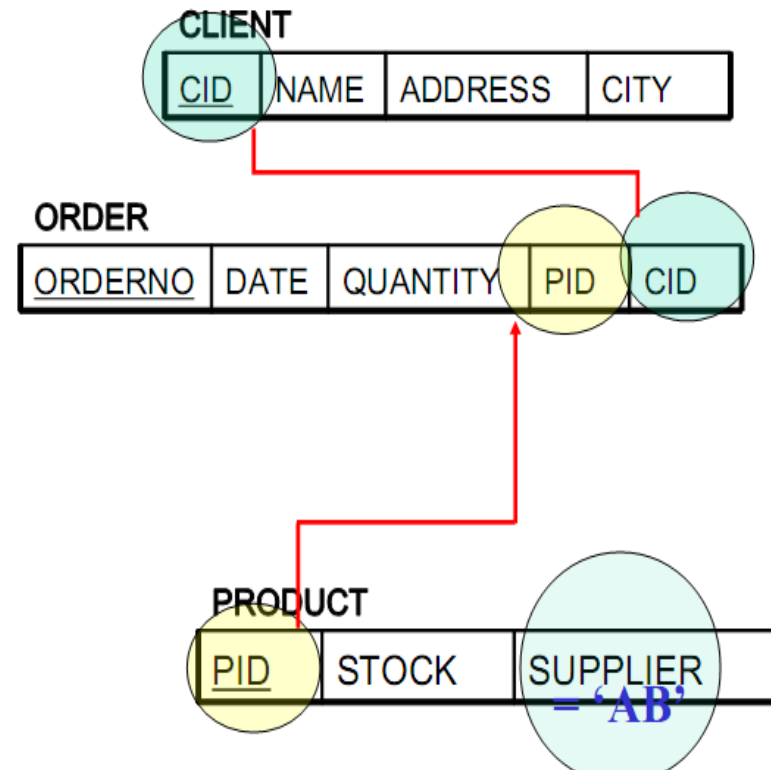


SQL(DML) - Query Example 6

- Query:6

- Βρείτε τις πόλεις στις οποίες μένουν πελάτες οι οποίοι δεν δίνουν παραγγελίες για προϊόντα που προμηθεύει ο «AB».

```
SELECT city
FROM client
WHERE cid <> all (
  SELECT cid
  FROM order
  WHERE pid = any (
    SELECT pid
    FROM product
    WHERE supplier = 'AB'
  )
);
```



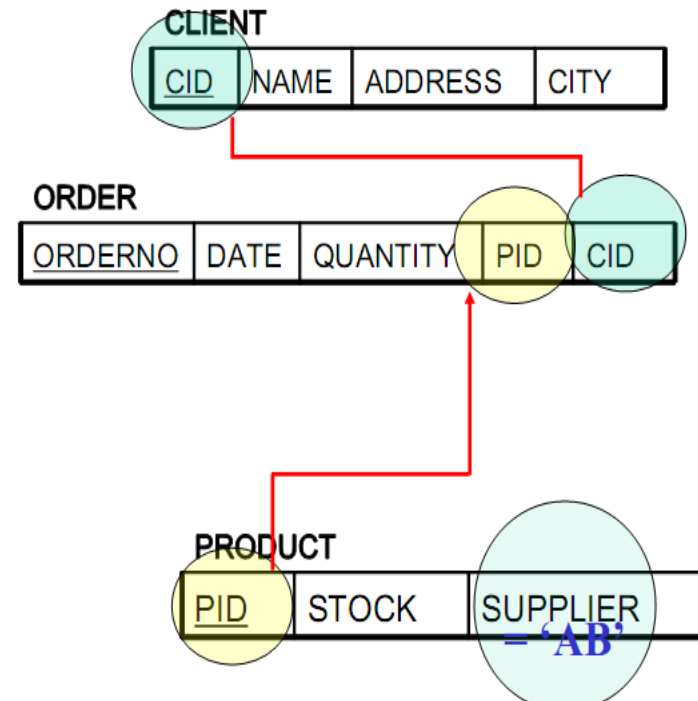


SQL(DML) - Query Example 6

- Query:6

- Βρείτε τις πόλεις στις οποίες μένουν πελάτες οι οποίοι δεν δίνουν παραγγελίες για προϊόντα που προμηθεύει ο «AB».

```
- SELECT city  
FROM client C  
WHERE NOT EXISTS (  
    SELECT *  
    FROM order O, product P  
    WHERE C.cid = O.cid  
          AND O.pid = P.pid  
          AND P.supplier = 'AB'  
)  
;
```

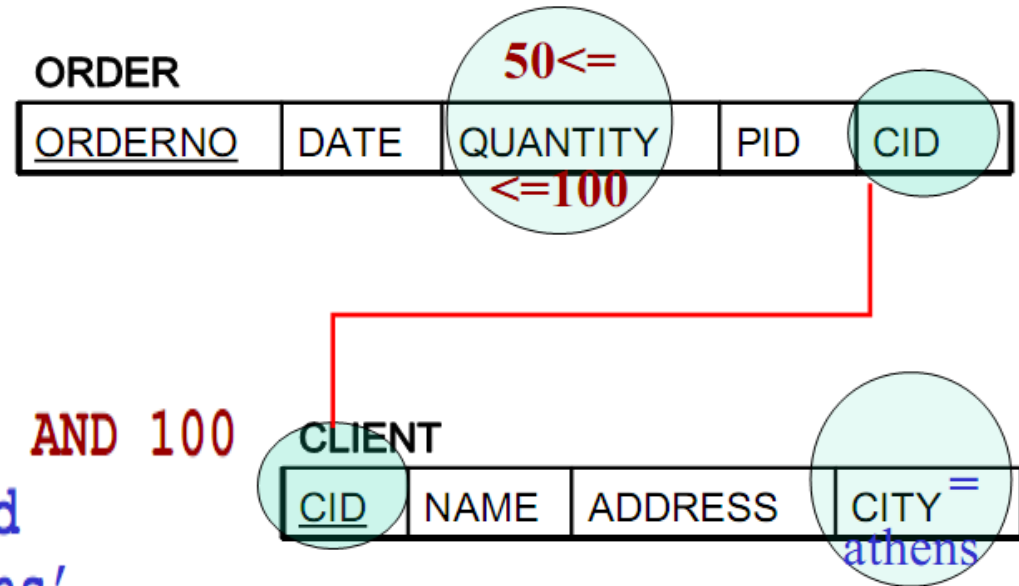




SQL(DML) - Query Example 7

- Query:7

- Βρείτε τους αριθμούς των παραγγελιών για τα προϊόντα που παραγγέλνονται σε ποσότητα από 50 μέχρι 100 τεμάχια και από πελάτες που βρίσκονται στην Αθήνα.



```
SELECT orderno  
FROM order O, client C  
WHERE quantity BETWEEN 50 AND 100  
      AND C.cid = O.cid  
      AND C.city='athens'
```

- quantity BETWEEN 50 AND 100
 ή
- quantity >=50 AND quantity <= 100

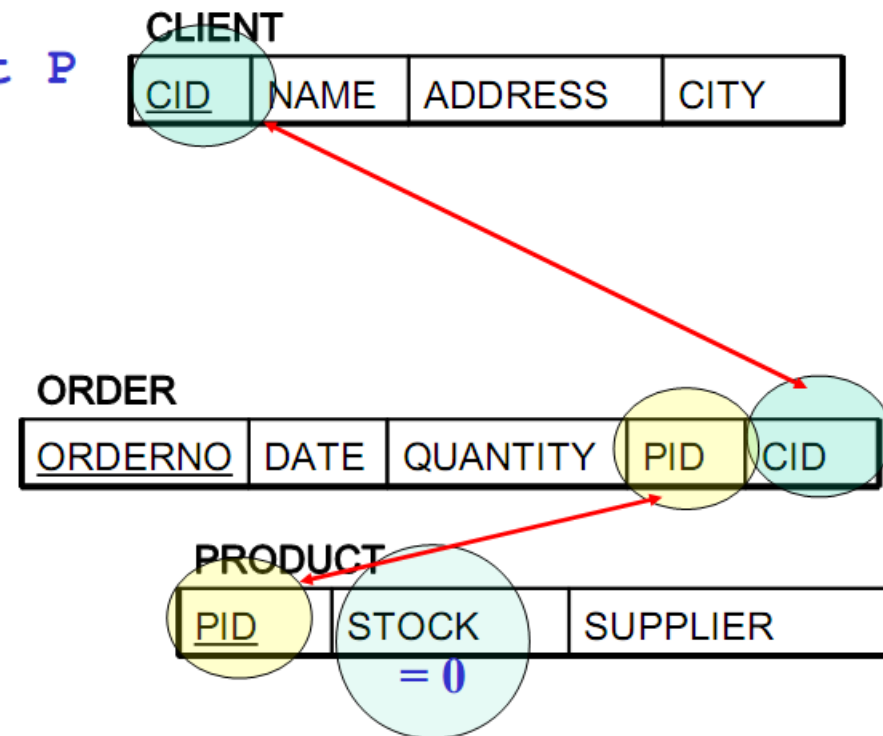


SQL(DML) - Query Example 8

- Query:8

- Βρείτε τα ονόματα και τις διευθύνσεις των πελατών οι οποίοι δίνουν παραγγελίες για προϊόντα για τα οποία δεν υπάρχει stock.

```
SELECT name, address  
FROM client C, order O, product P  
WHERE C.cid = O.cid  
      AND O.pid = P.pid  
      AND P.stock=0  
;
```





SQL(DML) – Aggregate functions

- Παραδείγματα συναρτήσεων συνάθροισης :

Συνάρτηση συνάθροισης	Περιγραφή	Περιορισμοί
<code>AVG(<i>expr</i>)</code>	Δίνει το μέσο όρο των τιμών μιας στήλης	Εφαρμόζεται μόνο σε γνωρίσματα αριθμητικών τύπων
<code>COUNT(<i>expr</i>), COUNT(*)</code>	Μετράει τις τιμές μιας στήλης (αν καθορίζετε το όνομα της στήλης είναι <i>expr</i>) ή όλων των γραμμών ενός πίνακα (αν καθορίζετε *). Η <code>COUNT(<i>expr</i>)</code> αγνοεί τις null τιμές, αλλά η <code>COUNT(*)</code> τις μετράει.	Εφαρμόζεται σε γνωρίσματα οποιουδήποτε τύπου



SQL(DML) – Aggregate functions

- Παραδείγματα συναρτήσεων συνάθροισης :

Συνάρτηση συνάθροισης	Περιγραφή	Περιορισμοί
MAX(<i>expr</i>)	Δίνει τη μεγαλύτερη τιμή σε μία στήλη (και την τελευταία τιμή για αλφαριθμητικούς τύπους). Αγνοεί τις null τιμές.	Εφαρμόζεται σε γνωρίσματα αριθμητικών ή αλφαριθμητικών τύπων
MIN(<i>expr</i>)	Δίνει τη μικρότερη τιμή σε μία στήλη (και την πρώτη τιμή για αλφαριθμητικούς τύπους). Αγνοεί τις null τιμές.	Εφαρμόζεται σε γνωρίσματα αριθμητικών ή αλφαριθμητικών τύπων
SUM(<i>expr</i>)	Δίνει το άθροισμα των τιμών σε μία στήλη	Εφαρμόζεται σε γνωρίσματα αριθμητικών τύπων



SQL(DML) – Aggregate functions

- Παραδείγματα:
 - Βρείτε το συνολικό αριθμό μαθητών που είναι κάτω από 18.
 - `SELECT count(*) FROM Student WHERE age < 18`
 - Βρείτε τον μέγιστο μισθό, τον ελάχιστο μισθό και ο μέσο μισθό από όλους τους υπαλλήλους.
 - `SELECT max (salary), min (salary), avg (salary) FROM employee`
 - Βρείτε το συνολικό αριθμό των πελατών.
 - `SELECT count (*) FROM customers`



SQL(DML) -Ομαδοποίηση πλειάδων (Group by)

- Ομαδοποίηση πλειάδων :

- Παρέχεται η δυνατότητα ομαδοποίησης των πλειάδων που αποτελούν την απάντηση σε μια ερώτηση σύμφωνα με τις κοινές τιμές κάποιων γνωρισμάτων (group by).
- Μπορούν επίσης να εφαρμοστούν συναρτήσεις συνάθροισης στις ομαδοποιημένες πλειάδες.
- Για παράδειγμα η ερώτηση :

- `SELECT deptno, sum(salary) FROM employee group by deptno`

- θα επιστρέψει τα διακεκριμένα deptno μαζί με το άθροισμα των μισθών του αντίστοιχου τμήματος

EMPLOYEE

SSN	NAME	MID	FNAME	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DEPTNO
-----	------	-----	-------	-------	---------	-----	--------	----------	--------



SQL(DML) -Ομαδοποίηση πλειάδων (Group by)

- Result

deptno	Sum
000	266643.00
100	155262.50
110	130442.81
115	13480000.00
121	110000.00
123	390500.00



SQL(DML) -Ομαδοποίηση πλειάδων (Group by)

- Ομαδοποίηση πλειάδων παράδειγμα :
 - Έστω ότι έχουμε το παρακάτω πίνακα:
 - Βρείτε το άθροισμα των αγορών για κάθε πελάτη
 - `SELECT customer, sum(OrderPrice)`
`FROM Orders group by Customer`

Orders

O_Id	OrderDate	OrderPrice	Customer
1	2008/11/12	1000	Hansen
2	2008/10/23	1600	Nilsen
3	2008/09/02	700	Hansen
4	2008/09/03	300	Hansen
5	2008/08/30	2000	Jensen
6	2008/10/04	100	Nilsen

Results:

Customer	SUM(OrderPrice)
Hansen	2000
Nilsen	1700
Jensen	2000



SQL(DML) -Ομαδοποίηση πλειάδων (Group by)

- Εάν στο προηγούμενο παράδειγμα δεν είχαμε βάλει το group by, δηλαδή είχαμε την επερώτηση
 - **SELECT** *customer*, *sum(OrderPrice)* **FROM** *Orders*;
- Τα αποτελέσματα που θα παίρναμε θα ήταν :

Customer	SUM(OrderPrice)
Hansen	5700
Nilsen	5700
Hansen	5700
Hansen	5700
Jensen	5700
Nilsen	5700



SQL (DML)-Ομαδοποίηση πλειάδων και περιορισμοί

- Οι περιορισμοί για κάθε ομάδα του αποτελέσματος, επιβάλλονται από το τμήμα HAVING της εντολής SELECT. Ο υπολογισμός αυτής της συνθήκης γίνεται μετά την ομαδοποίηση.

– Δομή σύνταξης :

- **SELECT** <λίστα πεδίων>
FROM <λίστα πινάκων>
[**WHERE** <σύνθετη συνθήκη>]
[**GROUP BY** <λίστα πεδίων>]
[**HAVING** <σύνθετη συνθήκη>]



SQL (DML)-Ομαδοποίηση πλειάδων και περιορισμοί

- Ομαδοποίηση πλειάδων και περιορισμοί παράδειγμα :
 - Έστω ότι έχουμε το παρακάτω πίνακα:
 - Βρείτε αν κάποιος πελάτης έχει άθροισμα αγορών κάτω από 2.000.
 - `SELECT Customer, sum(OrderPrice) FROM Orders group by Customer having sum(OrderPrice) < 2000;`

Orders

O_Id	OrderDate	OrderPrice	Customer
1	2008/11/12	1000	Hansen
2	2008/10/23	1600	Nilsen
3	2008/09/02	700	Hansen
4	2008/09/03	300	Hansen
5	2008/08/30	2000	Jensen
6	2008/10/04	100	Nilsen

Results:

Customer	SUM(OrderPrice)
Nilsen	1700



SQL (DML)-Ομαδοποίηση πλειάδων και περιορισμοί

- Ομαδοποίηση πλειάδων και περιορισμοί παράδειγμα :

- Έστω ότι έχουμε το παρακάτω πίνακα:

Για κάθε τμήμα επέστρεψε τον αριθμό του τμήματος, τον αριθμό των υπαλλήλων του τμήματος και τον μέσο όρο των μισθών του κάθε τμήματος, για τα τμήματα που έχουν πάνω από 10 υπαλλήλους.

- `SELECT dno, count(*), avg(salary)`
`FROM employee`
`group by dno`
`having count(*) > 10;`

EMPLOYEE

<u>SSN</u>	NAME	MID	FNAME	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DEPTNO
------------	------	-----	-------	-------	---------	-----	--------	----------	--------