



HY360 - ΑΡΧΕΙΑ ΚΑΙ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

ΦΡΟΝΤΙΣΤΗΡΙΟ ΣΤΗ ΓΛΩΣΣΑ SQL

OUTLINE

- ▶ ALL
- ▶ ANY
- ▶ (NOT) EXISTS
- ▶ BETWEEN
- ▶ GROUP BY
- ▶ HAVING

SQL (DML) - SELECT

EMPLOYEE

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

DEPARTMENT

<u>DEPTNO</u>	DEPTNAME	MGRSSN	MGRSTARDATE
---------------	----------	--------	-------------

DEPT_LOCATIONS

<u>DEPTNO</u>	<u>DEPTLOCATION</u>
---------------	---------------------

WORKS_ON

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

PROJECT

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

DEPENDENT

<u>ESSN</u>	<u>DEPENDNAME</u>	BDATE	SEX	RELATIONSHIP
-------------	-------------------	-------	-----	--------------

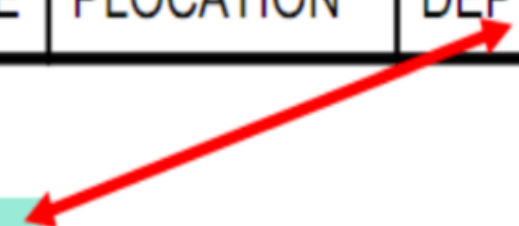
- Query:1

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

PROJECT

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

{2, 7}



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

- Query:1

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

- 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’

PROJECT

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

{2, 7}

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

- Query:1
 - Βρείτε τα ονόματα των έργων που δεν αφορούν το τμήμα “research” ή το τμήμα “design”.

- 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');

- Query:2

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

ή

WORKS_ON

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

EMPLOYEE

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

- Query:2

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

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
------------	------	-----	-------	-------	---------	-----	--------	----------	--------

- 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
------------	------	-----	-------	-------	---------	-----	--------	----------	--------

- Query:2

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

WORKS_ON

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

EMPLOYEE

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

- 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
------------	------	-----	-------	-------	---------	-----	--------	----------	--------

- Query:3

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

EMPLOYEE

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

- 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
------------	------	-----	-------	-------	---------	-----	--------	----------	--------

- 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 *E.ssn* **NOT IN** (**SELECT** *W.ssn*

FROM *works_on* W, *project* P

WHERE *W.ssn* = *E.ssn* **AND** *P.pno* = *W.pno*

AND *P.DeptNo* = 3);

- 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
------------	-------	-----------	--------

;

- 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
------------	------	-----	-------	-------	---------	-----	--------	----------	--------

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
------------	-------	-----------	--------

- 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
------------	-------	-----------	--------

- Query:5

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

EMPLOYEE

<u>SSN</u>	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

<u>ESSN</u>	<u>DEPENDNAME</u>	BDATE	SEX	RELATIONSHIP
-------------	-------------------	-------	-----	--------------

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

PRODUCT

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

ORDER

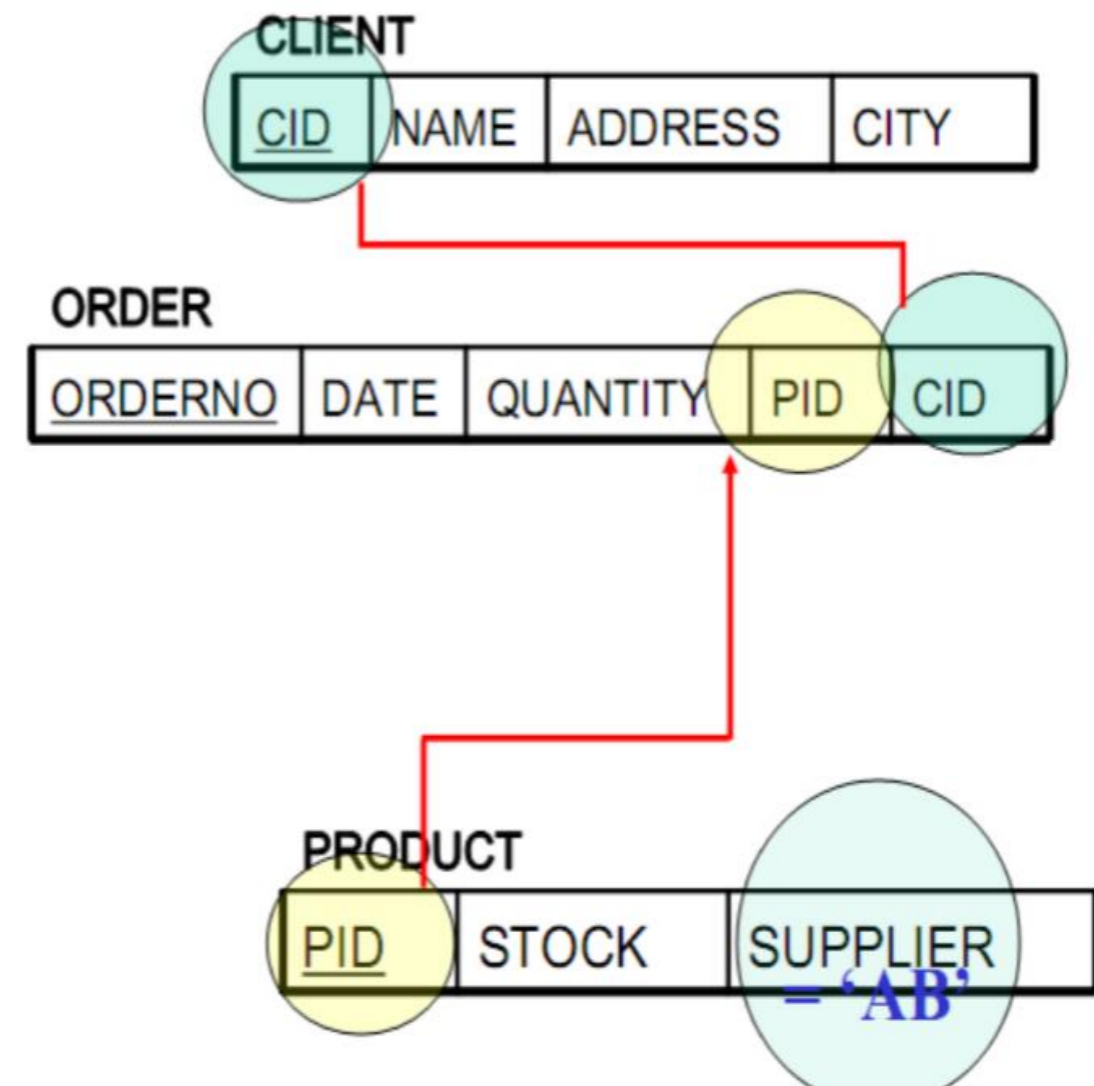
<u>ORDERNO</u>	DATE	QUANTITY	PID	CID
----------------	------	----------	-----	-----

CLIENT

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

- Query:6

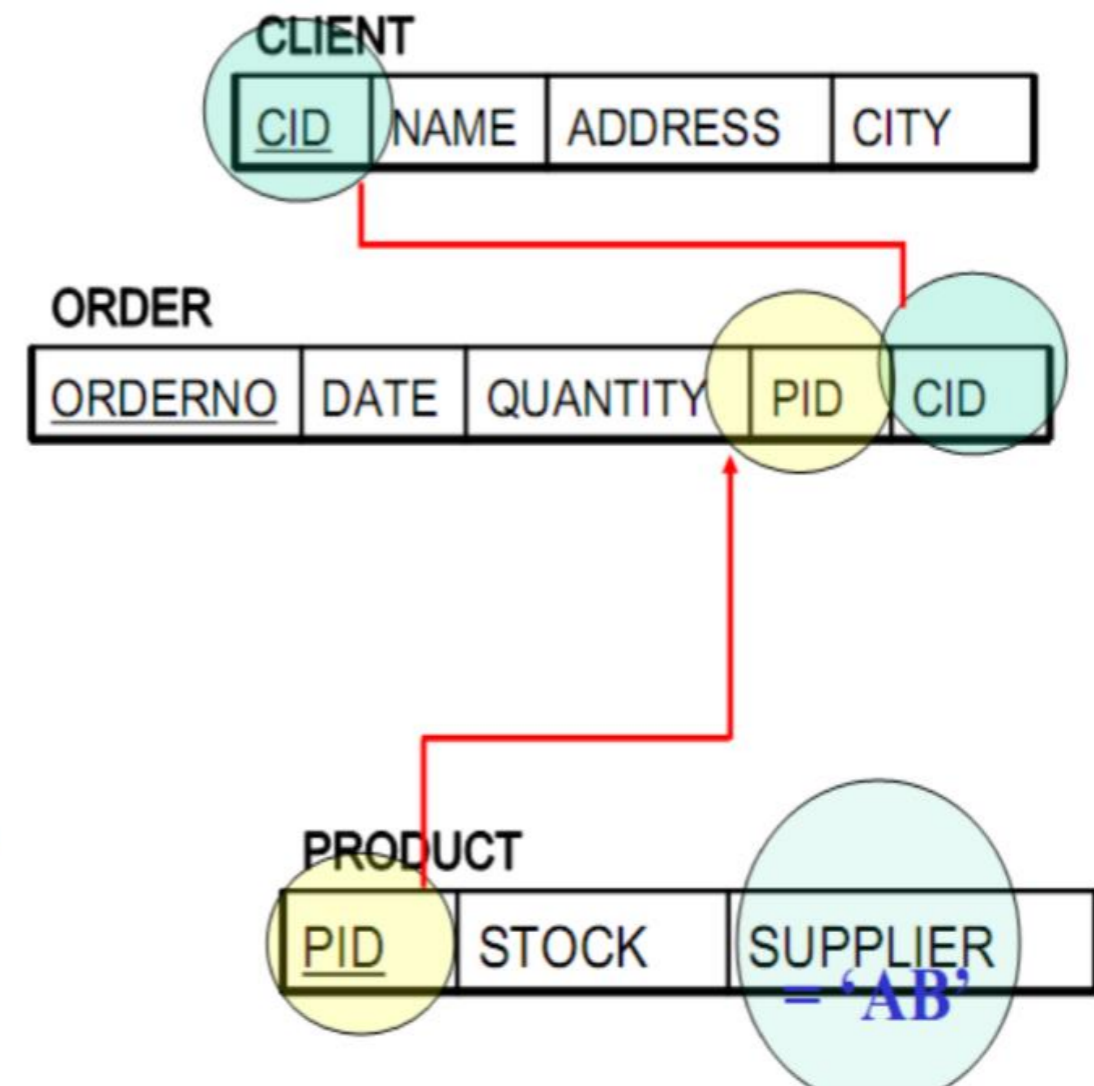
- Βρείτε τις πόλεις στις οποίες μένουν πελάτες οι οποίοι δεν δίνουν παραγγελίες για προϊόντα που προμηθεύει ο «AB».



- Query:6

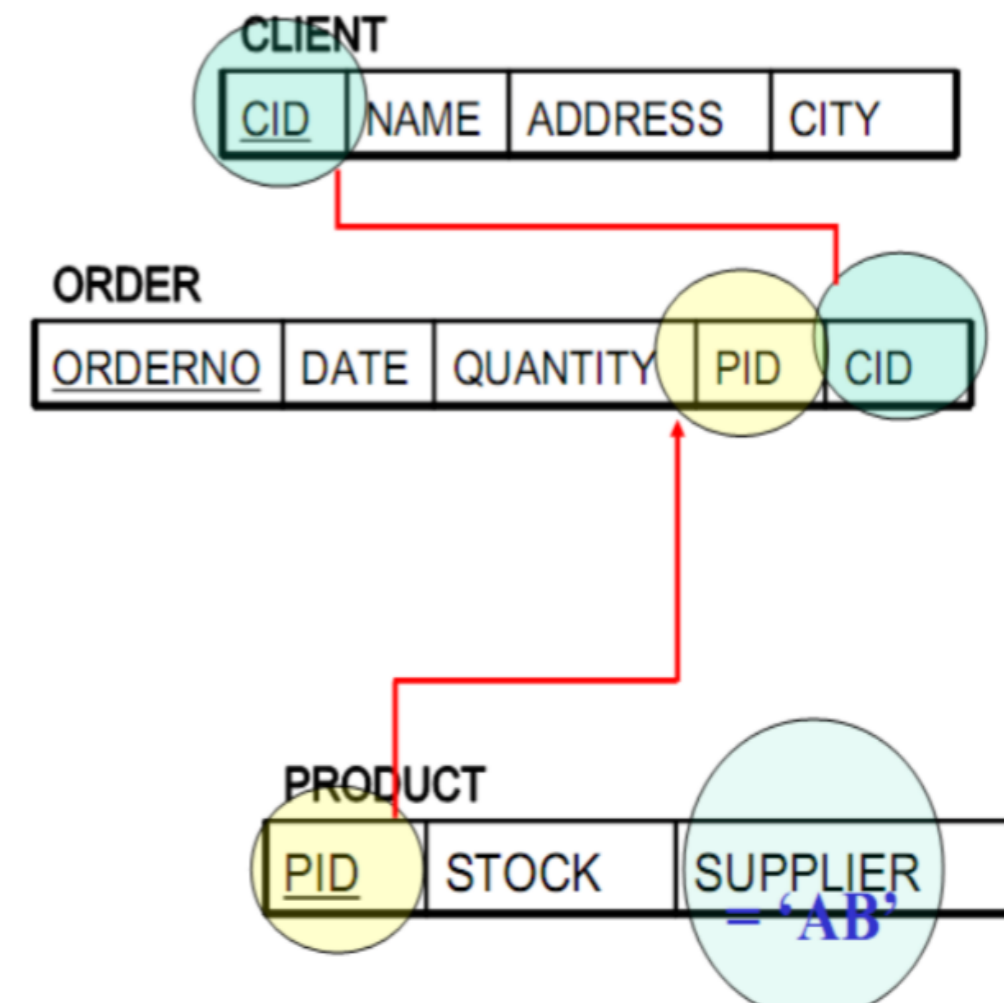
- Βρείτε τις πόλεις στις οποίες μένουν πελάτες οι οποίοι δεν δίνουν παραγγελίες για προϊόντα που προμηθεύει ο «AB».

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



- Query:6

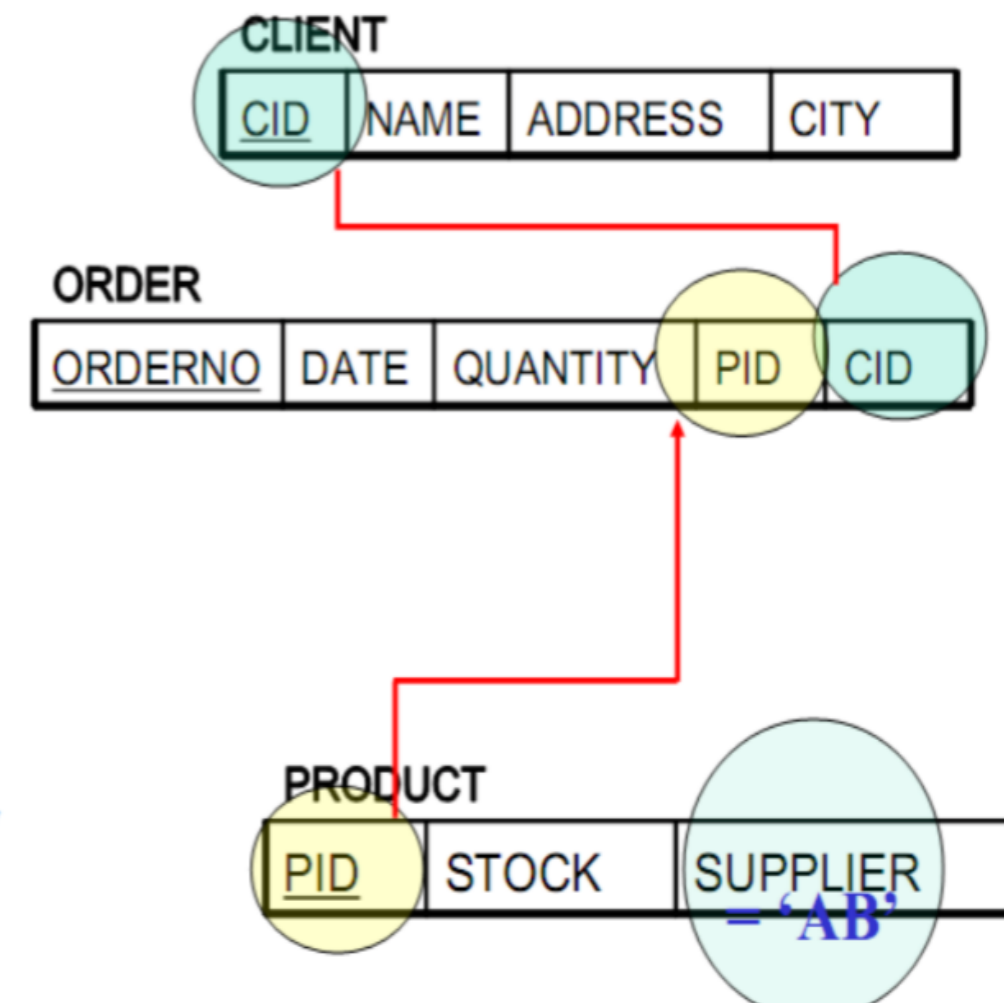
- Βρείτε τις πόλεις στις οποίες μένουν πελάτες οι οποίοι δεν δίνουν παραγγελίες για προϊόντα που προμηθεύει ο «AB».



- Query:6

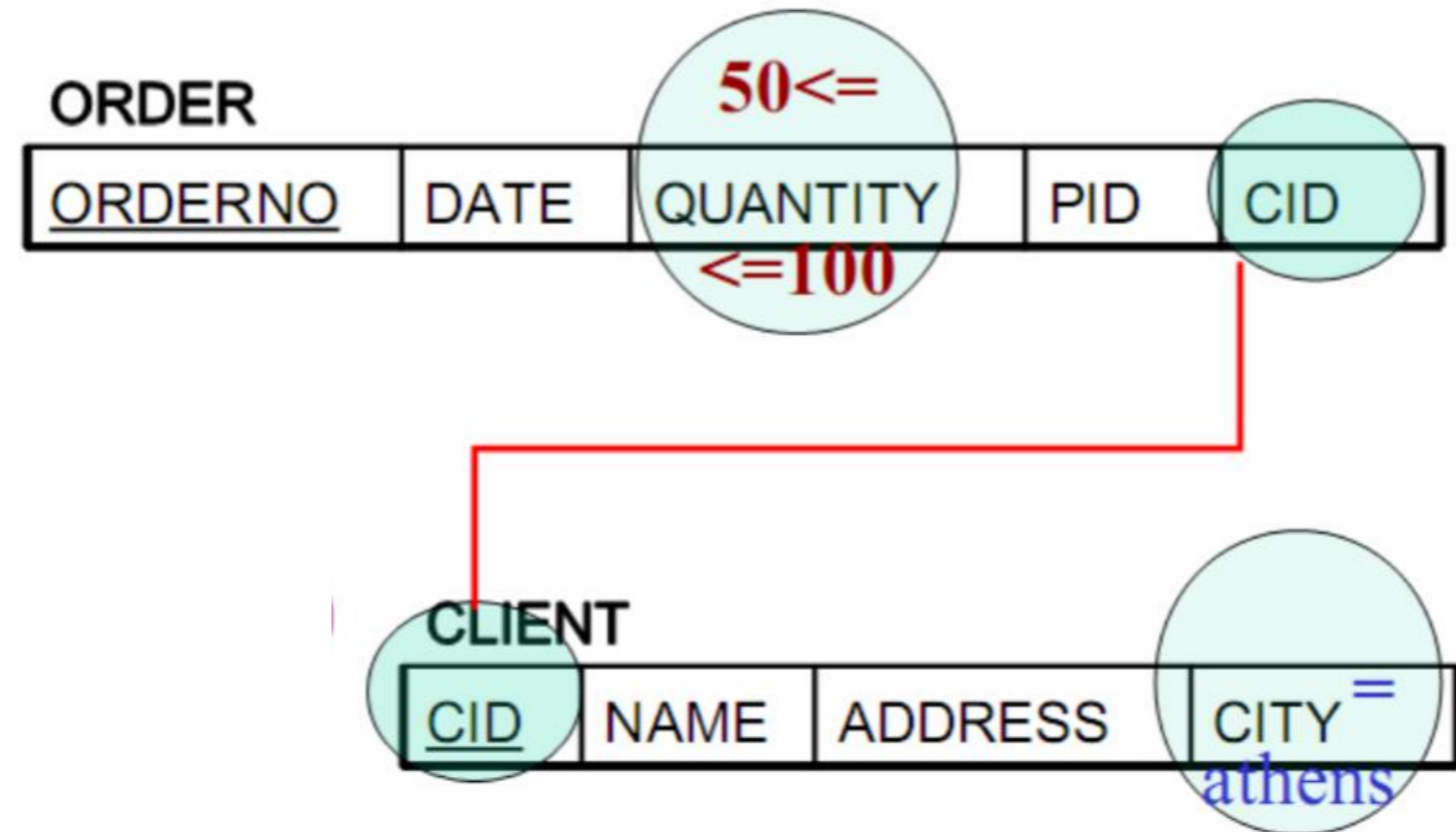
- Βρείτε τις πόλεις στις οποίες μένουν πελάτες οι οποίοι δεν δίνουν παραγγελίες για προϊόντα που προμηθεύει ο «AB».

```
– 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'
);
```



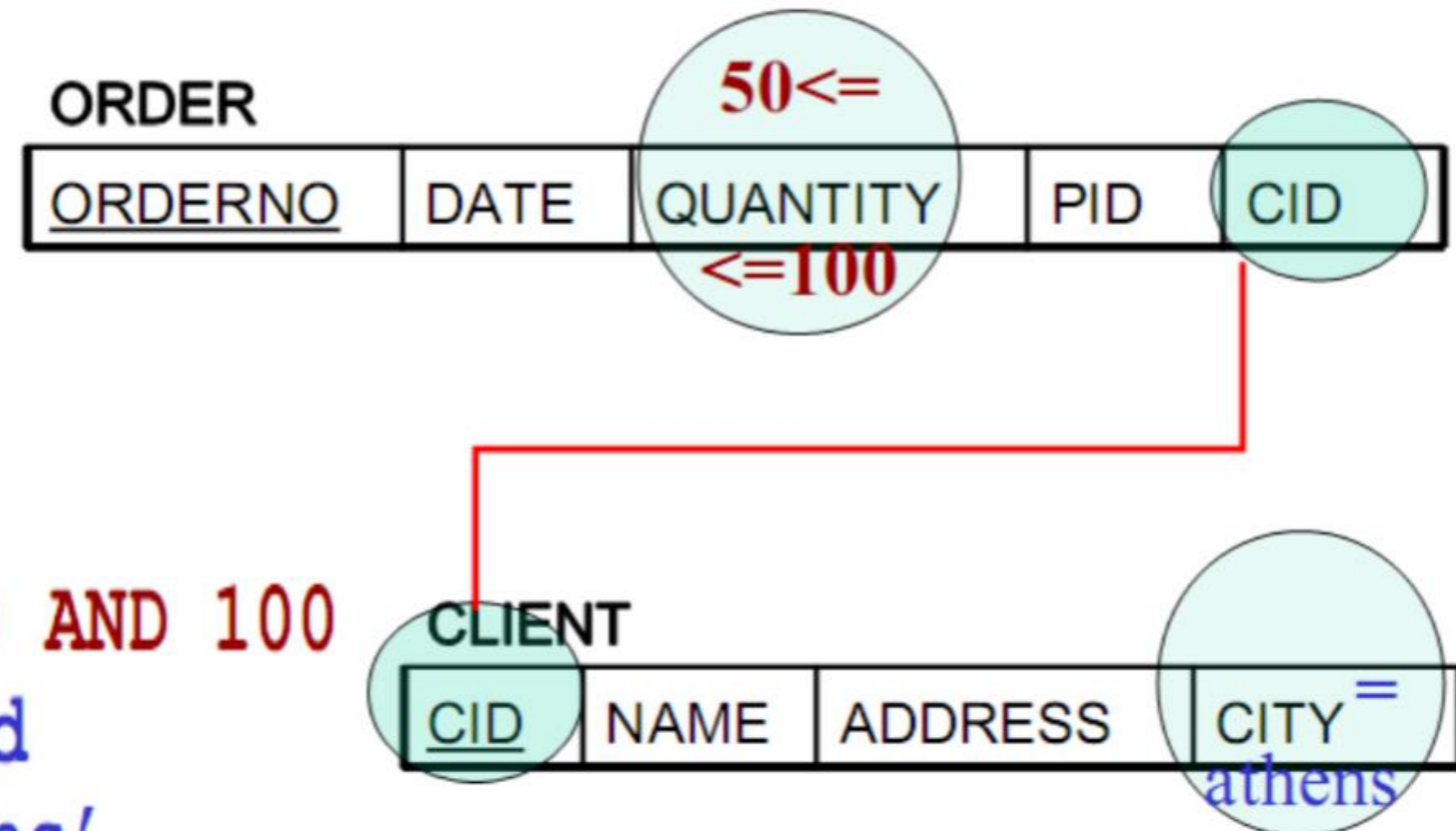
- Query:7

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



- Query:7

- Βρείτε τους αριθμούς των παραγγελιών για τα προϊόντα που παραγγέλλονται 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'`

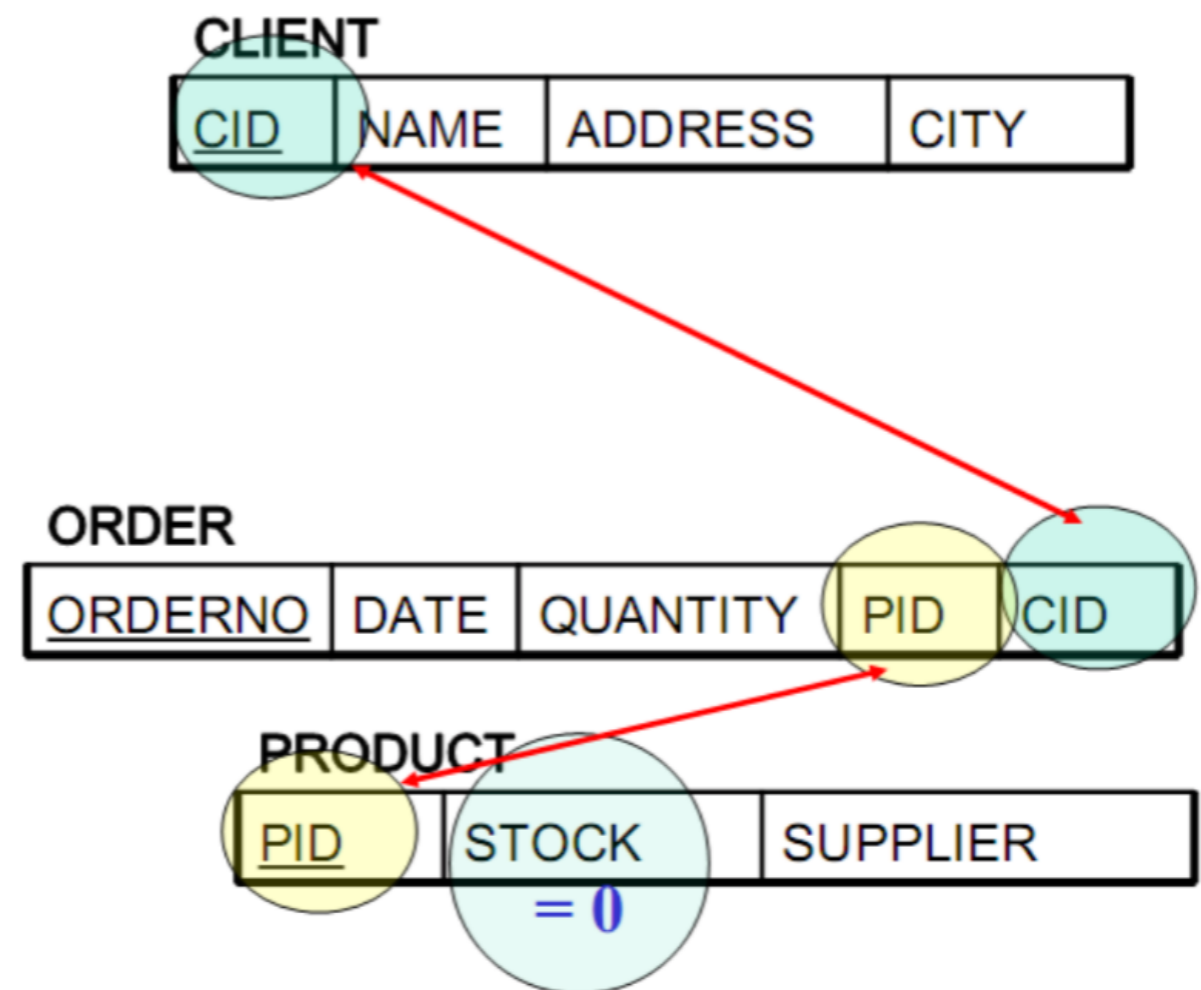
- `quantity BETWEEN 50 AND 100`

ή

- `quantity >= 50 AND quantity <= 100`

- Query:8

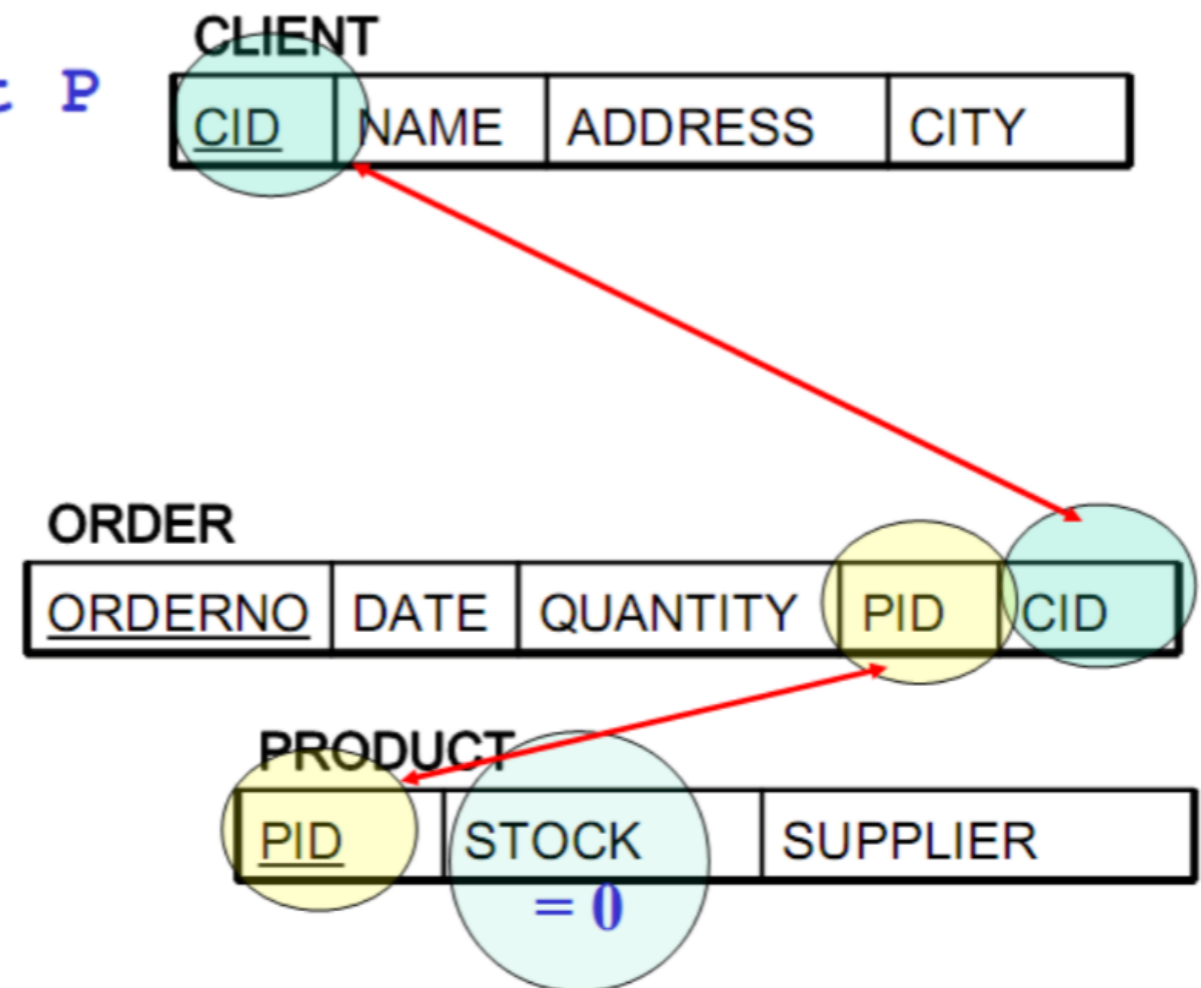
- Βρείτε τα ονόματα και τις διευθύνσεις των πελατών οι οποίοι δίνουν παραγγελίες για προϊόντα για τα οποία δεν υπάρχει stock.



- 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
;
```



- Παραδείγματα συναρτήσεων συνάθροισης :

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

- 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

- 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`

- 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

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

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

- 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`

Results:

Customer	SUM(OrderPrice)
Hansen	2000
Nilsen	1700
Jensen	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

- 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 :

- 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

- 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>]

- 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

- 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
------------	------	-----	-------	-------	---------	-----	--------	----------	--------

