Γλώσσα SQL
ακαδ. έτος 2021-22

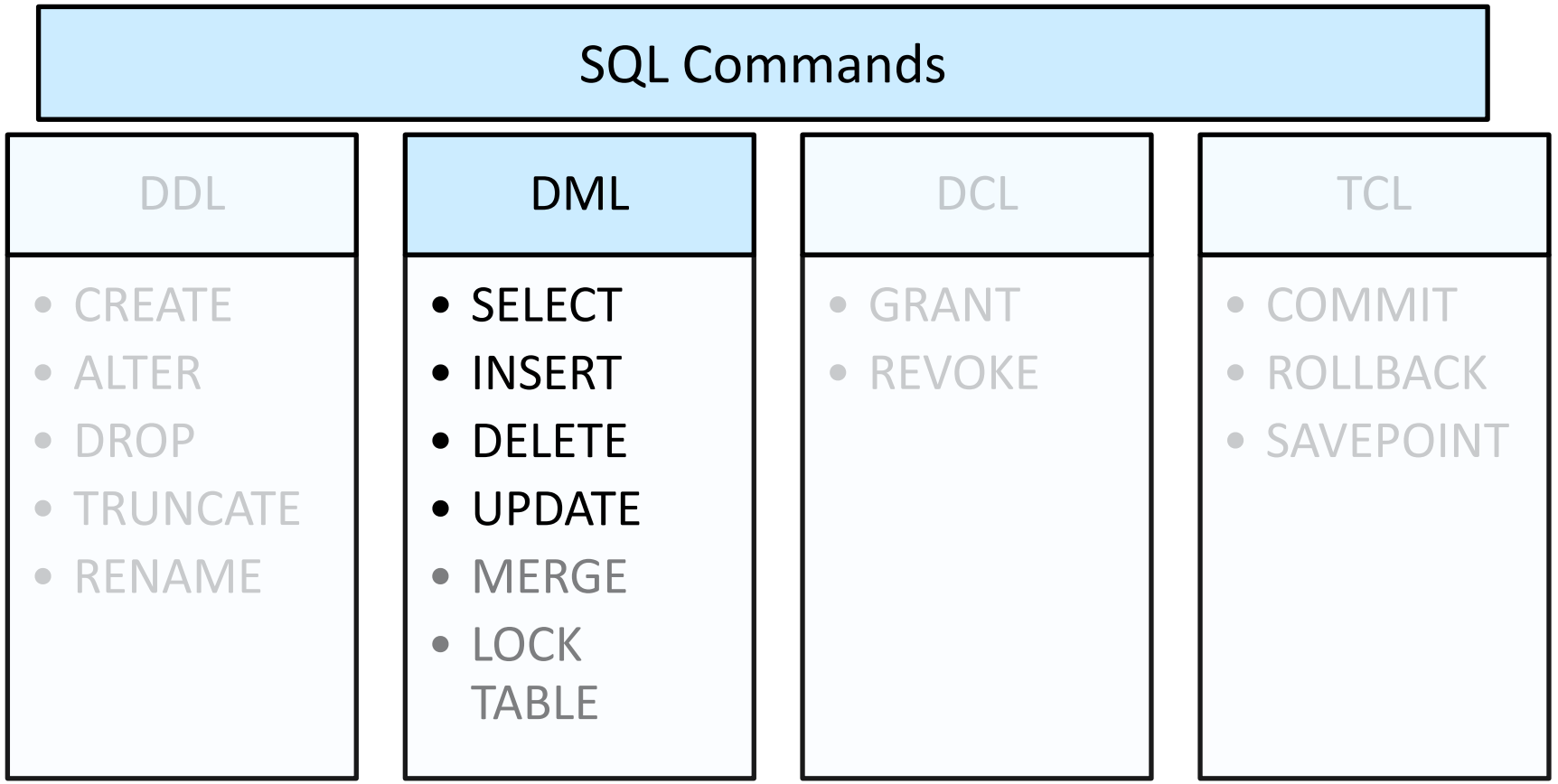
Διδάσκων:
καθ. Γιάννης Θεοδωρίδης

+Διδάσκουσα:
Δρ. Μαυροπόδη Ρόζα

Εργαστηριακοί βοηθοί:
Γιάννης Κοντούλης

Lectures on Databases: section I “Intro”, v. 2022.03
by
Data Science Lab. @ Univ. Piraeus (www.datastories.org)

SQL – DML data manipulation language



Εφαρμόζεται στα δεδομένων (data).
Διαχειρίζεται τις γραμμές μιας σχέσης, ενός πίνακα, και όχι π.χ τον ορισμό του πίνακα ή τα δικαιώματα πρόσβασης σε αυτόν….

Select

Το **select** παραθέτει τα χαρακτηριστικά (attributes) τα οποία είναι επιθυμητά ως αποτέλεσμα ενός ερωτήματος (query).

Παράδειγμα: find the names of all instructors:

select name
from instructor

Το αποτέλεσμα της ανωτέρω εντολής SQL είναι μια **σχέση.**
(επιστρέφει πάντα έναν πίνακα ο οποίος έχει μια ή περισσότερες στήλες.)

Προσοχή: Η SQL ως γλώσσα δεν πραγματοποιεί διάκριση μεταξύ πεζών/ κεφαλαίων. Οι αναφορές σε ονόματα σχέσεων (πινάκων), χαρακτηριστικών (στήλες) καθώς και τιμών καλό είναι να ακολουθούν πιστά τον ορισμό της σχέσης.

Select ... from .... where

Select

Εμφανίζει τα χαρακτηριστικά που είναι επιθυμητά στο αποτέλεσμα ενός ερωτήματος

From

Εμφανίζει τις σχέσεις που θα πρέπει να προσπελαστούν προκειμένου να εκτελεστεί το ερώτημα

Ορίζει ένα καρτεσιανό γινόμενο για τις σχέσεις που αναφέρονται στον όρο

Where

Κατηγόρημα που περιλαμβάνει τα χαρακτηριστικά των σχέσεων που εμφανίζονται στον όρο from

Περιορίζει τους συνδυασμούς που δημιουργούνται από το καρτεσιανό γινόμενο σε αυτούς που έχουν νόημα για την επιθυμητή απάντηση

SQL-DML select 1-1

Core SQL

SELECT select-list
[FROM table-expression **]**
[WHERE …]

Εφαρμογή

SELECT 'Hello world';
SELECT dept_name, budget **FROM** department;
SELECT dept_name **FROM** department **WHERE** budget>50000;

Postgres

```
Command:     SELECT
Description: retrieve rows from a table or view
Syntax:
[ WITH [ RECURSIVE ] with_query [, ...] ]
SELECT [ ALL | DISTINCT [ ON ( expression [, ...] ) ] ]
    [ * | expression [ [ AS ] output_name ] [, ...] ]
    [ FROM from_item [, ...] ]
    [ WHERE condition ]
    [ ORDER BY expression [ ASC | DESC | USING operator ] [ NULLS { FIRST | LAST } ] [, ...] ]
    [ GROUP BY grouping_element [, ...] ]
    [ HAVING condition ]
    [ LIMIT { count | ALL } ]
....
where from_item can be one of:

    [ ONLY ] table_name [ * ] [ [ AS ] alias [ ( column_alias [, ...] ) ] ]
........
    from_item [ NATURAL ] join_type from_item [ ON join_condition | USING ( join_column [, ...] ) ]
....
```

SQL-DML select 1-2

postgres (π.χ)

```
/* select ….*/
select * from department ;
select distinct dept_name from instructor;
select distinct * from instructor;
select dept_name, budget  from department ;
/* column alias*/
select dept_name, budget as proipologismos  from department ;
select budget*2 as double_budget  from department ;
select budget, budget*2 as double_budget  from department ;
/*  functions*/
select max(budget) from department ;
```

SQL-DML select 1-3

postgres (π.χ)

```
/* from .... */
select name, dept_name  from instructor;
/* καρτεσιανό γινόμενο*/
select name, course_id  from instructor , teaches;

select name, course_id  from instructor , teaches
where instructor.ID = teaches.ID;
/* table.column_name  */
select instructor.name, teaches.course_id, teaches.year
from instructor , teaches where instructor.ID = teaches.ID;
/* table alias  */
select i.name, t.course_id, t.year
from instructor as i, teaches as t
where i.ID = t.ID;
```

SQL-DML select 1-4

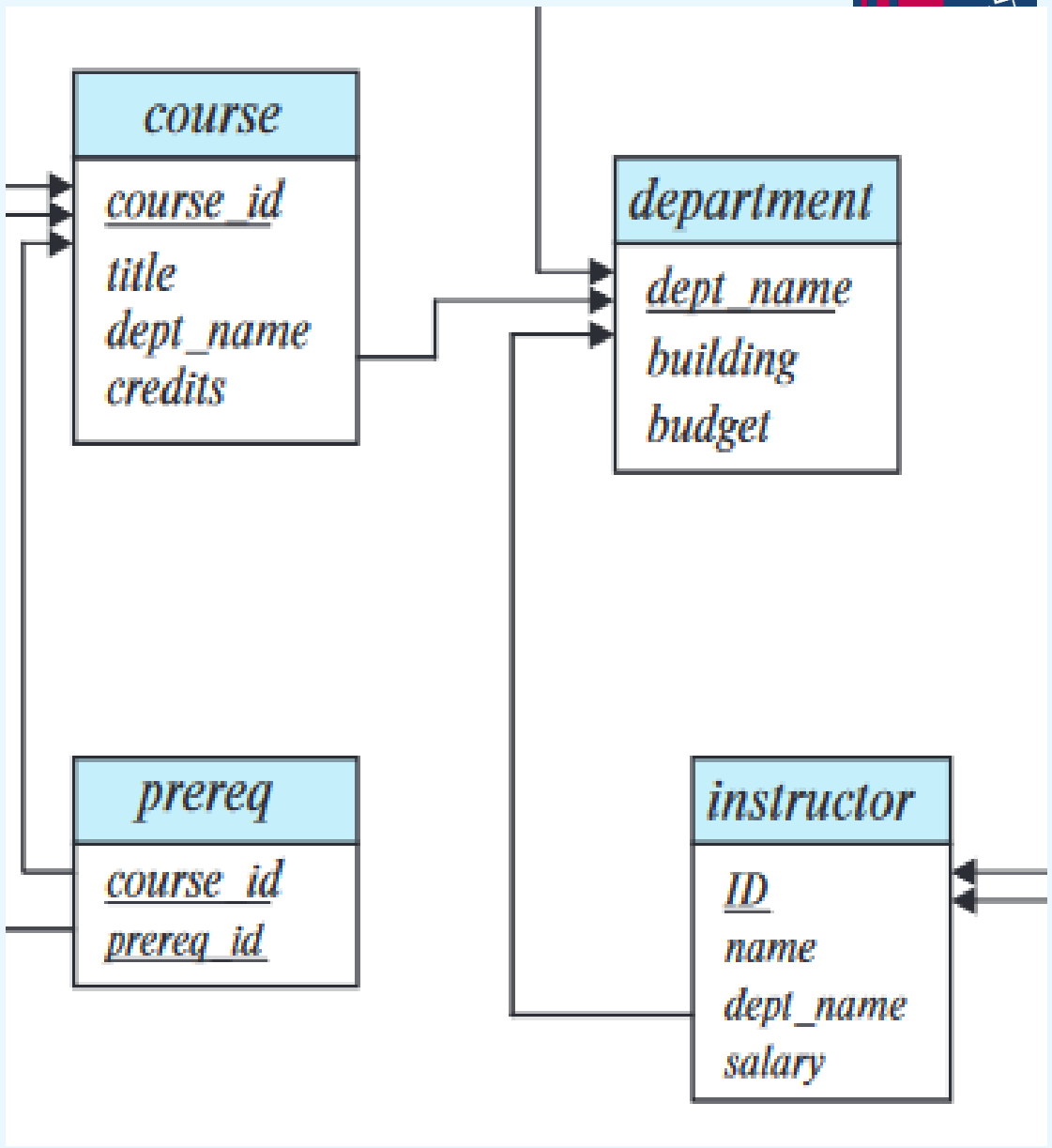
postgres (π.χ)

- **Τελεστές σύγκρισης**
  =, <, <=, >, >=,
  και <> για το διάφορο (not equal)
- **Λογικοί τελεστές**
  and, not, or

```
/* Βρείτε τα ονόματα και το μισθό όλων όσων διδάσκουν στο τμήμα Biology είτε όλων όσων έχουν μισθό μεγαλύτερο από 92000 */
SELECT name, salary
FROM instructor
WHERE dept_name = 'Biology' OR salary >92000;
/* Βρείτε τα ονόματα και το μισθό όλων όσων διδάσκουν σε οποιοδήποτε τμήμα εκτός της Biology */
SELECT name, salary
FROM instructor
WHERE dept_name <> 'Biology';
```

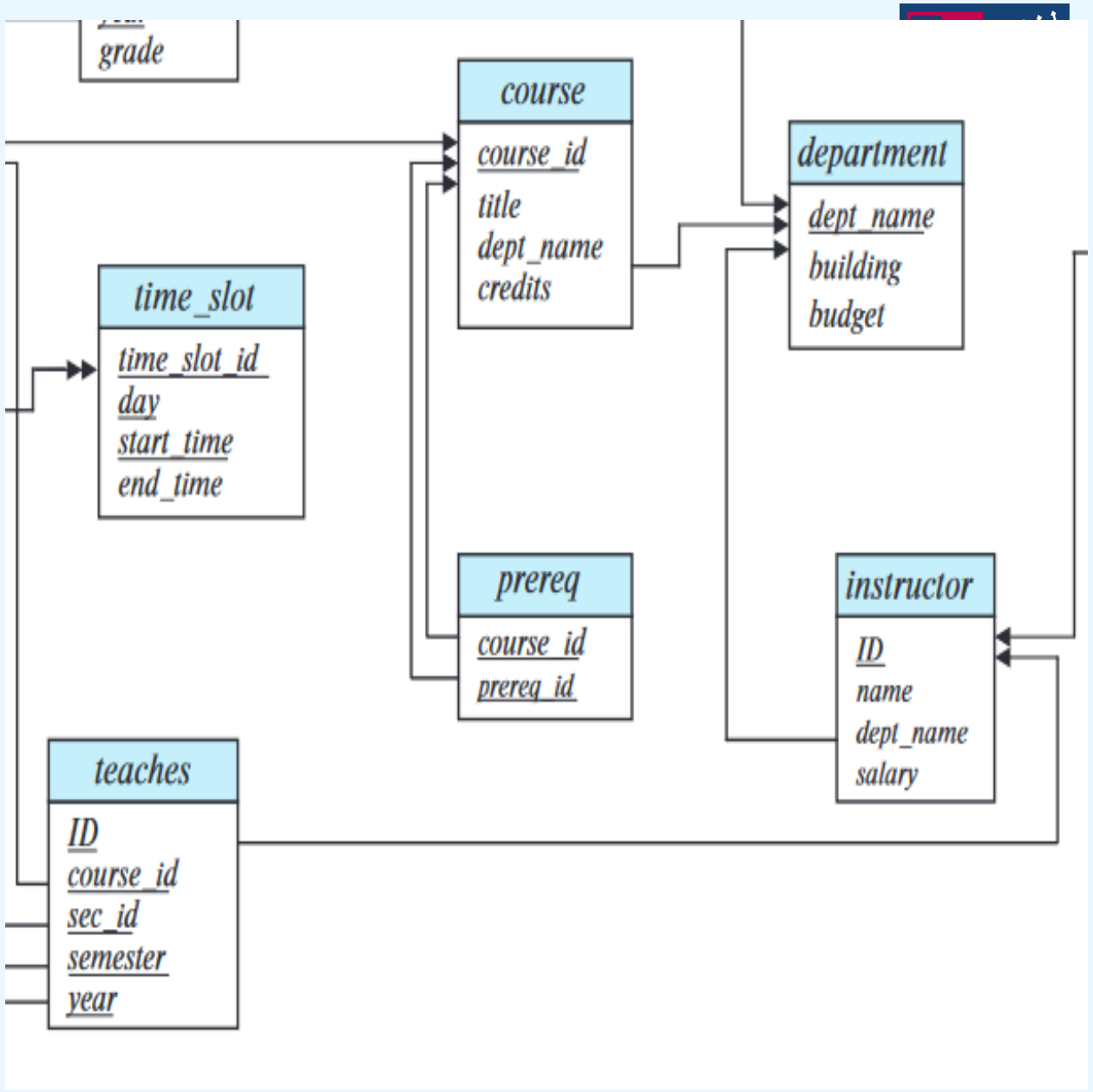
Βρες….

- ✓ Βρείτε τους τίτλους των μαθημάτων στο τμήμα Computer Science που έχουν 3 πιστωτικές μονάδες.
- ✓ Ανακαλέστε τα ονόματα των καθηγητών, μαζί με τα ονόματα των τμημάτων τους, καθώς και το όνομα του κτιρίου των τμημάτων τους.
- ✓ Βρείτε τον κωδικό καθηγητή και το όνομα του τμήματος των καθηγητών που σχετίζονται με ένα τμήμα με προϋπολογισμό μεγαλύτερο από 95000 €.



Βρες….

- ✓ Βρείτε τους καθηγητές του τμήματος Πληροφορικής που έχουν διδάξει κάποια μαθήματα.
- ✓ Για τους καθηγητές στο Πανεπιστήμιο που έχουν διδάξει κάποιο μάθημα, βρείτε τα ονόματά τους & το ID του μαθήματος των μαθημάτων που δίδαξαν.
- ✓ Αναφέρατε τα ονόματα των καθηγητών μαζί με τους τίτλους των μαθημάτων που διδάσκουν.



SQL-DML select 1-5

postgres (π.χ)

```
/* where ….*/
/* Βρείτε όλα τα μοναδικά ονόματα των καθηγητών οι οποίοι έχουν μισθό μεγαλύτερο από οποιονδήποτε καθηγητή του τμήματος  Comp. Sci.…. Εξηγήστε γιατί λειτουργεί το παρακάτω. */

select distinct T.name
from instructor as T, instructor as S
where T.salary > S.salary and S.dept_name = 'Comp. Sci.'
/*  ποιο είναι γρηγορότερο και γιατί;*/
select distinct T.name
from instructor as T
where T.salary > (select min(salary)
                  from instructor
                  where dept_name = 'Comp. Sci.') ;
```

SQL-DML select 1-5

postgres (π.χ)

/* where ….*/

select * **from** instructor **where** salary **in (**65000, 40000, 75000**) ;**

select * **from** instructor **where** ID **in (**select i_id from advisor **) ;**

select name **from** instructor
where salary **between** 90000 **and** 100000;
αντί του
select name **from** instructor
where salary <= 100000 **and** salary >= 90000

with teach_id **as (select** ID **from** teaches **where** teaches.year=2018**)**
select instructor.name **from** instructor, teach_id **where**
instructor.ID=teach_id.ID;

Πράξεις με συμβολοσειρές 1-1

- Η SQL παρέχει έναν τελεστή ταιριάσματος συμβολοσειρών **like** για συγκρίσεις με χρήση δύο ειδικών χαρακτήρων '**%**' και '**_**':
  - (%). Ο χαρακτήρας **%** ταιριάζει 0 ή οποιαδήποτε υπο-συμβολοσειρά.
  - (_). Ο χαρακτήρας **_** ταιριάζει ένα ή οποιοδήποτε χαρακτήρα

```
select distinct name
from instructor
where name like 'E%e__' ;
```

```
select name, dept_name
from instructor
where name like '_ _ n _ _ _ _ _';
```

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

Πράξεις με συμβολοσειρές 1-2

- Τα μοτίβα κάνουν διάκριση σε πεζά και κεφαλαία (case sensitive).
- π.χ.:
  - 'Intro%' θα επιστρέψει οποιαδήποτε τιμή αρχίζει με “Intro”.
  - '%Comp%' θα επιστρέψει οποιαδήποτε τιμή περιέχει “Comp”.
  - '_ _ _' θα επιστρέψει οποιαδήποτε τιμή με ακριβώς 3 χαρακτήρες.
  - '_ _ _ %' θα επιστρέψει οποιαδήποτε τιμή η οποία θα έχει τουλάχιστον 3 χαρακτήρες και μετά ακολουθεί οτιδήποτε.

SQL-DML order by 1-1

postgres (π.χ)

/* Βρείτε όλα τα μοναδικά ονόματα των καθηγητών ταξινομημένα κατά αύξουσα/φθίνουσα σειρά*/

select distinct name **from** instructor **order by** name **ASC;**
select distinct name **from** instructor **order by** name **DESC;**

select name, dept_name **from** instructor **order by** name **ASC,** dept_name **DESC;**

SQL-DML limit 1-1

postgres (π.χ)

```
/* Βρείτε τα 5 μοναδικά ονόματα των καθηγητών*/
select distinct name from instructor limit 5 ;
/* Βρείτε τα 5 πρώτα μοναδικά ονόματα των καθηγητών,
αλλά εμφανίστε 3ο,4ο, 5ο*/
select distinct name from instructor limit 5  offset 2;

/* 3 πρώτοι με το μικρότερο μισθό */
select name, dept_name , salary from instructor order by salary asc limit 3 ;
/* 3 πρώτοι με το μεγαλύτερο μισθό */
select name, dept_name , salary from instructor order by salary desc limit 3 ;
```

Βρες….

Θέλετε να φτιάξετε μια εφαρμογή τηλεφωνικού καταλόγου. Η εφαρμογή διαθέτει πεδίο αναζήτησης ονομάτων/ ή και τηλεφώνων. Όταν ο χρήστης συμπληρώσει το πεδίο αναζήτησης και επιλέξει το κουμπί τότε εκτελείται μια εντολή SQL. Έστω ότι η εφαρμογή χρησιμοποιεί την παρακάτω σχέση. Βρείτε και γράψτε την ανωτέρω δήλωση SQL.
Σημείωση: Ο χρήστης συμπληρώνει ένα πεδίο κάθε φορά το οποίο η εφαρμογή δε γνωρίζει εάν είναι τηλέφωνο ή όνομα.

name	dept_name	status	tel
Theodoridis Ioannis	Department of Informatics	Professor	+30 210 4142449
Sapounakis Aristeidis	Department of Informatics	Professor	+30 210 4142262
Douligeris Christos	Department of Informatics	Professor	+30 210 4142137
Prentza Andriana	Department of Digital Systems	Professor	+30 210 4142768
Kanatas Athanasios	Department of Digital Systems	Professor	+30 210 4142759
Konstantopoulos Charalampos	Department of Informatics	Professor	+30 210 4142124
Despotis Dimitrios	Department of Informatics	Professor	+30 210 4142315
Magklogiannis Ilias	Department of Digital Systems	Professor	+30 210 4142517
Metaxiotis Konstantinos	Department of Informatics	Professor	+30 210 4142578

Θεωρήστε, επιπλέον, ότι τα αποτελέσματα είναι πάνω από 30 γραμμές και ο χρήστης έχει διαθέσιμες σελίδες των 10 γραμμών.

SQL-DML group by 1-1

(π.χ)

/* ομαδοποιήσεις στα δεδομένα ανάλογα με κοινά τους χαρακτηριστικά σε μια στήλη*/

```
select dept_name, avg (salary) as avg_salary
from instructor
group by dept_name;
```

```
myunipi=# select dept_name, avg (salary) as avg_salary
myunipi-# from instructor
myunipi-# group by dept_name;
+------------+--------------------+
| dept_name  |     avg_salary     |
+------------+--------------------+
| Finance    | 85000.000000000000 |
| History    | 61000.000000000000 |
| Physics    | 91000.000000000000 |
| Music      | 40000.000000000000 |
| Comp. Sci. | 77333.333333333333 |
| Biology    | 72000.000000000000 |
| Elec. Eng. | 80000.000000000000 |
+------------+--------------------+
(7 rows)
```

SQL-DML group by 1-2

(π.χ)

```
/* προβολή του πλήθους των μελών ενός group*/
select dept_name, avg (salary) as avg_salary, count(*) as members_in_group
from instructor
group by dept_name;
```

```
myunipi=# select dept_name, avg (salary) as avg_salary, count(*) as members_in_group
myunipi-# from instructor
myunipi-# group by dept_name;
+------------+--------------------+------------------+
| dept_name  |     avg_salary     | members_in_group |
+------------+--------------------+------------------+
| Finance    | 85000.000000000000 |                2 |
| History    | 61000.000000000000 |                2 |
| Physics    | 91000.000000000000 |                2 |
| Music      | 40000.000000000000 |                1 |
| Comp. Sci. | 77333.333333333333 |                3 |
| Biology    | 72000.000000000000 |                1 |
| Elec. Eng. | 80000.000000000000 |                1 |
+------------+--------------------+------------------+
(7 rows)
```

SQL-DML group by 1-3

(π.χ)

```
/* group by με where */
select dept_name, avg (salary) as avg_salary, count(*) as members_in_group
from instructor
where salary>60000
group by dept_name;
```

```
myunipi=# select dept_name, avg (salary) as avg_salary, count(*) as members_in_group
myunipi-# from instructor
myunipi-# where salary>60000
myunipi-# group by dept_name;
+------------+--------------------+------------------+
| dept_name  |     avg_salary     | members_in_group |
+------------+--------------------+------------------+
| Finance    | 85000.000000000000 |                2 |
| History    | 62000.000000000000 |                1 |
| Physics    | 91000.000000000000 |                2 |
| Comp. Sci. | 77333.333333333333 |                3 |
| Biology    | 72000.000000000000 |                1 |
| Elec. Eng. | 80000.000000000000 |                1 |
+------------+--------------------+------------------+
(6 rows)
```

SQL-DML group by 1-4

(π.χ)

/* εμφάνιση των μοναδικών δεδομένων της στήλης*/

select dept_name, **count**(*) **as** members_in_group
from instructor **group by** dept_name**;**

Ερώτηση: Πως μπορώ να εμφανίσω, διαφορετικά, τα μοναδικά ονόματα

```
myunipi=# select dept_name, count(*) as members_in_group
myunipi-# from instructor group by dept_name;
+------------+------------------+
| dept_name  | members_in_group |
+------------+------------------+
| Finance    |                2 |
| History    |                2 |
| Physics    |                2 |
| Music      |                1 |
| Comp. Sci. |                3 |
| Biology    |                1 |
| Elec. Eng. |                1 |
+------------+------------------+
(7 rows)
```

SQL-DML group by 1-5

(π.χ)

/* group by multiple columns*/

select dept_name, budget, **count**(*) **from** department **group by** dept_name, budget;

```
myunipi=# select budget, count(*) from department group by budget;
+-----------+-------+
|  budget   | count |
+-----------+-------+
|  80000.00 |     2 |
|  85000.00 |     1 |
| 100000.00 |     1 |
|  90000.00 |     1 |
|  70000.00 |     1 |
|  50000.00 |     1 |
+-----------+-------+
(6 rows)

myunipi=# select dept_name,budget, count(*) from department group by dept_name, budget;
+------------+-----------+-------+
| dept_name  |  budget   | count |
+------------+-----------+-------+
| History    |  50000.00 |     1 |
| Physics    |  70000.00 |     1 |
| Music      |  80000.00 |     1 |
| Biology    |  90000.00 |     1 |
| Finance    |  80000.00 |     1 |
| Elec. Eng. |  85000.00 |     1 |
| Comp. Sci. | 100000.00 |     1 |
+------------+-----------+-------+
(7 rows)
```

SQL-DML group by 1-6

(π.χ)

```
/* filter. conditions on aggregated functions */
select dept_name, budget,
count(*) filter (where building='Taylor'),
sum(budget) filter (where budget>60000)
from department group by dept_name, budget;
```

```
myunipi=# select dept_name,budget,
myunipi-# count(*)filter(where building='Taylor'),
myunipi-# sum(budget) filter (where budget>60000)
myunipi-# from department group by dept_name, budget;
+------------+-----------+-------+-----------+
| dept_name  |  budget   | count |    sum    |
+------------+-----------+-------+-----------+
| History    |  50000.00 |     0 |           |
| Physics    |  70000.00 |     0 |  70000.00 |
| Music      |  80000.00 |     0 |  80000.00 |
| Biology    |  90000.00 |     0 |  90000.00 |
| Finance    |  80000.00 |     0 |  80000.00 |
| Elec. Eng. |  85000.00 |     1 |  85000.00 |
| Comp. Sci. | 100000.00 |     1 | 100000.00 |
+------------+-----------+-------+-----------+
(7 rows)
```

SQL-DML having 1-1

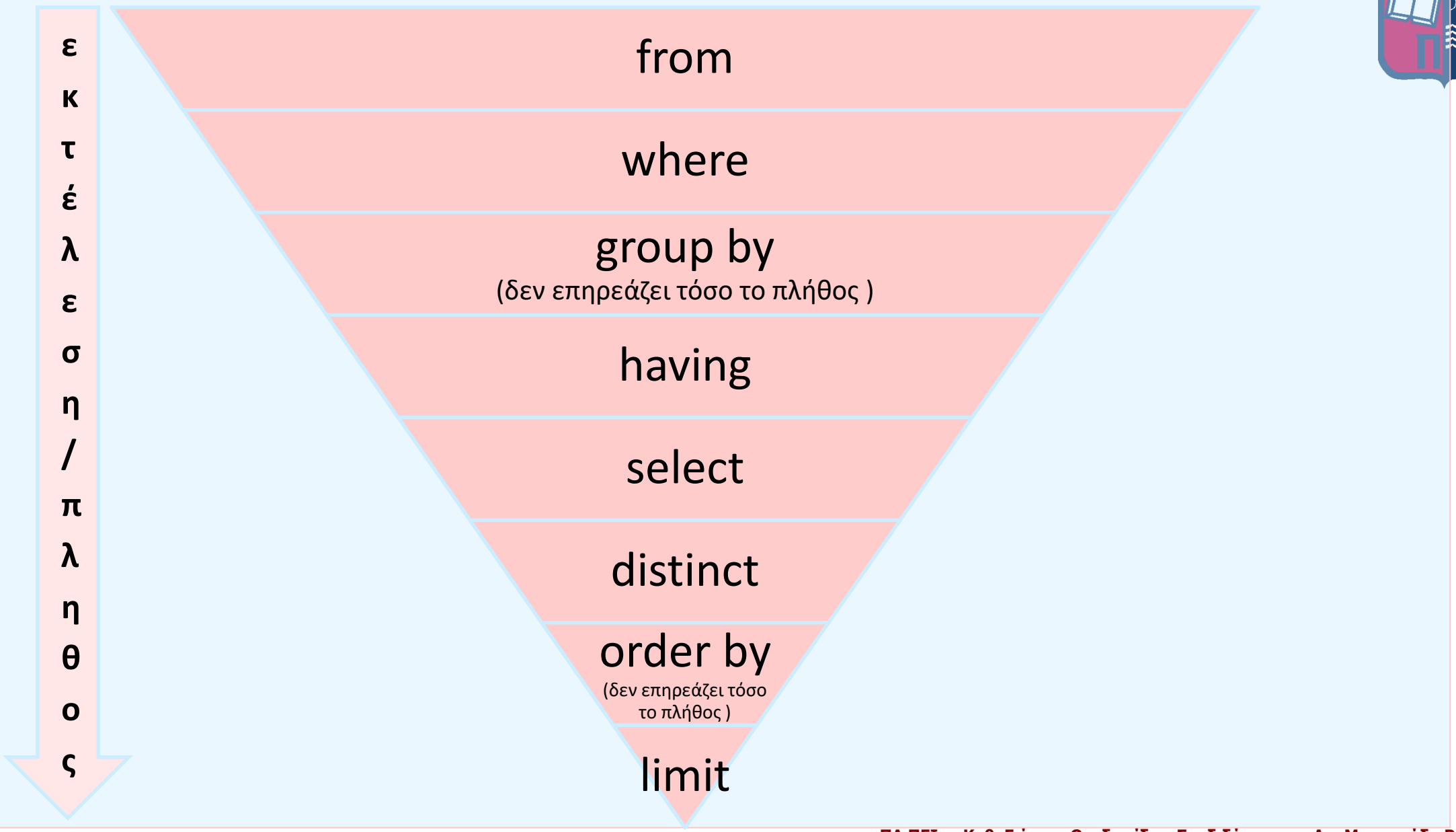
(π.χ)

```
/* group by με περιοριστικές συνθήκες -> having  */
select course_id, semester, year, sec_id, avg (tot_cred)
from student, takes
where student.ID= takes.ID and year = 2017
group by course_id, semester, year, sec_id
having count (student.ID) >= 2;
```

```
myunipi=# select course_id, semester, year, sec_id, avg (tot_cred)
myunipi-# from student, takes
myunipi-# where student.ID= takes.ID and year = 2017
myunipi-# group by course_id, semester, year, sec_id
myunipi-# having count (student.ID) >= 2;
+-----------+----------+------+--------+--------------------+
| course_id | semester | year | sec_id |        avg         |
+-----------+----------+------+--------+--------------------+
| CS-101    | Fall     | 2017 | 1      | 65.0000000000000000 |
| CS-190    | Spring   | 2017 | 2      | 43.0000000000000000 |
| CS-347    | Fall     | 2017 | 1      | 67.0000000000000000 |
+-----------+----------+------+--------+--------------------+
(3 rows)
```

- πρώτα τρέχει το **from**
- μετά το **where** , εάν υπάρχει, περιορίζει τις γραμμές των αποτελεσμάτων
- στη συνέχεια, εάν υπάρχει, το **group by** ομαδοποιεί τις γραμμές των αποτελεσμάτων (εάν δεν υπάρχει group by τότε θεωρείται ένα group όλες οι γραμμές)
- στη συνέχεια, εάν υπάρχει, το **having** περιορίζει τις γραμμές των αποτελεσμάτων
- και τέλος το **select** κάνει την προβολή, εφαρμόζοντας τις, τυχόν, **αθροιστικές** συναρτήσεις

SQL-DML σειρά εκτέλεσης / πλήθος αποτελεσμάτων



SQL-DML having 1-3

(π.χ)

```
/* group by με περιοριστικές συνθήκες -> having  */
select dept_name, avg (salary) as avg_salary
from instructor
group by dept_name
having avg (salary) > 42000;
```

```
myunipi=# select dept_name, avg (salary) as avg_salary
myunipi-# from instructor
myunipi-# group by dept_name
myunipi-# having avg (salary) > 42000;
+------------+--------------------+
| dept_name  |     avg_salary     |
+------------+--------------------+
| Finance    | 85000.000000000000 |
| History    | 61000.000000000000 |
| Physics    | 91000.000000000000 |
| Comp. Sci. | 77333.333333333333 |
| Biology    | 72000.000000000000 |
| Elec. Eng. | 80000.000000000000 |
+------------+--------------------+
(6 rows)
```

Παρατηρήστε ότι όλα τα ονόματα χαρακτηριστικών τα οποία εμφανίζονται στο group by, εμφανίζονται και στο select. Οτιδήποτε επιπλέον πρέπει να περιέχεται σε αθροιστική συνάρτηση.

Το having χρησιμοποιείται πάντα με το group by.
Δεν ισχύει και το αντίθετο, όμως.

SQL-DML συναρτήσεις 1-1

postgres (π.χ)

```
/* αθροιστικές συναρτήσεις */


select
sum (salary) as total_sal, /* άθροισμα όλων των τιμών στη στήλη salary */
max (salary) as highest_sal, /* μέγιστη τιμή στη στήλη salary */
min (salary) as lowest_sal, /* ελάχιστη τιμή στη στήλη salary */
avg (salary) as average_sal, /* μέση τιμή όλων των τιμών στη στήλη salary */
count(*) as total_rows /* άθροισμα του πλήθους των γραμμών της σχέσης
instructor*/
from instructor;
```

SQL-DML count 1-1

(π.χ)

/* NULL values and more on count() */

Όταν μια συναθριστική συνάρτηση εφαρμόζεται στις τιμές των χαρακτηριστικών π.χ. avg (salary) τότε αρχικά αφαιρούνται οι γραμμές οι οποίες έχουν null ως τιμή και στη συνέχεια, στις εναπομείνασες γραμμές εφαρμόζεται η συναθροιστική συνάρτηση.

Η count (*) επιστρέφει το πλήθος των γραμμών της σχέσης

count (salary) επιστρέφει το πλήθος των γραμμών στις οποίες το χαρακτηριστικό salary δεν είναι null.

SQL-DML aggregate functions 1-2

(π.χ)

/* error */

select distinct dept_name, **count**(dept_name) **from** instructor;

```
myunipi=# select distinct dept_name, count(dept_name) from instructor;
ERROR:  column "instructor.dept_name" must appear in the GROUP BY clause
 or be used in an aggregate function
LINE 1: select distinct dept_name, count(dept_name) from instructor;
                        ^
```

/* no problem. Θα μετρήσει μόνο τα μοναδικά dept_name */

select count(**distinct** dept_name) **from** instructor;

SQL-DML aggregate functions 1-3

(π.χ)

```
/* πλήθος all (null and not null) */
select count(*) from instructor;

/* πλήθος dept_name that it is not null */
select count(dept_name) from instructor;

select count(all dept_name) from instructor;

/* πλήθος των μοναδικών dept_name (not null) */
select count(distinct dept_name) from instructor;
```

```
myunipi=# select * from instructor;
+-------+------------+------------+-----------+
|  id   |    name    | dept_name  |  salary   |
+-------+------------+------------+-----------+
| 10101 | Srinivasan | Comp. Sci. |  65000.00 |
| 12121 | Wu         | Finance    |  90000.00 |
| 15151 | Mozart     | Music      |  40000.00 |
| 22222 | Einstein   | Physics    |  95000.00 |
| 32343 | El Said    | History    |  60000.00 |
| 33456 | Gold       | Physics    |  87000.00 |
| 45565 | Katz       | Comp. Sci. |  75000.00 |
| 58583 | Califieri  | History    |  62000.00 |
| 76543 | Singh      | Finance    |  80000.00 |
| 76766 | Crick      | Biology    |  72000.00 |
| 83821 | Brandt     | Comp. Sci. |  92000.00 |
| 98345 | Kim        | Elec. Eng. |  80000.00 |
| 1     | rosa       |            | 100000.00 |
+-------+------------+------------+-----------+
(13 rows)
```

```
myunipi=# select count(*) from instructor;
+-------+
| count |
+-------+
|    13 |
+-------+
(1 row)
```

```
myunipi=# select count(dept_name) from instructor;
+-------+
| count |
+-------+
|    12 |
+-------+
(1 row)

myunipi=# select count(all dept_name) from instructor;
+-------+
| count |
+-------+
|    12 |
+-------+
(1 row)
```

```
myunipi=# select count(distinct dept_name) from instructor;
+-------+
| count |
+-------+
|     7 |
+-------+
(1 row)
```

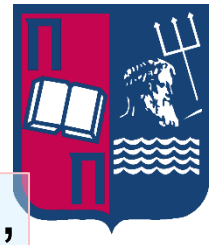
Βρες....

Θεωρήστε την παρακάτω σχεσιακή βάση δεδομένων, όπου τα πρωτεύοντα κλειδιά είναι υπογραμμισμένα. Δώστε μια δήλωση σε SQL για καθένα από τα ακόλουθα ερωτήματα:

- Βρείτε το ID, name, και city κάθε υπαλλήλου ο οποίος εργάζεται για στην εταιρείαΑ.
- Βρείτε το ID, name, και city κάθε υπαλλήλου ο οποίος εργάζεται για την εταιρείαΑ και κερδίζει πάνω από $10000.
- Βρείτε το ID κάθε υπαλλήλου ο οποίος δεν εργάζεται για την εταιρείαΑ.
- Βρείτε το ID κάθε υπαλλήλου ο οποίος και κερδίζει περισσότερα από κάθε άλλον υπάλληλο της εταιρείαςΑ.

employee (ID, person_name, street, city)
works (ID, company_name, salary)
company (company name, city)

SQL-DML insert into 1-1



Core SQL

Insert into table_name **values (**col1_value1, col2_value2, ….coln-valuen**);**

Insert into table_name (col1, col2 ….coln) **values (**value1, value2, …. valuen**);**

Εφαρμογή

insert into department **values** ('Biology', 'Watson', '90000');

insert into department (dept_name, building, budget) **values** ('Comp. Sci.', 'Taylor', '100000');

insert into department (building , dept_name, budget) **values** ('Taylor', 'Comp. Sci.', '100000');

SQL-DML insert into 1-2

Postgres

```
myunipi=# \h insert
Command:     INSERT
Description: create new rows in a table
Syntax:
[ WITH [ RECURSIVE ] with_query [, ...] ]
INSERT INTO table_name [ AS alias ] [ ( column_name [, ...] ) ]
    [ OVERRIDING { SYSTEM | USER } VALUE ]
    { DEFAULT VALUES | VALUES ( { expression | DEFAULT } [, ...] ) [, ...] | query }
    [ ON CONFLICT [ conflict_target ] conflict_action ]
    [ RETURNING * | output_expression [ [ AS ] output_name ] [, ...] ]
```

SQL-DML insert into 1-3

Postgres
(π.χ.)

/*πρώτες n στήλες*/

insert into department **values** ('plh');

```
create table department
   (dept_name  varchar(20),
    building  varchar(15),
    budget  numeric(12,2) check (budget > 0),
primary key (dept_name) );
```

insert into department (dept_name,building, budget)
values ('dept1', 'Taylor', '100000'),
('dept2', 'Taylor', '100000'),
('dept3', 'Taylor', '100000');

insert into department **values** ('plh'),(**null**), (**default**),('psi',**null**,**null**);
insert into department **default values**;

SQL-DML insert into 1-4

Postgres (π.χ.)

```
/*τρόποι εισαγωγής πρωτογενών δεδομένων*/
copy department from '/private/tmp/country_data.csv' delimiters ',' csv header;

insert into department (dept_name,building, budget )
values ('dept1', 'Taylor', '100000'),
       ('dept2', 'Taylor', '100000'),
       ('dept3', 'Taylor', '100000');

insert into department (dept_name,building, budget ) values ('dept1', 'Taylor',
'100000');
insert into department (dept_name,building, budget ) values ('dept2', 'Taylor',
'100000');
```

SQL-DML insert into 1-5

Postgres
(π.χ.)

```
/*αντιγραφή δεδομένων από ήδη υπάρχον πίνακα */
insert into department (dept_name, building)
(select name, ktirio from new_department
where budget = 1000 );

/*ατέρμονο loop ή μήπως όχι; */
insert into department
(select * from department);
```

SQL-DML insert into 1-6

Postgres (π.χ.)

```
/*αντιγραφή ήδη υπάρχον πίνακα  σε άλλον με ή χωρίς data*/
/*new table with data  or with no data*/
create table department_temp as
(select d.*
from department as d
where d.budget > 80000) with data;
/*copy table structure (with constriants) with no data*/
create table department_temp (like department including all);
/*copy table structure dependable from parent */
create table department_temp () inherits (department);
insert into department_temp values ('mydept','smith',1); /*insert some data*/
select * from department_temp; /*επιστρέφει data από department_temp*/
select * from department; /*επιστρέφει data από department και department_temp*/
select * from only department; /*επιστρέφει data από department*/
```

SQL-DML insert into 1-7

Postgres
(π.χ.)

insert into department **values** ('dept1'), ('dept2') **returning** *;

```
myunipi=# insert into department values ('dept1'),('dept2') returning *;
+-----------+----------+--------+
| dept_name | building | budget |
+-----------+----------+--------+
| dept1     |          |        |
| dept2     |          |        |
+-----------+----------+--------+
(2 rows)
```

insert into department
values ('dept1'), ('dept2') **returning** dept_name;

```
myunipi=# insert into department values ('dept1'), ('dept2')
myunipi-# returning  dept_name;
+-----------+
| dept_name |
+-----------+
| dept1     |
| dept2     |
+-----------+
(2 rows)
```

SQL-DML insert into 1-8

Postgres
(π.χ.)

```
insert into department
values ('dept1'), ('dept2')
returning  dept_name as my_newly_inserted_dept_name;
```

```
myunipi=# insert into department
myunipi-# values ('dept1'), ('dept2')
myunipi-# returning  dept_name as my_newly_inserted_dept_name;
+-----------------------------+
| my_newly_inserted_dept_name |
+-----------------------------+
| dept1                       |
| dept2                       |
+-----------------------------+
(2 rows)
```

SQL-DML insert into (upsert) 1-9

Postgres (π.χ.)

```
/*duplicated pk*/
insert into department (dept_name, building)
values ('Biology', null),
('plh', 'Room106,201,203');

upsert = update and insert

/*on duplicated pk do nothing*/
insert into department (dept_name, building)
values ('Biology', null),
('plh', 'Room106,201,203') ;
on conflict (dept_name) do nothing;
```

```
myunipi=# select * from department;
+------------+----------+-----------+
| dept_name  | building |  budget   |
+------------+----------+-----------+
| Biology    | Watson   |  90000.00 |
| Comp. Sci. | Taylor   | 100000.00 |
| Elec. Eng. | Taylor   |  85000.00 |
| Finance    | Painter  | 120000.00 |
| History    | Painter  |  50000.00 |
| Music      | Packard  |  80000.00 |
| Physics    | Watson   |  70000.00 |
+------------+----------+-----------+
(7 rows)
```

```
myunipi=# select * from department;
+------------+-----------------+-----------+
| dept_name  |    building     |  budget   |
+------------+-----------------+-----------+
| Biology    | Watson          |  90000.00 |
| Comp. Sci. | Taylor          | 100000.00 |
| Elec. Eng. | Taylor          |  85000.00 |
| Finance    | Painter         | 120000.00 |
| History    | Painter         |  50000.00 |
| Music      | Packard         |  80000.00 |
| Physics    | Watson          |  70000.00 |
| plh        | Room106,201,203 |           |
+------------+-----------------+-----------+
(8 rows)
```

SQL-DML insert into (upsert) 1-10

Postgres (π.χ.)

```
/*duplicated pk*/
insert into department (dept_name, budget)
values ('plh', 0.1);
```

```
/*on duplicated pk do update */
insert into department (dept_name, budget)
values ('plh', 0.1)
on conflict (dept_name) do update
set dept_name='testme'  ;
```

```
myunipi=# select * from department;
+------------+-----------------+----------+
| dept_name  |    building     |  budget  |
+------------+-----------------+----------+
| Biology    | Watson          |  90000.00 |
| Comp. Sci. | Taylor          | 100000.00 |
| Elec. Eng. | Taylor          |  85000.00 |
| Finance    | Painter         | 120000.00 |
| History    | Painter         |  50000.00 |
| Music      | Packard         |  80000.00 |
| Physics    | Watson          |  70000.00 |
| plh        | Room106,201,203 |           |
+------------+-----------------+----------+
(8 rows)
```

```
myunipi=# select * from department;
+------------+-----------------+----------+
| dept_name  |    building     |  budget  |
+------------+-----------------+----------+
| Biology    | Watson          |  90000.00 |
| Comp. Sci. | Taylor          | 100000.00 |
| Elec. Eng. | Taylor          |  85000.00 |
| Finance    | Painter         | 120000.00 |
| History    | Painter         |  50000.00 |
| Music      | Packard         |  80000.00 |
| Physics    | Watson          |  70000.00 |
| testme     | Room106,201,203 |           |
+------------+-----------------+----------+
(8 rows)
```

SQL-DML insert into (upsert) 1-11

Postgres (π.χ.)

Select * from department;

```
myunipi=# select * from department;
+------------+-----------------+-----------+
| dept_name  |    building     |  budget   |
+------------+-----------------+-----------+
| Biology    | Watson          |  90000.00 |
| Comp. Sci. | Taylor          | 100000.00 |
| Elec. Eng. | Taylor          |  85000.00 |
| Finance    | Painter         | 120000.00 |
| History    | Painter         |  50000.00 |
| Music      | Packard         |  80000.00 |
| Physics    | Watson          |  70000.00 |
| plh        | Room106,201,203 |           |
+------------+-----------------+-----------+
(8 rows)
```

insert into department (dept_name, budget)
values ('plh',0.1)
on conflict (dept_name)
do update set dept_name='testme',
budget=0.1 ;

```
myunipi=# select * from department;
+------------+-----------------+-----------+
| dept_name  |    building     |  budget   |
+------------+-----------------+-----------+
| Biology    | Watson          |  90000.00 |
| Comp. Sci. | Taylor          | 100000.00 |
| Elec. Eng. | Taylor          |  85000.00 |
| Finance    | Painter         | 120000.00 |
| History    | Painter         |  50000.00 |
| Music      | Packard         |  80000.00 |
| Physics    | Watson          |  70000.00 |
| testme     | Room106,201,203 |      0.10 |
+------------+-----------------+-----------+
(8 rows)
```

SQL-DML insert into (upsert) 1-12

Postgres (π.χ.)

Select * from department;

```
myunipi=# select * from department;
+------------+-----------------+-----------+
| dept_name  |    building     |  budget   |
+------------+-----------------+-----------+
| Biology    | Watson          |  90000.00 |
| Comp. Sci. | Taylor          | 100000.00 |
| Elec. Eng. | Taylor          |  85000.00 |
| Finance    | Painter         | 120000.00 |
| History    | Painter         |  50000.00 |
| Music      | Packard         |  80000.00 |
| Physics    | Watson          |  70000.00 |
| plh        | Room106,201,203 |           |
+------------+-----------------+-----------+
(8 rows)
```

insert into department (dept_name, budget)
values ('plh',10000), ('Physics',0.01)
on conflict (dept_name)
do update set budget=**EXCLUDED**. budget;

```
myunipi=# select * from department;
+------------+-----------------+-----------+
| dept_name  |    building     |  budget   |
+------------+-----------------+-----------+
| Biology    | Watson          |  90000.00 |
| Comp. Sci. | Taylor          | 100000.00 |
| Elec. Eng. | Taylor          |  85000.00 |
| Finance    | Painter         | 120000.00 |
| History    | Painter         |  50000.00 |
| Music      | Packard         |  80000.00 |
| plh        | Room106,201,203 |  10000.00 |
| Physics    | Watson          |      0.01 |
+------------+-----------------+-----------+
(8 rows)
```

SQL-DML update 1-1

Core SQL

update table_name **set** column1=value1, column2=value2
[**where** some_condition]

Εφαρμογή

update instructor **set** salary= 0.1;
update instructor **set** salary= salary * 1.05;
update instructor **set** salary = salary * 1.05 **where** salary < 70000;

Postgres

```
myunipi=# \h update
Command:     UPDATE
Description: update rows of a table
Syntax:
[ WITH [ RECURSIVE ] with_query [, ...] ]
UPDATE [ ONLY ] table_name [ * ] [ [ AS ] alias ]
    SET { column_name = { expression | DEFAULT } |
          ( column_name [, ...] ) = [ ROW ] ( { expression | DEFAULT } [, ...] ) |
          ( column_name [, ...] ) = ( sub-SELECT )
        } [, ...]
    [ FROM from_item [, ...] ]
    [ WHERE condition | WHERE CURRENT OF cursor_name ]
    [ RETURNING * | output_expression [ [ AS ] output_name ] [, ...] ]

URL: https://www.postgresql.org/docs/13/sql-update.html
```

SQL-DML update 1-2

postgres (π.χ)

```
-- ένας τρόπος σύνταξης της εντολής
update department set building='room 21', budget=100 where dept_name='plh';

-- διαφορετική σύνταξη
update department set(building, budget) =('room 21', 100) where
dept_name='plh';

--upadate από ήδη υπάρχον πίνακα
create table depart_temp (like department); /* create a temp table */
insert into depart_temp values ('plh','nikaia',555); /* insert some data to temp table */
update department
  set building = depart_temp. building, budget= depart_temp. budget
  from depart_temp
  where department.dept_name = depart_temp.dept_name;
```

SQL-DML update 1-3

postgres (π.χ)

```
-- with …
with subquery as (
select name, dept_name
from instructor
)
update department
set building = subquery.name
from subquery
where department.dept_name = subquery.dept_name;

/* το κτήριο με dept_name='Comp. Sci.' ποιανού το όνομα πήρε τελικά*/
```

Συγχρονισμός (Concurrency Control) 1-1

Η αντιμετώπιση των ταυτόχρονων/σύγχρονων ενεργειών στη βάση:

- προστατεύει την ατομικότητα των ενεργειών (isolation)
- διατηρεί την συνέπεια των δεδομένων (consistency)
- αντιμετωπίζει τις συγκρούσεις σε ενέργειες όπως read-write και write-read

Συγχρονισμός (Concurrency Control) 1-2

Τρεις γενικά τρόποι προσέγγισης

- Locking Protocol
  (i). Lock Acquisition (ii). Modification of Data (iii). Release Lock
- Time Stamp Ordering Protocol
  (i) W-timestamp(X), (ii) R-timestamp(X)
- Multiversion Concurrency Control (postgres)
  Διατηρεί εκδόσεις της κατάστασης της βάσης διακρινόμενες με τη χρήση timestamps.
  Κάθε ενέργεια write αλλάζει το timestamp.
  Όταν συμβαίνει μια ενέργεια read τότε ανάλογα τη χρονική στιγμή το read διαβάζει την κατάλληλη έκδοση των δεδομένων.
  (update στην πραγματικότητα είναι delete και μετά insert)
- Validation Concurrency Control
  (i) Read phase (ii) Validation phase (iii) Write phase
  (αισιόδοξη προσέγγιση, ποια ή πιθανότητα να χρειαστούν περισσότερες ενέργειες στα ίδια δεδομένα;)

SQL-DML delete 1-1

Core SQL

delete from table_name
[**where** some-condition] **;**

Εφαρμογή

delete from instructor; /*vs truncate instructor*/
delete from instructor **where** dept name = 'Finance';

Postgres

```
mýunipi=# \h delete
Command:     DELETE
Description: delete rows of a table
Syntax:
[ WITH [ RECURSIVE ] with_query [, ...] ]
DELETE FROM [ ONLY ] table_name [ * ] [ [ AS ] alias ]
    [ USING from_item [, ...] ]
    [ WHERE condition | WHERE CURRENT OF cursor_name ]
    [ RETURNING * | output_expression [ [ AS ] output_name ] [, ...] ]

URL: https://www.postgresql.org/docs/13/sql-delete.html
```

SQL-DML delete 1-2

postgres
(π.χ)

```
/* επιστρέφει όλες τις γραμμές που έχουν υποστεί διαγραφή*/
delete from department where building = 'Taylor' returning *;
```

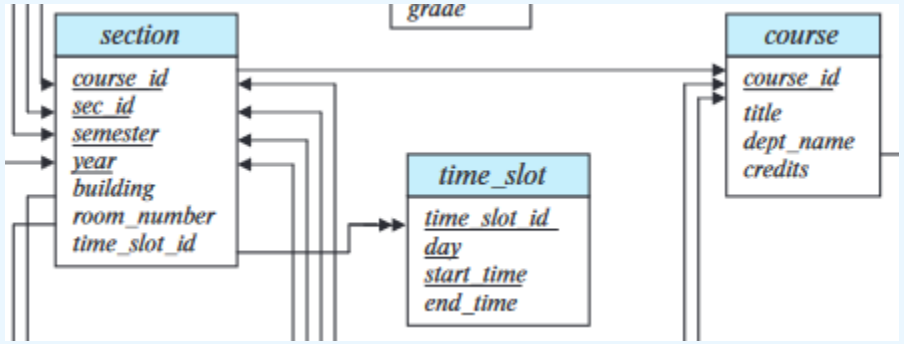
```
/* δημιουργία log table*/
create table teaches_log (like teaches);
/* ενημέρωση log table*/
with moved_rows as (
    delete from teaches where year > '2016' and year < '2018' returning * )
insert into teaches_log
select * from moved_rows;
```

SQL-DML delete 1-3

postgres (π.χ)

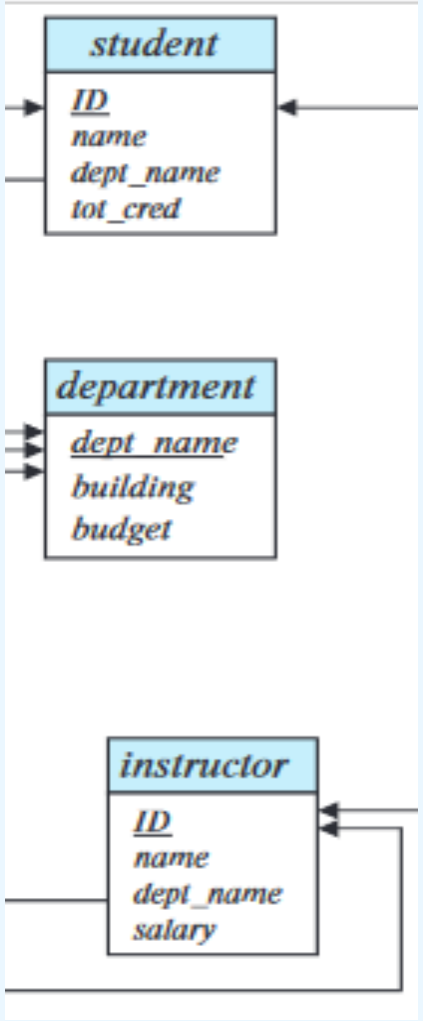
```
/* διαγραφή με αναφορά σε άλλο table*/
delete from department
using instructor, student, course
where
department.dept_name = instructor.dept_name and
department.dept_name = student.dept_name and
department.dept_name = course.dept_name  and
 department.dept_name <> 'Biology' ;
```

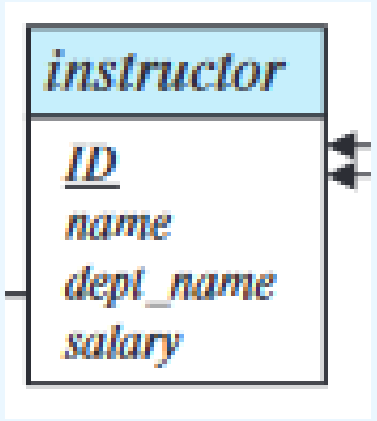
- ✓ Γράψτε ερώτημα SQL το οποίο θα πραγματοποιεί (και αποθηκεύει) αύξηση του μισθού κάθε εκπαιδευτή στο τμήμα Comp. Sci. κατά 10%



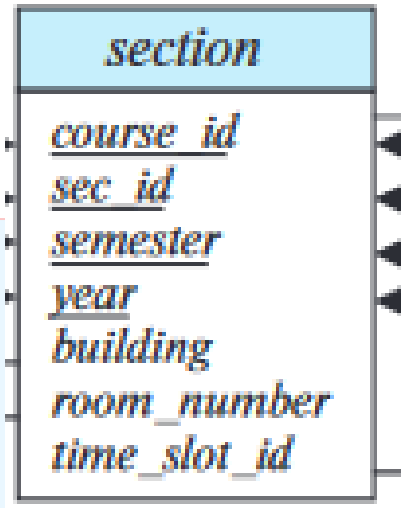
- ✓ Διαγράψτε όλα τα μαθήματα τα οποία δεν ήταν ποτέ διαθέσιμα προς διδασκαλία σε οποιαδήποτε διδακτική ενότητα (σχέση section).

- ✓ Εισαγάγετε κάθε μαθητή του οποίου το χαρακτηριστικό tot_cred είναι μεγαλύτερο από 100 ως εκπαιδευτής στο ίδιο τμήμα, με μισθό $10.000.





SQL-DML select πράξεις συνόλων 1-1



(π.χ)

Βρείτε όλους τους κωδικούς των μαθήματων τα οποία διδάχθηκαν το Φθινόπωρο του 2017 **ή** την Άνοιξη του 2018.

select course_id, semester, year **from** section
where (semester = 'Fall' **and** year= 2017) **or**
(semester = 'Spring' **and** year= 2018);

```
myunipi=# select course_id, semester, year from section
where (semester = 'Fall' and year= 2017) or (semester =
'Spring' and year= 2018);
+-----------+----------+------+
| course_id | semester | year |
+-----------+----------+------+
| CS-101    | Fall     | 2017 |
| CS-101    | Spring   | 2018 |
| CS-315    | Spring   | 2018 |
| CS-319    | Spring   | 2018 |
| CS-319    | Spring   | 2018 |
| CS-347    | Fall     | 2017 |
| FIN-201   | Spring   | 2018 |
| HIS-351   | Spring   | 2018 |
| MU-199    | Spring   | 2018 |
| PHY-101   | Fall     | 2017 |
+-----------+----------+------+
(10 rows)
```

Βρείτε όλους τους κωδικούς των μαθήματων τα οποία διδάχθηκαν το Φθινόπωρο του 2017 **ή/και** την Άνοιξη του 2018.

select course_id, semester, year **from** section
where (semester = 'Fall' **and** year= 2017) **and**
(semester = 'Spring' **and** year= 2018); (λάθος)

```
myunipi=# select course_id, semester, year
 from section
where (semester = 'Fall' and year= 2017)
and (semester = 'Spring' and year= 2018);
+-----------+----------+------+
| course_id | semester | year |
+-----------+----------+------+
+-----------+----------+------+
```

SQL-DML select πράξεις συνόλων 1-2

(π.χ)

Βρείτε όλους τους κωδικούς των μαθημάτων τα οποία διδάχθηκαν ή/και το Φθινόπωρο του 2017 **ή/και** την Άνοιξη του 2018.

```
/*ένωση στο ένα ή στο άλλο ή/και στα δυο.
Αυτόματη απαλοιφή των διπλότυπων*/
(select course_id, semester, year
from section
where semester = 'Fall' and year= 2017)
union
(select course_id, semester, year
from section
where semester = 'Spring' and year= 2018);
```

```
myunipi=# (select course_id, semester, year
myunipi(# from section
myunipi(# where semester = 'Fall' and year= 2017)
myunipi-# union
myunipi-# (select course_id, semester, year
myunipi(# from section
myunipi(# where semester = 'Spring' and year= 2018);
+-----------+----------+------+
| course_id | semester | year |
+-----------+----------+------+
| CS-101    | Fall     | 2017 |
| CS-101    | Spring   | 2018 |
| CS-315    | Spring   | 2018 |
| CS-319    | Spring   | 2018 |
| CS-347    | Fall     | 2017 |
| FIN-201   | Spring   | 2018 |
| HIS-351   | Spring   | 2018 |
| MU-199    | Spring   | 2018 |
| PHY-101   | Fall     | 2017 |
+-----------+----------+------+
(9 rows)
```

SQL-DML select πράξεις συνόλων 1-3

(π.χ)

Βρείτε όλους τους κωδικούς των μαθημάτων τα οποία διδάχθηκαν το Φθινόπωρο του 2017 **και** την Άνοιξη του 2018.

```
/*τομή και το ένα και το άλλο.
Αυτόματη απαλοιφή των διπλότυπων */
(select course_id from section
where semester = 'Fall' and year= 2017)
intersect
(select course_id from section
where semester = 'Spring'
and year= 2018);
```

```
myunipi=# select course_id, semester, year from section;
+-----------+----------+------+
| course_id | semester | year |
+-----------+----------+------+
| BIO-101   | Summer   | 2017 |
| BIO-301   | Summer   | 2018 |
| CS-101    | Fall     | 2017 |
| CS-101    | Spring   | 2018 |
| CS-190    | Spring   | 2017 |
| CS-190    | Spring   | 2017 |
| CS-315    | Spring   | 2018 |
| CS-319    | Spring   | 2018 |
| CS-319    | Spring   | 2018 |
| CS-347    | Fall     | 2017 |
| EE-181    | Spring   | 2017 |
| FIN-201   | Spring   | 2018 |
| HIS-351   | Spring   | 2018 |
| MU-199    | Spring   | 2018 |
| PHY-101   | Fall     | 2017 |
+-----------+----------+------+
(15 rows)
```

```
myunipi=# (select course_id from section
myunipi(# where semester = 'Fall' and year= 2017)
myunipi-# intersect
myunipi-# (select course_id from section
myunipi(# where semester = 'Spring' and year= 2018);
+-----------+
| course_id |
+-----------+
| CS-101    |
+-----------+
(1 row)

Time: 0.423 ms
```

SQL-DML select πράξεις συνόλων 1-4

```
myunipi=# select course_id, semester, year from section;
+-----------+----------+------+
| course_id | semester | year |
+-----------+----------+------+
| BIO-101   | Summer   | 2017 |
| BIO-301   | Summer   | 2018 |
| CS-101    | Fall     | 2017 |
| CS-101    | Spring   | 2018 |
| CS-190    | Spring   | 2017 |
| CS-190    | Spring   | 2017 |
| CS-315    | Spring   | 2018 |
| CS-319    | Spring   | 2018 |
| CS-319    | Spring   | 2018 |
| CS-347    | Fall     | 2017 |
| EE-181    | Spring   | 2017 |
| FIN-201   | Spring   | 2018 |
| HIS-351   | Spring   | 2018 |
| MU-199    | Spring   | 2018 |
| PHY-101   | Fall     | 2017 |
+-----------+----------+------+
(15 rows)
```

(π.χ)

Βρείτε όλους τους κωδικούς των μαθημάτων τα οποία διδάχθηκαν το Φθινόπωρο του 2017 **και όχι** την Άνοιξη του 2018.

```
/*διαφορά μόνο στο ένα και όχι στο άλλο*/
(select course_id from section
where semester = 'Fall' and year= 2017)
except
(select course_id from section
where semester = 'Spring' and year= 2018);
```

```
+-----------+
| course_id |
+-----------+
| PHY-101   |
| CS-347    |
+-----------+
(2 rows)
```

Βρείτε όλους τους κωδικούς των μαθημάτων τα οποία διδάχθηκαν την Άνοιξη του 2018 **και όχι** το Φθινόπωρο του 2017.

```
/*η σειρά των select παίζει ρόλο*/
```

```
+-----------+
| course_id |
+-----------+
| HIS-351   |
| CS-319    |
| FIN-201   |
| CS-315    |
| MU-199    |
+-----------+
(5 rows)
```

SQL-DML select πράξεις συνόλων 1-5

Η πράξεις union, intersect, except αφαιρούν αυτόματα όλες τις διπλότυπες εγγραφές.
με **union all**, **intersect all**, **except all** εμφανίζονται και τα διπλότυπα.

(π.χ)

Μπορεί να εφαρμοστούν σε σχέσεις οι οποίες έχουν ίδιο αριθμό χαρακτηριστικών (στηλών), τα χαρακτηριστικά είναι ίδιου τύπου και είναι όμοια ταξινομημένα.
(Τα select τα οποία πραγματοποιούνται θα πρέπει να επιστρέφουν ίδιο αριθμό στηλών, ιδίου τύπου δεδομένα και με ίδια σειρά.
Οι υπόλοιπες στήλες οι οποίες τυχόν υπάρχουν στις σχέσεις αλλά δεν αναφέρονται στα select είναι αδιάφορες)

SQL-DML select πράξεις συνόλων 1-6

(π.χ)

Άλλες πράξεις συνόλων είναι:
in: παρέχει έλεγχο εάν ένα στοιχείο υπάρχει σε ένα σύνολο,
select * from department where dept_name **in** ('Biology', 'Finance', 'Music');

τελεστής (π.χ. =, <>, <, <=, > , >= κλπ) **any/some**,
τελεστής (π.χ. =, <>, <, <=, > , >= κλπ) **all** : παρέχει έλεγχο σύγκρισης μεταξύ ενός στοιχείου και ενός συνόλου από τιμές,

exists : παρέχει έλεγχο εάν το σύνολο είναι άδειο.

Στα in και exists η άρνηση δηλώνεται με το **not**, π.χ. not in οποιαδήποτε τιμή εκτός από αυτές του συνόλου.
select * from department where dept_name **not in** ('Biology', 'Finance', 'Music');

SQL-DML Εμφωλευμένα υποερωτήματα (nested queries) 1-1

Εμφωλευμένα ερωτήματα (nested queries) επιστρέφουν ένα σύνολο από τιμές.
Τα δυο queries (εξωτερικό και εμφωλευμένο) συνδέονται με κάποιου είδους πράξη μεταξύ συνόλων (in, exists, all, some) ή/και τελεστή σύγκρισης (=, <>, <, <=, > , >=).

(π.χ) Αποτελούν ένα πλήρες ερώτημα select_from_where το οποίο μπορεί να τρέξει ανεξάρτητα είτε σε συνδυασμό (Correlated Nested Queries)
Αποτελούν χρονοβόρες διαδικασίες και καλό είναι να αποφεύγονται.

Μπορεί να υπάρξουν μέσα στο τμήμα:

- where
- from
- select

SQL-DML Εμφωλευμένα υποερωτήματα (nested queries) 1-2

- μέσα στο where

Βρείτε όλους τους κωδικούς των μαθημάτων τα οποία διδάχθηκαν το Φθινόπωρο του 2017 **και** την Άνοιξη του 2018.

(π.χ)

```
/*τομή- intersect και το ένα και το άλλο*/
/*το intersect αφαιρεί τα διπλότυπα αυτόματα εδώ distinct */
select distinct course_id from section
where semester = 'Fall' and year= 2017 and course_id
in
(select course_id
from section
where semester = 'Spring' and year= 2018);
```

```
myunipi=# select distinct course_id
myunipi-# from section
myunipi-# where semester = 'Fall' and year= 2017 and
myunipi-# course_id in (select course_id
myunipi(# from section
myunipi(# where semester = 'Spring' and year= 2018);
+-----------+
| course_id |
+-----------+
| CS-101    |
+-----------+
(1 row)


Time: 0.594 ms
```

SQL-DML Εμφωλευμένα υποερωτήματα (nested queries) 1-3

- μέσα στο where

Βρείτε όλους τους κωδικούς των μαθήματων τα οποία διδάχθηκαν το Φθινόπωρο του 2017 **και όχι** την Άνοιξη του 2018.

/*διαφορά **except** μόνο στο ένα και όχι στο άλλο*/

(π.χ)

```
select distinct course_id
from section
where semester = 'Fall' and year= 2017 and
course_id not in (select course_id
from section
where semester = 'Spring' and year= 2018);
```

```
myunipi=# select distinct course_id
myunipi-# from section
myunipi-# where semester = 'Fall' and year= 2017 and
myunipi-# course_id not in (select course_id
myunipi(# from section
myunipi(# where semester = 'Spring' and year= 2018);
+-----------+
| course_id |
+-----------+
| CS-347    |
| PHY-101   |
+-----------+
(2 rows)
```

SQL-DML Εμφωλευμένα υποερωτήματα (nested queries) 1-4

(π.χ)

➢ μέσα στο where

Βρείτε όλους τους διδάσκοντες οι οποίοι έχουν μισθό μεγαλύτερο από τουλάχιστον έναν διδάσκοντα του τμήματος 'Biology'.

```
select name
from instructor
where salary > some (select salary
from instructor
where dept_name = 'Biology');
```

"greater than at least one"
στην SQL είναι ισοδύναμο με "> some"

```
myunipi=# select name
myunipi-# from instructor
myunipi-# where salary > some (select salary
myunipi(# from instructor
myunipi(# where dept_name = 'Biology');
+----------+
|   name   |
+----------+
| Wu       |
| Einstein |
| Gold     |
| Katz     |
| Singh    |
| Brandt   |
| Kim      |
| rosa     |
+----------+
(8 rows)
```

SQL-DML Εμφωλευμένα υποερωτήματα (nested queries) 1-5

(π.χ)

- μέσα στο where

Βρείτε όλους τους διδάσκοντες οι οποίοι έχουν μισθό μεγαλύτερο από όλους διδάσκοντες του τμήματος 'Biology'.

```
select name
from instructor
where salary > all (select salary
from instructor
where dept_name = 'Biology');
```

"greater than all"
στην SQL είναι ισοδύναμο με "> all"

```
myunipi=# select name
myunipi-# from instructor
myunipi-# where salary > all (select salary
myunipi(# from instructor
myunipi(# where dept_name = 'Biology');
+----------+
|   name   |
+----------+
| Wu       |
| Einstein |
| Gold     |
| Katz     |
| Singh    |
| Brandt   |
| Kim      |
| rosa     |
+----------+
(8 rows)


Time: 0.579 ms
```

SQL-DML Εμφωλευμένα υποερωτήματα (nested queries) 1-6

(π.χ)

```
myunipi=# select * from instructor;
+-------+------------+------------+-----------+
|  id   |    name    | dept_name  |  salary   |
+-------+------------+------------+-----------+
| 10101 | Srinivasan | Comp. Sci. |  65000.00 |
| 12121 | Wu         | Finance    |  90000.00 |
| 15151 | Mozart     | Music      |  40000.00 |
| 22222 | Einstein   | Physics    |  95000.00 |
| 32343 | El Said    | History    |  60000.00 |
| 33456 | Gold       | Physics    |  87000.00 |
| 45565 | Katz       | Comp. Sci. |  75000.00 |
| 58583 | Califieri  | History    |  62000.00 |
| 76543 | Singh      | Finance    |  80000.00 |
| 76766 | Crick      | Biology    |  72000.00 |
| 83821 | Brandt     | Comp. Sci. |  92000.00 |
| 98345 | Kim        | Elec. Eng. |  80000.00 |
| 1     | rosa       |            | 100000.00 |
+-------+------------+------------+-----------+
(13 rows)
```

SQL-DML Εμφωλευμένα υποερωτήματα (nested queries) 1-7

- μέσα στο where και συσχετιζόμενα εμφωλευμένα ερωτήματα (Correlated Nested Queries). Προσοχή στα ονόματα των κοινών χαρακτηριστικών μεταξύ των σχέσων

Βρείτε όλους τους κωδικούς των μαθημάτων τα οποία διδάχθηκαν το Φθινόπωρο του 2017 **και** την Άνοιξη του 2018.

(π.χ)

```
/*τομή- intersect και το ένα και το άλλο*/
select distinct course_id from section as S
where semester = 'Fall' and year= 2017 and
exists
(select * from section as T
where semester = 'Spring' and year= 2018
and  S.course_id= T.course_id);
```

```
myunipi=# select distinct course_id
myunipi-# from section as S
myunipi-# where semester = 'Fall' and year= 2017 and
myunipi-# exists (select *
myunipi(# from section as T
myunipi(# where semester = 'Spring' and year= 2018
myunipi(# and  S.course_id= T.course_id);
+-----------+
| course_id |
+-----------+
| CS-101    |
+-----------+
(1 row)


Time: 0.599 ms
```

SQL-DML Εμφωλευμένα υποερωτήματα (nested queries) 1-8

- μέσα στο from

Βρείτε τους μέσους μισθούς των διδασκόντων ανά τμήμα, αλλά αυτών των τμημάτων όπου ο μέσος μισθός είναι μεγαλύτερος από $42.000.

(π.χ)

```
select dept_name, avg_salary
from (select dept_name, avg (salary) as avg_salary
from instructor
group by dept_name)
where avg_salary > 42000;
```

```
.....
group by dept_name) as dept_avg
.....
```

/*error στην postgres τα ερωτήματα στο from πρέπει να έχουν όνομα*/

```
myunipi=# select dept_name, avg_salary
myunipi-# from (select dept_name, avg (salary) as avg_salary
myunipi(# from instructor
myunipi(# group by dept_name)
myunipi-# where avg_salary > 42000;
ERROR:  subquery in FROM must have an alias
LINE 2: from (select dept_name, avg (salary) as avg_salary
              ^
```

SQL-DML Εμφωλευμένα υποερωτήματα (nested queries) 1-9

(π.χ)

➢ μέσα στο from

Βρείτε τους μέσους μισθούς των διδασκόντων ανά τμήμα, αλλά αυτών των τμημάτων όπου ο μέσος μισθός είναι μεγαλύτερος από $42.000.

```
select dept_name, avg_salary

from (select dept_name, avg(salary)
from instructor
group by dept_name)
as dept_avg (dept_name, avg_salary)

where avg_salary > 42000;
```

```
myunipi=# select dept_name, avg_salary
myunipi-# from (select dept_name, avg(salary)
myunipi(# from instructor
myunipi(# group by dept_name)
myunipi-# as dept_avg (dept_name, avg_salary)
myunipi-# where avg_salary > 42000;
+------------+----------------------+
| dept_name  |     avg_salary       |
+------------+----------------------+
|            | 100000.000000000000 |
| Finance    |  85000.000000000000 |
| History    |  61000.000000000000 |
| Physics    |  91000.000000000000 |
| Comp. Sci. |  77333.333333333333 |
| Biology    |  72000.000000000000 |
| Elec. Eng. |  80000.000000000000 |
+------------+----------------------+
(7 rows)
```

SQL-DML Εμφωλευμένα υποερωτήματα (nested queries) 1-10

- μέσα στο select

Βρείτε όλα τα τμήματα μαζί με τον αριθμό των εκπαιδευτών σε κάθε τμήμα.

(π.χ)

```
/*πρέπει να είναι σίγουρο ότι θα επιστρέψει μια τιμή. Διαφορετικά δίνει error.*/
select dept_name,
(select count(*)
from instructor
where department.dept_name = instructor.dept_name)
as num_instructors
from department;
```

```
myunipi=# select dept_name,
myunipi-# (select count(*)
myunipi(# from instructor
myunipi(# where department.dept_name = instructor.dept_name)
myunipi-# as num_instructors
myunipi-# from department;
+------------+-----------------+
| dept_name  | num_instructors |
+------------+-----------------+
| Biology    |               1 |
| Comp. Sci. |               3 |
| Elec. Eng. |               1 |
| History    |               2 |
| Music      |               1 |
| Physics    |               2 |
| Finance    |               2 |
| dept1      |               0 |
| dept2      |               0 |
+------------+-----------------+
(9 rows)
```

SQL-DML Εμφωλευμένα υποερωτήματα (nested queries) 1-11

Τα εμφωλευμένα υποερωτήματα μπορεί να συνδυαστούν με:

- insert into

(π.χ)

```
insert into department (dept_name, building)
 ( select name, ktirio from new_department  where budget = 1000 );
```

- update (μέσα στο where)

```
update department set building= building ||', room 21' where budget >
( select sum(salary) from instructor where
instructor.dept_name=deptartment.dept_name );
```

- delete (μέσα στο where)

```
 delete from department where budget >
( select sum(salary) from instructor where
instructor.dept_name=deptartment.dept_name );
```

- select το πιο συνηθισμένο από όλα

SQL-DML with… 1-1

(π.χ)

- Δημιουργεί μια προσωρινή σχέση/πίνακα η οποία είναι διαθέσιμη μόνο εντός του ερωτήματος/εντολής SQL στην οποία χρησιμοποιείται το with.
- Η προσωρινή σχέση μπορεί να χρησιμοποιηθεί σε πολλαπλά σημεία εντός του ερωτήματος/ εντολής της SQL.
- Μπορεί να συνδυαστεί με select, delete, insert, update

with onoma **as** (**query_of_with** select/delete/ update…)

query_main select/delete/ update…

SQL-DML with… 1-2

```
with max_budget (value) as
   (select max(budget) from department)
select D.dept_name
from department as D, max_budget as M
where D.budget = M.value ;
```

```
myunipi=# with max_budget (value) as
myunipi-# (select max(budget) from department)
myunipi-# select D.dept_name
myunipi-# from department as D, max_budget as M
myunipi-# where D.budget = M.value ;
+------------+
| dept_name  |
+------------+
| Comp. Sci. |
+------------+
(1 row)
```

(π.χ)

```
/*Παρατηρείτε διαφορά στην ονομασία ; */
with max_budget as
   (select max(budget) as tata from department)
select D.dept_name
from department as D, max_budget as M
where D.budget = M.tata ;
```

```
with max_budget as
   (select budget from department
         where dept_name='Biology')
select D.dept_name
from department as D, max_budget as M
where D.budget = M.budget ;
```

SQL-DML with… 1-3

➢ Μπορεί να συνδυαστεί με select, delete, insert, update

(π.χ)

```
with moved_rows as (
/*εσωτερικά οποιοδήποτε query*/
    delete from teaches where year > '2016' and year < '2018' returning * )

/*οποιοδήποτε query και εξωτερικά */
insert into teaches_log
select * from moved_rows;
```

SQL-DML views… 1-1

Core SQL

create view v **as** <query expression>;

Εφαρμογή

```
create view faculty as
select ID, name, dept_name from instructor where
dept_name='Comp. Sci.' ;
select * from faculty;
```

Postgres

```
myunipi=# \h create view
Command:     CREATE VIEW
Description: define a new view
Syntax:
CREATE [ OR REPLACE ] [ TEMP | TEMPORARY ] [ RECURSIVE ] VIEW name [ ( column_name [, ...] ) ]

    [ WITH ( view_option_name [= view_option_value] [, ... ] ) ]
    AS query
    [ WITH [ CASCADED | LOCAL ] CHECK OPTION ]

URL: https://www.postgresql.org/docs/13/sql-createview.html
```

SQL-DML views… 1-2

postgres (π.χ.)

```
create view departments_total_salary(dept_name, total_salary) as
select dept name, sum (salary)
from instructor
group by dept name;

select * from  departments_total_salary;
/*ονόματα στηλών*/
```

```
myunipi=# select * from departments_total_salary;
+------------+--------------+
| dept_name  | total_salary |
+------------+--------------+
|            |    100000.00 |
| Finance    |    170000.00 |
| History    |    122000.00 |
| Physics    |    182000.00 |
| Music      |     40000.00 |
| Comp. Sci. |    232000.00 |
| Biology    |     72000.00 |
| Elec. Eng. |     80000.00 |
+------------+--------------+
(8 rows)
```

SQL-DML views… 1-3

postgres (π.χ.)

drop view, alter view, create or replace

- Η διαγραφή μιας όψης πραγματοποιείται με το drop
  drop view [if exists] faculty **[cascade |restrict]**; /*διαγράφει την όψη [εφόσον υπάρχει] και εφόσον αναφέρεται διαγράφει και όσες όψεις εξαρτώνται από αυτή [cascade] ή όχι μόνον τη συγκεκριμένη [restrict]*/
- Η αλλαγή των ιδιοτήτων μιας όψης μπορεί να πραγματοποιηθεί με τη χρήση του
  alter view departments_total_salary **alter column** total_salary **set** salary_total**;**
  Η όψη χρησιμοποιείται σαν όνομα πίνακα.
- Το παρακάτω αντικαθιστά τον ορισμό της όψης
  create or replace view faculty **as**
  select ID, name, dept_name **from** instructor **where** dept_name='Music' ;

SQL-DML views… 1-4

postgres (π.χ.)

insert, update

Η ενημέρωση δεδομένων με τη χρήση όψης επιτρέπεται υπό προϋποθέσεις και κατά κανόνα θα πρέπει να αποφεύγεται.

- Στην περιοχή from αναφέρεται μόνον μια σχέση
- Στην περιοχή select περιέχονται μόνον χαρακτηριστικά της μίας και μοναδικής σχέσης και δεν υπάρχουν συναθροιστικές συναρτήσεις, εκφράσεις και το distinct
- Οποιοδήποτε χαρακτηριστικό δεν αναφέρεται στην περιοχή select θα τεθεί στην τιμή null. Δεν περιέχει περιορισμό not null και δεν μπορεί να είναι μέρος ενός primary key
- Δεν περιέχονται group by ή having ορισμοί.

SQL-DML views… 1-5

postgres (π.χ.)

```
create view faculty as
select ID, name, dept_name from instructor where
dept_name='Comp. Sci.' ;
select * from faculty;
```

```
myunipi=# select * from faculty;
+-------+-----------+------------+
|  id   |    name   | dept_name  |
+-------+-----------+------------+
| 10101 | Srinivasan | Comp. Sci. |
| 45565 | Katz       | Comp. Sci. |
| 83821 | Brandt     | Comp. Sci. |
+-------+-----------+------------+
(3 rows)
```

```
insert into faculty
values ('30765', 'Green', 'Music');

select * from faculty; /*ίδια αποτελέσματα*/
select * from instructor;
```

```
myunipi=# select * from instructor;
+-------+-----------+------------+-----------+
|  id   |    name   | dept_name  |  salary   |
+-------+-----------+------------+-----------+
| 10101 | Srinivasan | Comp. Sci. |  65000.00 |
| 12121 | Wu         | Finance    |  90000.00 |
| 15151 | Mozart     | Music      |  40000.00 |
| 22222 | Einstein   | Physics    |  95000.00 |
| 32343 | El Said    | History    |  60000.00 |
| 33456 | Gold       | Physics    |  87000.00 |
| 45565 | Katz       | Comp. Sci. |  75000.00 |
| 58583 | Califieri  | History    |  62000.00 |
| 76543 | Singh      | Finance    |  80000.00 |
| 76766 | Crick      | Biology    |  72000.00 |
| 83821 | Brandt     | Comp. Sci. |  92000.00 |
| 98345 | Kim        | Elec. Eng. |  80000.00 |
| 1     | rosa       |            | 100000.00 |
| 30765 | Green      | Music      |           |
+-------+-----------+------------+-----------+
(14 rows)
```

SQL-DML views… 1-6

```
myunipi=# select * from faculty;
+-------+------------+------------+
|  id   |    name    | dept_name  |
+-------+------------+------------+
| 10101 | Srinivasan | Comp. Sci. |
| 45565 | Katz       | Comp. Sci. |
| 83821 | Brandt     | Comp. Sci. |
+-------+------------+------------+
(3 rows)
```

postgres (π.χ.)

```
create view faculty as
select ID, name, dept_name from instructor where
dept_name='Comp. Sci.' ;
select * from faculty;

create view faculty_name as
select * from faculty
where name  like 'K%'
with local check option;
/*σε κάθε insert or update θα ελεγχθεί μόνο η συνθήκη name  like 'K%' */
create view faculty_name as
select * from faculty
where name  like 'K%'
with cascaded check option;
/*σε κάθε insert or update θα ελεγχθεί η συνθήκη name  like 'K%'  αλλά και η
dept_name='Comp. Sci.' */
```

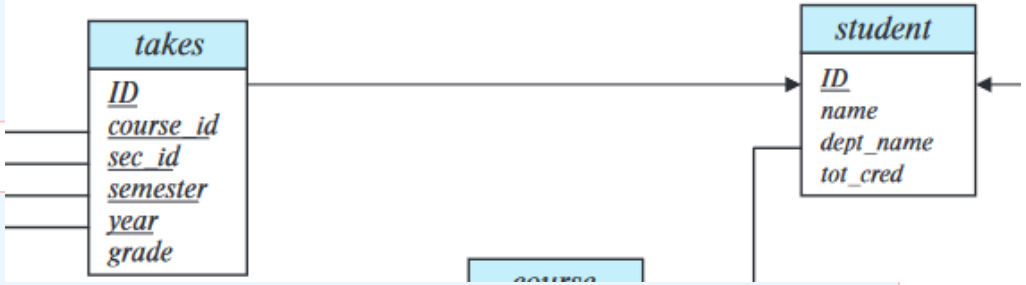
SQL-DML joins 1-1

- Μια λειτουργία SQL JOIN συνδυάζει γραμμές από δύο ή περισσότερους πίνακες.
- Δημιουργεί ένα σύνολο γραμμών σε έναν προσωρινό πίνακα.
- Μια λειτουργία JOIN επενεργεί σε δύο ή περισσότερους πίνακες, εάν έχουν τουλάχιστον ένα **κοινό χαρακτηριστικό** (στήλη) και **έχουν σχέση** μεταξύ τους (relationship).
- Η λειτουργία JOIN διατηρεί τους βασικούς πίνακες (δομή και δεδομένα) αμετάβλητους.

SQL-DML joins 1-2

Core SQL

```
select  […]
from
first_table join_type second_table [ join_condition ]
where […..];
```



Εφαρμογή

```
select name, course id
from student natural join takes;
/*ή με τη χρήση του on να γράψω τη σύνδεση */
select *
from student join takes on student.ID = takes.ID;
```

SQL-DML joins 1-3

join τύποι postgres

- **cross** join (καρτεσιανό γινόμενο)
- **inner** join (τομή)
- **left outer** join (αριστερός πίνακας + τομή)
- **right outer** join (δεξιός πίνακας + τομή)
- **full outer** join (ένωση)

table1

numid integer
12
14
10
11

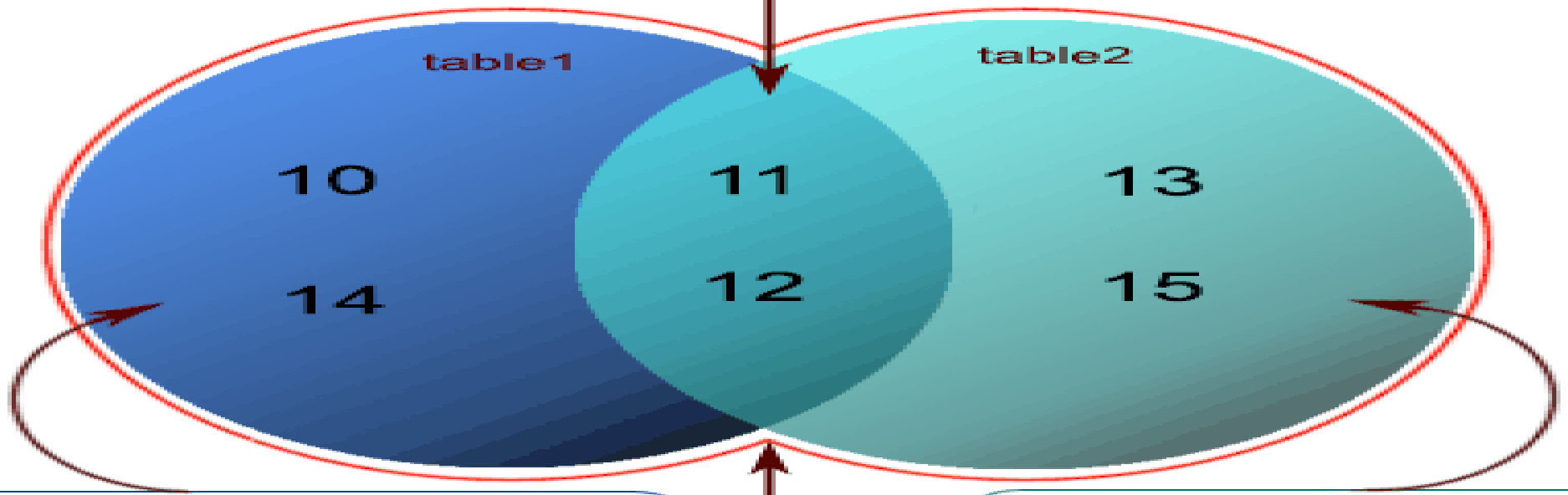
INNER JOIN

```
select * from table1
inner join table2
on table1.numid=table2.numid;
```

Result : [11,12]

table2

numid integer
13
15
11
12




```
select * from table1
left outer join table2
on table1.numid=table2.numid;
```

Result : [10,14,11,12]

LEFT OUTER JOIN

```
select * from table1
right outer join table2
on table1.numid=table2.numid;
```

Result : [11,12,13,15]

RIGHT OUTER JOIN

```
select * from table1
full outer join table2
on table1.numid=table2.numid;
```

Result : [10,14,11,12,13,15]

FULL OUTER JOIN

SQL-DML joins 1-5

join postgres (π.χ.)

```
myunipi=# select * from student;
+-------+----------+------------+----------+
|  id   |   name   | dept_name  | tot_cred |
+-------+----------+------------+----------+
| 00128 | Zhang    | Comp. Sci. |      102 |
| 12345 | Shankar  | Comp. Sci. |       32 |
| 19991 | Brandt   | History    |       80 |
```

```
myunipi=# select * from takes;
+-------+-----------+--------+----------+------+-------+
|  id   | course_id | sec_id | semester | year | grade |
+-------+-----------+--------+----------+------+-------+
| 00128 | CS-101    | 1      | Fall     | 2017 | A     |
| 00128 | CS-347    | 1      | Fall     | 2017 | A-    |
| 12345 | CS-101    | 1      | Fall     | 2017 | C     |
| 12345 | CS-190    | 2      | Spring   | 2017 | A     |
```

select * **from** student **cross join** takes**;**

/*ισοδύναμο με */ **select** * **from** student**,** takes**;**

```
myunipi=# select *
myunipi-# from student cross join takes ;
+-------+----------+------------+----------+-------+-----------+--------+----------+------+-------+
|  id   |   name   | dept_name  | tot_cred |  id   | course_id | sec_id | semester | year | grade |
+-------+----------+------------+----------+-------+-----------+--------+----------+------+-------+
| 00128 | Zhang    | Comp. Sci. |      102 | 00128 | CS-101    | 1      | Fall     | 2017 | A     |
| 12345 | Shankar  | Comp. Sci. |       32 | 00128 | CS-101    | 1      | Fall     | 2017 | A     |
| 19991 | Brandt   | History    |       80 | 00128 | CS-101    | 1      | Fall     | 2017 | A     |
```

SQL-DML joins 1-6

join postgres (π.χ.)

```
myunipi=# select * from student;
+-------+----------+------------+----------+
|  id   |   name   | dept_name  | tot_cred |
+-------+----------+------------+----------+
| 00128 | Zhang    | Comp. Sci. |      102 |
| 12345 | Shankar  | Comp. Sci. |       32 |
| 19991 | Brandt   | History    |       80 |
| 23121 | Chavez   | Finance    |      110 |
| 44553 | Peltier  | Physics    |       56 |
| 45678 | Levy     | Physics    |       46 |
| 54321 | Williams | Comp. Sci. |       54 |
| 55739 | Sanchez  | Music      |       38 |
| 70557 | Snow     | Physics    |        0 |
| 76543 | Brown    | Comp. Sci. |       58 |
| 76653 | Aoi      | Elec. Eng. |       60 |
| 98765 | Bourikas | Elec. Eng. |       98 |
| 98988 | Tanaka   | Biology    |      120 |
+-------+----------+------------+----------+
(13 rows)
```

```
myunipi=# select * from takes;
+-------+-----------+--------+----------+------+-------+
|  id   | course_id | sec_id | semester | year | grade |
+-------+-----------+--------+----------+------+-------+
| 00128 | CS-101    | 1      | Fall     | 2017 | A     |
| 00128 | CS-347    | 1      | Fall     | 2017 | A-    |
| 12345 | CS-101    | 1      | Fall     | 2017 | C     |
| 12345 | CS-190    | 2      | Spring   | 2017 | A     |
| 12345 | CS-315    | 1      | Spring   | 2018 | A     |
| 12345 | CS-347    | 1      | Fall     | 2017 | A     |
| 19991 | HIS-351   | 1      | Spring   | 2018 | B     |
| 23121 | FIN-201   | 1      | Spring   | 2018 | C+    |
| 44553 | PHY-101   | 1      | Fall     | 2017 | B-    |
| 45678 | CS-101    | 1      | Fall     | 2017 | F     |
| 45678 | CS-101    | 1      | Spring   | 2018 | B+    |
| 45678 | CS-319    | 1      | Spring   | 2018 | B     |
| 54321 | CS-101    | 1      | Fall     | 2017 | A-    |
| 54321 | CS-190    | 2      | Spring   | 2017 | B+    |
| 55739 | MU-199    | 1      | Spring   | 2018 | A-    |
| 76543 | CS-101    | 1      | Fall     | 2017 | A     |
| 76543 | CS-319    | 2      | Spring   | 2018 | A     |
| 76653 | EE-181    | 1      | Spring   | 2017 | C     |
| 98765 | CS-101    | 1      | Fall     | 2017 | C-    |
| 98765 | CS-315    | 1      | Spring   | 2018 | B     |
| 98988 | BIO-101   | 1      | Summer   | 2017 | A     |
| 98988 | BIO-301   | 1      | Summer   | 2018 |       |
+-------+-----------+--------+----------+------+-------+
(22 rows)
```

```
myunipi=# select * from student inner join takes on student.id=takes.id;
+-------+----------+------------+----------+-------+-----------+--------+----------+------+-------+
|  id   |   name   | dept_name  | tot_cred |  id   | course_id | sec_id | semester | year | grade |
+-------+----------+------------+----------+-------+-----------+--------+----------+------+-------+
| 00128 | Zhang    | Comp. Sci. |      102 | 00128 | CS-101    | 1      | Fall     | 2017 | A     |
| 00128 | Zhang    | Comp. Sci. |      102 | 00128 | CS-347    | 1      | Fall     | 2017 | A-    |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-101    | 1      | Fall     | 2017 | C     |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-190    | 2      | Spring   | 2017 | A     |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-315    | 1      | Spring   | 2018 | A     |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-347    | 1      | Fall     | 2017 | A     |
| 19991 | Brandt   | History    |       80 | 19991 | HIS-351   | 1      | Spring   | 2018 | B     |
| 23121 | Chavez   | Finance    |      110 | 23121 | FIN-201   | 1      | Spring   | 2018 | C+    |
| 44553 | Peltier  | Physics    |       56 | 44553 | PHY-101   | 1      | Fall     | 2017 | B-    |
| 45678 | Levy     | Physics    |       46 | 45678 | CS-101    | 1      | Fall     | 2017 | F     |
| 45678 | Levy     | Physics    |       46 | 45678 | CS-101    | 1      | Spring   | 2018 | B+    |
| 45678 | Levy     | Physics    |       46 | 45678 | CS-319    | 1      | Spring   | 2018 | B     |
| 54321 | Williams | Comp. Sci. |       54 | 54321 | CS-101    | 1      | Fall     | 2017 | A-    |
| 54321 | Williams | Comp. Sci. |       54 | 54321 | CS-190    | 2      | Spring   | 2017 | B+    |
| 55739 | Sanchez  | Music      |       38 | 55739 | MU-199    | 1      | Spring   | 2018 | A-    |
| 76543 | Brown    | Comp. Sci. |       58 | 76543 | CS-101    | 1      | Fall     | 2017 | A     |
| 76543 | Brown    | Comp. Sci. |       58 | 76543 | CS-319    | 2      | Spring   | 2018 | A     |
| 76653 | Aoi      | Elec. Eng. |       60 | 76653 | EE-181    | 1      | Spring   | 2017 | C     |
| 98765 | Bourikas | Elec. Eng. |       98 | 98765 | CS-101    | 1      | Fall     | 2017 | C-    |
| 98765 | Bourikas | Elec. Eng. |       98 | 98765 | CS-315    | 1      | Spring   | 2018 | B     |
| 98988 | Tanaka   | Biology    |      120 | 98988 | BIO-101   | 1      | Summer   | 2017 | A     |
| 98988 | Tanaka   | Biology    |      120 | 98988 | BIO-301   | 1      | Summer   | 2018 |       |
+-------+----------+------------+----------+-------+-----------+--------+----------+------+-------+
(22 rows)
```

select * **from** student **inner join** takes **on** student.id=takes.id;

SQL-DML joins 1-7

join postgres
(π.χ.)

```
myunipi=# select * from student;
+-------+----------+------------+----------+
|  id   |   name   | dept_name  | tot_cred |
+-------+----------+------------+----------+
| 00128 | Zhang    | Comp. Sci. |      102 |
| 12345 | Shankar  | Comp. Sci. |       32 |
| 19991 | Brandt   | History    |       80 |
| 23121 | Chavez   | Finance    |      110 |
| 44553 | Peltier  | Physics    |       56 |
| 45678 | Levy     | Physics    |       46 |
| 54321 | Williams | Comp. Sci. |       54 |
| 55739 | Sanchez  | Music      |       38 |
| 70557 | Snow     | Physics    |        0 |
| 76543 | Brown    | Comp. Sci. |       58 |
| 76653 | Aoi      | Elec. Eng. |       60 |
| 98765 | Bourikas | Elec. Eng. |       98 |
| 98988 | Tanaka   | Biology    |      120 |
+-------+----------+------------+----------+
(13 rows)
```

```
myunipi=# select * from takes;
+-------+-----------+--------+----------+------+-------+
|  id   | course_id | sec_id | semester | year | grade |
+-------+-----------+--------+----------+------+-------+
| 00128 | CS-101    | 1      | Fall     | 2017 | A     |
| 00128 | CS-347    | 1      | Fall     | 2017 | A-    |
| 12345 | CS-101    | 1      | Fall     | 2017 | C     |
| 12345 | CS-190    | 2      | Spring   | 2017 | A     |
| 12345 | CS-315    | 1      | Spring   | 2018 | A     |
| 12345 | CS-347    | 1      | Fall     | 2017 | A     |
| 19991 | HIS-351   | 1      | Spring   | 2018 | B     |
| 23121 | FIN-201   | 1      | Spring   | 2018 | C+    |
| 44553 | PHY-101   | 1      | Fall     | 2017 | B-    |
| 45678 | CS-101    | 1      | Fall     | 2017 | F     |
| 45678 | CS-101    | 1      | Spring   | 2018 | B+    |
| 45678 | CS-319    | 1      | Spring   | 2018 | B     |
| 54321 | CS-101    | 1      | Fall     | 2017 | A-    |
| 54321 | CS-190    | 2      | Spring   | 2017 | B+    |
| 55739 | MU-199    | 1      | Spring   | 2018 | A-    |
| 76543 | CS-101    | 1      | Fall     | 2017 | A     |
| 76543 | CS-319    | 2      | Spring   | 2018 | A     |
| 76653 | EE-181    | 1      | Spring   | 2017 | C     |
| 98765 | CS-101    | 1      | Fall     | 2017 | C-    |
| 98765 | CS-315    | 1      | Spring   | 2018 | B     |
| 98988 | BIO-101   | 1      | Summer   | 2017 | A     |
| 98988 | BIO-301   | 1      | Summer   | 2018 |       |
+-------+-----------+--------+----------+------+-------+
(22 rows)
```

```
myunipi=# select * from student left outer join takes on student.id=takes.id;
+-------+----------+------------+----------+-------+-----------+--------+----------+------+-------+
|  id   |   name   | dept_name  | tot_cred |  id   | course_id | sec_id | semester | year | grade |
+-------+----------+------------+----------+-------+-----------+--------+----------+------+-------+
| 00128 | Zhang    | Comp. Sci. |      102 | 00128 | CS-101    | 1      | Fall     | 2017 | A     |
| 00128 | Zhang    | Comp. Sci. |      102 | 00128 | CS-347    | 1      | Fall     | 2017 | A-    |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-101    | 1      | Fall     | 2017 | C     |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-190    | 2      | Spring   | 2017 | A     |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-315    | 1      | Spring   | 2018 | A     |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-347    | 1      | Fall     | 2017 | A     |
| 19991 | Brandt   | History    |       80 | 19991 | HIS-351   | 1      | Spring   | 2018 | B     |
| 23121 | Chavez   | Finance    |      110 | 23121 | FIN-201   | 1      | Spring   | 2018 | C+    |
| 44553 | Peltier  | Physics    |       56 | 44553 | PHY-101   | 1      | Fall     | 2017 | B-    |
| 45678 | Levy     | Physics    |       46 | 45678 | CS-101    | 1      | Fall     | 2017 | F     |
| 45678 | Levy     | Physics    |       46 | 45678 | CS-101    | 1      | Spring   | 2018 | B+    |
| 45678 | Levy     | Physics    |       46 | 45678 | CS-319    | 1      | Spring   | 2018 | B     |
| 54321 | Williams | Comp. Sci. |       54 | 54321 | CS-101    | 1      | Fall     | 2017 | A-    |
| 54321 | Williams | Comp. Sci. |       54 | 54321 | CS-190    | 2      | Spring   | 2017 | B+    |
| 55739 | Sanchez  | Music      |       38 | 55739 | MU-199    | 1      | Spring   | 2018 | A-    |
| 76543 | Brown    | Comp. Sci. |       58 | 76543 | CS-101    | 1      | Fall     | 2017 | A     |
| 76543 | Brown    | Comp. Sci. |       58 | 76543 | CS-319    | 2      | Spring   | 2018 | A     |
| 76653 | Aoi      | Elec. Eng. |       60 | 76653 | EE-181    | 1      | Spring   | 2017 | C     |
| 98765 | Bourikas | Elec. Eng. |       98 | 98765 | CS-101    | 1      | Fall     | 2017 | C-    |
| 98765 | Bourikas | Elec. Eng. |       98 | 98765 | CS-315    | 1      | Spring   | 2018 | B     |
| 98988 | Tanaka   | Biology    |      120 | 98988 | BIO-101   | 1      | Summer   | 2017 | A     |
| 98988 | Tanaka   | Biology    |      120 | 98988 | BIO-301   | 1      | Summer   | 2018 |       |
| 70557 | Snow     | Physics    |        0 |       |           |        |          |      |       |
+-------+----------+------------+----------+-------+-----------+--------+----------+------+-------+
(23 rows)
```

select * **from** student **left outer join** takes
on student.id=takes.id;

SQL-DML joins 1-8

join postgres (π.χ.)

```
myunipi=# select * from student;
+-------+----------+------------+----------+
|  id   |   name   | dept_name  | tot_cred |
+-------+----------+------------+----------+
| 00128 | Zhang    | Comp. Sci. |      102 |
| 12345 | Shankar  | Comp. Sci. |       32 |
| 19991 | Brandt   | History    |       80 |
| 23121 | Chavez   | Finance    |      110 |
| 44553 | Peltier  | Physics    |       56 |
| 45678 | Levy     | Physics    |       46 |
| 54321 | Williams | Comp. Sci. |       54 |
| 55739 | Sanchez  | Music      |       38 |
| 70557 | Snow     | Physics    |        0 |
| 76543 | Brown    | Comp. Sci. |       58 |
| 76653 | Aoi      | Elec. Eng. |       60 |
| 98765 | Bourikas | Elec. Eng. |       98 |
| 98988 | Tanaka   | Biology    |      120 |
+-------+----------+------------+----------+
(13 rows)
```

```
myunipi=# select * from takes;
+-------+-----------+--------+----------+------+-------+
|  id   | course_id | sec_id | semester | year | grade |
+-------+-----------+--------+----------+------+-------+
| 00128 | CS-101    | 1      | Fall     | 2017 | A     |
| 00128 | CS-347    | 1      | Fall     | 2017 | A-    |
| 12345 | CS-101    | 1      | Fall     | 2017 | C     |
| 12345 | CS-190    | 2      | Spring   | 2017 | A     |
| 12345 | CS-315    | 1      | Spring   | 2018 | A     |
| 12345 | CS-347    | 1      | Fall     | 2017 | A     |
| 19991 | HIS-351   | 1      | Spring   | 2018 | B     |
| 23121 | FIN-201   | 1      | Spring   | 2018 | C+    |
| 44553 | PHY-101   | 1      | Fall     | 2017 | B-    |
| 45678 | CS-101    | 1      | Fall     | 2017 | F     |
| 45678 | CS-101    | 1      | Spring   | 2018 | B+    |
| 45678 | CS-319    | 1      | Spring   | 2018 | B     |
| 54321 | CS-101    | 1      | Fall     | 2017 | A-    |
| 54321 | CS-190    | 2      | Spring   | 2017 | B+    |
| 55739 | MU-199    | 1      | Spring   | 2018 | A-    |
| 76543 | CS-101    | 1      | Fall     | 2017 | A     |
| 76543 | CS-319    | 2      | Spring   | 2018 | A     |
| 76653 | EE-181    | 1      | Spring   | 2017 | C     |
| 98765 | CS-101    | 1      | Fall     | 2017 | C-    |
| 98765 | CS-315    | 1      | Spring   | 2018 | B     |
| 98988 | BIO-101   | 1      | Summer   | 2017 | A     |
| 98988 | BIO-301   | 1      | Summer   | 2018 |       |
+-------+-----------+--------+----------+------+-------+
(22 rows)
```

```
myunipi=# select * from student inner join takes on student.id=takes.id;
+-------+----------+------------+----------+-------+-----------+--------+----------+------+-------+
|  id   |   name   | dept_name  | tot_cred |  id   | course_id | sec_id | semester | year | grade |
+-------+----------+------------+----------+-------+-----------+--------+----------+------+-------+
| 00128 | Zhang    | Comp. Sci. |      102 | 00128 | CS-101    | 1      | Fall     | 2017 | A     |
| 00128 | Zhang    | Comp. Sci. |      102 | 00128 | CS-347    | 1      | Fall     | 2017 | A-    |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-101    | 1      | Fall     | 2017 | C     |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-190    | 2      | Spring   | 2017 | A     |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-315    | 1      | Spring   | 2018 | A     |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-347    | 1      | Fall     | 2017 | A     |
| 19991 | Brandt   | History    |       80 | 19991 | HIS-351   | 1      | Spring   | 2018 | B     |
| 23121 | Chavez   | Finance    |      110 | 23121 | FIN-201   | 1      | Spring   | 2018 | C+    |
| 44553 | Peltier  | Physics    |       56 | 44553 | PHY-101   | 1      | Fall     | 2017 | B-    |
| 45678 | Levy     | Physics    |       46 | 45678 | CS-101    | 1      | Fall     | 2017 | F     |
| 45678 | Levy     | Physics    |       46 | 45678 | CS-101    | 1      | Spring   | 2018 | B+    |
| 45678 | Levy     | Physics    |       46 | 45678 | CS-319    | 1      | Spring   | 2018 | B     |
| 54321 | Williams | Comp. Sci. |       54 | 54321 | CS-101    | 1      | Fall     | 2017 | A-    |
| 54321 | Williams | Comp. Sci. |       54 | 54321 | CS-190    | 2      | Spring   | 2017 | B+    |
| 55739 | Sanchez  | Music      |       38 | 55739 | MU-199    | 1      | Spring   | 2018 | A-    |
| 76543 | Brown    | Comp. Sci. |       58 | 76543 | CS-101    | 1      | Fall     | 2017 | A     |
| 76543 | Brown    | Comp. Sci. |       58 | 76543 | CS-319    | 2      | Spring   | 2018 | A     |
| 76653 | Aoi      | Elec. Eng. |       60 | 76653 | EE-181    | 1      | Spring   | 2017 | C     |
| 98765 | Bourikas | Elec. Eng. |       98 | 98765 | CS-101    | 1      | Fall     | 2017 | C-    |
| 98765 | Bourikas | Elec. Eng. |       98 | 98765 | CS-315    | 1      | Spring   | 2018 | B     |
| 98988 | Tanaka   | Biology    |      120 | 98988 | BIO-101   | 1      | Summer   | 2017 | A     |
| 98988 | Tanaka   | Biology    |      120 | 98988 | BIO-301   | 1      | Summer   | 2018 |       |
+-------+----------+------------+----------+-------+-----------+--------+----------+------+-------+
(22 rows)
```

select * from student **right outer join** takes **on** student.id=takes.id;

/*Γιατί στο συγκεκριμένο παράδειγμα δίνει το ίδιο αποτέλεσμα με το inner join;
*/

SQL-DML joins 1-9

join postgres
(π.χ.)

```
myunipi=# select * from student;
+-------+----------+------------+----------+
|  id   |   name   | dept_name  | tot_cred |
+-------+----------+------------+----------+
| 00128 | Zhang    | Comp. Sci. |      102 |
| 12345 | Shankar  | Comp. Sci. |       32 |
| 19991 | Brandt   | History    |       80 |
| 23121 | Chavez   | Finance    |      110 |
| 44553 | Peltier  | Physics    |       56 |
| 45678 | Levy     | Physics    |       46 |
| 54321 | Williams | Comp. Sci. |       54 |
| 55739 | Sanchez  | Music      |       38 |
| 70557 | Snow     | Physics    |        0 |
| 76543 | Brown    | Comp. Sci. |       58 |
| 76653 | Aoi      | Elec. Eng. |       60 |
| 98765 | Bourikas | Elec. Eng. |       98 |
| 98988 | Tanaka   | Biology    |      120 |
+-------+----------+------------+----------+
(13 rows)
```

```
myunipi=# select * from takes;
+-------+-----------+--------+----------+------+-------+
|  id   | course_id | sec_id | semester | year | grade |
+-------+-----------+--------+----------+------+-------+
| 00128 | CS-101    | 1      | Fall     | 2017 | A     |
| 00128 | CS-347    | 1      | Fall     | 2017 | A-    |
| 12345 | CS-101    | 1      | Fall     | 2017 | C     |
| 12345 | CS-190    | 2      | Spring   | 2017 | A     |
| 12345 | CS-315    | 1      | Spring   | 2018 | A     |
| 12345 | CS-347    | 1      | Fall     | 2017 | A     |
| 19991 | HIS-351   | 1      | Spring   | 2018 | B     |
| 23121 | FIN-201   | 1      | Spring   | 2018 | C+    |
| 44553 | PHY-101   | 1      | Fall     | 2017 | B-    |
| 45678 | CS-101    | 1      | Fall     | 2017 | F     |
| 45678 | CS-101    | 1      | Spring   | 2018 | B+    |
| 45678 | CS-319    | 1      | Spring   | 2018 | B     |
| 54321 | CS-101    | 1      | Fall     | 2017 | A-    |
| 54321 | CS-190    | 2      | Spring   | 2017 | B+    |
| 55739 | MU-199    | 1      | Spring   | 2018 | A-    |
| 76543 | CS-101    | 1      | Fall     | 2017 | A     |
| 76543 | CS-319    | 2      | Spring   | 2018 | A     |
| 76653 | EE-181    | 1      | Spring   | 2017 | C     |
| 98765 | CS-101    | 1      | Fall     | 2017 | C-    |
| 98765 | CS-315    | 1      | Spring   | 2018 | B     |
| 98988 | BIO-101   | 1      | Summer   | 2017 | A     |
| 98988 | BIO-301   | 1      | Summer   | 2018 |       |
+-------+-----------+--------+----------+------+-------+
(22 rows)
```

```
myunipi=# select * from student left outer join takes on student.id=takes.id;
+-------+----------+------------+----------+-------+-----------+--------+----------+------+-------+
|  id   |   name   | dept_name  | tot_cred |  id   | course_id | sec_id | semester | year | grade |
+-------+----------+------------+----------+-------+-----------+--------+----------+------+-------+
| 00128 | Zhang    | Comp. Sci. |      102 | 00128 | CS-101    | 1      | Fall     | 2017 | A     |
| 00128 | Zhang    | Comp. Sci. |      102 | 00128 | CS-347    | 1      | Fall     | 2017 | A-    |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-101    | 1      | Fall     | 2017 | C     |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-190    | 2      | Spring   | 2017 | A     |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-315    | 1      | Spring   | 2018 | A     |
| 12345 | Shankar  | Comp. Sci. |       32 | 12345 | CS-347    | 1      | Fall     | 2017 | A     |
| 19991 | Brandt   | History    |       80 | 19991 | HIS-351   | 1      | Spring   | 2018 | B     |
| 23121 | Chavez   | Finance    |      110 | 23121 | FIN-201   | 1      | Spring   | 2018 | C+    |
| 44553 | Peltier  | Physics    |       56 | 44553 | PHY-101   | 1      | Fall     | 2017 | B-    |
| 45678 | Levy     | Physics    |       46 | 45678 | CS-101    | 1      | Fall     | 2017 | F     |
| 45678 | Levy     | Physics    |       46 | 45678 | CS-101    | 1      | Spring   | 2018 | B+    |
| 45678 | Levy     | Physics    |       46 | 45678 | CS-319    | 1      | Spring   | 2018 | B     |
| 54321 | Williams | Comp. Sci. |       54 | 54321 | CS-101    | 1      | Fall     | 2017 | A-    |
| 54321 | Williams | Comp. Sci. |       54 | 54321 | CS-190    | 2      | Spring   | 2017 | B+    |
| 55739 | Sanchez  | Music      |       38 | 55739 | MU-199    | 1      | Spring   | 2018 | A-    |
| 76543 | Brown    | Comp. Sci. |       58 | 76543 | CS-101    | 1      | Fall     | 2017 | A     |
| 76543 | Brown    | Comp. Sci. |       58 | 76543 | CS-319    | 2      | Spring   | 2018 | A     |
| 76653 | Aoi      | Elec. Eng. |       60 | 76653 | EE-181    | 1      | Spring   | 2017 | C     |
| 98765 | Bourikas | Elec. Eng. |       98 | 98765 | CS-101    | 1      | Fall     | 2017 | C-    |
| 98765 | Bourikas | Elec. Eng. |       98 | 98765 | CS-315    | 1      | Spring   | 2018 | B     |
| 98988 | Tanaka   | Biology    |      120 | 98988 | BIO-101   | 1      | Summer   | 2017 | A     |
| 98988 | Tanaka   | Biology    |      120 | 98988 | BIO-301   | 1      | Summer   | 2018 |       |
| 70557 | Snow     | Physics    |        0 |       |           |        |          |      |       |
+-------+----------+------------+----------+-------+-----------+--------+----------+------+-------+
(23 rows)
```

select * **from** student **full outer join** takes
on student.id=takes.id;
/*Γιατί στο συγκεκριμένο παράδειγμα δίνει το ίδιο αποτέλεσμα με το left join; */

SQL-DML joins 1-10

join condition postgres

Συνθήκες σύνδεσης στη λειτουργία SQL JOIN

- **natural**
  Χρησιμοποιεί τα κοινά χαρακτηριστικά (στήλες) **χωρίς** αυτά να χρειάζεται να **αναγράφονται (υπονοεί σύνδεση ισότητας)**
- **on**
  Χρησιμοποιεί κοινά χαρακτηριστικά (στήλες) **τα οποία αναγράφονται ρητά στο query και συνδέονται με τελεστή σύγκρισης (=,>,< κλπ)**
- **using**
  Χρησιμοποιεί κοινά χαρακτηριστικά (στήλες) **τα οποία αναγράφονται ρητά στο query (υπονοεί σύνδεση ισότητας)**

SQL-DML joins 1-11

```
myunipi=# select * from student left outer join takes on student.id=takes.id;
```

id	name	dept_name	tot_cred	id	course_id	sec_id	semester	year	grade
00128	Zhang	Comp. Sci.	102	00128	CS-101	1	Fall	2017	A
00128	Zhang	Comp. Sci.	102	00128	CS-347	1	Fall	2017	A-
12345	Shankar	Comp. Sci.	32	12345	CS-101	1	Fall	2017	C
12345	Shankar	Comp. Sci.	32	12345	CS-190	2	Spring	2017	A
12345	Shankar	Comp. Sci.	32	12345	CS-315	1	Spring	2018	A
12345	Shankar	Comp. Sci.	32	12345	CS-347	1	Fall	2017	A
19991	Brandt	History	80	19991	HIS-351	1	Spring	2018	B
23121	Chavez	Finance	110	23121	FIN-201	1	Spring	2018	C+
44553	Peltier	Physics	56	44553	PHY-101	1	Fall	2017	B-
45678	Levy	Physics	46	45678	CS-101	1	Fall	2017	F
45678	Levy	Physics	46	45678	CS-101	1	Spring	2018	B+
45678	Levy	Physics	46	45678	CS-319	1	Spring	2018	B
54321	Williams	Comp. Sci.	54	54321	CS-101	1	Fall	2017	A-
54321	Williams	Comp. Sci.	54	54321	CS-190	2	Spring	2017	B+
55739	Sanchez	Music	38	55739	MU-199	1	Spring	2018	A-
76543	Brown	Comp. Sci.	58	76543	CS-101	1	Fall	2017	A
76543	Brown	Comp. Sci.	58	76543	CS-319	2	Spring	2018	A
76653	Aoi	Elec. Eng.	60	76653	EE-181	1	Spring	2017	C
98765	Bourikas	Elec. Eng.	98	98765	CS-101	1	Fall	2017	C-
98765	Bourikas	Elec. Eng.	98	98765	CS-315	1	Spring	2018	B
98988	Tanaka	Biology	120	98988	BIO-101	1	Summer	2017	A
98988	Tanaka	Biology	120	98988	BIO-301	1	Summer	2018	
70557	Snow	Physics	0						

```
myunipi=# select * from student left outer join takes using (id);
```

id	name	dept_name	tot_cred	course_id	sec_id	semester	year	grade
00128	Zhang	Comp. Sci.	102	CS-101	1	Fall	2017	A
00128	Zhang	Comp. Sci.	102	CS-347	1	Fall	2017	A-
12345	Shankar	Comp. Sci.	32	CS-101	1	Fall	2017	C
12345	Shankar	Comp. Sci.	32	CS-190	2	Spring	2017	A
12345	Shankar	Comp. Sci.	32	CS-315	1	Spring	2018	A
12345	Shankar	Comp. Sci.	32	CS-347	1	Fall	2017	A
19991	Brandt	History	80	HIS-351	1	Spring	2018	B
23121	Chavez	Finance	110	FIN-201	1	Spring	2018	C+
44553	Peltier	Physics	56	PHY-101	1	Fall	2017	B-
45678	Levy	Physics	46	CS-101	1	Fall	2017	F
45678	Levy	Physics	46	CS-101	1	Spring	2018	B+
45678	Levy	Physics	46	CS-319	1	Spring	2018	B
54321	Williams	Comp. Sci.	54	CS-101	1	Fall	2017	A-
54321	Williams	Comp. Sci.	54	CS-190	2	Spring	2017	B+
55739	Sanchez	Music	38	MU-199	1	Spring	2018	A-
76543	Brown	Comp. Sci.	58	CS-101	1	Fall	2017	A
76543	Brown	Comp. Sci.	58	CS-319	2	Spring	2018	A
76653	Aoi	Elec. Eng.	60	EE-181	1	Spring	2017	C
98765	Bourikas	Elec. Eng.	98	CS-101	1	Fall	2017	C-
98765	Bourikas	Elec. Eng.	98	CS-315	1	Spring	2018	B
98988	Tanaka	Biology	120	BIO-101	1	Summer	2017	A
98988	Tanaka	Biology	120	BIO-301	1	Summer	2018	
70557	Snow	Physics	0					

(23 rows)

```
myunipi=# select * from student;
```

id	name	dept_name	tot_cred
00128	Zhang	Comp. Sci.	102
12345	Shankar	Comp. Sci.	32
19991	Brandt	History	80
23121	Chavez	Finance	110
44553	Peltier	Physics	56
45678	Levy	Physics	46
54321	Williams	Comp. Sci.	54
55739	Sanchez	Music	38
70557	Snow	Physics	0
76543	Brown	Comp. Sci.	58
76653	Aoi	Elec. Eng.	60
98765	Bourikas	Elec. Eng.	98
98988	Tanaka	Biology	120

(13 rows)

```
myunipi=# select * from takes;
```

id	course_id	sec_id	semester	year	grade
00128	CS-101	1	Fall	2017	A
00128	CS-347	1	Fall	2017	A-
12345	CS-101	1	Fall	2017	C
12345	CS-190	2	Spring	2017	A
12345	CS-315	1	Spring	2018	A
12345	CS-347	1	Fall	2017	A
19991	HIS-351	1	Spring	2018	B
23121	FIN-201	1	Spring	2018	C+
44553	PHY-101	1	Fall	2017	B-
45678	CS-101	1	Fall	2017	F
45678	CS-101	1	Spring	2018	B+
45678	CS-319	1	Spring	2018	B
54321	CS-101	1	Fall	2017	A-
54321	CS-190	2	Spring	2017	B+
55739	MU-199	1	Spring	2018	A-
76543	CS-101	1	Fall	2017	A
76543	CS-319	2	Spring	2018	A
76653	EE-181	1	Spring	2017	C
98765	CS-101	1	Fall	2017	C-
98765	CS-315	1	Spring	2018	B
98988	BIO-101	1	Summer	2017	A
98988	BIO-301	1	Summer	2018	

(22 rows)

select * from student
left outer join takes **on**
student.id=takes.id;

/*Ίδιο αποτέλεσμα. Διαφορετικές στήλες παρουσίασης */

select * from student
left outer join takes
using (id);

select * from student
natural left outer join
takes;

SQL-DML joins 1-12

join postgres (π.χ.)

```
myunipi=# select * from student;
+-------+----------+------------+----------+
|  id   |   name   | dept_name  | tot_cred |
+-------+----------+------------+----------+
| 00128 | Zhang    | Comp. Sci. |      102 |
| 12345 | Shankar  | Comp. Sci. |       32 |
| 19991 | Brandt   | History    |       80 |
| 23121 | Chavez   | Finance    |      110 |
| 44553 | Peltier  | Physics    |       56 |
| 45678 | Levy     | Physics    |       46 |
| 54321 | Williams | Comp. Sci. |       54 |
| 55739 | Sanchez  | Music      |       38 |
| 70557 | Snow     | Physics    |        0 |
| 76543 | Brown    | Comp. Sci. |       58 |
| 76653 | Aoi      | Elec. Eng. |       60 |
| 98765 | Bourikas | Elec. Eng. |       98 |
| 98988 | Tanaka   | Biology    |      120 |
+-------+----------+------------+----------+
(13 rows)
```

```
myunipi=# select * from takes;
+-------+-----------+--------+----------+------+-------+
|  id   | course_id | sec_id | semester | year | grade |
+-------+-----------+--------+----------+------+-------+
| 00128 | CS-101    | 1      | Fall     | 2017 | A     |
| 00128 | CS-347    | 1      | Fall     | 2017 | A-    |
| 12345 | CS-101    | 1      | Fall     | 2017 | C     |
| 12345 | CS-190    | 2      | Spring   | 2017 | A     |
| 12345 | CS-315    | 1      | Spring   | 2018 | A     |
| 12345 | CS-347    | 1      | Fall     | 2017 | A     |
| 19991 | HIS-351   | 1      | Spring   | 2018 | B     |
| 23121 | FIN-201   | 1      | Spring   | 2018 | C+    |
| 44553 | PHY-101   | 1      | Fall     | 2017 | B-    |
| 45678 | CS-101    | 1      | Fall     | 2017 | F     |
| 45678 | CS-101    | 1      | Spring   | 2018 | B+    |
| 45678 | CS-319    | 1      | Spring   | 2018 | B     |
| 54321 | CS-101    | 1      | Fall     | 2017 | A-    |
| 54321 | CS-190    | 2      | Spring   | 2017 | B+    |
| 55739 | MU-199    | 1      | Spring   | 2018 | A-    |
| 76543 | CS-101    | 1      | Fall     | 2017 | A     |
| 76543 | CS-319    | 2      | Spring   | 2018 | A     |
| 76653 | EE-181    | 1      | Spring   | 2017 | C     |
| 98765 | CS-101    | 1      | Fall     | 2017 | C-    |
| 98765 | CS-315    | 1      | Spring   | 2018 | B     |
| 98988 | BIO-101   | 1      | Summer   | 2017 | A     |
| 98988 | BIO-301   | 1      | Summer   | 2018 |       |
+-------+-----------+--------+----------+------+-------+
(22 rows)
```

```
myunipi=# select student.*,takes.course_id
myunipi-# from student full outer join takes on student.id=takes.id
myunipi-# where takes.course_id like 'CS%';
+-------+----------+------------+----------+-----------+
|  id   |   name   | dept_name  | tot_cred | course_id |
+-------+----------+------------+----------+-----------+
| 00128 | Zhang    | Comp. Sci. |      102 | CS-101    |
| 00128 | Zhang    | Comp. Sci. |      102 | CS-347    |
| 12345 | Shankar  | Comp. Sci. |       32 | CS-101    |
| 12345 | Shankar  | Comp. Sci. |       32 | CS-190    |
| 12345 | Shankar  | Comp. Sci. |       32 | CS-315    |
| 12345 | Shankar  | Comp. Sci. |       32 | CS-347    |
| 45678 | Levy     | Physics    |       46 | CS-101    |
| 45678 | Levy     | Physics    |       46 | CS-101    |
| 45678 | Levy     | Physics    |       46 | CS-319    |
| 54321 | Williams | Comp. Sci. |       54 | CS-101    |
| 54321 | Williams | Comp. Sci. |       54 | CS-190    |
| 76543 | Brown    | Comp. Sci. |       58 | CS-101    |
| 76543 | Brown    | Comp. Sci. |       58 | CS-319    |
| 98765 | Bourikas | Elec. Eng. |       98 | CS-101    |
| 98765 | Bourikas | Elec. Eng. |       98 | CS-315    |
+-------+----------+------------+----------+-----------+
```

select student.*,takes.course_id
from student **full outer join** takes **on** student.id=takes.id
where takes.course_id **like** 'CS%';

/* σειρά εκτέλεσης:
1. join, 2. where, 3. select */

SQL-DML joins 1-13

Join vs Subquery

- Τα JOINs είναι ταχύτερα από τα υπο ερωτήματα (subqueries). Το αντίθετο είναι πολύ σπάνιο.
- Στα JOINs το RDBMS υπολογίζει ένα σχέδιο εκτέλεσης, το οποίο μπορεί να προβλέψει, ποια δεδομένα πρέπει να φορτωθούν και πόσο χρόνο χρειάζεται η επεξεργασία τους. Στα υπο ερωτήματα δεν υφίσταται κάτι τέτοιο.
- Ένα JOIN ελέγχεται πρώτα η συνθήκη και στη συνέχεια εφαρμόζεται σε πίνακα. Στα υπο ερωτήματα πρώτα δημιουργείται εσωτερικός προσωρινός πίνακας και στη συνέχεια εφαρμόζονται οι συνθήκες.
- Όταν χρησιμοποιούνται JOINs, θα πρέπει να υπάρχει σύνδεση μεταξύ δύο ή περισσότερων από δύο πινάκων και κάθε πίνακας έχει σχέση με άλλους, ενώ στα υπο ερωτήματα (δηλαδή στα ερώτημα μέσα σε άλλο ερώτημα), δεν χρειάζεται να υφίσταται σχέση, καθώς λειτουργούν σε στήλες και συνθήκες.

πηγή:
comic.browserling.com