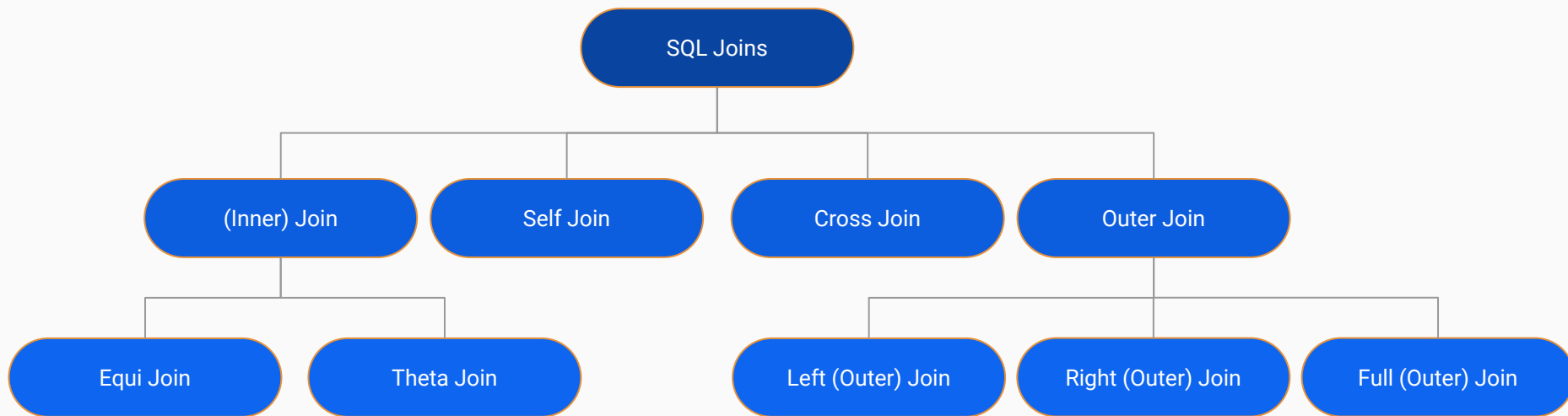


Συνένωση - Συνάθροιση - Ομαδοποίηση - Όψη

Εργαστήριο ΒΔ - 2020

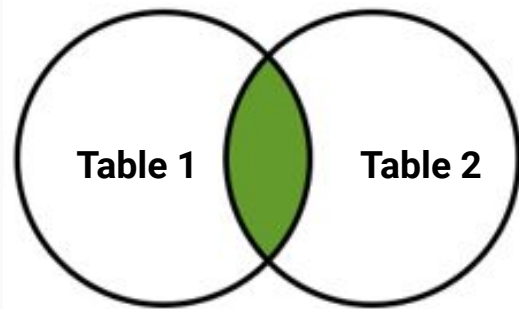
Τύποι συνένωσης (Join)



Inner Join

Επιστρέφει τις εγγραφές που έχουν κοινές τιμές **και στους δύο** πίνακες. $\Rightarrow$ Equi Join όταν χρησιμοποιείται ο τελεστής ισότητας.

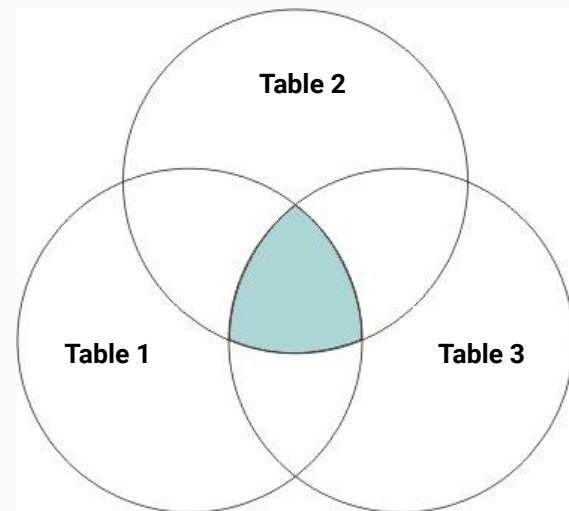
```
SELECT * FROM Table_1 AS t1  
INNER JOIN Table_2 AS t2  
ON t1.Column=t2.Column
```



Join πολλών σχέσεων

Επιστρέφει τις εγγραφές που έχουν κοινές τιμές **συνενώνοντας περισσότερους** πινάκων.

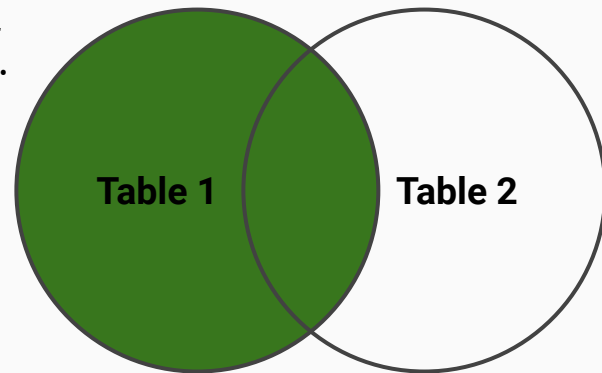
```
SELECT *  
FROM Table_1 AS t1  
JOIN Table_2 AS t2 ON t1.Column1 = t2.Column2  
JOIN Table_3 AS t3 ON t2.Column2 = t3.Column3  
WHERE t3.Column4 = 'Houston';
```



Left Outer Join

Επιστρέφει **όλες τις εγγραφές του αριστερού πίνακα** του join και τις εγγραφές **από τον δεξί πίνακα όπου η συνθήκη ικανοποιείται**.

```
SELECT * FROM Table_1 AS t1  
LEFT OUTER JOIN Table_2 AS t2  
ON t1.Column=t2.Column;
```

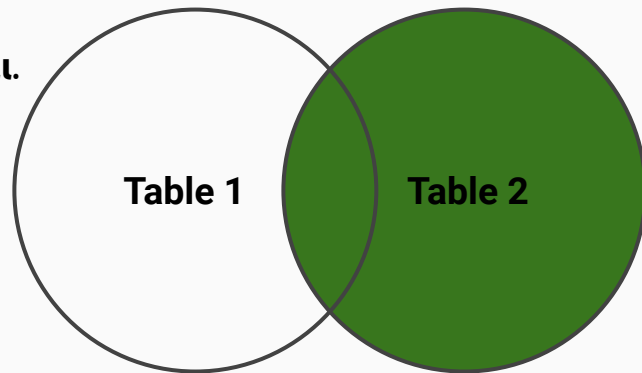


first_name	last_name	order_date	order_amount
George	Washington	07/04/1776	\$234.56
John	Adams	05/23/1784	\$124.00
Thomas	Jefferson	03/14/1760	\$78.50
Thomas	Jefferson	09/03/1790	\$65.50
James	Madison	NULL	NULL
James	Monroe	NULL	NULL

Right Outer Join

Επιστρέφει **όλες τις εγγραφές από τον δεξί πίνακα** του join και **τις εγγραφές από τον αριστερά πίνακα όπου η συνθήκη ικανοποιείται**.

```
SELECT * FROM Table_1 AS t1
RIGHT OUTER JOIN Table_2 AS t2
ON t1.Column=t2.Column;
```



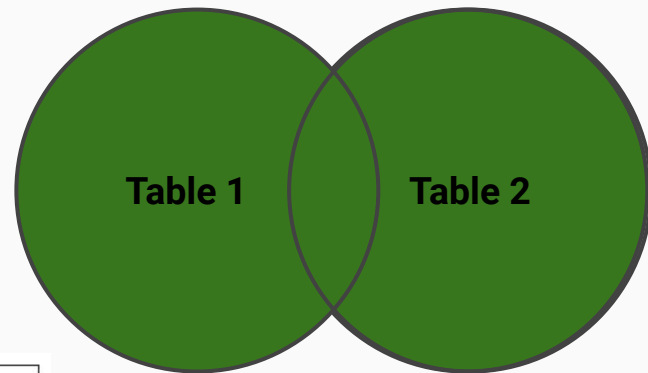
first_name	last_name	order_date	order_amount
George	Washington	07/04/1776	\$234.56
Thomas	Jefferson	03/14/1760	\$78.50
John	Adams	05/23/1784	\$124.00
Thomas	Jefferson	09/03/1790	\$65.50
NULL	NULL	07/21/1795	\$25.50
NULL	NULL	11/27/1787	\$14.40

Full Outer Join

Επιστρέφει **όλες τις εγγραφές** από τον δεξί ή τον αριστερό **πίνακα** για τις οποίες ικανοποιείται η συνθήκη.

```
SELECT * FROM Table_1 AS t1
FULL OUTER JOIN Table_2 AS t2
ON t1.Column=t2.Column;
```

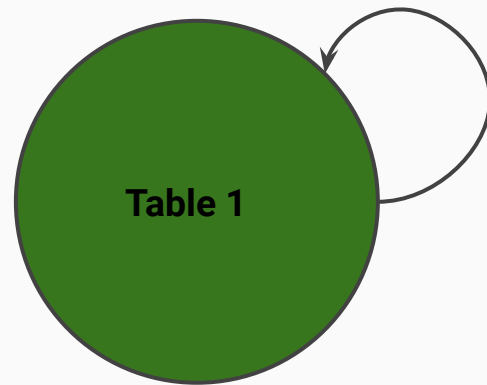
first_name	last_name	order_date	order_amount
George	Washington	07/04/1776	\$234.56
Thomas	Jefferson	03/14/1760	\$78.50
John	Adams	05/23/1784	\$124.00
Thomas	Jefferson	09/03/1790	\$65.50
NULL	NULL	07/21/1795	\$25.50
NULL	NULL	11/27/1787	\$14.40
James	Madison	NULL	NULL
James	Monroe	NULL	NULL



Self Join

Είναι σαν το κλασικό join, συνενώνοντας έναν πίνακα **με τον εαυτό του**.

```
SELECT *  
FROM Table1 t1, Table1 t2  
WHERE t1.Column1 = t2.Column2
```



Συνάθροιση

Υπάρχουν πέντε βασικές συναρτήσεις συνάθροισης:

- **COUNT** - καταμέτρηση πλήθους
- **SUM** - άθροισμα
- **AVG** - μέσο όρο
- **MIN** - ελάχιστη τιμή
- **MAX** - μέγιστη τιμή

```
SELECT SUM(costs)
FROM orders
WHERE product_category = 'Clothing';
```

Ομαδοποίηση δεδομένων

Αφορά την **ομαδοποίηση της προβολής** δεδομένων και συχνά συνδυάζεται με κάποια συνάρτηση συνάθροισης.

```
SELECT AVERAGE(salary), department  
FROM Employee  
GROUP BY Employee.department;
```

Επιλογή μετά τη συνάθροιση

Μετά την ομαδοποίηση των δεδομένων μπορεί να χρειαστεί να περιορίσουμε (επιλέξουμε) τις εγγραφές που προκύπτουν **μετά την ομαδοποίηση** με κάποια συνθήκη. Χρήση του τελεστή **HAVING**.

```
SELECT department_id, COUNT(employee_id)
FROM employee
GROUP BY department_id
HAVING COUNT(employee_id) > 5;
```

```
SELECT department_id, COUNT(employee_id)
FROM employee
WHERE department_id IN (4, 5)
GROUP BY department_id
HAVING COUNT(employee_id) > 5;
```

Όψεις

- Μία όψη είναι ένας **εικονικός πίνακας** που προκύπτει από συνδυασμό άλλων πινάκων και μπορεί να χρησιμοποιηθεί σαν να είναι κανονικός πίνακας σε ερωτήματα προβολής.
- Τα δεδομένα της όψης δεν είναι αποθηκευμένα στο δίσκο αλλά υπολογίζονται **κατά τον χρόνο κλήσης** της όψης.
- Υπάρχουν και τα “materialized views” οι οποίες αποθηκεύουν τα δεδομένα του αποτελέσματος τους στο δίσκο.

```
CREATE VIEW Emp_Dept as
SELECT fname || ' ' || lname as ypallilos, dname as tmima
FROM Employee, Department
WHERE Employee.dno=Department.dnumber
```