MULTIPLES TABLAS

INNER JOIN

```
SELECT region_name
FROM regions
NATURAL JOIN countries
WHERE country_name = ‘Canada’
...

SELECT country_name
FROM countries
NATURAL JOIN regions
WHERE region_name = ‘Americas’
```

MULTIPLES TABLAS

INNER JOIN

```
SELECT region_name
FROM regions
JOIN countries
USING (region_id)
WHERE country_name = ‘Canada’
...

SELECT country_name
FROM countries
JOIN regions
ON (countries.region_id = regions.region_id)
WHERE region_name = ‘Americas’
```

MULTIPLES TABLAS

OUTER JOIN

```
SELECT count(*)
FROM regions
OUTER JOIN countries;
```

MULTIPLES TABLAS

CROSS JOIN

```
SELECT count(*)
FROM regions
CROSS JOIN countries;
```

MULTIPLES TABLAS

EJEMPLOS

```
SELECT regions.region_name, countries.country_name
FROM regions, countries
WHERE regions.region_id = countries.region_id;

SELECT last_name, department_name
FROM employees, departments
WHERE employees.department_id (+) = departments.department_id;

SELECT * FROM regions, countries;
```

MULTIPLES TABLAS

```
SELECT table1.column, table2.column
FROM table1     [NATURAL JOIN table2] |
                [JOIN table2 USING (column_name)] |
                [JOIN table2 ON (table1.column_name = table2.column_name)] |
                [LEFT | RIGHT | FULL OUTER JOIN table2
                      ON (table1.column_name = table2.column_name)] |
                [CROSS JOIN table2];
```

MULTIPLES TABLAS

```
SELECT table1.column, table2.column
FROM table1     [NATURAL JOIN table2] |
                [JOIN table2 USING (column_name)] |
                [JOIN table2 ON (table1.column_name = table2.column_name)] |
                [LEFT | RIGHT | FULL OUTER JOIN table2
                      ON (table1.column_name = table2.column_name)] |
                [CROSS JOIN table2];
```

MULTIPLES TABLAS

NATURAL JOIN - EJEMPLOS

```
SELECT * FROM locations NATURAL JOIN countries;

SELECT * FROM locations, countries
WHERE locations.country_id = countries.country_id;

SELECT * FROM jobs NATURAL JOIN countries;

SELECT * FROM jobs, countries;
```

MULTIPLES TABLAS

NATURAL JOIN USING - EJEMPLOS

```
SELECT *
FROM locations
        JOIN countries USING (country_id);

SELECT *
FROM locations, countries
WHERE locations.country_id = countries.country_id;
```

MULTIPLES TABLAS

NATURAL JOIN ON - EJEMPLOS

```
SELECT *
FROM departments d
        JOIN employees e ON (e.employee_id=d.department_id);

SELECT *
FROM employees e, departments d
WHERE e.employee_id=d.department_id;
```

MULTIPLES TABLAS

NATURAL JOIN - EJEMPLOS

```
SELECT region_id, country_id, c.country_name, l.city, d.department_name
FROM  departments d
        NATURAL JOIN locations l
        NATURAL JOIN countries c
        NATURAL JOIN regions r;

SELECT r.region_name, c.country_name, l.city, d.department_name
FROM  departments d
        JOIN locations l ON (l.location_id=d.location_id)
        JOIN countries c ON (c.country_id=l.country_id)
        JOIN regions r ON (r.region_id=c.region_id);

SELECT r.region_name, c.country_name, l.city, d.department_name
FROM  departments d
        JOIN locations l USING (location_id)
        JOIN countries c USING (country_id)
        JOIN regions r USING (region_id);
```

MULTIPLES TABLAS

NATURAL JOIN - EJEMPLOS

```
SELECT d.department_name
FROM  departments d
          JOIN locations l ON (l.LOCATION_ID=d.LOCATION_ID)
WHERE d.department_name LIKE 'P%';

SELECT d.department_name
FROM  departments d
          JOIN locations l ON (l.LOCATION_ID=d.LOCATION_ID AND
                                    d.department_name like 'P%');
```

MULTIPLES TABLAS

NON EQUIJOINS

SELECT table1.column, table2.column FROM table1
[JOIN table2 ON (table1.column_name < table2.column_name)]|
[JOIN table2 ON (table1.column_name > table2.column_name)]|
[JOIN table2 ON (table1.column_name <= table2.column_name)]|
[JOIN table2 ON (table1.column_name >= table2.column_name)]|
[JOIN table2 ON (table1.column BETWEEN table2.col1 AND table2.col2)]

MULTIPLES TABLAS

SELFJOINS

```
SELECT id, name, father_id
FROM family;

SELECT name
FROM family
WHERE id=&father_id;

SELECT f1.name Dad, f2.name Child
FROM  family f1
        JOIN family f2 ON(f1.id=f2.father_id);
```

MULTIPLES TABLAS

LEFT OUTER JOIN - EJEMPLOS

```
SELECT e.employee_id,
                e.department_id EMP_DEPT_ID,
                d.department_id DEPT_DEPT_ID,
                d.department_name
FROM      departments d
                LEFT OUTER JOIN employees e ON (d.DEPARTMENT_ID=e.DEPARTMENT_ID)
WHERE d.department_name like 'P%';

SELECT e.employee_id,
                e.department_id EMP_DEPT_ID,
                d.department_id DEPT_DEPT_ID,
                d.department_name
FROM departments d JOIN employees e ON (d.DEPARTMENT_ID=e.DEPARTMENT_ID)
WHERE d.department_name like 'P%';
```

MULTIPLES TABLAS

RIGHT OUTER JOIN - EJEMPLOS

```
SELECT e.last_name, d.department_name
FROM    departments d
            RIGHT OUTER JOIN employees e ON (e.department_id=d.department_id)
WHERE e.last_name LIKE 'G%';
```

MULTIPLES TABLAS

FULL OUTER JOIN - EJEMPLOS

```
SELECT e.last_name, d.department_name
FROM    departments d
            FULL OUTER JOIN employees e ON (e.department_id = d.department_id)
WHERE e.department_id IS NULL;
```

MULTIPLES TABLAS

CROSS OUTER JOIN - EJEMPLOS

```
SELECT *
FROM   jobs
       CROSS JOIN job_history;

SELECT *
FROM   jobs j
       CROSS JOIN job_history jh
WHERE j.job_id='AD_PRES';
```