



Taller de Administración de Base de Datos Oracle

Coordinadora: Cintia Verónica Gioia

Docentes:

- Cintia Verónica Gioia
- Carina Pérez
- Nicolás Bracigliano
- Natalia Crespo

SQL STANDARD

*SOLUCIONES DE TRABAJO PRÁCTICO
BASICO y AVANZADO*

ANSI SQL – Básico y Avanzado

SQL STANDARD BASICO

Empleado(EMPNO, ENAME, SAL, DEPTNO)

Departamento(DEPTNO, DNAME)

Proyecto(PROYNO, PROYNOM)

TrabajaEn(EMPNO, PROYNO, horas)

Empleado

| EMPNO | ENAME | SAL | DEPTNO |
|-------|-------|------|--------|
| 1 | ne1 | 300 | d1 |
| 2 | ne1 | 500 | d2 |
| 3 | ne2 | 1000 | d2 |
| 4 | ne3 | 1500 | d3 |

Departamento

| DEPTNO | DNAME |
|--------|-------|
| d1 | nd1 |
| d2 | nd2 |
| d3 | nd3 |
| d4 | nd4 |

Proyecto

| PROYNO | PROYNOM |
|--------|---------|
| p1 | np1 |
| p2 | np2 |
| p3 | np3 |

Trabaja_En

| EMPNO | PROYNO |
|-------|--------|
| e1 | p3 |
| e2 | p1 |
| e3 | p2 |

Select <lista de columnas>

From <lista de tablas>

Where <condicion>

1. Seleccionar números de empleados

```
Select num_e  
From empleado
```

2. Seleccionar número y nombre de empleados

```
Select num_e, nom_e  
From empleado
```

3. Seleccionar todos los datos de los empleados

```
Select *  
From empleado
```

4. Seleccionar nombre de empleados no repetidos

```
SELECT distinct nom_e  
FROM empleado
```

5. Seleccionar número de empleados con código mayor a 3

```
Select num_e  
From empleado  
Where num_e>3
```

6. Seleccionar número de empleados con código mayor a 3 y trabaje en el depto d3

```
Select num_e  
From empleado  
Where num_e>3 and num_d=d3
```

Junta natural

7. Seleccionar nombre de departamento del empleado 4

```
Select nom_d  
From empleado e, departamento d  
Where e.num_e=4 and e.num_d=d.num_d
```

8. Seleccionar nombre de proyecto que trabaja el empleado 1

```
Select p.nom_p  
from trabajaEn t, proyecto p
```

where p.num_e=1 and t.num_p=p.num_p

9. Seleccionar nombre de empleado que trabaja en el proyecto “np1”

```
Select e.nom_e
from empleado e, trabajaEn t, proyecto p
where p.nom_p="np1" and t.num_p=p.num_p and t.num_e=e.num_e
```

Funciones agregativas (no mezclar funciones con columnas, solo con group by y having)

10.Determinar la cantidad de empleados que ganan mas de 100 \$

```
select count(*)
from empleado
where sue_e > 100
```

11.Determinar el sueldo total, el menor, el máximo y el promedio de todos los empleados

```
select sum(sue_e), min(sue_e), max(sue_e), avg(sue_e)
from empleado
```

12.Determinar el sueldo total de los empleados que trabajan en el departamento ‘nd2’

```
select sum(sue_e)
from empleado e, departamento d
where d.nom_d='nd2' and d.num_d=e.num_d
```

13.Determinar el sueldo de los empleados por departamento

```
SELECT num_d, Sum(sue_e)
FROM empleado
GROUP BY num_d
```

14.Determinar los nombres de empleados que empiecen con “n”

```
SELECT nom_e
FROM empleado
where nom_e like 'n%'
```

15.Determinar los nombres de empleados que empiecen con “ne” y terminen con cualquier letra

```
SELECT nom_e
FROM empleado
where nom_e like 'ne?'
```

Consulta 1:

Seleccionar los números de empleados de todos los empleados que trabajan en igual proyecto que el empleado con número 7521.

```
SELECT num_emp
FROM trabaja_en
WHERE num_proy in
      (select num_proy
       from trabaja_en
       where num_emp=7521);
```

Consulta 2:

Seleccionar los números de empleados de todos los empleados que trabajan en igual proyecto que el empleado con número 7521.No mostrar al empleado 7521 en la lista.

```
SELECT num_emp
FROM trabaja_en
WHERE num_proy in
      (select num_proy
       from trabaja_en
       where num_emp=7521)
and num_emp<>7521;
```

Consulta 3:

Seleccionar los números y nombres de empleados de todos los empleados que trabajan en igual proyecto que el empleado BLAKE.

```
SELECT t.num_emp, nom_emp
FROM trabaja_en t, empleado e
WHERE num_proy in
      (select num_proy
       from trabaja_en t, empleado e
       where t.num_emp=e.num_emp and nom_emp=' BLAKE')
and t.num_emp=e.num_emp;
```

Consulta 4:

Seleccionar los nombres de proyectos donde haya empleados que ganen más de 1000.

```
SELECT nom_proy
FROM proyecto p, trabaja_en t, empleado e
WHERE p.num_proy=t.num_proy and t.num_emp=e.num_emp and e.sueldo_emp>1000;
```

Consulta 4a:

Seleccionar los nombres de proyectos (no repetidos) donde haya empleados que ganen más de 1000.

```
SELECT DISTINCT nom_proy
FROM proyecto p, trabaja_en t, empleado e
WHERE p.num_proy=t.num_proy and t.num_emp=e.num_emp and e.sueldo_emp>=1000;
```

Consulta 4b:

Seleccionar los nombres de proyectos (no repetidos) donde haya empleados que sean "SALESMAN".

```
SELECT DISTINCT nom_proy
FROM proyecto p, trabaja_en t, departamento d, empleado e
WHERE p.num_proy=t.num_proy and t.num_emp=e.num_emp
And e.job = 'SALESMAN' ;
```

Consulta 4c:

Unión de consulta 4a y consulta 4b.

```
(SELECT nom_proy
FROM proyecto p, trabaja_en t, empleado e
WHERE p.num_proy=t.num_proy and t.num_emp=e.num_emp and e.sueldo_emp>1000)
UNION (SELECT DISTINCT nom_proy
FROM proyecto p, trabaja_en t, departamento d, empleado e
WHERE p.num_proy=t.num_proy and t.num_emp=e.num_emp
And e.job = 'SALESMAN');
```

Consulta 4d:

Otra versión de consulta 4c usando in.

```
SELECT DISTINCT nom_proy
FROM proyecto
WHERE nom_proy in
    (select nom_proy
     FROM proyecto p, trabaja_en t, empleado e
     WHERE p.num_proy=t.num_proy and t.num_emp=e.num_emp and
     e.sueldo_emp>1000)
or nom_proy in
    (SELECT DISTINCT nom_proy
     FROM proyecto p, trabaja_en t, departamento d, empleado e
     WHERE p.num_proy=t.num_proy and t.num_emp=e.num_emp
     And e.job = 'SALESMAN');
```

Consulta 5:

Seleccionar el nombre del empleado, nombre del departamento y sueldo del empleado con el mayor sueldo.

```
SELECT nom_emp, nom_dep, sueldo_emp
FROM empleado e, departamento d
WHERE e.num_dep=d.num_dep
and sueldo_emp >=all( select sueldo_emp
                     from empleado);
```

Consulta 6:

Seleccionar el nombre del empleado, nombre del departamento y sueldo de los empleados que tengan mayor sueldo que todos los empleados del departamento 10.

```
SELECT nom_emp, nom_dep, sueldo_emp
FROM empleado e, departamento d
WHERE e.num_dep=d.num_dep
and sueldo_emp > All( select sueldo_emp
                     from empleado
                     where num_dep=10);
```

Consulta 7:

Seleccionar el nombre del empleado, nombre del departamento y sueldo de los empleados que tengan mayor sueldo de por lo menos un empleado del departamento 10.

```
SELECT nom_emp, nom_dep, sueldo_emp
FROM empleado e, departamento d
WHERE e.num_dep=d.num_dep
and sueldo_emp > any( select sueldo_emp
                      from empleado
                      where num_dep=10);
```

Consulta 8:

Seleccionar el nombre del empleado, nombre del departamento y sueldo del empleado con menor sueldo.

```
SELECT nom_emp, nom_dep, sueldo_emp
FROM empleado e, departamento d
WHERE e.num_dep=d.num_dep
and sueldo_emp <=all( select sueldo_emp
                     from empleado
                     );
```

Consulta 9:

Seleccionar el nombre y número del empleado, nombre del departamento y sueldo del empleado que no trabaje en ningún proyecto.

```
SELECT num_emp, nom_emp, nom_dep, sueldo_emp
FROM empleado e, departamento d
WHERE e.num_dep=d.num_dep
and e.num_emp not in ( select num_emp
                      from trabaja_en
                      );
```


Consulta 10:

Seleccionar el nombre y número de los empleados que trabajan en algún proyecto.

```
SELECT num_emp, nom_emp
FROM empleado e
WHERE exists (select *
              from trabaja_en t
              where e.num_emp=t.num_emp);
```

Consulta 11:

Otra versión de consulta 9 usando NOT EXISTS.

```
SELECT num_emp, nom_emp
FROM empleado e
WHERE not exists (select *
                  from trabaja_en t
                  where e.num_emp=t.num_emp);
```

Consulta 12:

Seleccionar los números de empleados que trabajen en proyectos y sean MANAGER.

```
SELECT e.num_emp
FROM empleado e
WHERE exists
      (select *
       from trabaja_en t
       where t.num_emp=e.num_emp)
and e.job='MANAGER'
```

Consulta 13:

Seleccionar los números de empleados que trabajen en proyectos y NO sean MANAGER.

```
SELECT e.num_emp
FROM empleado e
WHERE exists
      (select *
       from trabaja_en t
       where t.num_emp=e.num_emp)
and e.job<>'MANAGER';
```

Consulta 14:

Seleccionar los nombres y números de empleados que trabajan en proyectos (junta usando condición de junta en el where).

```
SELECT nom_emp, e.num_emp, t.num_proy
FROM empleado e, trabaja_en t
WHERE e.num_emp=t.num_emp;
```

Consulta 15:

Consulta15 usando LEFT OUTER JOIN.

```
SELECT nom_emp, e.num_emp, t.num_proy
FROM empleado e, trabaja_en t
WHERE e.num_emp(+)=t.num_emp;
```

Consulta 16:

Consulta15 usando RIGHT OUTER JOIN.

```
SELECT nom_emp, e.num_emp, t.num_proy
FROM empleado e, trabaja_en t
WHERE e.num_emp=t.num_emp(+);
```

Consulta 17:

Seleccionar número de empleado, número y nombre del proyecto donde trabaja cada empleado.

```
SELECT t.num_emp, t.num_proy, p.nom_proy
FROM trabaja_en t, proyecto p
WHERE t.num_proy=p.num_proy;
```

Consulta 18:

Consulta19 con RIGHT JOIN.

```
SELECT t.num_emp, t.num_proy, p.nom_proy
FROM trabaja_en t, proyecto p
WHERE t.num_proy=p.num_proy(+);
```