



**GUSTAVO CORONEL**  
DESARROLLA SOFTWARE

# TALLER DE ORACLE SQL



Una buena base de datos, asegura tu inversión.

## LABORATORIO 01 CONSULTAS A UNA TABLA

**GUSTAVO CORONEL**  
*desarrollasoftware.com*



## CONTENIDO

<b>CONSULTAS A UNA TABLA .....</b>	<b>3</b>
SINTAXIS.....	3
SELECCIÓN DE COLUMNAS.....	3
CAMPOS CALCULADOS.....	3
<b>FILTRO DE FILAS.....</b>	<b>4</b>
SINTAXIS.....	4
OPERADORES DE COMPARACIÓN .....	4
OPERADORES LOGICOS.....	4
OPERADORES LIKE, BETWEEN, IN, IS NULL .....	5
<b>ORDENAMIENTO DEL CONJUNTO DE RESULTADOS .....</b>	<b>5</b>
SINTAXIS.....	5
<b>MANIPULACIÓN DE VALORES NULL .....</b>	<b>5</b>
<b>FUNCIONES DE FECHA Y HORA.....</b>	<b>6</b>



## CONSULTAS A UNA TABLA

### Sintaxis

```
SELECT * | < lista_de_columnas y/o expresiones >  
FROM [ esquema . ] < nombre_de_tabla >;
```

### Selección de columnas

Ejercicios:

1. Desarrolle una sentencia SELECT para consultar el id, nombre, apellido y email de todos los empleados. Esquema **RECURSOS**.
2. Desarrolle una sentencia SELECT para consultar el id, nombre, dirección y email de todos los alumnos. Esquema **EDUCA**.
3. Desarrolle una sentencia SELECT para consultar el id, nombre, precio y profesor de cada curso. Esquema **EDUCA**.
4. Desarrolle una sentencia SELECT para consultar el id y nombre de todos los departamentos. Esquema **RECURSOS**.
5. Desarrolle una sentencia SELECT para consultar el nombre, vacantes y precio de cada curso. Esquema **EDUCA**.

### Campos calculados

6. Desarrollar una sentencia SELECT que permita obtener la cantidad y porcentaje de matriculados por cada curso. Esquema **EDUCA**.
7. Desarrollar una sentencia SELECT para consultar el nombre y apellido de un empleado en una sola columna. Base de datos **RECURSOS**.
8. Desarrollar una sentencia SELECT que permita obtener el importe que se obtendría si se logra vender todas las vacantes por cada curso. Base de datos **EDUCA**.
9. Desarrolle una sentencia SELECT que permita obtener el importe que se obtendría por cada producto si se venden todas las unidades que hay en stock. Esquema **VENTAS**.
10. Desarrollar una sentencia SELECT que permita consultar el salario, y el salario proyectado con un aumento del 15% de cada empleado. Esquema **RECURSOS**.



## FILTRO DE FILAS

### Sintaxis

```
WHERE < condición_de_filtro >
```

### Operadores de Comparación

#### Ejercicios

1. Consultar los empleados que ingresaron a trabajar el primer trimestre del año 1981. Esquema **SCOTT** y **RECURSOS**
2. Desarrollar una sentencia SELECT para consultar los empleados del departamento de contabilidad. Esquemas **SCOTT** y **RECURSOS**.
3. Desarrolle una sentencia SELECT para consultar los cursos que no tienen alumnos matriculados. Esquema **EDUCA**.
4. Desarrollar una sentencia SELECT para consultar los productos que su precio en mayor a 50 Soles. Esquema **VENTAS**.
5. Desarrollar una sentencia SELECT para consultar los empleados que su salario en mayor a 3000.00. Esquema **RECURSOS**.

### Operadores Logicos

#### Ejercicios

1. Desarrollar una sentencia SELECT para consultar los empleados que su salario esta entre 3500.00 y 10000.00. Esquema **RECURSOS**.
2. Desarrollar una sentencia SELECT para consultar los empleados de contabilidad cuyo sueldo e mayor a 10,000.00. Esquema **RECURSOS**.
3. Desarrollar una sentencia SELECT para obtener un listado de los empleados que su nombre tiene en la segunda posición la letra "A" o la letra "O". Esquemas **SCOTT** y **RECURSOS**.
4. Desarrollar una sentencia SELECT para obtener un listado de los productos de la categoría "LINEA BLANCA" y que su precio sea mayor a 1000.00. Esquema **VENTAS**.
5. Desarrollar una sentencia SELECT para obtener un listado de los empleados de los departamentos de VENTAS y recursos humanos. Esquemas **SCOTT** y **RECURSOS**.
6. Consultar los empleados que la segunda letra de su nombre puede ser A, O u I. Esquemas **SCOTT** y **RECURSOS**.



## Operadores LIKE, BETWEEN, IN, IS NULL

Ejercicios:

7. Desarrollar una sentencia SELECT que permita consultar los empleados cuya penúltima letra de su nombre es E. Esquema **SCOTT** y **RECURSOS**.
8. Desarrollar una sentencia SELECT que permita consultar los empleados que su nombre finaliza con la letra "O". Esquema **RECURSOS**.
9. Desarrollar una sentencia SELECT que permita consultar los empleados que su apellido tiene en la segunda posición la letra "A" o la letra "O". Esquemas **SCOTT** y **RECURSOS**.
10. Desarrollar una sentencia SELECT que permita averiguar que empleados tienen un sueldo mayor de 3,000.0 y menor de 10,000.0. Esquemas **SCOTT** y **RECURSOS**.
11. Desarrollar una sentencia SELECT que permita averiguar quiénes son los empleados de los departamentos de contabilidad, recursos humanos y ventas. Esquemas **SCOTT** y **RECURSOS**.
12. Consultar los empleados del departamento de ventas que no tienen comisión. Esquemas **SCOTT** y **RECURSOS**.

## ORDENAMIENTO DEL CONJUNTO DE RESULTADOS

### Sintaxis

```
ORDER BY < column1 > [ ASC | DESC ]  
        [, < column2 > [ ASC | DESC ] [, . . . ] ]
```

Ejercicios

13. Desarrollar una sentencia SELECT que permita mostrar una lista de los empleados ordenada por fecha de ingreso. Esquema **RECURSOS**.
14. Desarrollar una sentencia SELECT para consultar los productos ordenados por categoría y precio. Esquema **VENTAS**.

## MANIPULACIÓN DE VALORES NULL

Ejercicios:

15. Desarrollar una sentencia SELECT que permita averiguar los cursos que aún no tienen profesor. Esquema **EDUCA**.
16. Desarrollar una sentencia SELECT para consultar el ingreso total de cada empleado. Esquema **RECURSOS**.
17. Desarrollar una sentencia SELECT para consultar los empleados cuyos ingresos totales es menor a 8,000.00. Esquema **RECURSOS**.
18. Consultar los empleados cuyo ingreso (salario + comisión) supera los 3500.00. Esquema **SCOTT** y **RECURSOS**.
- 19.



## FUNCIONES DE FECHA Y HORA

Ejercicios:

20. Desarrollar una sentencia SELECT para consultar los empleados que ingresaron a la empresa un mes de Enero. Esquemas **SCOTT** y **RECURSOS**.
21. Desarrollar una sentencia SELECT para consultar las matriculas del último mes. Esquema **EDUCA**.