
1. ESTRUCTURA BÁSICA DE SENTENCIAS SQL

- 1.1. La sentencia SELECT
- 1.2. Etiquetas de columnas
- 1.3. Expresiones aritméticas
- 1.4. Valor NULL
- 1.5. Alias de columnas
- 1.6. Concatenaciones
- 1.7. Visualización de la estructura de una tabla
- 1.8. Comandos de edición y de ficheros

1.1. - LA SENTENCIA SELECT

Recupera la información de la base de datos. Con la sentencia SELECT podemos hacer lo siguiente:

- Selección: Podemos utilizar la capacidad de selección de SQL para seleccionar los registros de la tabla que queramos, devueltos por una consulta. Podemos utilizar varios criterios para restringir de manera selectiva los registros que queremos ver.
- Proyección: Podemos utilizar la capacidad de proyección para seccionar las columnas en la tabla que se deseen, devueltas por nuestra consulta. Podemos seleccionar tantas tablas como necesitemos.
- Unión: Permite extraer información de distintas tablas mediante la creación de enlaces entre una columna y las dos tablas compartidas.

Sintaxis:

```
SELECT [DISTINCT] { * | COLUMN[ALIAS],...}  
FROM TABLA;
```

Ejemplos:

```
SELECT *  
FROM EMP;
```

Extrae todos los elementos de la tabla empleado

```
SELECT JOB, SAL  
FROM EMP;
```

Muestra las columnas JOB y Sal de la tabla EMP

Escritura de una sentencia SQL:

- Los comandos pueden constar de una o varias líneas.
- Las tabulaciones pueden ser usadas por comodidad.
- Las abreviaturas y separaciones de palabras no están permitidas.

-
- Los comandos no son sensibles a las mayúsculas y a las minúsculas.

Ejecución de una sentencia SQL

- Colocar punto y coma al final de la última cláusula.
- Colocar un punto y coma o / al final de la última línea del búfer.
- Colocar una / en el prompt de SQL.
- Escribir RUN de SQL*Plus desde el prompt de SQL.

1.2. – ETIQUETAS DE LAS COLUMNAS

Por defecto, las etiquetas en una tabla, se muestran siempre en mayúsculas.

Las de tipo cadena y fecha se justifican a la izquierda y las numéricas a la derecha.

1.3. – EXPRESIONES ARITMÉTICAS

Los operadores básicos son: +, -, *, /. Solo con datos numéricos y fechas.

Ejemplo:

```
SELECT ENAME, SAL, SAL+30  
FROM EMP;
```

El orden general de prioridad es el siguiente * / + - . Los operadores * y / tienen prioridad sobre los operadores + y - . Los operadores de la misma prioridad se evalúan de izquierda a derecha. Los paréntesis () se utilizan para romper las prioridades.

1.4. – VALOR NULL

Un valor NULL es un valor que es inaccesible, sin valor desconocido o inaplicable. No representa ni un cero ni un espacio en blanco, ya que el cero es un número y blanco un carácter.

Si operamos con un NULL la expresión tomará el valor nulo.

1.5. – ALIAS DE COLUMNAS

- Un alias de columna renombra un encabezamiento de columna.
- Es útil, especialmente en cálculos.
- Sigue inmediatamente al nombre de la columna. La palabra clave opcional AS entre el nombre de la columna y el alias.
- Se requiere encerrar el alias entre comillas dobles, si contienen espacios en blanco, caracteres especiales y palabras clave.

Ejemplo.

```
SELECT      ENAME NOMBRE, JOB AS TRABAJO ,  
            SAL*12 AS "SALARIO ANNUAL",  
            SAL*6  "SALARIO SEMESTRAL"  
FROM EMP;
```

1.6.- EL OPERADOR CONCATENACIÓN

El operador de concatenación:

- Está representado por dos barras verticales (||)
- Vincula columnas o cadenas de caracteres con otras columnas.
- Crea una columna resultado que es una expresión de tipo carácter.
- Se pueden utilizar alias de columnas para la concatenación.

Ejemplo:

```
SELECT ENAME || JOB AS "MEZCLA"  
FROM EMP;
```

Cadenas de caracteres:

- Un literal es un carácter, expresión o número incluido en la lista de la cláusula SELECT.
- Los valores literales de tipo fecha y carácter debe ir entre comillas simples.
- Por cada fila devuelve se genera una cadena de caracteres.

Ejemplo:

```
SELECT ENAME || ' ' || 'ES' || ' ' || JOB AS "Nombre y trabajos"  
FROM EMP;
```

Las filas duplicadas se eliminan usando DISTINCT en la cláusula SELECT.

Ejemplo:

```
SELECT DISTINCT deptno  
FROM emp;
```

1.7.- VISUALIZACION DE LA ESTRUCTURA DE UNA TABLA

Se realiza mediante el comando SQL*Plus DESCRIBE (DESC), muestra la descripción de las columnas de una tabla, pudiendo existir los siguientes tipos de campos:

- NUMBER (n,s) : dónde **n** es el número máximo de caracteres enteros y **s** el número de dígitos decimales
- VARCHAR2(n): hace referencia a cadenas de caracteres de longitud variable de tamaño **n**.
- DATE: fecha/hora
- CHAR(s): hace referencia a cadenas de caracteres de longitud fija de tamaño **s**.

Sintaxis:

DESC Nombre_tabla
