

TEMA 4: INTRODUCCIÓN A SQL

VALORES LITERALES EN ORACLE

1.- DEFINICIÓN DE LITERAL.

En general, en los lenguajes informáticos, **un LITERAL es un valor explícito constante que no se representa mediante un identificador**. Los términos literal y valor constante se podrían considerar sinónimos. Existen literales de tipo numérico, carácter, cadena, fecha, etc.

Aunque este concepto es genérico, cada lenguaje representa los literales de cada tipo en función de su sintaxis. **Por ejemplo**: en Oracle un literal de tipo cadena sería 'soy una cadena' y en cualquier otro lenguaje la misma cadena se podría representar como "soy una cadena".


2.- LITERALES EN ORACLE.

Veamos a continuación como se representan los literales de cada tipo en Oracle:

TIPOS DE DATOS	DESCRIPCIÓN	EJEMPLOS
NUMÉRICOS	<ul style="list-style-type: none">Enteros (con o sin signo).Reales (con o sin signo)Notación científica	5, +5, -5 5.2, +5.2, -5.2, 5.0 ... 5E3 (=5x10 ³)
CADENA	<ul style="list-style-type: none">Se representan entre comillas simples.	'SOY UNA CADENA' 'hola' '2500'
CARÁCTER INDIVIDUAL	<ul style="list-style-type: none">Conviene observar que en Oracle no existe el tipo carácter como tal.Este tipo se consigue definiendo cadenas de tamaño 1.	'a' 'A' '0' '*'
FECHA	<ul style="list-style-type: none">Se representan entre comillas simples.Existen distintos formatos de fecha en Oracle (los veremos con detalle). El más típico es el llamado formato de fecha corta dd-mm-yyyyNo hay que confundir los literales de tipo fecha con los de tipo cadena. Dependiendo del contexto estaremos usando unos u otros.	'01-12-2005' '01/12/2005'
TODOS	<ul style="list-style-type: none">VALOR NULO: tal y como hemos visto el valor nulo es un valor especial que denota la ausencia de valor para cualquier tipo de datos.En Oracle el literal que representa el valor nulo para cualquier tipo de datos es NULL	NULL

3.- USO DE LITERALES EN SQL.

En SQL se pueden utilizar literales en cualquier lugar donde sintácticamente se permitan o esperen. Por ejemplo, si miramos la sintaxis de SQL, podemos utilizar un valor constante cada vez que se necesita una **ref_columna**:

ref_columna::= atributo|
constante| 
expresion_aritmetica|
ref_funcion

Siempre que se utiliza un literal, su tipo de datos debe ser el adecuado según el contexto. Por ejemplo, en una comparación básica en una sentencia SELECT:

VER DEFINICIÓN DE LAS TABLAS EMPLE Y DEPART EN TEMA 3	
SELECT *	SELECT *
FROM DEPART	FROM EMPLE
WHERE DNOMBRE='VENTAS'	WHERE FECHA ALT<'01-01-2000'
'VENTAS' debe ser un literal de tipo cadena	'01-01-2000' debe ser un literal de tipo fecha

4.- CONVERSIÓN IMPLÍCITA DE TIPOS EN ORACLE.

Oracle realiza de manera automática, si lo considera necesario, lo que se denomina una CONVERSIÓN IMPLÍCITA DE TIPO, para evitar que se produzca un error. Esto quiere decir que, si en un contexto determinado se espera un valor de un tipo de datos concreto, y no se utiliza explícitamente, Oracle convertirá automáticamente, SI PUEDE, ese valor a un valor del tipo de datos esperado.

Por ejemplo, supongamos que realizamos la consulta siguiente:

ALUMNOS (DNI: VARCHAR2(10), NOMBRE: VARCHAR2(30), CURSO: NUMBER(1), NOTA: NUMBER(4,2), TELEF: VARCHAR2(10)) CP= {DNI}
SELECT NOTA FROM ALUMNOS WHERE DNI !=12344345
12344345 es un literal de tipo numérico que Oracle convierte automáticamente a cadena SIN AVISARNOS En realidad se debería haber especificado el literal de tipo cadena '12344345'

Hay que tener en cuenta que esta circunstancia NO es la más apropiada para aprender SQL, ya que cosas que en realidad son erróneas (así serían consideradas en cualquier editor de consultas de cualquier otro SGBD), son correctas en un editor de Oracle.

NOTA IMPORTANTE

Nosotros trabajaremos como si esta conversión implícita no se realizara, es decir:
NOS OBLIGAREMOS A UTILIZAR ADECUADAMENTE LOS LITERALES EN FUNCIÓN DEL CONTEXTO.