# FUNCIONES EN ORACLE

* 1. FUNCIONES DE CARACTERES
  2. FUNCIONES DE FILTRO
  3. FUNCIONES DE MANIPULACIÓN DE CARACTERES

1. FUNCIONES NUMÉRICAS
2. FUNCIONES DE FECHA
3. FUNCIONES DE CONVERSIÓN
   * 1. IMPLÍCITA AL TIPO DE DATO
     2. EXPLÍCITA AL TIPO DE DATO

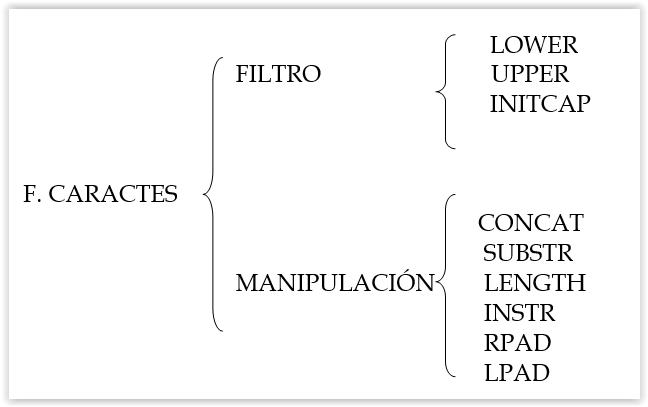
* TO\_ CHAR
* TO\_ NUMBER
* TO\_ DATE

5 FUNCIONES GENERALES

* NVL
* DECODE

6 FUNCIONES ANIDADAS

**1. FUNCIONES DE CARACTERES**



**1.1 FUNCIONES DE FILTRO**

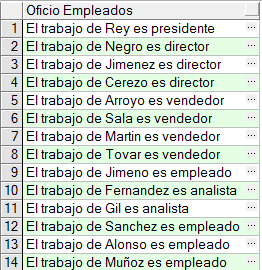
* **LOWER**: Convierte a minúsculas las cadenas de caracteres mayúsculas o mixtas.
* **UPPER:** Convierte a mayúsculas las cadenas de caracteres minúsculas o mixtas.
* **INITCAP:** Convierte la primera letra de una cadena a mayúsculas y el resto a minúsculas

Ejemplos:

* 1.- Obtener de la tabla empleados una columna encabezada por “Oficio empleados” y cuyas filas contengan el texto El trabajo de Nombre es oficio, es decir.

SELECT 'El trabajo de ' || INITCAP(apellido) || ' es ' || LOWER (oficio) as "Oficio Empleados"

FROM EMP



* 2.- Obtener el número de empleado, nombre y número de departamento del empleado cuyo apellido es Cerezo. Utilizar la función UPPER.

SELECT \*

FROM EMP

WHERE UPPER(apellido) ='CEREZO'

**

**1.2 FUNCIONES DE MANIPULACIÓN DE CARACTERES.**

* **CONCAT:** Dadas dos cadenas de caracteres cad1, cad2 introducidas como argumentos, devuelve una cadena de caracteres formada por la unión de cad1 y cad2.

Sintaxis:

**CONCAT (Cad1, Cad2)**

Ejemplos:

SELECT CONCAT('REFRES','CANTE') AS CONCATENANDO FROM DUAL;



SELECT CONCAT('ESTE EMPLEADO SE LLAMA, ',APELLIDO) AS CONCATENAR FROM EMP

WHERE UPPER(apellido)='GIL';



* **SUBSTR:** Extrae una cadena de caracteres de una determinada medida, según se especifique en sus argumentos.

Sintaxis:

**SUBSTR (Cad, m[,n])**

* Cad: cadena de caracteres
* m representa el número de orden del carácter de comienzo.
* n representa el número de caracteres a recuperar.

Si se omite n devuelve la cadena desde la posición indicada por m.

Ejemplos:

Extrae 4 caracteres de la cadena FLORERO, comenzando por el primer carácter.

SELECT SUBSTR('FLORERO',1,4) as dato FROM DUAL



Seleccionar el nombre de los empleados que empiezan por ‘S’

SELECT apellido FROM EMP WHERE SUBSTR(apellido,1,1)='s'



* **LENGTH:** Devuelve la longitud de una cadena de caracteres.

Sintaxis:

LENGTH(Cad);

Ejemplos:

select LENGTH('LIBRO') as longitud from dual



Seleccionar los nombres de los empleados con más de 5 caracteres.

SELECT apellido FROM EMP WHERE LENGTH(apellido)>5



* **INSTR:** Busca una cadena de caracteres dentro de otra y devuelve la posición numérica de comienzo de ésta, si la encuentra.

Sintaxis:

**INSTR(Cad,Subcad)**

* Cad cadena de caracteres donde vamos a buscar.
* Subcad carácter o cadena de caracteres buscada.

Ejemplo:

select INSTR ('BENITO','N') as dato from dual



* **LPAD:** Ajusta una cadena de caracteres a la derecha, es decir, añade caracteres a la izquierda de la cadena, hasta que consigue una determinada longitud.

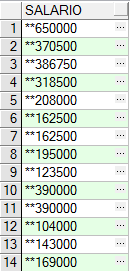
Sintaxis:

**LPAD(Cad1,n[,Cad2])**

* Devuelve cad1 con longitud n y ajustado a la derecha.
* Cad2 indica el carácter de relleno. Si se suprime Cad2 asume como carácter de relleno el espacio.

Ejemplo:

select LPAD (salario,8, '\*') as salario from emp



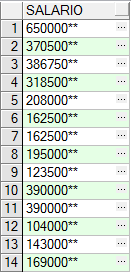
* **RPAD:** Ajusta una cadena de caracteres a la izquierda, es decir, añade caracteres a la derecha de la cadena, hasta que consigue una determinada longitud.

Sintaxis:

**RPAD(Cad1,n[,Cad2])**

* Devuelve cad1 con longitud n y ajustado a la izquierda.
* Cad2 indica el carácter de relleno. Si se suprime Cad2 asume como carácter de relleno el espacio.

select RPAD (salario,8, '\*') as salario from emp



**2. FUNCIONES NUMÉRICAS**

* **ROUND:** Redondea la columna o expresión a un valor n posiciones decimales. Si se omite n no se redondea con lugares decimales.

Sintaxis:

**ROUND(número[,n])**

Ejemplos:

select ROUND(45.923) as REDONDEO from dual



select ROUND(45.49) as REDONDEO from dual



select ROUND(45.50) as REDONDEO from dual



select ROUND(45.923,2) as REDONDEO from dual



* **TRUNC:** Elimina los dígitos indicados. Trunca la columna o valor en la enésima posición decimal, si se omite n, sin lugares decimales.

Sintaxis:

**TRUNC(número[,n])**

Ejemplos:

select TRUNC(45.923) as TRUNCA from dual



select TRUNC(45.923,2) as TRUNCA from dual



* **MOD:** Devuelve el resto de la división entre dos números ( m entre n)

Sintaxis:

MOD(m,n)

Ejemplo:

Calcular el resto resultante de dividir el salario a la comisión para todos los empleados cuyo oficio es VENDEDOR,

SELECT MOD(SALARIO, COMISION) as RESTO

FROM EMP

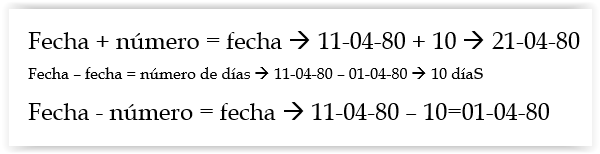
WHERE OFICIO ='VENDEDOR';



**3. FUNCIONES DE FECHAS.**

**3.1 OPERADORES ARITMÉTICOS CON FECHAS.**

Permiten sumar o restar un número o una fecha a otra fecha, dando como resultado lo siguiente:



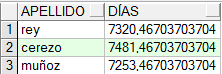
Ejemplo:

Obtener el apellido, y el tiempo que lleva un empleado en la empresa, expresado en días, para los empleados del departamento 10.

SELECT Apellido, SYSDATE-fecha\_alt as Días

FROM EMP

WHERE DEPT\_NO = 10

****

Devuelve los empleados y el número de días que hay entre la fecha del sistema y la fecha de entrada en la empresa. La parte no entera del número que devuelve indica horas.

###### 3.2 FUNCIONES DE FECHA.

* **MONTHS\_BETWEEN:**  Devuelve la cantidad de meses que hay entre la fecha 1 y la fecha 2.

Sintaxis:

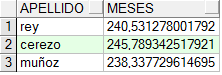
**MONTHS\_BETWEEN(Fecha1, Fecha2)**

Ejemplo:

SELECT Apellido, MONTHS\_BETWEEN(SYSDATE,fecha\_alt) as Meses

FROM EMP

WHERE DEPT\_NO = 10



* **ADD\_ MONTHS:** Agrega número de n meses a la fecha.

Sintaxis:

**ADD\_MONTHS(Fecha1,n)**

Ejemplo:

SELECT Apellido, Fecha\_alt, ADD\_MONTHS(fecha\_alt,2) as "Fecha+ 2 MESES"

FROM EMP



* **NEXT\_ DAY:** Devuelve la fecha del día, especificado en carácter, siguiente a la fecha, carácter puede ser un número de 1..7 o un literal correspondiente al día de la semana.

Sintaxis:

**NEXT\_DAY (FECHA, ‘CARÁCTER’)**

Ejemplo:

SELECT Apellido, Fecha\_alt, NEXT\_DAY(fecha\_alt,'LUNES') as "Próximo Lunes"

FROM EMP



SELECT Apellido, Fecha\_alt, NEXT\_DAY(fecha\_alt,2) as "Próximo Martes"

FROM EMP



* **LAST\_ DAY:** Devuelve la fecha del último día del mes que contiene la fecha.

Sintaxis:

**LAST\_DAY(FECHA)**

Ejemplo:

SELECT Apellido, Fecha\_alt, LAST\_DAY(fecha\_alt) as "Último día del mes"

FROM EMP



* **ROUND :** Redondea la fecha según el formato sea ‘MM’ o ‘YY’. Cuando no se especifica ningún formato devuelve la fecha como estaba. Si el formato es ‘MM’, redondea el mes dependiendo del día y si el formato es ‘YY’, redondea al año dependiendo del mes.

Sintaxis:

**ROUND(FECHA,[Formato])**

Ejemplos:

SELECT Apellido, Fecha\_alt, ROUND(FECHA\_ALT,'MM') as "Fecha redondeada al mes"



SELECT Apellido, Fecha\_alt, ROUND(FECHA\_ALT,'YY') as "Fecha redondeada al año"

FROM EMP



* **TRUNC:** Trunca la fecha dependiendo del formato. Redondea siempre a 01-xx-xx cuando es MM, y a 01-01-xx cuando es YY.

Sintaxis:

**TRUNC(FECHA,[Formato])**

SELECT Apellido, Fecha\_alt, TRUNC(FECHA\_ALT,'MM') as "Fecha truncada al mes"

from emp

****

SELECT Apellido, Fecha\_alt, TRUNC(FECHA\_ALT,'YY') as "Fecha truncada al año"

FROM EMP



**4. FUNCIONES DE CONVERSIÓN.**

Existen dos tipos de funciones de conversión: implícitas y explicitas

**4.1 IMPLÍCITAS AL TIPO DE DATO**

Las implícitas son las realizadas automáticamente por ORACLE.

ORACLE puede utilizar tipos de datos de ANSI, DB”... y es capaz de convertir estos datos a sus formatos.

**4.2 EXPLICITAS AL TIPO DE DATOS**

Son funciones que permiten al usuario que maneja la base de datos, modificar o cambiar el formato de los datos como TO\_NUMBER, TO\_DATE, TO\_CHAR.

TO\_NUMBER TO\_DATE

NUMÉRICO CARÁCTER FECHA

TO\_CHAR TO\_CHAR

* **TO\_CHAR:** Convierte datos de tipo DATE o NUMBER a cadenas de caracteres. Se suele utilizar para visualizar las fechas o valores numéricos en un formato diferente a como están almacenadas en la base de datos.

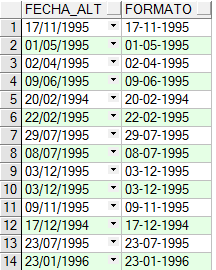
Sintaxis:

**TO\_CHAR(fecha,’formato’);**

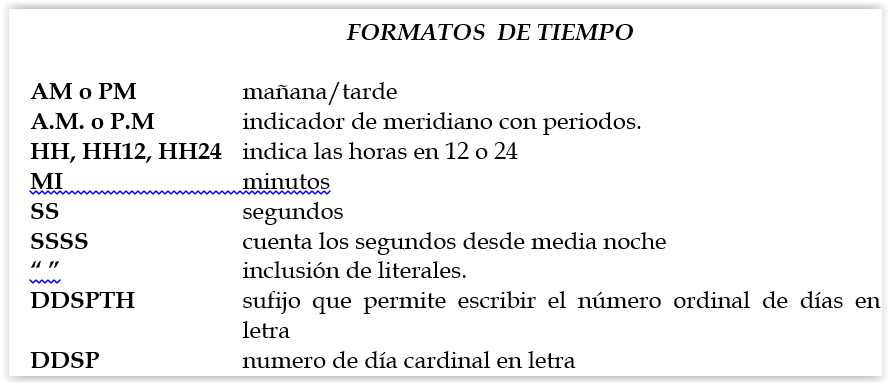
**TO\_CHAR(número,’formato’);**

**EJEMPLO:**

SELECT FECHA\_ALT,TO\_CHAR (FECHA\_ALT, 'DD-MM-YYYY') as FORMATO from emp;



## 



### 

Ejemplos:

1.- Obtener la hora del sistema

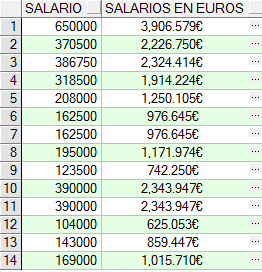
SELECT TO\_CHAR(SYSDATE,'HH24:MI:SS') as Hora

FROM DUAL



2.-Obtener el salario de los empleados de la tabla EMP en €.

SELECT SALARIO, TO\_CHAR(SALARIO/166.386,'9,999,999.999L') AS "SALARIOS EN EUROS" FROM EMP

****

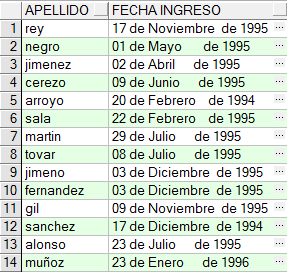
* 3.- Visualiza el nombre y la fecha de contratación de los empleados de la empresa. Formato: 17 DE MAYO DE 1987

SELECT APELLIDO

, TO\_CHAR(FECHA\_ALT, 'DD "de" Month "de" YYYY')

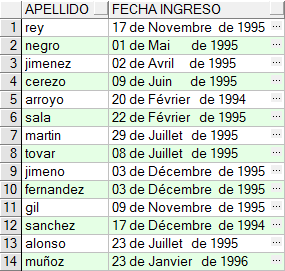
AS FECHA\_INGRESO

FROM EMP;



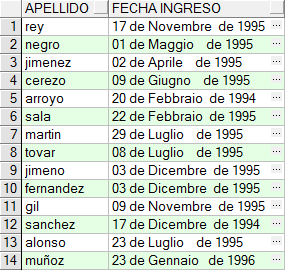
En francés:

SELECT Apellido , TO\_CHAR(FECHA\_ALT,'DD" de "Month" de "YYYY"','nls\_date\_language = FRENCH') AS "FECHA INGRESO" FROM EMP



En italiano:

SELECT Apellido , TO\_CHAR(FECHA\_ALT,'DD" de "Month" de "YYYY"','nls\_date\_language = Italian') AS "FECHA INGRESO" FROM EMP



* **TO\_ NUMBER:** Convierte una cadena de caracteres a un número.

Sintaxis:

**TO\_NUMBER(CADENA)**

SELECT TO\_NUMBER ('214324') as FORMATO from dual;



* **TO\_ DATE:** Convierte una cadena de caracteres a una fecha.

Sintaxis:

**TO\_DATE (CADENA,’formato’)**

**5. FUNCIONES GENERALES**

* **NVL** Se utiliza para convertir los valores nulos a un valor especificado.

Sintaxis:

**NVL(Columna,valor)**

Ejemplo:

SELECT apellido,salario, comision, salario + NVL(comision,0) as "salario mensual" from emp



* **DECODE** Es similar a las estructuras CASE o IF\_ THEN\_ ELSE, para facilitar las consultas condicionales.

Sintaxis:

**DECODE (COL / EXPRESIÓN, SELECCION1, RESULTADO1,**

**[SELECCION2, RESULTADO2]...[DEFAULT] )**

Ejemplo:

Seleccionar el salario con una subida del salario de un 10% a los Analistas, un 20% a los VENDEDORES y un 30% a los DIRECTORES, al resto dejar el salario tal y como lo tenían. NO MODIFICAMOS, SOLO QUEREMOS VER LOS CAMBIOS SI NOS DECIDIMOS A HACERLO

SELECT OFICIO, SALARIO

, DECODE(UPPER(OFICIO), 'ANALISTA', SALARIO \* 1.10

, 'VENDEDOR', SALARIO \* 1.20

, 'DIRECTOR', SALARIO \* 1.30

, SALARIO) AS NUEVO\_SALARIO

FROM EMP;



**6. FUNCIONES ANIDADAS**

Las funciones a nivel fila pueden anidarse hasta cualquier nivel y son ejecutadas desde el nivel central hasta el exterior.

Sintaxis:

**...F3(F2(F1(COL, ARG1), ARG2 ) , ARG3) ...**

Ejemplo:

Visualiza la fecha del próximo viernes que se lleva 6 meses con la fecha de contratación.

La fecha resultante debería tener el formato viernes, 12 marzo,1982. ordenar los resultados por fecha de contratación y crear un alias en el que figure “REVISIÓN DE LOS PRÓXIMOS 6 MESES”

SELECT APELLIDO, FECHA\_ALT

, TO\_CHAR(NEXT\_DAY(

ADD\_MONTHS(FECHA\_ALT, 6)

, 'viernes')

, 'day, dd month, yyyy')

AS REVISION

FROM EMP;

