

Le langage SQL

Les fonctions / ORACLE

La fonction TO_CHAR

Peut servir pour :

- Afficher une colonne de type numérique avec un format particulier
- Afficher une colonne de type date dans un format différent
- Extraire une partie du contenu d'une colonne de type date

Formats numériques de la fonction TO_CHAR

| Élément | Exemple | Explication |
|---------|---------|---|
| 9 | 999 | Le nombre de 9 détermine la longueur d'affichage |
| 0 | 0999 | Affiche un zéro en début de nombre |
| \$ | \$999 | Affiche \$ devant le nombre |
| B | B999 | Affiche un blanc à la place de 0 en début de nombre |
| Mi | 999Mi | Affiche - après un nombre négatif |
| Pr | 999Pr | Affiche les valeurs négatives entre < et > |
| , | 99,999 | Affiche , à ce endroit (équivalent notre .) |
| . | 999.99 | Affiche . à cet endroit (équivalent notre ,) |
| V | 999V99 | Positionne un point virtuel. |

Exemples:

1) `Select nom_emp, to_char(salaire, 'B9999')` from employé;

Affichera le salaire des employés en remplaçant le premier chiffre par un espace si celui-ci est 0.

Un salaire de 01200 serait donc affiché : ' 1200'

2) `Select nomcli, to_char(solde, '999999Pr')` from compte;

Affichera le solde des comptes en le mettant entre < et > celui-ci est négatif.

Un solde de 200 serait affiché : 200

Un solde de -200 serait affiché : <200>

Formats pour les fonctions TO_CHAR/TO_DATE

| Elément | Explication |
|------------|--|
| DD | 2 chiffres du jour |
| MM | 2 chiffres du mois |
| YYYY | 4 chiffres de l'année |
| D | Jour de la semaine (1 à 7) |
| Day | Nom du jour en 9 caractères |
| Dy | Nom du jour en 3 caractères |
| J | Julian Day (nombre de jours depuis 1/1/4712 av JC) |
| W | Le numéro de la semaine dans le mois (1 à 5) |
| WW | Le numéro de la semaine dans l'année (1 à 52) |
| Month | Nom du mois en 9 caractères |
| Mon | Nom du mois en 3 caractères |
| HH ou HH12 | Heure du jour (de 1 à 12) |
| HH24 | Heure du jour (de 1 à 24) |
| Mi | Minutes (de 0 à 59) |
| SS | Secondes (de 0 à 59) |
| SSSSS | Secondes après minuit (de 0 à 86399) |

Exemples :

- 1) Select to_char(datsal, 'MM/YYYY') from employe where noemp = 12;
En supposant que la colonne DATSAL recherchée contienne la valeur
'12/03/2009', la requête afficherait : 03/2009
- 2) Select to_char(datemb, 'Day, DD Month YYYY') from employe where
noemp = 12;
En supposant que la colonne DATEMB recherchée contienne la valeur
'05/01/2009', la requête afficherait : Lundi, 05 janvier 2009

Les fonctions numériques

| Fonction | Explication |
|-------------|---|
| Abs(n) | Donne la valeur absolue du paramètre n |
| Ceil(n) | Donne le nombre entier directement supérieur au paramètre n |
| Floor(n) | Donne le nombre entier directement inférieur au paramètre n |
| Mod(n,m) | Donne le reste de la division du paramètre n par m |
| Power (n,m) | Elève le paramètre à la puissance m |
| Round(n,m) | Arrondit la valeur du paramètre n en gardant m décimales |
| Trunc(n,m) | Tronque la valeur du paramètre n en gardant m décimales |
| Sign(n) | Evalue le signe du paramètre n (-1 négatif, 1 positif, 0) |
| Sqrt(n) | Donne la racine carrée du paramètre n |

Les paramètres (n et m) peuvent être des constantes numériques ou des colonnes de type numérique.

Exemples :

- 1) Select round(salaire, 2) from employe where noemp = 12;

En supposant que le salaire recherché soit 1533,126 Euros, la requête ci-dessous afficherait comme résultat : 1533,13 (soit arrondi au 2^{ième} décimale).

- 2) Select trunc(salaire, 2) from employe where noemp = 12;

En supposant que le salaire recherché soit 1533,126 Euros, la requête ci-dessous afficherait comme résultat : 1533,12 (soit tronqué au 2^{ième} décimale).

Les fonctions sur du texte (partie 1)

| Fonction | Explication |
|----------------|---|
| Chr(val_ascii) | Donne la caractères correspondant au code ascii |
| Ascii(t) | Donne le code ascii du paramètre t |
| Upper(t) | Transforme le paramètre t en majuscules |
| Lower(t) | Transforme le paramètre t en minuscules |
| Initcap(t) | Transforme le paramètre t avec uniquement les initiales en majuscules |
| Lpad(t,m,car) | Ajoute le caractère <i>car</i> à gauche du contenu du paramètre t jusqu'à obtention de <i>m</i> caractères. |
| Ltrim(t,car) | Supprime tous les <i>car</i> se trouvant à gauche dans le paramètre t |
| Rpad(t,m,car) | Ajoute le caractère <i>car</i> à droite du contenu du paramètre t jusqu'à obtention de <i>m</i> caractères. |
| Rtrim(t,car) | Supprime tous les <i>car</i> se trouvant à droite dans le paramètre t |

Le paramètre *t* peut être une constante ou une colonne de type texte.
Le paramètre *m* correspond à un nombre.
Le paramètre *car* correspond à 1 caractère.

Exemples :

- 1) Select upper(nom_emp), Initcap(prenom) from employe where noemp = 12;

La requête ci-dessus afficherait : DUPOND Jean

Soit, le nom de famille en majuscules et le prénom avec uniquement la première lettre en majuscule (peu importe de quelle façon le nom et le prénom aient été stockées dans la BD).

- 2) Select Rtrim(nomhotel, '*') from hotel;

La requête ci-dessus afficherait les noms d'hôtels en éliminant les éventuels caractères * qui se situeraient à la fin.

Les fonctions sur du texte (partie 2)

| Fonction | Explication |
|---------------------------|--|
| Replace(t,x,y) | Remplace la première occurrence de x par y dans le paramètre t |
| Translate(t,x,y) | Remplace toutes les occurrences de x par y dans le paramètre t |
| Substr(t,n,m) | Extrait du paramètre t la partie commençant à la position n sur une longueur de m |
| Length(t) | Donne la longueur du texte contenu dans le paramètre n |
| Decode(t,x1,y1,x2,y2,...) | Examine le contenu du paramètre t et en remplace le contenu d'après les valeurs indiquées : - si r1 - remplacement par s1 - si r2 alors remplacement par s2 - ... |

Le paramètre t peut être une constante ou une colonne de type texte.
Les paramètres n et m correspondent à un nombre.
Les paramètres x et y correspondent à une chaîne de caractère.

Exemples :

- 1) Select substr(nomemp, 1, 3) from employe where noemp = 12;

En supposant que l'employé recherché s'appelle Dupond,
la requête ci-dessus afficherait : Dup (soit les 3 premiers caractères).

- 2) Select fonction, decode(fonction, 'Commercial', 'CIAL', 'Comptable', 'COMP')

from employe where noemp= 12;

En supposant que l'employé recherché occupe la fonction de comptable,
la requête ci-dessus afficherait : COMP.

Les fonctions spécifiques pour dates

| Fonction | Explication |
|---------------------------|---|
| Add_months(dte,n) | Ajoute n mois à la date contenu dans le paramètre dte |
| Last_day(dte) | Donne le dernier jour du mois de la date |
| Months_between(dte1,dte2) | Donne le nombre de mois entre les 2 dates données en paramètre |
| Next_day(dte, jour) | Recherche la date suivante à celle donnée en paramètre qui correspond au jour indiqué |
| Sysdate | Retourne la date du jour |

Les paramètres dte et jour peuvent être une constante ou une colonne de type date.
Le paramètre n correspond à un nombre.

Exemples :

- 1) Select months_between(sysdate, datemb) from employe where noemp = 12;

La requête afficherait le nombre de mois entre la date système (date du jour) et la date d'embauche de l'employé.

- 2) Select add_months(dat_salaire, 6) date_augmentation from employe where noemp = 12;

La requête afficherait une date correspondant à la date de dernière modification de salaire + 6 mois en indiquant dans le titre que ceci constituerait la date d'augmentation.

Les fonctions complémentaires

| Fonction | Explication |
|-------------------|---|
| Greatest(n,m,...) | Retourne la valeur la plus grande parmi celles des paramètres donnés |
| Least(n,m,...) | Retourne la valeur la plus petite parmi celles des paramètres donnés |
| Nvl(t, val) | Examine le contenu du paramètre n : - s'il contient une valeur, celle-ci est restituée - sinon, la valeur de substitution <i>val</i> est donnée |

Les paramètres *n* et *m* peuvent être une constante ou une colonne de type numérique.

Le paramètre *t* peut être une colonne de n'importe quel type.

Le paramètre *val* peut être une constante de n'importe quel type.

Exemples :

- 1) Select greatest(salaire, commission) from employe where noemp = 12;

En supposant que pour l'employé recherché la colonne SALAIRE contienne la valeur 1533 et la colonne COMMISSION 1100, la requête afficherait la valeur la plus grande, soit 1533.

- 2) Select nom_emp, prenom, nvl(commission, 0) from employe where noemp IN (25, 26);

En supposant que les 2 employés recherchés soient :

Dupond Jean, ayant une commission de 1100 Euro

Frida Marie, n'ayant pas de commission (colonne ayant une valeur nulle)

La requête afficherait :

Dupond Jean 1100

Frida Marie 0