

TP 2 : Les fonctions d'oracle

OBJECTIFS :

Manipuler quelques fonctions prédéfinies d'oracle.

Partie I: La table DUAL

DUAL d'Oracle est une pseudo-table composée d'une ligne et d'une colonne. Elle permet d'effectuer des select sans utiliser de table particulière pour tester des fonctions (tests unitaires) ou récupérer des informations indépendantes des données mais liées à la base ou à Oracle (date système, séquence de valeurs, etc.)

```
SELECT * FROM DUAL;  
SELECT SYSDATE FROM DUAL;
```

Partie II: Fonctions pour les caractères

| Fonction | Définition | Exemple | Résultat |
|--|---|-----------------------------|-----------|
| LENGTH (<i>column</i> <i>expr</i>) | Permet de récupérer le nombre de caractères d'une chaîne. LENGTH retourne une valeur de type NUMBER. | LENGTH ('Bonjour') | 7 |
| SUBSTR (<i>column</i> <i>expr</i> , <i>m</i> [<i>n</i>]) | Permet d'extraire une chaîne de caractères de la chaîne de caractère <i>column</i> (ou issue de <i>expr</i>) sur une longueur <i>n</i> à partir de la position <i>m</i> . | SUBSTR ('Bonjour',1,3) | Bon |
| INSTR (<i>column</i> <i>expr</i> , <i>c</i>) | Permet de récupérer la position de la première occurrence du caractère <i>c</i> dans la chaîne de caractères <i>column</i> ou issue de <i>expr</i> . | INSTR('Bonjour','j') | 4 |
| LPAD (<i>column</i> <i>expr</i> , <i>n</i> , ' <i>string</i> ') | Permet de placer <i>n</i> caractères de type <i>string</i> à gauche de la valeur de <i>column</i> (ou <i>expr</i>). | LPAD(sal, 10, '*') | *****5000 |
| RPAD (<i>column</i> <i>expr</i> , <i>n</i> , ' <i>string</i> ') | Permet de placer <i>n</i> caractères de type <i>string</i> à droite de la valeur de <i>column</i> (ou <i>expr</i>). | RPAD(sal, 10, '*') | 5000***** |
| CONCAT (<i>column1</i> <i>expr1</i> , <i>column2</i> <i>expr2</i>) | Permet de concaténer la valeur de la première chaîne à la valeur de la seconde chaîne. (équivalent à l'opérateur " "). | CONCAT ('Bon', 'jour') | Bonjour |
| TRIM (<i>leading</i> <i>trailing</i> <i>both</i> , <i>trim_character</i> FROM <i>trim_source</i>) | Permet de couper les caractères <i>trim_character</i> en entête (<i>leading</i>), en fin (<i>trailing</i>) ou les deux (<i>both</i>) d'une chaîne de caractère <i>trim_source</i> . | TRIM ('S' FROM 'SSMITH') | MITH |

| Fonctions | Définition | Exemples | Resultat |
|--------------------------------|--|-------------------------|--------------|
| INITCAP (column expr) | Convertit la première lettre de chaque mot d'une chaîne de caractères en majuscule et les autres lettres en minuscule. | INITCAP('Cours de SQL') | Cours De Sql |
| LOWER (column expr) | Convertit une chaîne de caractères en minuscule. | LOWER('Cours de SQL') | cours de sql |
| UPPER (column expr) | Convertit une chaîne de caractères en majuscule. | UPPER ('Cours de SQL') | COURS DE SQL |

| Fonction | Objectif | Exemple |
|-----------|---|---------------------------------|
| ASCII (c) | Retourne le caractère ASCII équivalent. | ASCII ('A') donne 65 |
| CHR (n) | Retourne le caractère équivalent dans le jeu de caractères de la base ou du jeu national (NLS). | CHR (161) CHR (162) donne ;¢ |

Application:

1. Afficher les clients dont le longueur de leurs nom est supérieur à 5 caractères:

```
.....
.....
```

2. Afficher les trois premiers caractères de noms de tous les clients:

```
.....
.....
```

3. Afficher les produits dont la lettre 'a' des libellés se trouve à la position 4 ou bien 3 ou bien 7:

```
.....
.....
```

4. Afficher les adresses des clients sur 15 positions (compléter l'affichage avec 3 caractères '#' gauche et le reste avec des '*' à droite)

```
###Tunis*****
```

```
###Sousse*****
```

```
.....
.....
```

5. L'espace est un caractère invisible, supprimer le pendant l'affichage des adresses des clients.

```
.....
.....
```

6. Afficher la première lettre des adresses des clients en majuscule et le reste en minuscule:

```
.....
.....
```

7. Afficher tous les clients qui habitent à 'Tunis' (quelque soit l'écriture, minuscule ou bien majuscule: TUNIS, tunis, TuniS,...)

```
.....
.....
```

8. Afficher la deuxième lettre et son code ASCII des libellés des produits:

```
.....
.....
```

Partie III: Fonctions pour les numériques

| Fonction | Objectif | Exemple |
|-------------------------------|---|--------------------------------|
| ABS (<i>n</i>) | Valeur absolue de <i>n</i> . | |
| FLOOR (<i>n</i>) | Plus grand entier \leq à <i>n</i> . | FLOOR (15.7) donne 15. |
| MOD (<i>m</i> , <i>n</i>) | Division entière de <i>m</i> par <i>n</i> . | |
| POWER (<i>m</i> , <i>n</i>) | <i>m</i> puissance <i>n</i> . | |
| ROUND (<i>m</i> , <i>n</i>) | Arrondi à une ou plusieurs décimales. | ROUND (17.567, 2) donne 17,57. |
| SQRT (<i>n</i>) | Racine carrée de <i>n</i> . | |
| TRUNC (<i>n</i> , <i>m</i>) | Coupe de <i>n</i> à <i>m</i> décimales. | TRUNC (15.79, 1) donne 15.7. |

Application: Donner le resultat d'execution des requetes suivantes:

```
SELECT MOD(5, 2) * TRUNC(4.258, 2) + TRUNC(3.258, 0)
FROM DUAL;
.....
```

```
SELECT ABS(FLOOR(10.256) * (ROUND(-4.258, 2) - SQRT(4)))
FROM DUAL;
.....
```

Partie IV: Fonctions pour les valeurs nulles

| Fonction | Objectif | Exemple |
|--|--|---|
| NULLIF (<i>expr1</i> , <i>expr2</i>) | Si <i>expr1</i> = <i>expr2</i> retourne NULL, sinon retourne <i>expr1</i> . | NULLIF('Raffarine', 'Parafine') retourne 'Raffarine'. |
| NVL (<i>expr1</i> , <i>expr2</i>) | Convertit <i>expr1</i> susceptible d'être nulle en une valeur réelle (<i>expr2</i>). | NVL(grade, 'Aucun !') retourne 'Aucun !' si grade est NULL. |

Application:

1. Donner la requête permettant d'obtenir l'affichage suivant:

| |
|--------------------------|
| Client Sans Adresse ...! |
| Sousse |
| Client Sans Adresse ...! |
| Sfax |

```
.....
```

2. Afficher NULL si le longueur du nom de client est égale à son adresse:

```
.....
```

Partie V: Fonctions pour les dates

| Fonction | Objectif | Retour |
|--|---|--------|
| ADD_MONTHS | Ajoute des mois à une date | DATE |
| CURRENT_DATE | Retourne la date courante (calendrier grégorien) dans la session et le fuseau de la base. | DATE |
| EXTRACT ({ YEAR MONTH DAY HOUR MINUTE SECOND } FROM { d i }) | Extrait une partie donnée d'une date ou d'un intervalle. | NUMBER |
| LAST_DAY (d) | Retourne le dernier jour du mois | DATE |
| MONTHS_BETWEEN (d1 , d2) | Retourne le nombre de mois entre deux dates (d1 et d2 avec d1 > d2). | NUMBER |
| NEW_TIME (d , z1 , z2) | Retourne la date d exprimée en zone z1 dans la zone z2. | DATE |
| NEXT_DAY (d , jour) | Retourne la date du prochain jour ouvrable (exemple jour ' LUNDI ') à partir de d. | DATE |
| ROUND (d , format) | Arrondit une date d selon un format (exemple : ' YEAR '). | DATE |
| SYSDATE | Date courante (du système). | DATE |
| TRUNC (d , format) | Tronque une date d selon un format (exemple : ' YEAR '). | DATE |

Opérations arithmétiques sur les dates:

| Opération | Résultat | Description |
|------------------|-----------------|---|
| date + nombre | date | Ajoute un nombre de jours à une date |
| date - nombre | date | Soustrait un nombre de jours à une date |
| date - date | nombre de jours | Soustrait une date à une autre date |
| date + nombre/24 | date | Ajoute un nombre d'heures à une date |

Application:

1. Afficher la date du mardi prochain:

```
.....
```

2. Afficher la date après une semaine:

```
.....
```

3. Afficher le dernier jour et le dernier jour du mois en cours:

```
.....
```

4. Rendez-vous dans 4 mois:

```
.....
```

5. Numéro du mois d'il y a 65 jours ?

```
.....
```

6. Afficher l'année en cours ?

```
.....
```

Partie VI: Fonctions de conversion

Oracle autorise des conversions de types implicites ou explicites:

Implicites

| De | A |
|------------------|----------|
| VARCHAR2 ou CHAR | NUMBER |
| VARCHAR2 ou CHAR | DATE |
| NUMBER | VARCHAR2 |
| DATE | VARCHAR2 |

Le script suivant ne renvoie aucune erreur :

```
CREATE TABLE Test (c1 NUMBER, c2 DATE, c3 VARCHAR2(1), c4 CHAR);
INSERT INTO Test VALUES ('548,45', '13-05-2003', 3, 5);
```

Explicites

Les fonctions de conversion les plus connues sont TO_NUMBER, TO_CHAR et TO_DATE.

| Fonction | Définition |
|--|---|
| TO_CHAR(<i>number</i> [, <i>date</i> , [<i>fnt</i>], [<i>nlsparams</i>]]) | Convertit un nombre ou une date en une chaîne de caractères |
| TO_NUMBER (<i>char</i> [<i>'format'</i>]) | Convertit une chaîne de caractères en un nombre |
| TO_DATE (<i>char</i> [, <i>'format'</i>]) | Convertit une chaîne de caractères en une date |

Les formats:

| | | | |
|-------------|------------------------------------|--------------|--|
| YYYY | Année sur 4 chiffres | HH | Heure sur 12 heures |
| YYY | Les 3 derniers chiffres de l'année | HH24 | Heure sur 24 heures |
| YY | Les 2 derniers chiffres de l'année | MI | Minutes |
| Y | Le dernier chiffre de l'année | SS | Secondes |
| Q | Numéro du trimestre dans l'année | J | Jour du calendrier julien |
| WW | Numéro de semaine dans l'année | YEAR | Année en toute lettres |
| W | Numéro de semaine dans le mois | MONTH | Nom du mois |
| MM | Numéro du mois | MON | Nom du mois abrégé sur les 3 premières lettres |
| DDD | Numéro du jour dans l'année | DAY | Nom du jour |
| DD | Numéro du jour dans le mois | DY | Nom du jour abrégé sur les 3 premières lettres |
| D | Numéro du jour dans la semaine | AM | Indication AM |
| | | PM | Indication PM |

Application:

1. Afficher la date sous la forme '23 Octobre 2014':

.....

2. Afficher la date et l'heure complète sous la forme 19/10/2014 21:06:38:

.....

3. Afficher votre âge en nombre de jours:

.....

4. Afficher votre âge en nombre de mois:

.....

5. Afficher le jours de votre date de naissance:

.....

6. Afficher les commandes créés entre 2000 et 2010:

.....

7. Afficher 'Montant vide' pour les factures dont le montant est NULL:

.....