

# Administration de la Base de Données Oracle

## TP1 : Manipulation d'une BD à l'aide de SQL\*Plus et formatage des données

### Objectifs du TP

Les objectifs de ce TP sont les suivants :

- Manipuler les données de la BD en utilisant SQL\*Plus
- Manipuler le format des données et des rapports sous SQL\*Plus

NB : Le schéma utilisé dans le TP est le schéma HR prédéfini dans Oracle Database

---

### Exercice 1 : Préparation de l'utilisateur HR

- 1) Depuis CMD, se connecter à SQL\*Plus sans utilisateur

```
> sqlplus /nolog
```

- 2) Se connecter en tant que sysdba

```
SQL > connect / as sysdba
```

- 3) Afficher tous les utilisateurs

```
SQL > select * from all_users ;
```

- 4) Afficher les informations sur les comptes utilisateurs. Remarque ?

```
SQL > select * from dba_users ;
```

- 5) Afficher les paramètres d'affichage

```
SQL > show linesize pagesize
```

- 6) Modifier les paramètres d’affichage

```
SQL > set linesize 200  
SQL > set pagesize 1000
```

- 7) Afficher les informations sur l’utilisateur HR

```
SQL > select * from dba_users where username='HR' ;
```

- 8) Afficher la description de la table DBA\_USERS

```
SQL > desc dba_users
```

- 9) Afficher le statut de l’utilisateur HR

```
SQL > select account_status from dba_users where username= 'HR' ;
```

- 10) Activer l’utilisateur HR et définir son mot de passe

```
SQL > alter user HR account unlock;  
SQL > select account_status from dba_users where username= 'HR' ;  
SQL > alter user HR identified by HR;  
SQL > select account_status from dba_users where username= 'HR' ;
```

## Exercice 2 : Manipulation des données dans SQL\*Plus

- 1) Essayer de démarrer l’instance oracle

```
SQL > startup
```

**Remarque** : Une erreur va s’afficher si l’instance est déjà démarrée.

- 2) Se déconnecter de l’utilisateur sysdba

```
SQL > disconnect
```

- 3) Se connecter avec l'utilisateur HR

```
SQL > connect HR/HR
```

**Attention :** Le mot de passe est sensible à la casse.

- 4) Afficher l'utilisateur courant

```
SQL > show user
```

- 5) Afficher les tables de l'utilisateur HR

```
SQL > select * from tab ;
```

- 6) Afficher la description de la table employees

```
SQL > desc employees
```

- 7) Afficher les données de la table employees

```
SQL > select first_name, last_name, salary, hire_date from employees;
```

- 8) Afficher la description des tables jobs, departments et countries.

- 9) Afficher les noms et prénoms des employés qui travaillent comme 'Programmer'.

- 10) Quel est le département où travaillent les employés dont la fonction est 'Sales Manager' ?

- 11) Afficher les noms et prénoms des employés qui travaillent à 'London'.

- 12) Sortir de SQL\*Plus

```
SQL > exit
```

## Exercice 3 : Formatage des données et des rapports sous SQL\*Plus

### 1. Formatage des entêtes

1) Exécuter les commandes ci-dessous. Quels sont vos constats ?

```
SQL> COLUMN FIRST_NAME HEADING 'FIRST NAME'
SQL> COLUMN LAST_NAME HEADING 'LAST NAME '
SQL> COLUMN HIRE_DATE HEADING 'HIRE DATE'
SQL> SELECT FIRST_NAME, LAST_NAME, SALARY, HIRE_DATE FROM EMPLOYEES;
```

```
SQL> select * from countries;
SQL> Select * from regions;
SQL> COLUMN REGION_ID HEADING 'IDENTIFIANT REGION'
SQL> select * from countries;
SQL> select * from regions;
```

2) Exécuter les commandes ci-dessous. Quels sont vos constats ?

```
SQL> COLUMN REGION_ID HEADING 'IDENTIFIANT|REGION'
SQL> select * from countries;
SQL> select * from regions;
```

```
SQL> SET UNDERLINE +
SQL> select * from regions;
SQL> select * from jobs;
```

## 2. Formatage des colonnes :

- 1) Exécuter la commande ci-dessous. Qu'est-ce que vous remarquez ?

```
SQL> SELECT FIRST_NAME, LAST_NAME, SALARY, HIRE_DATE FROM EMPLOYEES;
```

Exécuter la commande ci-dessous, puis relancer la commande précédente. Qu'est-ce que vous constatez ?

```
SQL> COLUMN HIRE_DATE FORMAT A10  
SQL>/
```

- 2) Lancer la requête ci-dessous

```
SQL> SELECT LAST_NAME, SALARY, COMMISSION_PCT  
SQL> FROM EMP_DETAILS_VIEW  
SQL> WHERE JOB_ID='SA_MAN';
```

Modifier ensuite le format de la colonne SALARY pour afficher les valeurs de cette colonnes selon le modèle \$99,990

```
SQL> COLUMN SALARY FORMAT $99,990  
SQL> /
```

## 3. Commande WRAP:

- 1) Effacer les paramètres de colonne précédents

```
SQL> CLEAR COLUMNS
```

Vérifier le statut du paramètre RECSEP et le désactiver.

```
SQL> show RECSEP  
SQL> SET RECSEP OFF
```

Vérifier les paramètres d’affichage puis mettre le paramètre linesize à 100.

Exécuter la requête ci-dessous:

```
SQL> SELECT LAST_NAME, JOB_TITLE, CITY  
SQL> FROM EMP_DETAILS_VIEW  
SQL> WHERE SALARY>12000;
```

Positionner la largeur de la colonne JOB\_TITLE à 22 puis exécuter de nouveau la requête précédente. Qu’est-ce que vous remarquez ?

```
SQL> COLUMN JOB_TITLE FORMAT A22  
SQL> /
```

Activer le paramètre RECSEP avec l’option WRAPPED, puis réexécuter la requête. Qu’est-ce que vous remarquez ?

```
SQL> SET RECSEP WRAPPED  
SQL> /
```

Activer le paramètre WORD\_WRAPPED sur la colonne JOB\_TITLE puis relancer la même requête. Observer le résultat.

```
SQL> COLUMN JOB_TITLE WORD_WRAPPED  
SQL> /
```

Changer le paramètre RECSEP à EACH puis relancer la même requête. Observer le résultat.

```
SQL> SET RECSEP EACH  
SQL> /
```

Changer la valeur du paramètre RECSEPCHAR puis réexécuter la requête :

```
SQL> SET RECSEPCHAR "***"  
SQL> /
```

Positionner la valeur du paramètre RECSEPCHAR à vide et réexécuter la requête :

```
SQL> SET RECSEPCHAR ""  
SQL> /
```

#### 4. Commande BREAK

1) Désactiver le parameter RECSEP

```
SQL> SET RECSEP OFF
```

Exécuter la requête suivante :

```
SQL> SELECT DEPARTMENT_ID, LAST_NAME, SALARY  
SQL> FROM EMP_DETAILS_VIEW  
SQL> WHERE SALARY > 12000  
SQL> ORDER BY LAST_NAME;
```

Faire un BREAK sur la colonne DEPARTMENT\_ID et réexécuter la requête. Qu'est-ce que vous remarquez ?

```
SQL> BREAK ON DEPARTMENT_ID  
SQL> /
```

Exécuter la requête suivante :

```
SQL> SELECT DEPARTMENT_ID, LAST_NAME, SALARY  
SQL> FROM EMP_DETAILS_VIEW  
SQL> WHERE SALARY > 12000  
SQL> ORDER BY DEPARTMENT_ID;
```

2) Supprimer les paramètres BREAK

```
SQL> CLEAR BREAKS
```

Lancer successivement les requêtes et commandes ci-dessous :

```
SQL> SELECT DEPARTMENT_ID, JOB_ID, LAST_NAME, SALARY  
SQL> FROM EMP_DETAILS_VIEW  
SQL> WHERE SALARY>12000  
SQL> ORDER BY DEPARTMENT_ID, JOB_ID;
```

```
SQL> BREAK ON DEPARTMENT_ID  
SQL> /
```

```
SQL> BREAK ON JOB_ID  
SQL> /
```

Qu'est-ce que vous constatez ?

Lancer les commandes ci-dessous.

```
SQL> BREAK ON DEPARTMENT_ID ON JOB_ID  
SQL> /
```

```
SQL> BREAK ON DEPARTMENT_ID SKIP PAGE ON JOB_ID SKIP 1  
SQL> /
```

Ajouter un titre pour pouvoir identifier la page :

```
SQL> TTITLE 'Liste des salariés par fonction et par département'  
SQL> /
```

Supprimer les paramètres de titre et de break :

```
SQL> TTITLE OFF  
SQL> CLEAR BREAKS  
SQL> /
```



## 5. Commande COMPUTE

1) Exécuter la requête :

```
SQL> SELECT JOB_ID, LAST_NAME, SALARY  
SQL> FROM EMP_DETAILS_VIEW  
SQL> WHERE SALARY>12000  
SQL> ORDER BY JOB_ID;
```

Afficher le salaire moyen par fonction.

```
BREAK ON JOB_ID SKIP 1  
COMPUTE avg LABEL 'Salaire moyen' OF SALARY ON JOB_ID
```

2) Afficher le salaire moyen de l'entreprise.

```
BREAK ON REPORT  
COMPUTE avg LABEL 'Salaire moyen' OF SALARY ON REPORT
```

### Astuce

*Pour afficher correctement 'Salaire moyen' et pas en dessous de la colonne job, utiliser la colonne DUMMY comme suit :*

```
COLUMN DUMMY FORMAT A20 HEADING '  
SELECT NULL DUMMY, JOB_ID, LAST_NAME, SALARY  
2 FROM EMP_DETAILS_VIEW  
3 WHERE SALARY>12000  
4 ORDER BY JOB_ID;
```

## 6. Commande SPOOL

- 1) Activer le SPOOL vers un fichier au choix, en testant les 3 options CREATE, REPLACE, APPEND.

Faire des manipulations reprenant les différentes commandes de formatage vues et testées précédemment.

- 2) Arrêter le SPOOL et vérifier le fichier de sortie.