

EISTI 2009-2010 – TD/TP de BDD 1 – Séance 4

Client Oracle SQLPLUS – Script de création SQL

Exercice 1 : Prise en main Oracle :

1. Connection à la BDD Oracle de l'EISTI.

Vous utiliserez pour vos TP's et Projets la base de données Oracle de l'école. Un espace vous y est réservé pour créer vos tables et gérer vos données. Votre login est celui de l'école et votre mot de passe est identique à votre login.

Pour se connecter au serveur, on utilise un logiciel client SQL*PLUS qui vous permettra ensuite d'exécuter des séquences d'instructions du langage SQL (LDD, LMD, LCD, LCT). Tapez en ligne de commande :

```
sqlplus mon_login@bddetu
```

Puis entrer votre mot de passe. Une fois connecté, vous pouvez le changer :

```
alter user mon_login identified by mon_nouveau_mot_de_passe;
```

2. Exécution d'un script SQL.

Une fois connecté, vous pouvez taper directement vos ordres dans SQL*PLUS à la suite du prompt en validant chaque instruction par un ';'.

Il est cependant préférable de les éditer dans un fichier séparé et les charger ensuite dans SQL*PLUS avec la commande start ou @ :

- start script
- start script.sql
- @script
- @script.sql

Exercice 2 : Gestion de séjour, du MLD vers la BDD

1. Ecrire dans un fichier **tables_sejour.sql** le script de création des tables à partir du MLD du TD précédent sur les séjours en hôtel.

Exécuter votre script.

2. Pour connaître ou retrouver les différentes structures que vous avez créées, vous disposez de plusieurs vues :

- `user_objects`
- `user_tables`
- `user_users`
- `user_views`
- `user_constraints`

Affichez la description de chacune des vues avec la commande `desc`. Par exemple,
`desc user_objects`

Affichez le contenu des vues avec la commande SQL `select`. Par exemple,
`select * from user_objects ;`

3. Vous devez avoir retrouvé les tables créées avec le script. Afficher la description et le contenu de chaque table avec `desc` et `select`.

4. Insérer quelques données dans la table `Hotel` :

```
insert into Hotel values ('H1', 'Hôtel de la Plage') ;  
insert into Hotel values ('H2', 'Hôtel des Pyrénées') ;
```

Vérifier maintenant que la table `Hotel` contient bien des données.

5. Vérifier la contrainte de clé primaire en ajoutant un hôtel avec un identifiant déjà pris :

```
insert into Hotel values ('H2', 'Hôtel de la Gare') ;
```

6. Si vous constatez des problèmes avec la structure de vos tables, modifier votre script et recharger le. Pensez à supprimer vos tables au début du script pour tout reprendre à zéro.

Exercice 3 : Commandes Utiles sous SQL*PLUS

1. Rappel dernière commande.

Vous pouvez rappeler le dernier ordre SQL avec la commande `/`

Attention, cela n'est pas valable avec les commandes SQL*PLUS (`start`, `desc`, etc ...).

2. Edition dernière commande.

Vous pouvez éditer la dernière commande si elle est complexe avec la commande `ed`. Cela ouvre un éditeur externe (`vi`, `emacs`). Une fois vos modifications effectuées, sauver le fichier. Vous vous retrouvez sur SQL*PLUS où vous pouvez relancer la dernière commande modifiée en tapant `/`

Vous pouvez régler cet éditeur par défaut en tapant sous SQL*PLUS :

```
SQL> DEFINE _EDITOR=vi
SQL> DEFINE _EDITOR='xemacs -nw'
```

3. Personnalisation SQL*PLUS.

Vous disposez d'un certain nombre de variables d'environnements pour personnaliser SQL*PLUS. Parmi celles-ci, on peut citer :

- `pagesize` : longueur d'une page en nombre de lignes
- `linesize` : taille d'une ligne en nombre de caractères
- `sqlprompt` : chaîne de caractères de l'invite utilisateur

Pour voir le contenu d'une variable : `show nom_variable`

Pour changer la valeur d'une variable : `set nom_variable valeur`

4. Réglages automatiques.

On peut charger automatiquement tous ces réglages au démarrage de SQL*PLUS en les ajoutant dans un des fichiers suivants :

- `glogin.sql` : commun à toutes les sessions et utilisateurs, vous le trouvez dans le répertoire :
 - `$ORACLE_HOME/sqlplus`
- `login.sql` : fichier à placer dans le répertoire du projet courant. Les réglages ne seront valables que pour ce projet

5. Sauvegarde de Requêtes.

Vous avez la possibilité de tester vos instructions SQL dans SQL*PLUS jusqu'à ce qu'ils soient corrects et ensuite les insérer dans votre script.

```
SQL> save nom_fichier
```

La dernière requête exécutée est stockée dans le fichier `nom_fichier.sql`. Si le fichier existe déjà, vous pouvez ajouter la requête en fin de fichier (mode Append) :

```
SQL> save nom_fichier APP
```

6. Sauvegarde de Résultats de Requêtes.

Vous pouvez sauver le résultat de vos requêtes dans un fichier. Pour commencer la sauvegarde dans le fichier `nom_fichier.sql` :

```
SQL> spool nom_fichier
```

Pour arrêter la sauvegarde dans un fichier :

```
SQL> spool off
```