

TP1 Base de Données Avancée

Assurez-vous d'avoir pris connaissance du petit guide [SQLPlus](#).

Vous trouverez des rappels sur PL/SQL sur le site <http://www.techonthenet.com/oracle/>

Connexion

L'accès au serveur Oracle se fait depuis Turing ou master-info (même login, même mot de passe). Attention, votre compte oracle est différent de votre accès à turing. Par défaut, votre login oracle est votre login turing sans le chiffre 1 initial (et le tout en majuscule). Votre mot de passe initial pour le compte oracle est votre login oracle.

Pour pouvoir se connecter, il faut avoir les variables d'environnement suivantes:

```
ORACLE_HOME=/opt/oracle/product/10.2.0/
```

```
ORACLE_BASE=/opt/oracle
```

```
ORACLE_SID=ROSA
```

```
TWO_TASK=ROSA
```

Ces variables sont théoriquement présentes automatiquement pour tous les utilisateurs (mais en cas de problème il n'est pas inutile de vérifier).

1 Transactions et droits d'accès

Exercice 1

1. Se connecter au même compte à partir d'une autre fenêtre (ou écran sur un autre poste de travail).
 2. Créer dans l'une des deux fenêtres ouvertes une table et y insérer dans chacune des fenêtres une ligne. Que voit-on de la table créée et des lignes insérées à partir de chacun des deux écrans ? Conclusion.
 3. Annuler l'une des insertions précédentes en écrivant `ROLLBACK;`. Que s'est-il passé ?
 4. Valider la ligne insérée par un `COMMIT` à partir de l'écran par lequel n'a pas eu lieu son insertion. Que constate-t-on ?
 5. Valider la ligne insérée par un `COMMIT` à partir de l'écran de son insertion. Conclusion.
 6. Adjoindre à votre table une clé. Insérer dans la table à partir des deux fenêtres une ligne ayant la même valeur pour chaque attribut de la clé. Qu'arrive-t-il? Émettre un `ROLLBACK`. Que devient le blocage ?
 7. Clore la session dans la fenêtre d'insertion de la ligne par un `EXIT` ou un `QUIT`. Que constate-t-on dans la fenêtre restante? Faire encore une mise à jour dans la table à partir de la fenêtre restante et sortir.
 8. Ouvrir une nouvelle session en utilisant une seule fenêtre. La dernière transaction a-t-elle été validée ?
 9. Insérer encore une ligne, puis créer une nouvelle table, y insérer une ligne. Émettre un `ROLLBACK`. Que sont devenues les deux tables et les deux lignes insérées ?
 10. Successivement insérer encore une ligne dans la première table, éliminer la dernière table créée et émettre un nouveau `ROLLBACK`. Qu'est devenue la dernière ligne insérée ?
- En conclusion, comment est validée une transaction ?

2 Optimisation de Requêtes

On rappelle la syntaxe pour écrire et voir le plan d'exécution d'une requête avec Oracle :

```
explain plan for votre-requete-sql
```

Après l'exécution de EXPLAIN PLAN, Oracle écrit dans la table "plan_table" les informations sur le plan d'exécution. Pour la consulter, écrire :

```
select lpad(' ', 2*(level-1)) || operation ||' '||options||' ' || object_name
||' '||decode(id, 0, 'Cost = ' ||position)
"test plan"
from plan_table
start with id = 0
connect by prior id = parent_id;
```

Exercice 2

1. Créer une table `Livre(Nom varchar(32), Identifiant integer)` ;
2. Écrire une requête SQL pour savoir les n-uplets de `Livre` avec `Identifiant = 5`. Voir le plan d'exécution donné par Oracle et le commenter.
3. Comment est-ce que le plan d'exécution change si on ajoute un index `livre_idx` sur `Livre(Identifiant)` ?
4. Ajouter une contrainte d'unicité sur `Identifiant`. Comment est-ce que le plan d'exécution change ? Qu'est-ce qui se passe si, à la place d'ajouter un index, on met `Identifiant` comme clé principale ? Est-ce que le plan d'exécution change ?
5. Écrire une requête pour récupérer seulement les identifiants des livres avec `Identifiant` compris entre 5 et 50. Voir le plan d'exécution donné par Oracle et le commenter.
6. Qu'est-ce qui se passe si on demande tous les n-uplets avec `Identifiant` compris entre 5 et 50 ?

Exercice 3

Soient les tables `Livre(Nom varchar(32), Identifiant integer)` et `Acheteur(Nom varchar(32), Livre_Id integer)`.

1. Commenter le plan d'exécution de Oracle pour la requête suivante :

```
SELECT l.Indentifiant
FROM Livre l, Acheteur a
WHERE l.Indentifiant = a.Livre_Id
AND l.Indentifiant > 5;
```

2. Qu'est-ce qui se passe si on change la condition de jointure en demandant que l'attribut `Identifiant` soit plus grand que l'attribut `Livre_Id` ?
3. Qu'est-ce qui se passe, par rapport à (2), si on ajoute une contrainte d'unicité sur l'attribut `Livre_Id` ?