

## Administration des SGBD

### TP Oracle

---

Après chaque TP, un compte rendu doit être rédigé contenant les points intéressants abordés à chaque question :

- ✓ Requêtes SQL correspondantes aux questions posées
- ✓ Résultat de l'exécution de la requête
- ✓ Réponses aux questions soulevées dans le sujet,
- ✓ Explication des parties non triviales des requêtes complexes,
- ✓ Analyse et commentaire des jeux d'essais.

Les rapports de TP doivent être envoyés au plus le 25 Décembre 2025.  
rapports sont notés

---

### TP N° 1

#### **Création et manipulation d'une BD**

Ce TP concerne la création, la manipulation et l'interrogation d'une BD ainsi que les contraintes d'intégrité statiques, l'évolution de schéma ...etc.

Il est conseillé de créer un utilisateur ainsi que les objets nécessaires de façon à obtenir plus de souplesse dans l'administration.

La création d'un utilisateur nécessite au minimum deux tablespaces, un par défaut ainsi qu'un temporaire. Ces opérations doivent être effectuées en tant que DBA (Data Base Administrateur).

- **Création de Tablespace**

```
CREATE TABLESPACE xxx_tbs DATAFILE 'C:\tbs_xxx.dat' SIZE 100M AUTOEXTEND ON ONLINE;
```

Description :

|                                     |  |
|-------------------------------------|--|
| CREATE TABLESPACE xxx_tbs           | Spécifie le nom du tablespace de la base.                                      |
| DATAFILE 'C:\tbs_xxx.dat' SIZE 100M | Spécifie le nom complet du fichier système ainsi que sa taille en méga octets. |
| AUTOEXTEND ON                       | La taille est augmentée automatiquement en cas de saturation                   |
| ONLINE                              | Disponible immédiatement suite à sa création                                   |

- **Création de Tablespace Temporaire**

```
CREATE TEMPORARY TABLESPACE xxx_TempTBS TEMPFILE 'C:\temp_xxx.dat' SIZE 100M AUTOEXTEND ON;
```

- **Création des utilisateurs**

```
Create User Nom_User Identified by MotDePasse Default Tablespace Nom_Tablespace  
Temporary Tablespace Nom_Temp_Tablespace;
```

- **Attribution de privilèges**

```
GRANT ALL privileges to Nom_User
```

- **Création de Table**

```
CREATE TABLE [schéma.] nomTable (colonne1 type1 [DEFAULT valeur1] [NOT NULL] [, colonne2 type2 [DEFAULT valeur2]  
[NOT NULL] ] [CONSTRAINT nomContrainte1 typeContrainte1]);
```

## Contraintes :

-UNIQUE (colonne1 [,colonne2])  
-PRIMARY KEY (colonne1 [,colonne2])  
-FOREIGN KEY (colonne1 [,colonne2]) REFERENCES [schma.]nomTablePere (colonne1 [,colonne2]) [ON DELETE { CASCADE | SET NULL }]  
-CHECK (condition)

### Modification de schéma

ALTER TABLE nomTable DISABLE CONSTRAINT nomContrainte;  
ALTER TABLE nomTable ENABLE CONSTRAINT nomContrainte [EXCEPTIONS INTO tableErreurs];

### Création de la table Erreur

CREATE TABLE TableErreurs (adresse ROWID, utilisateur VARCHAR2(30), nomTable VARCHAR2(30), nomContrainte VARCHAR2(30));

## Travail demandé

### Partie I : Création des TableSpaces et des utilisateurs

1. Créer deux Tablespaces **VotreNom\_TBS** et *Temporary Tablespace* **VotreNom2\_TempTBS**
2. Créer un utilisateur **TP1\_VotreNom** en lui attribuant les deux tablespaces créés précédemment
3. Donner tous les privilèges à cet utilisateur.

### Partie II : Langage de définition de données

4. Proposer une base de données de 6 tables au minimum en précisant la clé primaire et les éventuelles clés étrangères de chacune de ces table. Au moins une table avec une clé primaire composée de deux attributs ou plus.  
**C1:** Posez une contraintes sur les différentes valeurs que peut prendre un attribut. **Exp1:** l'attribut «CodeSpécialité» peut être: {RSD, SII, SSI, IL}  
**C2:** Posez une contrainte sur la valeur d'un attribut comme par exemple Nombre étudiant dans une section < 100
5. Créer les relations de la base de données avec toutes les contraintes d'intégrité.
6. Ajouter un attribut à une de vos tables .
7. Ajouter la contrainte not null pour des attributs de deux relations différentes.
8. Modifier la longueur d'un attribut (agrandir, réduire).
9. Supprimer une colonne dans une table. Vérifier la suppression.
10. Renommer une colonne dans une table. Vérifier.
11. Ajouter la contrainte spécifiant qu'un attribut non déclaré clé étrangère fait référence à la clé d'une autre table
12. Ajouter la contrainte intra-relation (deux attributs de la même relation) sur une table. **Exp:** Employé : la date de naissance < date de recrutement.
13. Ajouter la contrainte inter-relation (un attribut d'une table est lié par une condition avec un attribut d'une autre table).

### Partie III : Langage de manipulation de données

14. Remplir toutes les tables par au minimum 7 instances dans chaque tables en incluant des tuples qui provoquent des erreurs et expliquer les problèmes dans chaque cas.
15. Écrire une requête qui permet d'augmenter pour un ensemble de tuples la valeur d'un attribut comme dans exemple **C2** ; ajouter 10 étudiants aux sections IL et RSD. Que remarquez vous?. Désactiver la contrainte pour autoriser la modification. Réactiver la contrainte.

16. Supprimer les tuples d'une des tables ayant des clés qui sont des clés étrangères dans une autre table. Quels sont les problèmes rencontrés. **Exp:** Supprimer des étudiants tandis qu'il existe le même matricule dans la table Note.
17. Assurez-vous qu'il existe dans votre TP: Des insertions, des modifications de tuples qui ne respectent pas les contraintes et des suppressions possibles et celles qui provoquent des erreurs.

#### Partie IV : Langage d'interrogation de données

18. Écrire une requête incluant la **jointure** de plusieurs tables ainsi que des **restrictions** .
19. Écrire une requête qui inclue un **groupe by** et une **fonction** de calcul
20. Transformer la requête précédente en **Vue** et **interroger** la .

## TP 2

### Privilèges d'accès à la base de données

Oracle permet à plusieurs utilisateurs de travailler sur la même base de données en toute sécurité.

Deux commandes sont particulièrement importantes : GRANT et REVOKE et permettent de définir les droits de chaque utilisateur sur les objets de la base.

Tout utilisateur accède à la base à l'aide de son nom utilisateur et de son mot de passe. C'est le nom utilisateur qui permet de déterminer les droits d'accès aux objets de la base de données.

Au cours des TP précédents vous avez travaillé seul dans un schéma de nom égal à votre nom d'utilisateur. Nous allons vérifier que le SGBD gère la concurrence d'accès à des objets de la base entre plusieurs utilisateurs différents. Tout utilisateur qui crée des objets est propriétaire de ces objets. Le créateur d'un objet peut décider de donner (ou de supprimer) certains droits d'accès à tout autre utilisateur de sa connaissance.

**GRANT privilège[ON table/vue] TO utilisateur [WITH GRANT OPTION]**

Cet ordre permet de donner le privilège concerné sur la table ou la vue à l'utilisateur.

Un utilisateur ayant accordé un privilège peut le reprendre à tout moment à l'aide de l'ordre RE-VOKE :

**REVOKE privilège ON [table/vue] FROM utilisateur**

Les privilèges qui peuvent être donnés sont les suivants :

**SELECT** droit de lecture, **INSERT** droit d'insertion de lignes, **UPDATE** droit de mise-à-jour de lignes, **DELETE** droit de suppression de lignes,

**ALTER** droit de modification de la définition de la table, **INDEX** droit de création d'index, **ALL** tous les droits ci-dessus.

Un utilisateur ayant reçu un privilège avec la mention facultative WITH GRANT OPTION peut les transmettre à son tour à un autre utilisateur (User3).

#### Questions :

1. Créez un autre utilisateur : **GererTP2**.
2. Connectez-vous à l'aide cet utilisateur. Que remarquez-vous ?
3. Donner le droit de création d'une session pour cet utilisateur (Create Session) et reconnecter vous.
4. Donner les privilèges suivants à **GererTP2**, créer des tables, des utilisateurs. Vérifier.
5. Exécutez la requête Q1 suivante : **Select \* from GererTP2..<Une de vos tables>**. Que remarquez-vous ?
6. Donner les droits de lecture à cet utilisateur pour cette table . Exécutez la requête Q1 maintenant.
7. Enlever les privilèges précédemment accordés.
8. Vérifier que les privilèges ont bien été supprimés.
9. Créer un profil « **Gerer\_DroitTP2** » qui est caractérisé par : ( 3 sessions simultanées autorisées, Un appel système ne peut pas consommer plus de 30 secondes de CPU, Chaque session ne peut excéder 30 minutes, Un appel système ne peut lire plus de 1 500 blocs de données en mémoire et sur le disque, Chaque session ne peut allouer plus de 25 ko de mémoire en SGA, Pour chaque session, 40 minutes d'inactivité maximum sont autorisées, 3 tentatives de connexion avant blocage du compte, Le mot de passe est valable pendant 80 jours et il faudra attendre 60 jours avant qu'il puisse être utilisé à nouveau, 1 seul jour d'interdiction d'accès après que les 3 tentatives de connexion ont été atteintes, La période de grâce qui prolonge l'utilisation du mot de passe avant son changement est de 25 jours.).
10. Affecter ce profil à l'utilisateur **GererTP2**.
11. Faites une modification au niveau d'une table .Que remarquez-vous ? Que faut-il faire ?
12. Donner les droits de mise à jour à cet utilisateur pour cette table et réessayer de refaire la modification.
13. Créer le rôle : « **GestionTP2** » qui peut voir deux tables et peut modifier une troisième table.
14. Assigner ce rôle à **GererTP2**. Vérifier que les autorisations assignées au rôle **GestionTP2**, ont été bien transférées sur l'utilisateur **GererTP2**.

15. Créer un index sur une attribut autre que la clé dans une table. Que remarquez-vous ?
16. Donner les droits de création d'index à **GestionTP2** pour Cette table, ensuite réessayer de créer l'index. Que se passe-t-il ?

## TP (3)

### Dictionnaire Oracle

ORACLE dispose d'une structure centralisée (le dictionnaire de données) contenant la description de tous les objets (tables, vues, utilisateurs, ...) gérés par ce SGBD. Ce dictionnaire regroupe donc toutes les informations nécessaires au fonctionnement du SGBD. Il présente la particularité d'être lui-même organisé comme une base de données (on parle de meta-base) et d'être par conséquent accessible directement à travers SQL. Pour des raisons d'intégrité aisément compréhensibles, la plupart des tables du dictionnaire ne sont pas modifiables par les utilisateurs (seul l'administrateur de la base SYS, a accès à ces dernières). A l'opposé, l'accès à leur contenu est généralement autorisé, ce qui va nous permettre d'explorer quelque peu le dictionnaire afin de mieux appréhender cette structure de donnée essentielle au SGBD. D'une manière plus précise, les tables du dictionnaire sont généralement cryptées. Leur contenu est par contre accessible par l'intermédiaire de vues. Elles sont de quatre grands types:

- Vues relatives aux objets d'un utilisateur : USER\_\*
- Vues relatives aux objets accessibles à un utilisateur : ALL\_\*
- Vues relatives aux administrateurs : DBA\_\*
- Vues relatives aux suivi des performances : V\$\*

Au cours de ce TP, nous allons nous intéresser plus particulièrement aux vues de type ALL et USER. Pour vous guider dans vos travaux, vous trouvez dans le tableau suivant l'ensemble des vues du dictionnaire de données Oracle de type USER.

| Nom de la vue           | Synonyme | Contenu   |
|-------------------------|----------|---|
| <b>DICTIONARY</b>       | DICT     | Toutes les vues du dictionnaire, pour le développeur ou le DBA : Nom de la vue, description                               |
| <b>USER_TABLES</b>      | TABS     | mes tables : nom, tablespace, stockage, statistiques, cluster éventuel  |
| <b>USER_TAB_COLUMNS</b> | COLS     | Colonnes de mes tables : Nom colonne, type, longueur, obligatoire   |
| <b>USER_VIEWS</b>       | -        | Mes vues : Nom, texte de l'ordre SQL associé, type  |
| <b>USER_INDEXES</b>     | IND      | Mes indexes : Nom, table indexée, unicité, stockage, statistiques   |
| <b>USER_IND_COLUMNS</b> | -        | Nom index, nom table, nom colonne, position et longueur   |
| <b>USER_CLUSTERS</b>    | CLU      | Mes clusters ; Nom, stockage, statistiques  |
| <b>USER_OBJECTS</b>     | OBJ      | Mes objets : tables, vues, indexes, clusters, synonymes, procédures, fonction, package, séquence                          |
| <b>USER_SEQUENCES</b>   | SEQ      | Mes séquences : Valeur min, max, incrément, cycle, cache  |
| <b>USER_SYNONYMS</b>    | SYN      | Mes synonymes : Nom du synonyme, de la table, propriétaire et dblink éventuel   |
| <b>USER_USERS</b>       | -        | Caractéristiques générales de l'utilisateur: Nom, tablespace par défaut, tablespace temporaire                            |
| <b>USER_CONSTRAINTS</b> | -        | mes contraintes : Nom, type, table d'accueil, statut  |
| <b>USER_DB_LINKS</b>    | -        | mes database links (liens base distantes) : Nom, user distant, mot de passe, serveur distant, date de creation            |
| <b>USER_TAB_PRIVS</b>   | -        | Des privilèges donnés ou reçus : Bénéficiaire, propriétaire, créateur   |
| <b>USER_EXTENTS</b>     | -        | Caractéristiques de stockage de mes objets : Nom du segment, de la partition, du tablespace, taille en octets et en blocs |
| <b>USER_TS_QUOTAS</b>   | -        | Quota d'écriture autorisé sur les tablespace : Nom du tablespace, taille max en octets et en blocs                        |

#### Question :

Connectez vous (les tables du TP précédent étant créées) ;

1. Connecter en tant que « **System** ». Lister le catalogue « DICT ». Il contient combien d'instances ? Donner sa structure ?
2. Donner le rôle et la structure des tables (ou vues) suivantes : ALL\_TAB\_COLUMNS, USER\_USERS, ALL\_CONSTRAINTS et USER\_TAB\_PRIVS.
3. Trouver le nom d'utilisateur avec lequel vous êtes connecté ?
4. Comparer la structure et le contenu des tables ALL\_TAB\_COLUMNS et USER\_TAB\_COLUMNS ?
5. Vérifiez que les tables du **TP1** ont été réellement créées ? Donner toutes les informations sur ces tables ?
6. Lister les tables de l'utilisateur « **system** » et celles de votre utilisateur
7. Donner la description des attributs de deux de vos tables (Exploiter la table **USER\_TAB\_COLUMNS**).
8. Comment peut-on vérifier qu'il y a une référence de clé étrangère entre deux tables ?
9. Donner toutes les contraintes créées lors du **TP1** et les informations qui les caractérisent (Exploitez la table **USER\_CONSTRAINTS**);
10. Retrouver toutes les informations permettant de recréer une de vos tables.
11. Trouver tous les privilèges accordés à **GestionTP2**.
12. Trouver les rôles donnés à l'utilisateur **GererTP2**.
13. Trouver tous les objets appartenant à **GestionTP2**.
14. L'administrateur cherche le propriétaire d'une table, comment il pourra le trouver ?
15. Donner la taille en Ko de cette table.
16. Vérifier l'effet produit par chacune des commandes de définition de données du **TP1** sur le dictionnaire.