

DDL - DML - DCL

1 Création de table

1. Exécutez la requête suivante :

```
CREATE TABLE Test (  
    tId int NOT NULL,  
    tLib varchar(50) NOT NULL,  
    tNb1 decimal(5,2) DEFAULT 12 NOT NULL,  
    tNb2 decimal(8,2) NOT NULL);
```

1. Créez quelques tuples en vérifiant que les différents attributs sont effectivement obligatoires.
2. Veillez à insérer 2 tuples ayant la même valeur pour **tId**.
3. Réalisez une insertion d'un tuple pour lequel vous ne spécifiez pas la valeur de **tNb1**. Vérifiez que l'attribut reçoit bien la valeur 12.

2 Ajout de contraintes

Table Test

1. Ajoutez chacune des contraintes suivantes :
 - **tId** est clé primaire ;
 - **tNb1** doit être supérieur à 0 ;
 - **tNb2** doit être supérieur à **tNb1** ;
 - deux tuples de **Test** ne peuvent pas avoir les mêmes valeurs pour la paire **tNb1** et **tNb2**.

Pour chaque contrainte, introduisez des tuples la satisfaisant et d'autres ne la satisfaisant pas. Remarquez que l'ajout d'une contrainte n'est accepté que si les données déjà présentes la respectent.

2. Modifiez un ensemble de tuples de **Test** en veillant à ce que cette modification entraîne, pour un tuple, la violation d'une contrainte. Vérifiez qu'aucun tuple n'aura été modifié.

Clé étrangère

3. Exécutez la requête:

```
CREATE TABLE Test2 (  
    ttId int,  
    t2Ref int);
```

4. Ajoutez la contrainte définissant l'attribut **t2Ref** comme clé étrangère vers la table **Test**.
5. Ajoutez des tuples à **Test2** pour vérifier la mise en œuvre de cette dernière contrainte.
6. Supprimez la contrainte de clé étrangère. Recréez-la avec l'option **ON DELETE CASCADE**. Testez.

Contraintes deferrable

7. Supprimez la contrainte *foreign key* définie à l'étape précédente.
8. Recréez-la avec l'option **DEFERRABLE INITIALLY DEFERRED**.
9. Donnez une suite d'instructions mettant en évidence la différence de comportement du SGBD avec ou sans l'option du point précédent. Terminez cette suite d'instructions par un **COMMIT**. Exécutez-la au moyen d'un script SQL.
10. Une fois de plus, supprimez la contrainte de clé étrangère. Recréez-la maintenant avec l'option **DEFERRABLE INITIALLY IMMEDIATE**. Remarquez que la mention **IMMEDIATE** n'est pas nécessaire car il s'agit de l'option par défaut.
11. Lors des tests, vous voyez que nous semblons nous retrouver dans la situation sans **DEFERRABLE**. Pour pouvoir profiter ou non du fait de différer le contrôle de la contrainte, une instruction particulière doit être émise en début de transaction. Accédez à la page 19-48 (p. 1306) du [References Guide SQL Oracle](#) qui vous est fourni sur poESI dans [Ressources](#). Testez.

3 Création du schéma conceptuel

Cet exercice vous donne l'occasion de disposer d'un script pour la création des tables de la base de donnée Département / Employé. Ce script est utilisé ensuite à plusieurs reprises. Le [schéma conceptuel des tables Département / Employé](#) est fourni sur poESI.

1. Créez les tables **Employe** et **Departement** dans votre schéma *sans* définir l'attribut **EmpSal**. Importez les données à partir du schéma ADT. N'oubliez pas de commenter chacune des tables et chacun des attributs. Veillez à garder toutes les commandes DDL dans un fichier texte qui constitue un script SQL.
2. Supprimez les tables créées et faites exécuter votre script SQL pour les recréer.
3. Ajoutez l'attribut **EmpSal** et affectez lui les valeurs en provenance de **ADT.Employe.EmpSal**.
4. Vérifiez au travers de quelques tests que les différentes CI (contraintes d'intégrité) explicitées sont correctement implémentées.

4 Schémas externes

1. Créez la vue **Manager**(mgrNo, mgrNom, dptDirigéLib, nbEmpDirigés). Elle permet d'accéder aux informations des managers de la société.
2. Créez la vue **EmployeDirection**(empno, empnom, empsal, empdpt). Elle reprend les employés du département de libellé « DIRECTION ».
3. Ces deux vues sont-elles modifiables ? En cas de réponse positive, vérifiez l'effet de la clause **WITH CHECK OPTION**.

5 Gestion des privilèges

Pour cette partie, nous vous demandons de travailler d'abord avec un · e de vos camarades de classe, puis avec plusieurs d'entre eux.

1. Donnez le privilège à votre camarade de classe de consulter la vue **Manager**.

```
GRANT SELECT ON MaVue TO SchemaCamarade1;
```

2. Donnez le privilège à votre camarade de consulter la vue **EmployeDirection** en lui permettant de propager ce droit. Que se passe-t-il pour les utilisateurs ayant reçu de votre camarade ce privilège lorsque vous révoquez le privilège de votre camarade ?
3. Donnez à votre camarade le privilège de « mise à jour » sur les employés masculins de votre table **Employe**.

6 Synonymes

Vous devez ici encore travailler avec un · e camarades de classe.

1. Créez un synonyme sur une des vues auxquelles votre camarade vous donne accès. Si votre camarade supprime sa vue ou vous retire le droit d'accès, le synonyme existe-t-il toujours ?
2. Supprimez le synonyme.

7 Schéma interne

1. Créez un index sur l'attribut **EmpNom** de votre table **Employe**.

8 Consultation du catalogue

1. Consultez les vues utilisateur **USER_TABLES**, **USER_TAB_COLUMNS**, **USER_COL_COMMENTS**, **USER_CONSTRAINTS** et **USER_CONS_COLUMNS**. Une description sommaire de ces vues sur le catalogue est consultable sur poESI dans le document [Catalogue Oracle](#). Vérifiez la présence des contraintes, commentaires, etc. que vous avez créés.

9 Mise en œuvre des transactions

Ouvrez deux sessions distinctes de votre console SQL et :

1. Ajoutez des tuples à la table **Test** à partir d'une session en contrôlant le remplissage de la table à partir de l'autre session.
2. Testez l'annulation de transaction.
3. Testez la gestion de conflit lorsque deux sessions modifient les mêmes données « en même temps ». Expliquez ce que cette locution imprécise « en même temps » veut dire.