**Compte rendu PLSQL**

Realizer par:

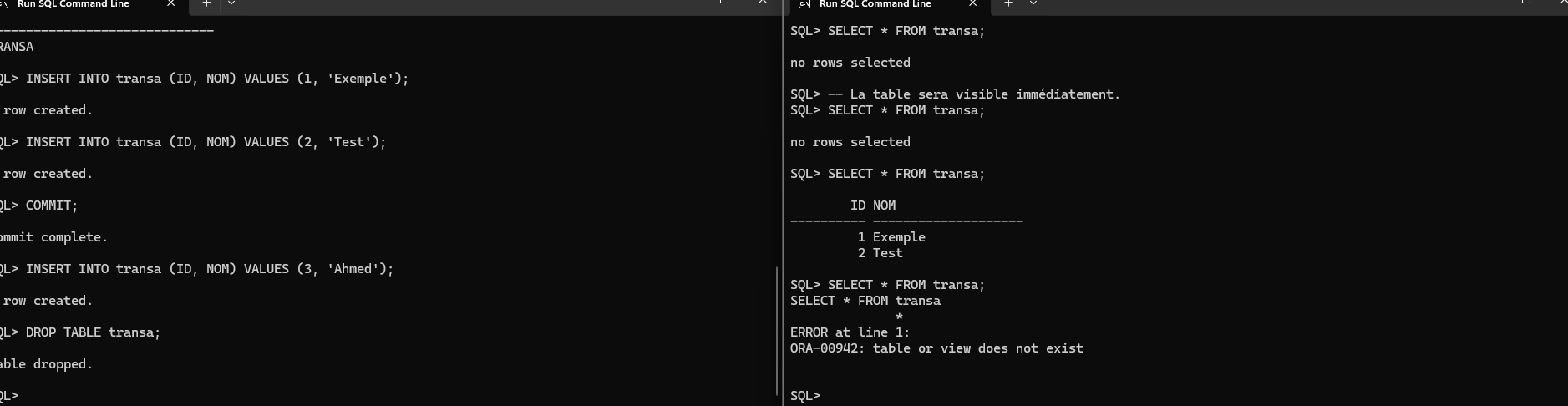
Ahmed TAOUYA

Yasser ELBAKRI

3IIR15

Partie 1: Transaction et ordre CREATE TABLE ET DROP TABLE

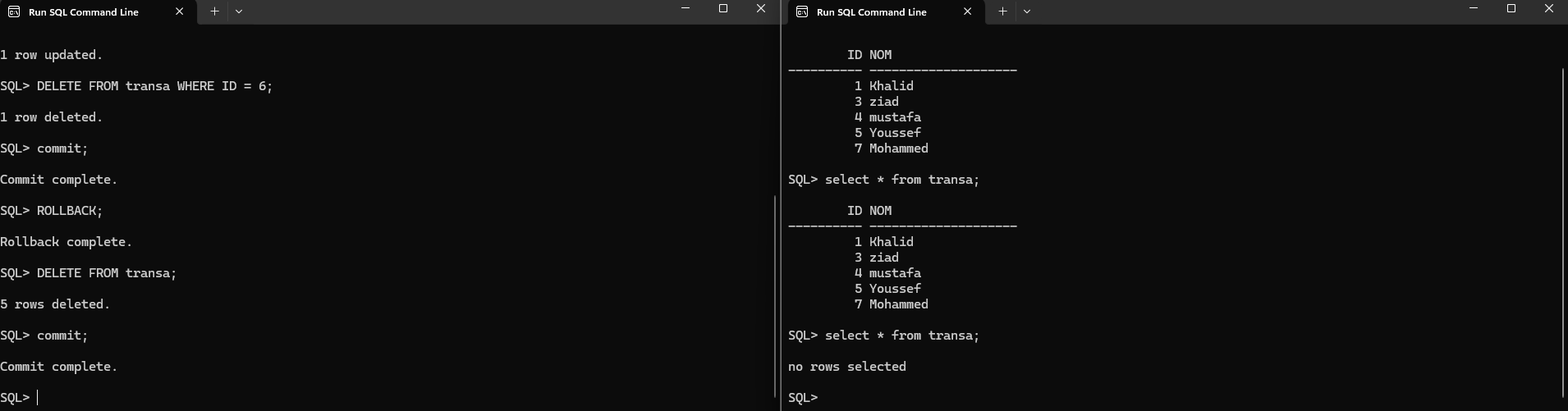
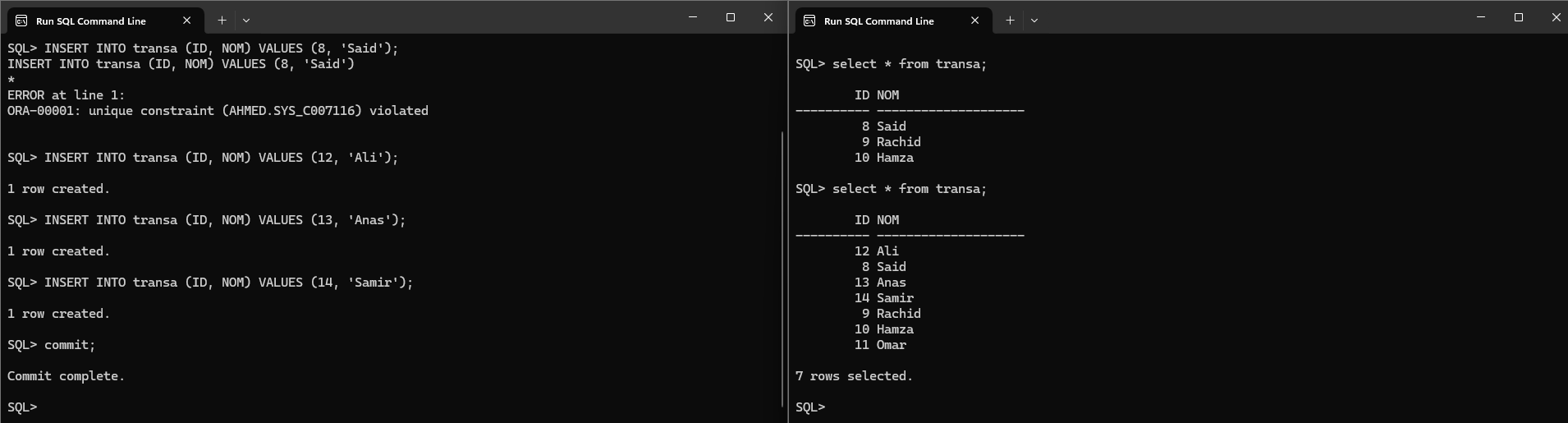
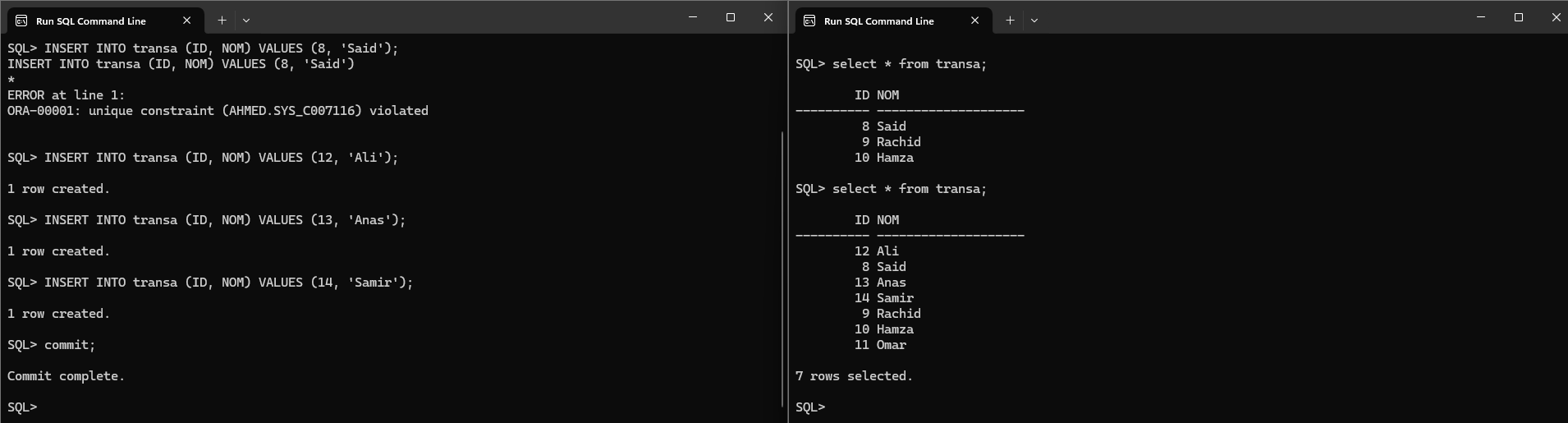
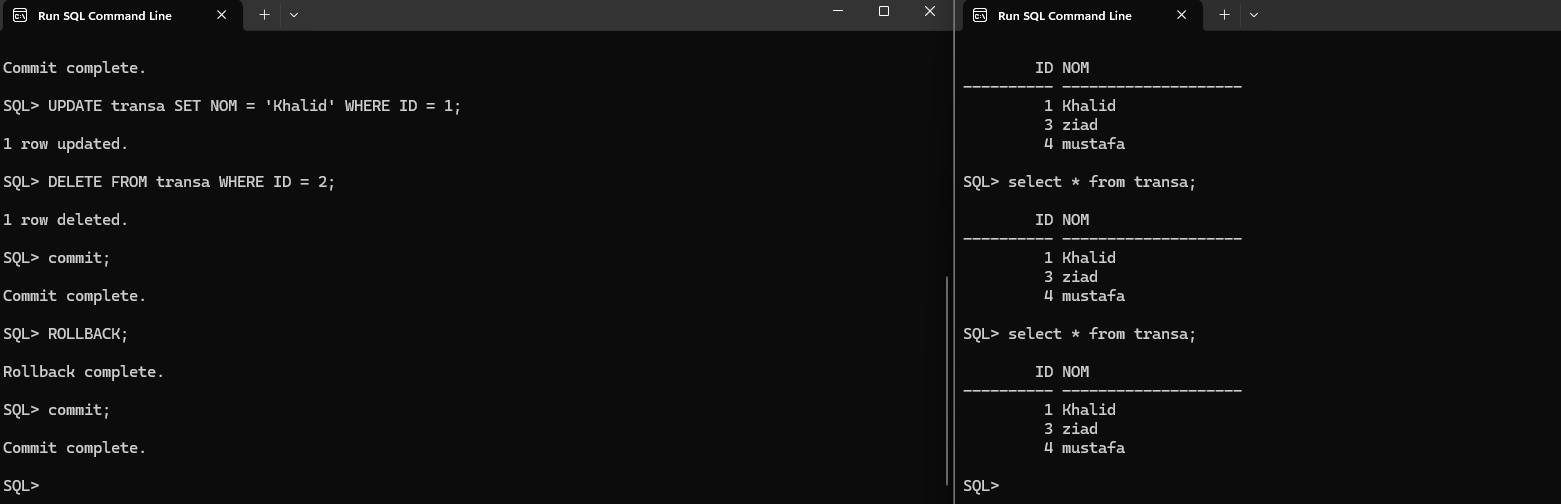
Les opérations CREATE et DROP sont des opérations DDL, qui par défaut ne sont pas transactionnelles dans le sens classique (comme les DML, qui peuvent être annulées par ROLLBACK). Elles sont instantanément validées à l'exécution et ne nécessitent pas de COMMIT pour être appliquées. Toutefois, tant que la session n'est pas fermée ou qu'une transaction n'est pas explicitement validée, les autres sessions ne peuvent pas voir ces modifications.



Partie 2 : atomicité d’une transaction courante

**Conclusion Générale**

* **Transactions et COMMIT/ROLLBACK :**
  + Les opérations de modification de données (insertion, modification, suppression) peuvent être annulées avec ROLLBACK si elles ne sont pas validées.
  + Une fois validées avec COMMIT, les modifications sont permanentes et ne peuvent pas être annulées.
* **Structure de la Table :**
  + Les changements structurels (ALTER TABLE) sont immédiatement validés et ne peuvent pas être annulés avec un ROLLBACK.
* **Fermeture de Session :**
  + Les modifications non validées avant la fermeture de la session sont perdues, tandis que celles validées par COMMIT sont conservées.



### Partie 2 : atomicité d’une transaction courante

#### **Étape 1 : Insertion et Visualisation**

* **Action :** Insertion de trois à quatre lignes dans la table transa.
* **Résultat :** Les lignes sont ajoutées et affichées avec la commande SELECT. Aucune erreur signalée.

#### **Étape 2 : Sauvegarde Partielle avec** SAVEPOINT

* **Action :** Deux séries de lignes sont insérées, suivies de la création de deux points de sauvegarde (SAVEPOINT point1 et SAVEPOINT point2).
* **Résultat :** Les sauvegardes partielles sont réussies, permettant un suivi précis des modifications.

#### **Étape 3 :** ROLLBACK **Partiel**

* **Action :** Annulation partielle avec ROLLBACK TO point1.
* **Résultat :** Les lignes insérées après point1 sont supprimées, mais celles insérées avant sont préservées. La table contient les données ajoutées jusqu’à point1.

**Conclusion :** Les SAVEPOINT permettent de gérer efficacement les transactions partielles.

#### **Étape 4 : Validation avec** COMMIT

* **Action :** Insertion de nouvelles lignes suivie de leur validation avec COMMIT.
* **Résultat :** Toutes les données sont enregistrées de manière permanente dans la table. Les tentatives ultérieures de ROLLBACK ne peuvent pas annuler ces modifications.

**Conclusion :** Le COMMIT marque un point de non-retour, rendant les données persistantes.

#### **Étape 5 : Suppression avec** DELETE

* **Action :** Suppression de toutes les lignes de la table, suivie d’un COMMIT.
* **Résultat :** La table est vidée avec succès.

**Conclusion :** La commande DELETE suivie d’un COMMIT permet de supprimer toutes les données de manière irréversible.

#### **Étape 6 : Annulation Partielle avec** ROLLBACK TO

* **Action :** Insertion de plusieurs lignes, création de points de sauvegarde (SAVEPOINT), puis annulation partielle avec ROLLBACK TO point5.
* **Résultat :** Seules les lignes ajoutées avant point5 sont conservées dans la table.

**Conclusion :** Le ROLLBACK TO permet d'annuler uniquement les modifications postérieures à un point précis.

#### Partie 3 : transaction et ordre DDL

#### **Étape 1 : Modification de la Structure avec** ALTER TABLE

* **Action :** Ajout d’une colonne AGE après l’insertion de données.
* **Résultat :**
  + La modification structurelle (ALTER TABLE) a effectué un COMMIT implicite.
  + Les données insérées avant la modification sont validées automatiquement.

#### **Étape 2 : Vérification du Contenu**

* **Action :** Affichage du contenu et de la structure de la table avec DESC et SELECT.
* **Résultat :**
  + La structure affiche la nouvelle colonne AGE.
  + Les données insérées avant l’opération sont préservées.

**Conclusion :** Les commandes DDL (ALTER TABLE) effectuent un COMMIT implicite, confirmant l’atomicité entre les données et la structure.

### ****Synthèse Générale****

1. **Gestion des Transactions :**
   * Les SAVEPOINT et ROLLBACK offrent une flexibilité dans le contrôle des transactions.
   * Les commandes COMMIT et ROLLBACK sont essentielles pour garantir ou annuler les modifications.
2. **Atomicité :**
   * Les modifications DDL (ALTER TABLE) effectuent un COMMIT implicite, validant automatiquement les données en cours.
3. **Impact des Fermetures :**
   * Une fermeture brutale sans COMMIT entraîne un ROLLBACK implicite, annulant toutes les modifications non validées.
4. **Opérations de Suppression :**
   * La commande DELETE nécessite un COMMIT pour valider la suppression des données.

