

#### Objectifs du TP :

- Manipulation de la fonctionnalité Oracle Flashback Version Query.
- Manipulation de la fonctionnalité Flashback table.

#### Exercice 1 : Manipulation de la fonctionnalité Oracle Flashback Version Query

- ⇒ **Oracle Flashback Version Query** : permet aux utilisateurs de récupérer les versions précédentes des lignes d'une table sur une période de temps spécifique.
- ⇒ Cela est particulièrement utile pour l'audit des modifications, la correction d'erreurs et la compréhension de l'évolution des données au fil du temps.

L'objectif de cet exercice est de vérifier les changements de salaire d'un employé.

1. Se connecter à SQLPLUS avec l'utilisateur HR.
2. Vérifier le salaire de l'employé ayant l'id 107 :

**SQL> Select salary from employees where employee\_id = 107;**

```
SQL> select salary from employees where employee_id = 107;

SALARY
-----
    4200
```

3. Mettre à jour le salaire de cet employé en rajoutant 100. Vérifier le résultat et faire un commit pour valider la transaction :

**SQL> update employees set salary = salary + 100 where employee\_id = 107;**

Vérification:

**SQL> Select salary from employees where employee\_id = 107;**

```
SQL> update employees
  2  set salary = salary + 100
  3  where employee_id = 107;

1 ligne mise à jour.

SQL> select salary from employees where employee_id = 107;

SALARY
-----
    4300
```

SQL> commit;

4. Vérifier les changements de valeur du salaire de l'employé :

SQL> Select salary from employees **versions between scn minvalue and maxvalue** where employee\_id = 107;

```
SQL> select salary from employees
  2  versions between scn minvalue and maxvalue
  3  where employee_id = 107;
```

SALARY
4300
4200

- ➔ **MINVALUE et MAXVALUE** spécifient la plage des SCN à considérer. **MINVALUE** représente le plus ancien SCN disponible et **MAXVALUE** le plus récent.
- ➔ Cette clause permet de voir l'historique des modifications des lignes dans la table entre les SCN spécifiés.

5. Afficher plus d'informations : ajouter les informations sur la date de début et de fin de chaque version du champ « salary » :

SQL> select **versions\_starttime, versions\_endtime**, salary from employees **versions between scn minvalue and maxvalue** where employee\_id = 107;

```
SQL> column VERSIONS_STARTTIME format A20
SQL> column VERSIONS_endTIME format A20
SQL> select versions_starttime, versions_endtime, salary
  2  from employees
  3  versions between scn minvalue and maxvalue
  4  where employee_id = 107;
```

VERSIONS_STARTTIME	VERSIONS_ENDTIME	SALARY
22/05/24 00:42:25		4300
	22/05/24 00:42:25	4200

- ➔ récupérer les différentes versions des données d'une table à différents points dans le temps.
- ➔ Cette commande permet de voir l'historique des modifications des données pour une ligne spécifique.

## Exercice 2 : Flashback table

1. Vérifier recyclebin. Cette table contient les différentes opérations de DROP effectuées sur les tables, tant que la table n'a pas été récupérée par la fonctionnalité Flashback.

**SQL> Select \* from recyclebin ;**

2. Créer une nouvelle table à partir de la table employees.

**SQL> Create table employee\_new as (select \* from employees);**

3. Vérifier que la nouvelle table est bien alimentée : **select \* from employee\_new;**

4. Supprimer la table employee\_new :

**SQL> drop table employee\_new;**

**SQL> commit;**

5. Vérifier que la table employee\_new n'existe plus :

**SQL> select count(\*) from employee\_new;**

```
SQL> select count(*) from employee_new;
select count(*) from employee_new
                        *
ERREUR à la ligne 1 :
ORA-00942: Table ou vue inexistante
```

6. Vérifier que nous avons une nouvelle entrée dans la table recyclebin.

**SQL> Select object\_name, original\_name, operation, type, ts\_name, createtime, droptime, dropscn from recyclebin;**

OBJECT_NAME	ORIGINAL_NAME	OPERATION	TYPE	TS_NAME	CREATETIME	DROPTIME	DROPSCN
BIN\$iFVCp9tsQgCx606KnCx58g==\$0	EMPLOYEE_NEW	DROP	TABLE	SYSAUX	2024-05-22:00:17:15	2024-05-22:00:19:44	5231349

7. Récupérer la table employee\_new à l'aide de la fonctionnalité flashback table :

**SQL> flashback table employee\_new to before drop;**

```
SQL> select count(*) from employee_new;

COUNT(*)
-----
       107
```

8. Vérifier de nouveau le contenu de la table recyclebin, elle doit être vide.

**SQL> Select**

**object\_name,original\_name,operation,type,ts\_name,createtime,droptime,dropsn from recyclebin;**

```
SQL> select object_name,original_name,operation,type,ts_name,createtime,droptime,dropsn from recyclebin;  
aucune ligne s
```