BDA / TP2 Oracle

Table des matières

| Tabl | le des matières | . 2 |
|------|----------------------------|-----|
| 1. | Introduction | 3 |
| | Déclencheur de mise a jour | |
| | Conclusion | |

1. Introduction

Dans ce TP, nous allons voir comment utiliser les TRIGGERS.

2. Déclencheur de mise a jour

1) Nous allons commencer par ajouter une colonne dans la table Continent :

```
-- 1.1
ALTER TABLE Continent
ADD NBP DECIMAL(3);
```

2) Nous allons maintenant mettre des valeurs dans la nouvelle colonne. Nous allons donc regarder le nombre de pays par continent :

```
-- 1.2

UPDATE Continent c

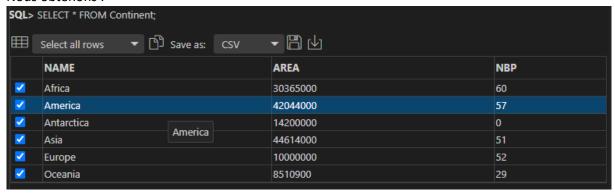
SET NBP = (SELECT COUNT(*)

FROM Country co

WHERE co.cname = c.name);

SELECT * FROM Continent;
```

Nous obtenons:



3) Nous allons maintenant créer un déclencher pour augmenter le compteur à chaque ajout de pays :

```
-- 1.3

CREATE OR REPLACE TRIGGER t_i_pays_continent_nbp

AFTER INSERT

ON Country

FOR EACH ROW

BEGIN

UPDATE Continent c

SET NBP = NBP + 1

WHERE :NEW.cname = c.name;

END;
```

Voici la syntaxe du trigger. On créer un trigger en lui donnant un nom. On lui dit ensuite qu'il s'exécute après un insert sur la table Country a chaque ligne. On lui donne ensuite le code.

4) On le test maintenant en ajoutant un pays et en affichant la table continent :

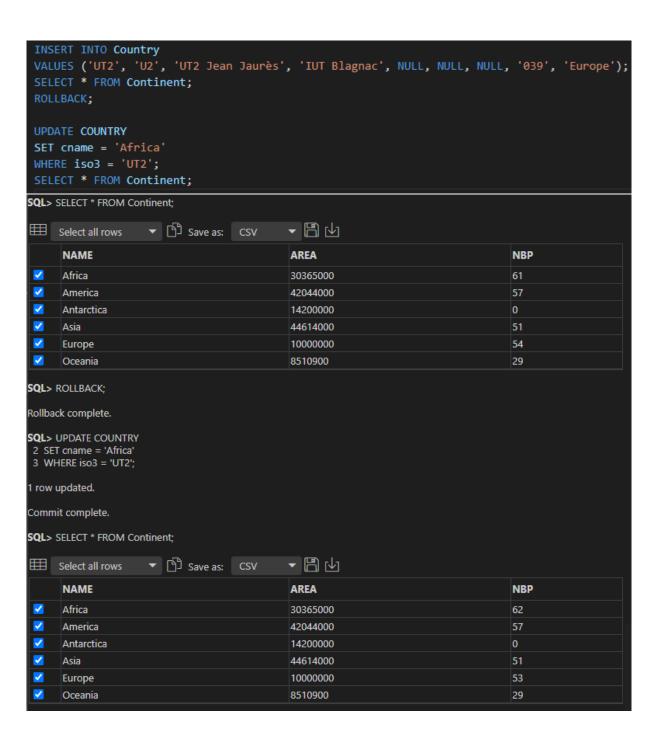
```
INSERT INTO Country
 VALUES ('UT2', 'U2', 'UT2 Jean Jaurès', 'IUT Blagnac', NULL, NULL, NULL, '039', 'Europe');
SELECT * FROM Continent;
ROLLBACK;
On obtient:
SQL> SELECT * FROM Continent;
                                           → 🖺 🕁
Select all rows
                    ▼ 🖺 Save as: CSV
                                            AREA
      NAME
                                                                                 NBP
     Africa
                                            30365000
                                                                                 60
      America
                                            42044000
     Antarctica
                                            14200000
                                            44614000
                                            10000000
     Europe
     Oceania
                                            8510900
                                                                                 29
```

5) Nous allons maintenant modifier notre trigger pour qu'il s'exécute après un delete et un update en plus du insert :

```
CREATE OR REPLACE TRIGGER t i pays continent nbp
AFTER INSERT OR DELETE OR UPDATE of cname
ON Country
FOR EACH ROW
    IF INSERTING THEN
        UPDATE Continent c
        SET NBP = NBP + 1
        WHERE : NEW.cname = c.name;
    ELSIF DELETING THEN
        UPDATE Continent c
        SET NBP = NBP - 1
        WHERE :NEW.cname = c.name;
    ELSIF UPDATING THEN
        UPDATE Continent c
        SET NBP = NBP +1
        WHERE : NEW.cname = c.name;
        UPDATE Continent c
        SET NBP = NBP -1
        WHERE :OLD.cname = c.name;
    END IF;
END;
```

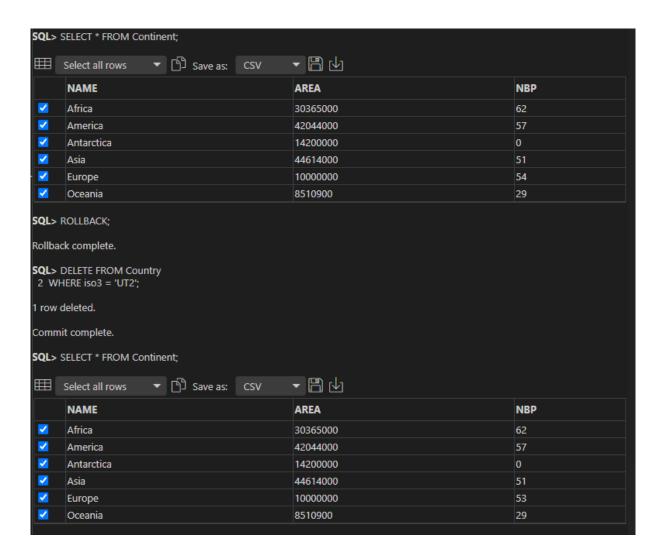
Nous pouvons voir les différents if avec INSERTING, DELETING et UPDATING qui permettent de savoir dans quelle requête on se trouve.

6) L'insert a déjà été tester, si on test avec Update, on obtient :



Après un test sur DELETE, on obtient :

```
DELETE FROM Country
WHERE iso3 = 'UT2';
SELECT * FROM Continent;
```



3. Conclusion

Dans ce TP, nous avons pu voir comment utiliser les triggers après plusieurs requêtes avec les INSERTING, DELETING et UPDATING.