

Morgado-Samagaio Jonathan

BDA / TP1 Oracle

Table des matières

Table des matières	2
1. Introduction.....	3
2. SQL.....	3
3. PL/SQL	5
4. Conclusion	7

1. Introduction

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

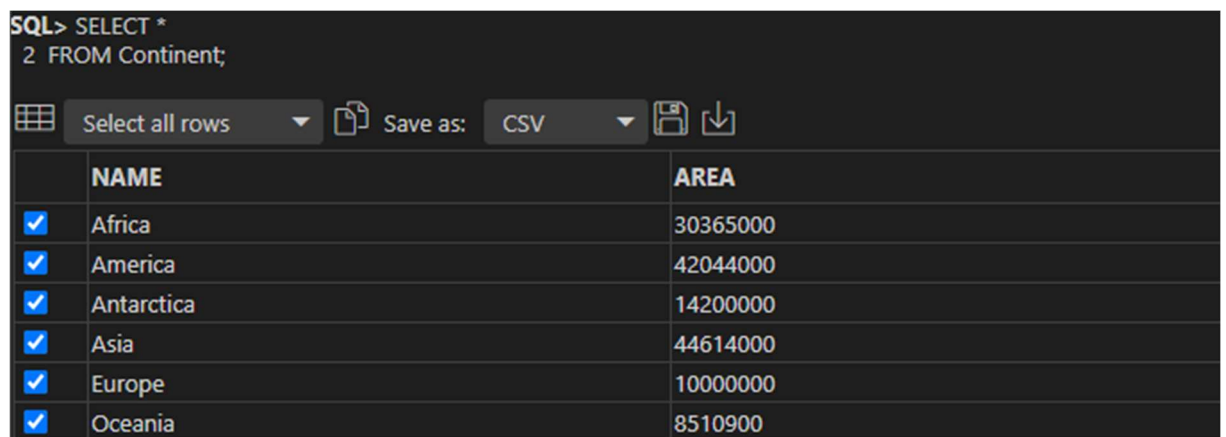
2. SQL

- 1) Nous commençons par exécuter le script createbase.sql et celui s'exécute sans problèmes.
- 2) Nous allons maintenant rajouter différentes contraintes sur la table Pandemic :

```
-- 2.2
ALTER TABLE Pandemic
ADD (CONSTRAINT pk_pandemic PRIMARY KEY (iso3, ddate),
     CONSTRAINT fk_pandemic_iso3 FOREIGN KEY (iso3) REFERENCES Country (iso3),
     CONSTRAINT ck_pandemic_cases CHECK (cases >= 0),
     CONSTRAINT ck_pandemic_deaths CHECK (deaths >= 0),
     CONSTRAINT ck_pandemic_rrate CHECK (rrate >= 0),
     CONSTRAINT ck_pandemic_icu CHECK (icu >= 0),
     CONSTRAINT ck_pandemic_hpatients CHECK (hpatients >= 0),
     CONSTRAINT ck_pandemic_tests CHECK (tests >= 0),
     CONSTRAINT ck_pandemic_prate CHECK (prate >= 0),
     CONSTRAINT ck_pandemic_vacc CHECK (vacc >= 0),
     CONSTRAINT ck_pandemic_fvacc CHECK (fvacc >= 0));
```

Elles sont rajoutées sans problèmes.

- 3) Nous lançons l'exécution du script insertbase.sql qui s'exécute sans problèmes aussi.
- 4) Nous affichons maintenant le contenu de la table continent :



```
SQL> SELECT *
2 FROM Continent;
```

	NAME	AREA
<input checked="" type="checkbox"/>	Africa	30365000
<input checked="" type="checkbox"/>	America	42044000
<input checked="" type="checkbox"/>	Antarctica	14200000
<input checked="" type="checkbox"/>	Asia	44614000
<input checked="" type="checkbox"/>	Europe	10000000
<input checked="" type="checkbox"/>	Oceania	8510900

- 5) Nous vérifions maintenant le nombre de tuples de chaque tables :

```
-- 2.5
SELECT COUNT(*)
FROM Continent;
SELECT COUNT(*)
FROM Region;
SELECT COUNT(*)
FROM Country;
SELECT COUNT(*)
FROM Border;
SELECT COUNT(*)
FROM Pandemic;
```

Nous obtenons :

	COUNT(*)
<input checked="" type="checkbox"/>	6

```
SQL> SELECT COUNT(*)
2 FROM Region;
```

	COUNT(*)
<input checked="" type="checkbox"/>	26

```
SQL> SELECT COUNT(*)
2 FROM Country;
```

	COUNT(*)
<input checked="" type="checkbox"/>	249

```
SQL> SELECT COUNT(*)
2 FROM Border;
```

	COUNT(*)
<input checked="" type="checkbox"/>	652

```
SQL> SELECT COUNT(*)
2 FROM Pandemic;
```

	COUNT(*)
<input checked="" type="checkbox"/>	9931

- 6) Nous allons maintenant afficher les tables créées et leur date de création :

```
-- 2.6
SET PAGESIZE 10000;
COLUMN OBJECT_NAME FORMAT A25;
COLUMN CREATED FORMAT A10;
SELECT OBJECT_NAME, CREATED
FROM USER_OBJECTS;
```

Nous obtenons :

3. PL/SQL

- 1) Nous allons créer une procédure PL/SQL permettant d'ajouter un pays :

```
create or replace PROCEDURE ajout_Country(
    vIso3 Country.Iso3%TYPE,
    vIso2 Country.Iso2%TYPE,
    vNom Country.Name%TYPE,
    vCapitale Country.Capital%TYPE,
    vPopulation Country.Population%TYPE,
    vArea Country.Area%TYPE,
    vRcode Country.RCode%TYPE,
    vConNom Country.CName%TYPE
) IS
    erreur_ck EXCEPTION;
    PRAGMA EXCEPTION_INIT (erreur_ck, -2290);
    erreur_fk EXCEPTION;
    PRAGMA EXCEPTION_INIT (erreur_fk, -2291);
BEGIN
    INSERT INTO Country (iso3, iso2, name, capital, population, area, rcode, cname)
    VALUES (vIso3, vIso2, vNom, vCapitale, vPopulation, vArea, vRcode, vConNom);
    DBMS_OUTPUT.PUT_LINE('Pays '||vNom||' ajouté');
    COMMIT;
```

```

EXCEPTION
  WHEN erreur_ck THEN
    ROLLBACK;
    IF (SQLERRM LIKE '%CK_COUNTRY_AREA%') THEN
      DBMS_OUTPUT.PUT_LINE('Superficie de '||vNom||' négative ou nul');
    ELSIF (SQLERRM LIKE '%CK_COUNTRY_POPULATION%') THEN
      DBMS_OUTPUT.PUT_LINE('Population de '||vNom||' négative ou nul');
    ELSIF (SQLERRM LIKE '%CK_COUNTRY_RCode%') THEN
      DBMS_OUTPUT.PUT_LINE('Code de région de '||vNom||' négatif ou nul');
    END IF;
  WHEN erreur_fk THEN
    ROLLBACK;
    IF (SQLERRM LIKE '%FK_COUNTRY_CNAME%') THEN
      DBMS_OUTPUT.PUT_LINE('Continent inconnue !');
    ELSIF (SQLERRM LIKE '%FK_COUNTRY_RCODE%') THEN
      DBMS_OUTPUT.PUT_LINE('Code de région inconnue !');
    END IF;
  WHEN DUP_VAL_ON_INDEX THEN
    ROLLBACK;
    IF (SQLERRM LIKE '%PK_COUNTRY%') THEN
      DBMS_OUTPUT.PUT_LINE('Pays '||vNom||' Existant !');
    ELSIF (SQLERRM LIKE '%UK_COUNTRY_ISO2%') THEN
      DBMS_OUTPUT.PUT_LINE('Code ISO2 '||vIso2||' Existant !');
    ELSIF (SQLERRM LIKE '%UK_COUNTRY_name%') THEN
      DBMS_OUTPUT.PUT_LINE('Code ISO2 '||vNom||' Existant !');
    END IF;
  WHEN OTHERS THEN
    ROLLBACK;
    DBMS_OUTPUT.PUT_LINE(SQLERRM);
    RAISE_APPLICATION_ERROR(-20000, 'ERREUR IMPREVUE !');
END;

```

- 2) Nous allons tester cette procédure et différents cas d'erreur.

```

SET SERVEROUTPUT ON;
EXECUTE ajout_Country('AAA', 'AA', 'Test', 'Test', 50, 50, '419', 'America');

```

Avec ça, la méthode fonctionne et renvoie :

```
Pays Test ajouté
```

Si on renvoie la même requête, on obtient :

```
Pays Test Existant !
```

Si on met une population négative :

```
Population de Test négative ou nul
```

Si on met une superficie négative :

```
Superficie de Test négative ou nul
```

Si on met un continent inexistant :

```
Continent inconnue !
```

Si on met un code inconnu :

Code de région inconnue !

4. Conclusion

Dans ce TP, nous avons pu revoir les bases du SQL et du PL SQL.