

Morgado-Samagaio Jonathan

BDA / TP4 Oracle

Table des matières

Table des matières	2
1. Introduction.....	3
2. Déclencheur de mise à jour directe	3
3. Déclencheur de mise à jour indirecte	4
4. Conclusion	6

1. Introduction

Dans ce TP, nous allons voir comment utiliser les TRIGGERS pour suivre les modifications qui ont lieu sur des tables.

2. Déclencheur de mise à jour directe

- 1) On crée une table Statistique :

```
--1.1
DROP TABLE Statistique;
CREATE TABLE Statistique
(
    iso3 CHAR(3),
    ddate DATE,
    cases NUMBER,
    vac NUMBER,
    fvacc NUMBER,
    CONSTRAINT pk_statistique PRIMARY KEY (iso3)
);
```

Cette table contient 5 colonnes et une clé primaire.

- 2) On met maintenant cette table à jour avec les derniers éléments de la table Pandemic :

```
--1.2
INSERT INTO Statistique (iso3, ddate, cases, vac, fvacc)
SELECT DISTINCT iso3, ddate, cases, vac, fvacc
FROM Pandemic P
WHERE ddate = (SELECT MAX(ddate)
               FROM Pandemic
               WHERE iso3 = P.iso3
               AND vac IS NOT NULL
               AND cases IS NOT NULL
               AND fvacc IS NOT NULL);

SELECT * FROM Statistique;
```

Nous pouvons voir l'utilisation d'un INSERT INTO ... SELECT.

On obtient :

ISO3	DDATE	CASES	VAC	FVACC
AUS	13-SEP-21	76901	14091380	8711193
BEL	12-SEP-21	1205516	8495260	8261001
BRA	12-SEP-21	20999779	142257204	72795741
ZAF	13-SEP-21	2860835	10860642	7330517
CHN	08-SEP-21	95127	1095000000	969720000
DEU	12-SEP-21	4087125	55315227	51710807
DNK	12-SEP-21	353803	4434115	4300780
ESP	12-SEP-21	4907461	37216507	35300384
FRA	20-SEP-21	1	1	1
GBR	12-SEP-21	7259752	48439272	44048093

Page 1 of 2 (1-10 of 17 rows)

- 3) Nous allons maintenant créer un déclencheur pour mettre à jour statistique à chaque mise à jour de pandemic :

```
--1.3
CREATE OR REPLACE TRIGGER t_iud_pandemic
AFTER INSERT OR UPDATE OF cases, vacc, fvacc
ON Pandemic
FOR EACH ROW
BEGIN
    UPDATE Statistique
    SET ddate = :NEW.ddate,
        cases = :NEW.cases,
        vac = :NEW.vacc,
        fvacc = :NEW.fvacc
    WHERE iso3 = :NEW.iso3
    AND ddate <= :NEW.ddate;
END;
/
```

Nous essayons maintenant de saisir une nouvelle valeur dans pandemic et nous vérifions dans Statistique :

```
INSERT INTO Pandemic (iso3, ddate, cases, vacc, fvacc)
VALUES ('FRA', TO_DATE('2021-10-20', 'YYYY-MM-DD'), 1, 1, 1);

SELECT * FROM Statistique;
```

```
SQL> INSERT INTO Pandemic (iso3, ddate, cases, vacc, fvacc)
2 VALUES ('FRA', TO_DATE('2021-10-20', 'YYYY-MM-DD'), 1, 1, 1);
1 row created.
```

```
SQL> SELECT * FROM Statistique;
```

ISO3	DDATE	CASES	VAC	FVACC
<input checked="" type="checkbox"/> AUS	13-SEP-21	76901	14091380	8711193
<input checked="" type="checkbox"/> BEL	12-SEP-21	1205516	8495260	8261001
<input checked="" type="checkbox"/> BRA	12-SEP-21	20999779	142257204	72795741
<input checked="" type="checkbox"/> ZAF	13-SEP-21	2860835	10860642	7330517
<input checked="" type="checkbox"/> CHN	06-SEP-21	95127	1095000000	969720000
<input checked="" type="checkbox"/> DEU	12-SEP-21	4087125	55315227	51710807
<input checked="" type="checkbox"/> DNK	12-SEP-21	353803	4434115	4300780
<input checked="" type="checkbox"/> ESP	12-SEP-21	4907461	37216507	35300384
<input checked="" type="checkbox"/> FRA	20-OCT-21	1	1	1
<input checked="" type="checkbox"/> GBR	12-SEP-21	7259752	48439272	44048093

3. Déclencheur de mise à jour indirecte

- 1) On crée une table trace pour garder en mémoire toutes les modifications de pandemic :

```
--2.1
DROP TABLE Trace;
CREATE TABLE Trace
(
    no NUMBER,
    ordre VARCHAR(250)
);
```

- 2) On crée ensuite une séquence SEQ_no pour les futurs ajouts dans Trace :

```
--2.2
DROP SEQUENCE SEQ_no;
CREATE SEQUENCE SEQ_no;
```

- 3) On créer maintenant le trigger permettant de stocker les commandes à exécuter après un ajout dans Pandemic :

```
--2.3
CREATE OR REPLACE TRIGGER t_iud_pandemic_trace
AFTER INSERT OR UPDATE OF cases, vacc, fvacc
ON Pandemic
FOR EACH ROW
DECLARE
    tDate DATE;
BEGIN
    SELECT ddate INTO tDate
    FROM Statistique
    WHERE iso3 = :NEW.iso3;

    IF (INSERTING AND tDate < :NEW.ddate) THEN
        INSERT INTO Trace
        VALUES (SEQ_no.nextval,
                'UPDATE Statistique
                 SET ddate = '''||:NEW.ddate||'',
                     cases = '''||:NEW.cases||'',
                     vac = '''||:NEW.vacc||'',
                     fvacc = '''||:NEW.fvacc||'
                 WHERE iso3 = '''||:NEW.iso3||'
                 AND ddate <= '''||:NEW.ddate||''';');
    ELSIF (UPDATING AND tDate < :NEW.ddate) THEN
        INSERT INTO Trace
        VALUES (SEQ_no.nextval,
                'UPDATE Statistique
                 SET ddate = '''||:NEW.ddate||'',
                     cases = '''||:NEW.cases||'',
                     vac = '''||:NEW.vacc||'',
                     fvacc = '''||:NEW.fvacc||'
                 WHERE iso3 = '''||:NEW.iso3||'
                 AND ddate <= '''||:NEW.ddate||''';');
    END IF;
END;
/
```

On drop ensuite le trigger précédent et on essaie d'ajouter dans Pandemic des valeurs :

```
DROP TRIGGER t_iud_pandemic;

INSERT INTO Pandemic (iso3, ddate, cases, vacc, fvacc)
VALUES ('FRA', TO_DATE('2021-09-21', 'YYYY-MM-DD'), 11, 11, 11);

SELECT * FROM Trace;
```

	NO	ORDRE
<input checked="" type="checkbox"/>	21	UPDATE Statistique SET ddate = '21-OCT-21', cases = 11, vac = 11, fvacc = 11 WHERE iso3 = FRA AND ddate <= '21-OCT-21';

4. Conclusion

Dans ce TP, nous avons pu voir comment utiliser les triggers pour suivre les mises à jours sur des tables et effectuer des requêtes après ces mises à jours.