

*RAPPORT*

*DEVOIR*

*ASGBD*

*TP :*

YOUSFI

ZAKARIA

171732026950

GRP TP : 3

## **1)Création de deux tableSpaces default tbs et temp tbs:**

```
SQL*Plus: Release 10.2.0.1.0 - Production on Mer. Nov. 30 19:09:57 2022
```

```
Copyright (c) 1982, 2005, Oracle. All rights reserved.
```

```
Entrez le mot de passe :
```

```
Connecté à :
```

```
Oracle Database 10g Express Edition Release 10.2.0.1.0 - Production
```

```
SQL> create tablespace default_tbs datafile 'c:\default_tbs.dat' size 100M autoextend on online;
```

```
Tablespace créée.
```

```
SQL>
```

```
SQL> create temporary tablespace temp_tbs tempfile 'c:\temp_tbs.dat' size 100M autoextend on;
```

```
Tablespace créée.
```

```
SQL>
```

## **2) Création d'un utilisateur DBAASSOCIATION:**

```
SQL> create user DBAASSOCIATION identified by password default tablespace default_tbs temporary tablespace temp_tbs;
```

```
Utilisateur créé.
```

```
SQL>
```

## **3) Donnent tous les privilèges à cet utilisateur:**

```
SQL> grant all privileges to DBAASSOCIATION;
```

```
Autorisation de privilèges (GRANT) acceptée.
```

```
SQL>
```

## **4) Création des tables:**

```
SQL> create table MEMBRES( NUMERO_MBR integer , NOM_MBR VARCHAR(50) , DATE_ADHESION_MBR date , ADRESSE_MBR varchar (100) , constraint pk_num primary key (NUMERO_MBR )) ;
```

```
Table créée.
```

```
SQL> create table CONGRES ( CODE_CGN integer , THEME_CGN varchar(30) , PAYS_CGN varchar(20) , DUREE_CGN varchar (30) ,NB_PARTICIPANT_CGN integer , MONTANT_FIXE number , MONTANT_TOTAL_COTISE number , constraint pk_code primary key (CODE_CGN ) ) ;
```

```
Table créée.
```

```
SQL>
```

```
SQL> CREATE table PARTICIPATIONS (NUMERO_MBR integer , CODE_CGN integer , MONTANT_COTISE number, constraint pk_par primary key (NUMERO_MBR ,CODE_CGN) ,constraint fk_mbr foreign key (NUMERO_MBR) references MEMBRES(NUMERO_MBR) ON DELETE CASCADE,constraint fk_cgn foreign key (CODE_CGN) references CONGRES (CODE_CGN) ON DELETE CASCADE ) ;
```

```
Table créée.
```

```
SQL>
```

## **5)implémentation des contraintes:**

### **5-a)**

```
SQL> ALTER TABLE CONGRES ADD CONSTRAINT CHK_NBPARG CHECK(NB_PARTICIPANT_CGN>0);
Table modifiée.
SQL>
```

### **5-b)**

```
SQL> alter table MEMBRES add constraint ck_MEMBRES_null check(NUMERO_MBR is not null);
Table modifiée.
SQL>
```

### **5-c)**

```
SQL> ALTER TABLE CONGRES ADD CONSTRAINT CHK_MONTANT CHECK( MONTANT_FIXE>=0 AND MONTANT_TOTAL_COTISE>=0);
Table modifiée.
SQL> alter table PARTICIPATIONS add constraint ck_MONTANTC check(MONTANT_COTISE>0);
Table modifiée.
SQL>
```

### **5-d)**

```
SQL> ALTER TABLE CONGRES ADD CONSTRAINT UC_CONGRES UNIQUE (THEME_CGN);
Table modifiée.
SQL>
```

### **5-e)**

```
SQL> create trigger montant_sup
2 before insert on PARTICIPATIONS
3 FOR EACH ROW
4 DECLARE
5 mon_fixe CONGRES.MONTANT_FIXE%type;
6 BEGIN
7 select MONTANT_FIXE into mon_fixe from CONGRES where CONGRES.CODE_CGN=:new.CODE_CGN;
8 if mon_fixe > :new.MONTANT_COTISE then
9 RAISE_APPLICATION_ERROR(20056,'MONTANT_COTISE >> montant fixé par participant pour un congrès.');
```

Dûclencheur crûû.

```
10 end if;
11 end;
12 /
SQL>
```

## 5-f)

```
SQL> create trigger mise_aj_part
  2 AFTER insert on PARTICIPATIONS
  3 FOR EACH ROW
  4 BEGIN
  5 UPDATE CONGRES SET MONTANT_TOTAL_COTISE=MONTANT_TOTAL_COTISE+:new.MONTANT_COTISE,
  6 NB_PARTICIPANT_CGN=NB_PARTICIPANT_CGN+1
  7 WHERE CODE_CGN=:new.CODE_CGN;
  8 end;
  9 /
```

D  clencheur cr   .

SQL>

## 6) Ins  rer 3 tuples dans chaque relation:

```
SQL> INSERT INTO MEMBRES VALUES (1,'YOUSFI ZAKARIA','28/11/2022','CITY 99 BLOC B ORAN');
```

1 ligne cr   e.

```
SQL> INSERT INTO MEMBRES VALUES (2,'KRIMLIN BANA','29/11/2022','MOSCOW CITY KHARKIV RUSSIA');
```

1 ligne cr   e.

```
SQL> INSERT INTO MEMBRES VALUES (3,'SNOOP DOGG','30/11/2022','60ST CALIFORNIA UNITED STATES');
```

1 ligne cr   e.

SQL>

```
SQL> INSERT INTO CONGRES VALUES (1,'A','ALGERIA','2h30mins',1,5000,6000);
```

1 ligne cr   e.

```
SQL> INSERT INTO CONGRES VALUES (2,'B','RUSSIA','4h',1,6000,7000);
```

1 ligne cr   e.

```
SQL> INSERT INTO CONGRES VALUES (3,'C','USA','6h',1,7000,8000);
```

1 ligne cr   e.

SQL>

```
SQL> INSERT INTO PARTICIPATIONS VALUES (1,1,6000);
```

1 ligne cr   e.

```
SQL> INSERT INTO PARTICIPATIONS VALUES (2,2,7000);
```

1 ligne cr   e.

```
SQL> INSERT INTO PARTICIPATIONS VALUES (3,3,8000);
```

1 ligne cr   e.

SQL>

**7) l'état suivant (NUMERO\_MBR, NOM\_MBR, THEME\_CGN, MONTANT\_COTISE):**

```
SQL> SELECT p.NUMERO_MBR,m.NOM_MBR ,c.THEME_CGN,p.MONTANT_COTISE
  2   from PARTICIPATIONS p, MEMBRES m, CONGRES c
  3   where p.NUMERO_MBR=m.NUMERO_MBR and p.CODE_CGN=c.CODE_CGN
  4   order by p.MONTANT_COTISE DESC;
```

| NUMERO_MBR | NOM_MBR          | THEME_CGN | MONTANT_COTISE |
|------------|------------------|-----------|----------------|
| C          | 3 SNOOP DOGG     |           | 8000           |
| B          | 2 KRIMLIN BANA   |           | 7000           |
| A          | 1 YOUSFI ZAKARIA |           | 6000           |

SQL>

**8) le nombre de membres (CODE\_CGN, THEME\_CGN, NBP):**

```
SQL> SELECT c.CODE_CGN,c.THEME_CGN,c.NB_PARTICIPANT_CGN AS NBP
  2   from CONGRES c, PARTICIPATIONS p
  3   where c.CODE_CGN = p.CODE_CGN;
```

| CODE_CGN | THEME_CGN | NBP |
|----------|-----------|-----|
| 1        | A         | 2   |
| 2        | B         | 2   |
| 3        | C         | 2   |

SQL>

9) **le congrès dont le mt total cotise est maximal:**

```
SQL> SELECT CODE_CGN, THEME_CGN, NB_PARTICIPANT_CGN, MONTANT_FIXE, MONTANT_TOTAL_COTISE
  2   from CONGRES
  3   WHERE MONTANT_TOTAL_COTISE = (SELECT MAX(MONTANT_TOTAL_COTISE) FROM CONGRES);
```

| CODE_CGN | THEME_CGN | NB_PARTICIPANT_CGN | MONTANT_FIXE |
|----------|-----------|--------------------|--------------|
| 3        | C         | 2                  | 7000         |

SQL>

THE END.