

Rapport TP3

ENDO

NOM : ACHACHI

Prénom : El hadj Ali

Mat : 181832008538

1) Connecté au dernier utilisateur MSETR :

```
Entrez le nom utilisateur : MSETR
Entrez le mot de passe :

Connecté Ó :
Oracle Database 10g Enterprise Edition Release 10.2.0.3.0 - Production
With the Partitioning, OLAP and Data Mining options
```

2) créer nouvelle utilisateur DATA et autoriser tous les privilèges:

```
SQL> CREATE USER DATA Identified by ENG;

Utilisateur créé.

SQL> GRANT ALL PRIVILEGES TO DATA;

Autorisation de privilèges (GRANT) acceptée.
```

3) création des tables avec leur remplissage :

- DPcclient :

```
SQL> CREATE TABLE DPCCClient (  
2   CodeClient int,  
3   NomClient VARCHAR(50),  
4   SexeClient VARCHAR(1),  
5   CodeVille int,  
6   NomVille VARCHAR(50),  
7   CodeWilaya int,  
8   NomWilaya VARCHAR(30),  
9   constraint nk_SexeClient check(SexeClient in ('M','F')),  
10  CONSTRAINT sk_client PRIMARY KEY (CodeClient)  
11 );  
  
Table créée.  
  
SQL> BEGIN  
2   FOR i IN  
3   (SELECT NUMCLIENT, NOMCLIENT, SEXE, V.CODEVILLE, V.NOMVILLE,  
4    FROM MSETR.Client C, MSETR.Ville V, MSETR.Wilaya W  
5    WHERE W.CODWIL = V.CODEWILAYA  
6    AND V.CODEVILLE = C.CODEVILLE)  
7   LOOP  
8   INSERT INTO DPCCClient VALUES (i.NUMCLIENT , i.NOMCLIENT, i.SEXE, i.CODEVILLE, i.NOMVILLE, i.CODWIL, i.NOMWILAYA);  
9   END LOOP;  
10  COMMIT;  
11  END;  
12  /  
  
Procédure PL/SQL terminée avec succès.  
  
SQL> SELECT COUNT(*) FROM DPCCClient;  
  
COUNT(*)  
-----  
1065566
```

- Dtypeligne :

```
SQL> CREATE TABLE DTypeLigne(  
2   CodeTypeLigne int,  
3   TypeLigne varchar(30),  
4   constraint pk_DTypeLigne PRIMARY KEY(CodeTypeLigne)  
5  );  
  
Table créée.  
  
SQL> BEGIN  
2   FOR i IN  
3   (SELECT codetp, Tligne  
4    FROM MSETR.typeligne )  
5   LOOP  
6   INSERT INTO DTypeLigne VALUES (i.codetp , i.Tligne);  
7   END LOOP;  
8   COMMIT;  
9   END;  
10  /  
  
Procédure PL/SQL terminée avec succès.  
  
SQL> SELECT COUNT(*) FROM DTypeLigne;  
  
COUNT(*)  
-----  
10
```

- Dtypeappel :

```
SQL> CREATE TABLE DTypeAppel(
2      CodeTypeAppel int,
3      TypeAppel varchar(30),
4      constraint pk_DTypeAppel PRIMARY KEY (CodeTypeAppel)
5  );
```

Table cr   e.

```
SQL> BEGIN
2  FOR i IN
3  (SELECT codeta, typeap
4    FROM MSETR.typeappel )
5  LOOP
6  INSERT INTO DTypeAppel VALUES (i.codeta , i.typeap);
7  END LOOP;
8  COMMIT;
9  END;
10 /
```

Proc  dure PL/SQL termin  e avec succ  s.

```
SQL> SELECT COUNT(*) FROM DTypeAppel;
```

COUNT(*)
2

- Ddestinataire :

```
SQL> CREATE TABLE DDestinataire (
2      CodeOp  rateurDestinataire int,
3      NomOp  rateurDestinataire varchar(30),
4      constraint pk_DDestinataire PRIMARY KEY (CodeOp  rateurDestinataire)
5  );
```

Table cr   e.

```
SQL> BEGIN
2  FOR i IN
3  (SELECT codedo, nomop
4    FROM MSETR.destinataire )
5  LOOP
6  INSERT INTO DDestinataire VALUES (i.codedo , i.nomop);
7  END LOOP;
8  COMMIT;
9  END;
10 /
```

Proc  dure PL/SQL termin  e avec succ  s.

```
SQL> SELECT COUNT(*) FROM DDestinataire;
```

COUNT(*)
522

- DTemps : (avec la création de la séquence)

```
SQL> CREATE TABLE DTemps (
  2     CodeTemps NUMBER(6),
  3     Jour VARCHAR(12),
  4     LibJour VARCHAR(15),
  5     Mois VARCHAR(8),
  6     Libmois VARCHAR(12),
  7     Annee VARCHAR(4),
  8     CONSTRAINT pk_DTemps PRIMARY KEY (CodeTemps)
  9 );
```

Table cr  e.

```
SQL> CREATE SEQUENCE seq
  2  MINVALUE 1
  3  MAXVALUE 10000
  4  START WITH 1
  5  INCREMENT BY 1;
```

S  quence cr  e.

```
SQL> BEGIN
  2  FOR i IN
  3  (SELECT DISTINCT TO_CHAR(DATEAPP,'DD/MM/YYYY') AS Jour,
  4  TO_CHAR( DATEAPP,'DAY') AS LibJour,
  5  TO_CHAR( DATEAPP,'MM/YYYY') AS Mois,
  6  TO_CHAR( DATEAPP,'MONTH') AS Libmois,
  7  TO_CHAR( DATEAPP,'YYYY') AS Annee
  8  FROM MSETR.Appel) LOOP
  9  INSERT INTO DTemps VALUES (seq.NEXTVAL, i.Jour, i.LibJour, i.Mois, i.Libmois, i.Annee);
 10  END LOOP;
 11  COMMIT;
 12  END;
 13  /
```

Proc  dure PL/SQL termin  e avec succ  s.

```
SQL> SELECT COUNT(*) FROM DTemps;
```

```
  COUNT(*)
-----
       731
```

- FAppel :

```
SQL> CREATE TABLE FAppel (  
 2   CodeClient int,  
 3   CodeTypeLigne int,  
 4   CodeTypeAppel int,  
 5   CodeTemps NUMBER(6),  
 6   NBAppels int,  
 7   Duree int,  
 8   CONSTRAINT PK_FAppel PRIMARY KEY (CodeClient, CodeTypeLigne, CodeTypeAppel, CodeTemps),  
 9   CONSTRAINT FK_FH_DP FOREIGN KEY (CodeClient) REFERENCES DPCClient,  
10   CONSTRAINT FK_FH_DS FOREIGN KEY (CodeTypeLigne) REFERENCES DTypeLigne ,  
11   CONSTRAINT FK_FH_DM FOREIGN KEY (CodeTypeAppel) REFERENCES DTypeAppel,  
12   CONSTRAINT FK_FH_DT FOREIGN KEY ( CodeTemps) REFERENCES DTemps  
13 );
```

Table cr  e.

```
SQL> BEGIN  
 2 FOR i IN  
 3 (SELECT L.numclient, L.codetp, A.codeta , A.codedo , D.CodeTemps, count(A.CodeAppel) as NBAppel, sum(A.Duree) as Dure  
 4 FROM MSETR.ligne L, DTemps D , MSETR.Appel A  
 5 WHERE D.Jour = A.DateApp  
 6 AND L.numligne = A.numligne  
 7 GROUP BY numclient, codetp, codeta , codedo, CodeTemps  
 8 )  
 9 LOOP  
10 INSERT INTO FAppel VALUES (i.numclient, i.codetp, i.codeta, i.codedo, i.CodeTemps, i.NBAppel , i.Dure) log errors into err$_FAppel reject limit unlimited;  
11 END LOOP;  
12 COMMIT;  
13 END;  
14 /
```

Proc  dure PL/SQL termin  e avec succ  s.

```
SQL> SELECT COUNT(*) FROM FAppel;
```

```
  COUNT(*)  
-----  
    2326630
```

Note :

J'ai utilise " log errors into err\$_FAppel reject limit unlimited ; "
apr  s le INSERT dans le remplissage du table FAppel a cause d'erreur
violation du constraint.

