Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université des Sciences et de la Technologie Houari Boumediene

Faculté d'Informatique

Spécialité : MASTER 1 Big Data Analytics

Rapport de TP3 ENDO

Travail présenté à Monsieur SELMOUNE Nazih

Travail présenté par :

AISSANI Anouar

161835024828

Année Universitaire: 2021/2022

1. **Créer un nouvel utilisateur:**

```
SQL> create user TP3ENDO identified by psw;


User created.


SQL> grant all privileges to TP3ENDO;


Grant succeeded.


SQL> connect TP3ENDO/ psw;
Connected.
```

2. **Créer le schéma (tables + contraintes):**

```
SQL> CREATE TABLE DClient (
  2          CodeClient NUMBER(10),
  3          NomClient VARCHAR(10),
  4          SexeClient VARCHAR(1),
  5          CodeVille VARCHAR(10),
  6          NomVille VARCHAR(10),
  7          CodeWilaya VARCHAR(10),
  8          NomWilaya VARCHAR(10),
  9          CONSTRAINT pk_DClient PRIMARY KEY (CodeClient)
 10  );


Table created.
```

```
SQL> CREATE TABLE DTypeLigne (
  2      CodeTypeLigne NUMBER(10),
  3      TypeLigne VARCHAR(10),
  4      CONSTRAINT pf_DTypeLigne PRIMARY KEY (CodeTypeLigne)
  5  );

Table created.

SQL> CREATE TABLE DTypeAppel (
  2      CodeTypeAppel NUMBER(10),
  3      TypeAppel VARCHAR(20),
  4      CONSTRAINT pk_DTypeAppel PRIMARY KEY (CodeTypeAppel)
  5  );

Table created.

SQL> CREATE TABLE DDestinataire (
  2      CodeOperateurDestinataire NUMBER(10),
  3      NomOperateurDestinataire VARCHAR(50),
  4      CONSTRAINT pk_DDestinataire PRIMARY KEY (CodeOperateurDestinataire)
  5  );

Table created.
```

```
SQL> CREATE TABLE DTemps (
  2      CodeTemps NUMBER(10),
  3      Jour VARCHAR(10),
  4      LibJour VARCHAR(10),
  5      Mois VARCHAR(7),
  6      Libmois VARCHAR(10),
  7      Annee VARCHAR(4),
  8      CONSTRAINT pk_DTemps PRIMARY KEY (CodeTemps)
  9  );

Table created.

SQL> CREATE TABLE FAppel (
  2      CodeClient NUMBER(10),
  3      CodeTypeLigne NUMBER(10),
  4      CodeTypeAppel NUMBER(10),
  5      CodeOperateurDestinataire NUMBER(10),
  6      CodeTemps NUMBER(10),
  7      NBAppels NUMBER,
  8      Duree NUMBER,
  9      CONSTRAINT fk_DClient FOREIGN KEY (CodeClient) REFERENCES DClient
         (CodeClient),
 10      CONSTRAINT fk_DTypeLigne FOREIGN KEY (CodeTypeLigne) REFERENCES
```

```
         DTypeLigne (CodeTypeLigne),
11       CONSTRAINT fk_DTypeAppel FOREIGN KEY (CodeTypeAppel) REFERENCES DTypeAppel
         (CodeTypeAppel),
12       CONSTRAINT fk_DDestinataire FOREIGN KEY (CodeOperateurDestinataire)
         REFERENCES DDestinataire (CodeOperateurDestinataire),
13       CONSTRAINT fk_DTemps FOREIGN KEY (CodeTemps) REFERENCES DTemps (CodeTemps),
14       CONSTRAINT pk_FAppel PRIMARY KEY (CodeClient, CodeTypeLigne, CodeTypeAppel,
          CodeOperateurDestinataire, CodeTemps)
15   );


Table created.
```

3. Remplir les tables:

```
SQL> begin
  2  for i in
  3  ( SELECT c.NumClient, c.NomClient, c.SexeClient, v.CodeVille, v.NomVille,
      w.CodeWilaya, w.NomWilaya
  4  FROM master.Client c, master.Ville v, master.Wilaya w
  5  WHERE c.CodeVille = v.CodeVille
  6  AND v.CodeWilaya = w.CodeWilaya) loop
  7  insert into DClient values(i.NumClient, i.NomClient, i.SexeClient,
     i.CodeVille, i.NomVille, i.CodeWilaya, i.NomWilaya);
  8  end loop;
  9  commit ;
 10  end ;
 11  /
```

```
PL/SQL procedure successfully completed.


SQL> begin
  2  for i in
  3  ( SELECT CodeTypeLigne, TypeLigne
  4  FROM master.TypeLigne) loop
  5  insert into DTypeLigne values(i.CodeTypeLigne, i.TypeLigne);
  6  end loop;
  7  commit ;
  8  end ;
  9  /


PL/SQL procedure successfully completed.


SQL> begin
  2  for i in
  3  ( SELECT CodeTypeAppel, TypeAppel
  4  FROM master.TypeAppel) loop
  5  insert into DTypeAppel values(i.CodeTypeAppel, i.TypeAppel);
  6  end loop;
  7  commit ;
  8  end ;
  9  /
```

```
PL/SQL procedure successfully completed.


SQL> begin
  2  for i in
  3  ( SELECT CodeOperateurDstinataire, NomOperateurDstinataire
  4  FROM master.Destinataire) loop
  5  insert into DDestinataire values(i.CodeOperateurDstinataire,
     i.NomOperateurDstinataire);
  6  end loop;
  7  commit ;
  8  end ;
  9  /


PL/SQL procedure successfully completed.


SQL> CREATE SEQUENCE seq
  2  MINVALUE 1
  3  MAXVALUE 100000
  4  START WITH 1
  5  INCREMENT BY 1;


Sequence created.
```

```
SQL> BEGIN
  2  For I in (SELECT distinct
  3  TO_CHAR(DateAppel,'DD/MM/YYYY') as jour,
  4  TO_CHAR(DateAppel,'DAY') as libjour,
  5  TO_CHAR(DateAppel,'MM/YYYY') as Mois,
  6  TO_CHAR(DateAppel,'MONTH') as Libmois,
  7  TO_CHAR(DateAppel,'YYYY') as Annee
  8  FROM master.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  /

PL/SQL procedure successfully completed.

SQL> begin
  2  for i in
  3  ( SELECT DISTINCT c.NumClient , l.CodeTypeLigne, a.CodeTypeAppel,
     a.CodeOperateurDstinataire, t.CodeTemps,
  4  count(*) as NBAppels, SUM(a.DureeAppel) as Duree
  5  FROM master.Client c, master.Ligne l,master.Appel a, DTemps t
```

```
  6  WHERE  c.NumClient = l.NumClient
  7  AND l.NumeroLigne = a.NumeroLigne
  8  AND t.Jour = TO_CHAR(a.DateAppel,'DD/MM/YYYY')
  9  GROUP BY c.NumClient, l.CodeTypeLigne, a.CodeTypeAppel,
     a.CodeOperateurDstinataire, t.CodeTemps)
 10  LOOP
 11  insert into FAppel values(i.NumClient, i.CodeTypeLigne, i.CodeTypeAppel,
     i.CodeOperateurDstinataire, i.CodeTemps, i.NBAppels, i.Duree);
 12  end loop;
 13  commit ;
 14  end ;
 15  /


PL/SQL procedure successfully completed.
```