



Faculté des Sciences d'Agadir – Ibn Zohr

Licence d'excellence
Filière : Analytique de Données et Intelligence
Artificielle

TP 3

COMPT RENDU

Réalisé Par :
Barakat Imane
Elhillali Naoual

1- Création de user TP4 :

```
SQL> alter session set "_oracle_script"=true;
Session altered.

SQL> create user TP4 identified by TP4 default tablespace users quota 100m on users;
User created.

SQL> grant all privileges to TP4;
Grant succeeded.
```

2- Création des tables :

```
SQL> conn TP4/TP4
Connected.
SQL> CREATE TABLE Usine (
  2     NU NUMBER PRIMARY KEY,
  3     NomU VARCHAR2(50),
  4     Ville VARCHAR2(50)
  5 );
Table created.

SQL> CREATE TABLE Produit (
  2     NP NUMBER PRIMARY KEY,
  3     NomP VARCHAR2(50),
  4     Couleur VARCHAR2(50),
  5     Poids NUMBER
  6 );
Table created.

SQL> CREATE TABLE Fournisseur (
  2     NF NUMBER PRIMARY KEY,
  3     NomF VARCHAR2(50),
  4     Statut VARCHAR2(50),
  5     Ville VARCHAR2(50)
  6 );
Table created.

SQL> CREATE TABLE Livraison (
  2     NP NUMBER REFERENCES Produit(NP),
  3     NU NUMBER REFERENCES Usine(NU),
  4     NF NUMBER REFERENCES Fournisseur(NF),
  5     Quantite NUMBER,
  6     PRIMARY KEY (NP, NU, NF)
  7 );
Table created.
```

3- Insertion au tables :

```
SQL> INSERT INTO Usine (NU, NomU, Ville) VALUES (1, 'UsineA', 'VilleA');  
1 row created.
```

```
SQL> INSERT INTO Produit (NP, NomP, Couleur, Poids) VALUES (101, 'ProduitA', 'Rouge', 10.5);  
1 row created.
```

```
SQL> INSERT INTO Fournisseur (NF, NomF, Statut, Ville) VALUES (201, 'FournisseurA', 'Actif', 'Agadir');  
1 row created.
```

```
SQL> INSERT INTO Livraison (NP, NU, NF, Quantite) VALUES (101, 1, 201, 50);  
1 row created.
```

4- le nom et la ville de toutes les usines :

```
SQL> SELECT UPPER(NomU) AS NomUsine, UPPER(Ville) AS VilleUsine  
2 FROM Usine;  
  
NOMUSINE  
-----  
VILLEUSINE  
-----  
USINEA  
VILLEA  
  
USINEB  
AGADIR  
  
USINEC  
CASA  
  
NOMUSINE  
-----  
VILLEUSINE  
-----  
USINED  
RABAT  
  
USINEE  
TAROUDANT  
  
USINEF  
LAAYOUNE  
  
6 rows selected.
```

- 5- le numéro, le nom et la ville de toutes les usines dont le nom contient plast :

```
SQL> select NomU,Ville from Usine where NomU LIKE '%plast%';
```

NOMU	VILLE
plastique	Agadir

- 6- les numéros des fournisseurs qui approvisionnent l'usine n°1

```
SQL> SELECT DISTINCT Fournisseur.NF AS NumeroFournisseur
2 FROM Fournisseur
3 JOIN Livraison ON Fournisseur.NF = Livraison.NF
4 WHERE Livraison.NU = 1;
```

NUMEROFOURNISSEUR
201
202

- 7- le nom des fournisseurs qui approvisionnent l'usine n°1 en produit n°1.

```
SQL> SELECT DISTINCT Fournisseur.NF AS NumeroFournisseur
2 FROM Fournisseur
3 JOIN Livraison ON Fournisseur.NF = Livraison.NF
4 WHERE Livraison.NU = 1 and Livraison.NP=101;
```

NUMEROFOURNISSEUR
201

- 8- r les numéros des fournisseurs qui approvisionnent les usines n°1 ou n°2.

```
SQL> SELECT DISTINCT Fournisseur.NF AS NumeroFournisseur
2 FROM Fournisseur
3 JOIN Livraison ON Fournisseur.NF = Livraison.NF
4 WHERE Livraison.NU = 1 or Livraison.NU=2;
```

NUMEROFOURNISSEUR
201
203
206
202

- 9- les numéros des usines qui ne reçoivent aucun produit du fournisseur n°1.

```
SQL> SELECT DISTINCT Usine.NU AS NumeroUsine
  2 FROM Usine
  3 WHERE NOT EXISTS (
  4     SELECT 1
  5     FROM Livraison
  6     WHERE Livraison.NU = Usine.NU
  7         AND Livraison.NF = 201
  8 );
```

NUMEROUSINE
6
7
4
5
3

- 10- les numéros des usines qui s'approvisionnent chez le fournisseur n°3.

```
SQL> select distinct Usine.NU as NumeroUsine
  2 from Usine
  3 join Livraison on Usine.NU = Livraison.NU
  4 where Livraison.NF=203;
```

NUMEROUSINE
2

- 11- les numéros des fournisseurs qui fournissent au moins un produit fourni par le fournisseur n°1

```
SQL> select distinct fournisseur.NF as NumeroFournisseur
  2 from fournisseur
  3 join Livraison on Livraison.NF = fournisseur.NF
  4 where Livraison.NP in (
  5     select Livraison.NP
  6     from Livraison
  7     where NF=201);
```

NUMEROFOURNISSEUR
201
203
206
202

- 12- les numéros des usines qui s'approvisionnent uniquement chez le fournisseur n°3.

```
SQL> select Livraison.NU as NumeroUsine
  2 from Livraison
  3 where Livraison.NF=203
  4 and not exists(select 1
  5     from Livraison
  6     where Livraison.NU= Livraison.NU and Livraison.NF<>203);
```

no rows selected

13- le nom et le nombre d'usines qu'il approvisionne. L'affichage sera ordonné par nombre d'usines

```
SQL> SELECT Fournisseur.NF AS NumeroFournisseur,  
2 Fournisseur.NomF AS NomFournisseur,  
3 COUNT(DISTINCT Livraison.NU) AS NombreUsinesApprovisionnees  
4 FROM Fournisseur  
5 JOIN Livraison ON Fournisseur.NF = Livraison.NF  
6 GROUP BY Fournisseur.NF, Fournisseur.NomF  
7 ORDER BY NombreUsinesApprovisionnees DESC;
```

NUMEROFOURNISSEUR	NOMFOURNISSEUR	NOMBREUSINESAPPROVISIONNEES
201	FournisseurA	2
206	FournisseurF	2
204	FournisseurD	1

NUMEROFOURNISSEUR	NOMFOURNISSEUR	NOMBREUSINESAPPROVISIONNEES
202	FournisseurB	1
205	FournisseurE	1
203	FournisseurC	1

6 rows selected.

14- le nom des fournisseurs qui approvisionne plus de 4 usines

```
SQL> SELECT Fournisseur.NomF AS NomFournisseur  
2 FROM Fournisseur  
3 JOIN Livraison ON Fournisseur.NF = Livraison.NF  
4 GROUP BY Fournisseur.NF, Fournisseur.NomF  
5 HAVING COUNT(DISTINCT Livraison.NU) > 4;
```

no rows selected

15- le nom des produits fournis à plus de 4 usines d'Agadir

```
SQL> SELECT Produit.NomP AS NomProduit
  2  FROM Produit
  3  JOIN Livraison ON Produit.NP = Livraison.NP
  4  JOIN Usine ON Livraison.NU = Usine.NU
  5  WHERE Usine.Ville = 'Agadir'
  6  GROUP BY Produit.NP, Produit.NomP
  7  HAVING COUNT(DISTINCT Livraison.NU) > 4;
```

no rows selected

- 16- le nom (seule l'initiale en majuscule) et la quantité totale fournie aux usines**

```
SQL> SELECT INITCAP(Produit.NomP) AS NomProduit,
  2  SUM(Livraison.Quantite) AS QuantiteTotale
  3  FROM Produit
  4  JOIN Livraison ON Produit.NP = Livraison.NP
  5  GROUP BY Produit.NP, Produit.NomP;
```

NOMPRODUIT	QUANTITETOTALE
Produita	170
Produitb	40
Produitc	70
Produitd	66

- 17- les fournisseurs qui approvisionnent toutes les usines ordonner par nom**

```
SQL> SELECT Fournisseur.NF AS NumeroFournisseur,
  2  Fournisseur.NomF AS NomFournisseur
  3  FROM Fournisseur
  4  JOIN Livraison ON Fournisseur.NF = Livraison.NF
  5  GROUP BY Fournisseur.NF, Fournisseur.NomF
  6  HAVING COUNT(DISTINCT Livraison.NU) = (SELECT COUNT(DISTINCT NU) FROM Usine)
  7  ORDER BY NomFournisseur;
```

no rows selected