

THOUNY Alexandre  
LARACINE Thibaut

## TP1 BD1 3A APP

### Exercice 1 :

#### Partie 1 :

On a eu un problème avec la création de la table Commandes, il nous est impossible de rajouter des tuples dedans.

Nous avons donc créé la table Commande que nous allons utiliser pour la suite du tp.

i)

```
CREATE TABLE Commande (  
    NoCom    NUMBER(5),  
    Client   VARCHAR2(20),  
    Appellation VARCHAR2(20),  
    NomFour  VARCHAR2(20),  
    Qte      NUMBER(4)  
);
```

```
INSERT INTO Commande VALUES (1535,'Jean', 'Agata', 'BonPom', 6) ;  
INSERT INTO Commande VALUES (1854,'Jean', 'Agata', 'Vincent', 20) ;  
INSERT INTO Commande VALUES (1254,'Paul','Jeannette','Charlie', 20) ;  
INSERT INTO Commande VALUES (1258,'Paul','Jeannette','Charlie', 20) ;  
INSERT INTO Commande VALUES (1596,'Paul','Amandine','Charlie', 12) ;  
INSERT INTO Commande VALUES (2000,'Jean', 'Agata', 'BonPom', 12) ;
```

iii)

DESC Commande;

Et on reçoit :

Nom	NULL ?	Type
NOCOM		NUMBER(5)
CLIENT		VARCHAR2(20)

```
APPELLATION    VARCHAR2(20)
NOMFOUR        VARCHAR2(20)
QTE            NUMBER(4)
```

iv)

```
SELECT * FROM Commande;
```

b)

i)

```
CREATE TABLE Produit(
  Appelation VARCHAR(20),
  NomFour VARCHAR(20),
  Prix DECIMAL(10,2)
)
```

ii)

```
INSERT INTO Produit VALUES('Agata','BonPom',20);
INSERT INTO Produit VALUES('Amandine','Charlie',18);
INSERT INTO Produit VALUES('Agata','Vincent',8.2);
INSERT INTO Produit VALUES('Monalisa','Vincent',4.3);
INSERT INTO Produit VALUES('Agata','Charlie',18.5);
INSERT INTO Produit VALUES('Bintje','Charlie',5.1);
INSERT INTO Produit VALUES('Jeannette','Charlie',5);
INSERT INTO Produit VALUES('Agata','SaV',10);
```

iii)

Avec DESC Produit; on obtient :

```
Nom    NULL ? Type
```

-----

```
APPELLATION    VARCHAR2(20)
NOMFOUR        VARCHAR2(20)
PRIX           NUMBER(10,2)
```

Avec SELECT \* FROM PRODUIT; on vérifie que les tuples ont bien été insérés.

c)

i) SELECT \* FROM BDR.FOURNISSEUR;

ii)

CREATE TABLE Fournisseurs AS

SELECT \* FROM BDR.FOURNISSEUR;

iii)

Avec DESC Fournisseurs; on obtient :

Nom NULL ? Type

-----

NOMFOUR NOT NULL VARCHAR2(20)

STATUT VARCHAR2(20)

VILLE VARCHAR2(20)

Avec SELECT \* FROM PRODUIT; on vérifie que les tuples ont bien été copiés.

## Partie 2 :

/\* 1. Afficher toutes les Commande \*/

SELECT \* FROM Commande;

/\* 2. Afficher les appellations des pommes commandées \*/

SELECT DISTINCT Appellation FROM COMMANDE;

/\* 3. Afficher les fournisseurs de Jean\*/

SELECT DISTINCT Appellation FROM COMMANDE WHERE Client = 'Jean';

/\* 4. Afficher les fournisseurs des pommes Agata et Amandine\*/

SELECT DISTINCT NomFour FROM PRODUIT WHERE Appellation IN ('Agata', 'Amandine')  
AND PRIX < 10;

/\* 5. Afficher les produits commandés par Jean \*/

SELECT Distinct P.\*

FROM PRODUIT P

JOIN COMMANDE C ON

P.Appellation = C.Appellation

```
AND P.NomFour = C.NomFour  
WHERE CLIENT = 'Jean';
```

/\* 6. Afficher les paires de fournisseurs habitant dans la même ville \*/

```
SELECT F1.NOMFOUR, F2.NOMFOUR  
FROM FOURNISSEURS F1  
JOIN FOURNISSEURS F2 ON  
F1.Ville = F2.Ville  
AND F1.NOMFOUR != F2.NOMFOUR;
```

/\* 7. Afficher les produits commandés par Jean qui coûtent plus de 15euros \*/

```
(SELECT Distinct P.*  
FROM PRODUIT P  
WHERE PRIX > 15)  
UNION  
(SELECT Distinct P.*  
FROM PRODUIT P  
JOIN COMMANDE C ON  
P.Appellation = C.Appellation  
AND P.NomFour = C.NomFour  
WHERE CLIENT = 'Jean');
```

/\* 8. Produits qui n'ont pas été commandé \*/

```
SELECT P.*  
FROM PRODUIT P  
LEFT JOIN COMMANDE C ON  
P.Appellation = C.Appellation  
AND P.NomFour = C.NomFour  
WHERE C.NoCom IS NULL;
```

/\* 9. Produits commandés en quantité supérieure à 10 et dont le prix est inférieur à 15 \*/

```
SELECT P.*  
FROM PRODUIT P JOIN COMMANDE C  
ON P.APPELLATION = C.APPELLATION AND P.NOMFOUR = C.NOMFOUR  
WHERE C.QTE > 10 AND P.PRIX < 15;
```

/\* 10. Appellations qui sont fournies par tous les fournisseurs

```
SELECT APPELLATION  
FROM PRODUIT  
GROUP BY APPELLATION
```

HAVING COUNT(DISTINCT NOMFOUR) = (SELECT COUNT(\*) FROM FOURNISSEURS);

## Exercise 2 :

### Partie 1 :

```
CREATE TABLE JOUEUR AS  
SELECT * FROM BDR.JOUEUR;
```

```
CREATE TABLE RENCONTRE AS  
SELECT * FROM BDR.RENCONTRE;
```

```
CREATE TABLE GAIN AS  
SELECT * FROM BDR.GAIN;
```

### Partie 2 :

a)

```
CREATE OR REPLACE VIEW JoueurFR  
AS  
SELECT NuJoueur,Prenom,Nom  
FROM Joueur  
WHERE NATIONALITE='France';
```

b)

```
CREATE OR REPLACE VIEW JoueurFRsponPEUG  
AS  
SELECT UNIQUE J.NuJoueur,J.Prenom,J.Nom  
FROM JoueurFR J JOIN GAIN G on J.NUJOUEUR = G.NUJOUEUR  
WHERE G.NOMSPONSOR='Peugeot';
```

c)

```
CREATE OR REPLACE VIEW PrimeJoueurFRsponPEUG AS  
SELECT J.NuJoueur, SUM(G.PRIME) AS SPrime
```

```
FROM JOUEURFRSPONPEUG J
    JOIN GAIN G ON J.NuJoueur = G.NUJOUEUR
GROUP BY J.NuJoueur;
```

d)

```
CREATE OR REPLACE VIEW RencGagneeJoueurFRriche AS
SELECT NUGAGNANT, NUPERDANT, LIEUTOURNOI, ANNEE
FROM RENCONTRE R
WHERE NUGAGNANT IN (SELECT J.NuJoueur
    FROM JOUEURFR J
        JOIN GAIN G ON J.NuJoueur = G.NUJOUEUR
    GROUP BY J.NuJoueur
    HAVING SUM(G.PRIME) > 1000000);
```

### Partie 3 :

1)

```
SELECT * FROM JOUEUR;
SELECT * FROM GAIN;
SELECT * FROM RENCONTRE;
```

2)

```
SELECT J.NuJoueur, R.LIEUTOURNOI, R.ANNEE
FROM JoueurFR J JOIN GAIN G on J.NUJOUEUR = G.NUJOUEUR
JOIN RENCONTRE R ON J.NUJOUEUR = R.NUGAGNANT OR J.NUJOUEUR =
R.NUPERDANT
WHERE G.NOMSPONSOR='Peugeot' AND R.ANNEE >=1990 AND R.ANNEE <=1994;
```

3)

```
SELECT J.Nom, J.Annais
FROM JOUEUR J JOIN RENCONTRE R ON J.NUJOUEUR = R.NUGAGNANT OR
J.NUJOUEUR = R.NUPERDANT
WHERE R.LIEUTOURNOI='Roland Garros' AND R.ANNEE=1994;
```

4)

```
SELECT UNIQUE J.Nom, J.NATIONALITE
FROM JOUEUR J JOIN RENCONTRE R ON J.NUJOUEUR = R.NUGAGNANT
JOIN GAIN G ON J.NUJOUEUR = G.NUJOUEUR
```

WHERE R.LIEUTOURNOI='Roland Garros' AND G.NOMSPONSOR='Peugeot';

5)

```
SELECT UNIQUE J.Nom, J.NATIONALITE
FROM JOUEUR J
JOIN RENCONTRE R1 ON J.NUJOUEUR = R1.NUGAGNANT OR J.NUJOUEUR =
R1.NUPERDANT
JOIN RENCONTRE R2 ON J.NUJOUEUR = R2.NUGAGNANT OR J.NUJOUEUR =
R2.NUPERDANT
WHERE R1.LIEUTOURNOI = 'Roland Garros' AND R2.LIEUTOURNOI = 'Wimbledon' AND
R1.ANNEE = 1992 AND R2.ANNEE = 1992;
```

6)

```
SELECT COUNT(DISTINCT *)
FROM JOUEUR J JOIN RENCONTRE R ON J.NUJOUEUR = R.NUGAGNANT OR
J.NUJOUEUR = R.NUPERDANT
WHERE R.LIEUTOURNOI='Wimbledon' AND R.ANNEE=1993;
```

7)

```
SELECT ANNEE, AVG(PRIME) AS MOYENNE_PRIME
FROM GAIN
GROUP BY ANNEE;
```

8)

```
SELECT J.Nom
FROM JOUEUR J
JOIN GAIN G ON J.NuJoueur = G.NUJOUEUR
JOIN RENCONTRE R ON J.NUJOUEUR = R.NUGAGNANT OR J.NUJOUEUR =
R.NUPERDANT
WHERE R.LIEUTOURNOI = 'Roland Garros'
GROUP BY J.NuJoueur, J.Nom
HAVING SUM(G.PRIME) > 1000000;
```

9)

```
SELECT J.Nom
FROM JOUEUR J
WHERE J.NuJoueur IN (
    SELECT J.NuJoueur
    FROM JOUEUR J
    JOIN RENCONTRE R ON J.NuJoueur = R.NUPERDANT
    WHERE R.LIEUTOURNOI = 'Wimbledon')
```

```

MINUS
SELECT J.NuJoueur
FROM JOUEUR J
JOIN RENCONTRE R ON J.NuJoueur = R.NUGAGNANT
WHERE R.LIEUTOURNOI = 'Wimbledon'
)
AND J.NuJoueur IN (
    SELECT J.NuJoueur
    FROM JOUEUR J
    JOIN RENCONTRE R ON J.NuJoueur = R.NUGAGNANT
    WHERE R.LIEUTOURNOI = 'Roland Garros'
MINUS
SELECT J.NuJoueur
FROM JOUEUR J
JOIN RENCONTRE R ON J.NuJoueur = R.NUPERDANT
WHERE R.LIEUTOURNOI = 'Roland Garros'
);

10)
SELECT NOM
FROM JOUEUR J
JOIN RENCONTRE R
ON J.NUJOUEUR = R.NUGAGNANT OR J.NUJOUEUR = R.NUPERDANT
WHERE LIEUTOURNOI = 'Roland Garros'
GROUP BY J.NUJOUEUR, J.NOM
HAVING COUNT(DISTINCT ANNEE) = (
    SELECT COUNT(DISTINCT ANNEE)
    FROM RENCONTRE
    WHERE LIEUTOURNOI = 'Roland Garros'
);

11)
SELECT MAX(PRIME), NOM
FROM GAIN G
JOIN JOUEUR J
ON G.NUJOUEUR = J.NUJOUEUR
WHERE ANNEE = 1992
AND PRIME = (SELECT MAX(PRIME) FROM GAIN WHERE ANNEE = 1992)
GROUP BY NOM, PRIME;

```



```
SELECT NOMSPONSOR
FROM GAIN
GROUP BY NOMSPONSOR
HAVING COUNT(DISTINCT LIEUTOURNOI) = (SELECT COUNT(DISTINCT LIEUTOURNOI)
FROM GAIN);
```