

TP 1 - SQL

Correction

Exercice : La base de données "Société"

I)

```
CREATE TABLE Usine(NU SMALLINT CONSTRAINT pkUsine PRIMARY KEY,  
                    NomU VARCHAR2(20) NOT NULL,  
                    Ville VARCHAR2(40));
```

```
CREATE TABLE Produit(NP SMALLINT CONSTRAINT pkProduit PRIMARY KEY,  
                      NomP VARCHAR2(20) NOT NULL,  
                      Couleur VARCHAR2(15),  
                      Poids NUMBER(5,3) CONSTRAINT poidsProduit CHECK (Poids>0));
```

```
CREATE TABLE Fournisseur(  
                        NF SMALLINT CONSTRAINT pkFournisseur PRIMARY KEY,  
                        NomF VARCHAR2(20) NOT NULL,  
                        Statut VARCHAR2(20),  
                        Ville VARCHAR2(40));
```

```
CREATE TABLE Livraison(NP SMALLINT,  
                        NU SMALLINT,  
                        NF SMALLINT,  
                        Quantite NUMBER(5),  
                        CONSTRAINT pkLivraison PRIMARY KEY (NP,NU,NF),  
                        CONSTRAINT refNPlivraison FOREIGN KEY (NP) REFERENCES Produit (NP),  
                        CONSTRAINT refNulivraison FOREIGN KEY (NU) REFERENCES Usine (NU),  
                        CONSTRAINT refNFlivraison FOREIGN KEY (NF) REFERENCES Fournisseur (NF),  
                        CONSTRAINT QuantiteLivraison CHECK (Quantite>0));
```

II)

1) Donner le numéro, le nom et la ville de toutes les usines.

```
SELECT * FROM Usine;
```

2) Donner le numéro et le nom de toutes les usines de Sochaux.

```
SELECT NU, NomU  
FROM Usine  
WHERE Ville = 'Sochaux';
```

3) Donner les numéros de fournisseurs qui approvisionnent l'usine n°1 en produit n°3.

```
SELECT NF  
FROM Livraison  
WHERE NU = 1 AND NP = 3;
```

4) Donner le numéro et le nom des produits qui n'ont pas de couleur renseignée.

SELECT NP, NomP FROM Produit WHERE Couleur IS NULL;

5) Donner le nom des usines sans doublon (trié par ordre croissant).

SELECT DISTINCT NomU FROM Usine ORDER BY NomU;

6) Donner le numéro des usines dont le nom commence par C.

SELECT NU FROM Usine WHERE NomU LIKE 'C%';

7) Donner le numéro des produits dont le nom contient s ou S.

SELECT NP FROM Produit WHERE LOWER(NomP) LIKE '%s%';

8) Donner le nombre total de fournisseurs.

SELECT COUNT() FROM Fournisseur;*

9) Donner le nombre de produits ayant une couleur renseignée.

SELECT COUNT(Couleur) FROM Produit;

ou SELECT COUNT() FROM Produit WHERE Couleur IS NOT NULL;*

10) Donner le poids moyen des produits.

SELECT AVG(Poids) FROM Produit;

11) Donner la somme des poids des produits verts.

SELECT SUM(Poids) FROM Produit WHERE Couleur='vert';

12) Donner le plus petit poids parmi les produits dont on connaît la couleur.

SELECT MIN(Poids) FROM Produit WHERE Couleur IS NOT NULL;

13) Donner le nombre de produits livrés par chaque fournisseur (décrit par leur numéro).

SELECT NF, COUNT(DISTINCT NP) FROM Livraison

GROUP BY Livraison.NF;

14) Donner le poids moyen des produits selon leur couleur.

SELECT Couleur, AVG(Poids) FROM Produit

GROUP BY Couleur;

15) Donner la couleur des produits dont le poids moyen est supérieur à 10kg.

SELECT Couleur, AVG(Poids) FROM Produit

GROUP BY Couleur

HAVING AVG(POIDS) > 10;

16) Donner les noms des fournisseurs qui approvisionnent l'usine n°1 en produit n°3.

SELECT DISTINCT NomF FROM Fournisseur, Livraison

WHERE Fournisseur.NF = Livraison.NF

AND Livraison.NU=1 AND Livraison.NP=3;

ou

```
SELECT DISTINCT NomF
FROM Fournisseur
WHERE NF IN (SELECT NF FROM Livraison WHERE NU=1 AND NP=3);
```

17) Donner le nom et la couleur des produits livrés par le fournisseur n°2.

```
SELECT DISTINCT NomP, Couleur
FROM Produit, Livraison
WHERE Livraison.NP = Produit.NP AND NF = 2;
```

ou

```
SELECT NomP, Couleur FROM Produit
WHERE NP IN (SELECT NP FROM Livraison WHERE NF=2);
```

18) Donner les numéros des fournisseurs qui approvisionnent l'usine n°1 en produit rouge.

```
SELECT DISTINCT NF
FROM Livraison, Produit
WHERE Couleur = 'rouge' AND Livraison.NP=Produit.NP AND NU=1;
```

ou

```
SELECT DISTINCT NF
FROM Livraison
WHERE NP IN (SELECT NP FROM Produit WHERE Couleur='rouge') AND NU = 1;
```

19) Donner les noms des fournisseurs qui approvisionnent une usine de Sochaux ou de Paris en produit rouge.

```
SELECT DISTINCT NomF
FROM Livraison, Produit, Fournisseur, Usine
WHERE Couleur='rouge' AND Livraison.NP=Produit.NP AND Livraison.NF=Fournisseur.NF
AND Livraison.NU=Usine.NU AND (Usine.Ville='Sochaux' OR Usine.Ville='Paris');
```

ou

```
SELECT DISTINCT NomF
FROM Fournisseur
WHERE NF IN (SELECT NF FROM Livraison WHERE NP IN
(SELECT NP FROM Produit WHERE Couleur='rouge') AND
NU IN (SELECT NU FROM Usine WHERE Ville='Sochaux' OR Ville='Paris'));
```

20) Donner les numéros des produits livrés à une usine par un fournisseur de la même ville.

```
SELECT DISTINCT NP
FROM Livraison, Fournisseur, Usine
WHERE Livraison.NF=Fournisseur.NF AND Livraison.NU=Usine.NU
AND Usine.Ville=Fournisseur.Ville;
```

21) Donner les numéros des usines qui ont au moins un fournisseur qui n'est pas de la même ville.

```
SELECT DISTINCT Livraison.NU
FROM Livraison, Fournisseur, Usine
WHERE Livraison.NF=Fournisseur.NF AND Livraison.NU=Usine.NU
AND Usine.Ville<>Fournisseur.Ville;
```

22) Donner les numéros des fournisseurs qui approvisionnent à la fois les usines n°1 et n°2.

```
SELECT DISTINCT first.NF
FROM Livraison first, Livraison second
WHERE first.NF=second.NF AND first.NU=1 AND second.NU=2;
ou
SELECT DISTINCT NF FROM Livraison
WHERE NF IN (SELECT NF FROM Livraison WHERE NU=1) AND NU=2;
```

23) Donner les numéros des usines qui utilisent au moins un produit disponible chez le fournisseur n°3 (c'est à dire un produit qu'il livre mais pas nécessairement à cette usine).

```
SELECT DISTINCT NU
FROM Livraison
WHERE NP IN (SELECT NP FROM Livraison WHERE NF=3);
```

24) Donner les numéros des usines qui s'approvisionnent uniquement chez le fournisseur n°3.

```
SELECT NU
FROM Livraison
WHERE NU NOT IN (SELECT NU FROM Livraison WHERE NF<>3);
```

25) Donner les numéros des usines qui ne reçoivent aucun produit rouge d'un fournisseur parisien.

```
SELECT NU FROM Livraison
WHERE NU NOT IN (SELECT NU FROM Livraison, Fournisseur, Produit
                WHERE Livraison.NP=Produit.NP AND Livraison.NF=Fournisseur.NF
                AND Couleur='rouge' AND Ville='Paris');
```

26) Donner le nombre de produits livrés par un fournisseur de Paris.

```
SELECT COUNT(DISTINCT NP) FROM Livraison, Fournisseur
WHERE Livraison.NF=Fournisseur.NF AND Ville='Paris';
```

27) Donner le numéro du produit le plus léger (ou les numéros des produits les plus légers si plusieurs produits ont ce même poids).

```
SELECT NP FROM Produit
WHERE poids IN (SELECT MIN(poids) FROM Produit);
ou
SELECT NP FROM Produit P1
WHERE NOT EXISTS (SELECT * FROM Produit P2 WHERE P1.poids>P2.poids);
```

28) Donner le nombre de produits livrés par chaque fournisseur (décrit par leur nom).

```
SELECT NomF, COUNT(DISTINCT NP) FROM Livraison, Fournisseur
WHERE Livraison.NF=Fournisseur.NF
GROUP BY NomF;
```

29) Donner les numéros des usines qui achètent au fournisseur n°3 tous les produits qu'il fournit.

```
SELECT DISTINCT NU
FROM Livraison
WHERE NF=3
GROUP BY NU
HAVING COUNT(DISTINCT NP)=(SELECT COUNT(DISTINCT NP)
                           FROM Livraison WHERE NF=3);
```

30) Donner les numéros des produits qui sont livrés à toutes les usines de Paris.

```
SELECT DISTINCT Livraison.NP
FROM Livraison, Usine
WHERE Livraison.NU=Usine.NU AND Usine.Ville='Paris'
GROUP BY Livraison.NP
HAVING COUNT(DISTINCT Livraison.NU)=(SELECT COUNT(NU)
                                     FROM Usine WHERE Ville='Paris');
```

ou

```
SELECT NP
FROM Produit
WHERE NOT EXISTS (SELECT NU FROM Usine WHERE
NOT EXISTS (SELECT * FROM Livraison WHERE NOT (Ville='Paris')
            OR (Produit.NP=Livraison.NP AND Usine.NU=Livraison.NU)));
```

31) Donner les numéros des fournisseurs qui approvisionnent toutes usines avec un même produit.

```
SELECT DISTINCT NF
FROM Livraison
GROUP BY NF, NP
HAVING COUNT(DISTINCT NU)=(SELECT COUNT(NU) FROM Usine);
```

ou

```
SELECT NF
FROM Fournisseur
WHERE EXISTS (SELECT NP FROM Produit
              WHERE NOT EXISTS (SELECT NU FROM Usine
                                WHERE NOT EXISTS (SELECT * FROM Livraison
                                                    WHERE Fournisseur.NF=Livraison.NF AND Usine.NU=Livraison.NU
                                                    AND Produit.NP=Livraison.NP)))) ;
```

32) Donner pour chaque usine (décrit par leur numéro), la quantité totale de produits livrés (attention, les usines n'ayant aucune livraison doivent apparaître dans les résultats).

```
SELECT Usine.NU, SUM(Quantite) FROM Usine  
LEFT JOIN Livraison ON Usine.NU=Livraison.NU  
GROUP BY Usine.NU;
```

33) Donner les numéros des fournisseurs qui fournissent au moins un produit fourni par un fournisseur qui fournit au moins un produit rouge.

```
SELECT DISTINCT Livraison.NF  
FROM Livraison, Livraison Livraison1, Livraison Livraison2, Produit  
WHERE Couleur='rouge' AND Produit.NP=Livraison2.NP  
AND Livraison2.NF=Livraison1.NF AND Livraison1.NP=Livraison.NP;  
ou  
SELECT DISTINCT NF  
FROM Livraison  
WHERE NP IN (SELECT NP FROM Livraison WHERE NF IN  
              (SELECT NF FROM Livraison WHERE NP IN (SELECT NP  
              FROM Produit WHERE Couleur='rouge'))));
```

34) Ajouter un nouveau fournisseur : < 4, Dupont, sous-traitant, Saint-Etienne >.

```
INSERT INTO Fournisseur VALUES (4, 'Dupont', 'sous-traitant', 'Saint-Etienne');
```

35) Supprimer tous les produits de couleur noire et de numéro compris entre 1 et 3.

Remarque : à cause de la contrainte d'intégrité référentielle (clé étrangère) sur NP dans "Livraison", il faut supprimer les livraisons concernant ces produits avant de pouvoir les supprimer de "Produit".

```
DELETE FROM Livraison WHERE NP IN (SELECT NP FROM Produit WHERE NP>=1 AND  
NP<=3 AND Couleur='noir');  
DELETE FROM Produit WHERE NP>=1 AND NP<=3 AND Couleur='noir';
```

36) Changer la ville du fournisseur n°4 : il a déménagé à Lyon.

```
UPDATE Fournisseur SET Ville='Lyon'; WHERE NF=4;
```