

Feuille de T. D. 3 : MySQL

Exercice 1 : La base de données Société

I) Donner les instructions MySQL permettant de créer ces tables (ne pas oublier les clés primaires et les contraintes d'intégrité référentielle.)

II) Exprimer en MySQL les requêtes suivantes :

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

```
SELECT *  
FROM Usine;
```

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

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

3) Donner les numéros de fournisseurs qui approvisionnent l'usine no 1 en produit no 3.

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

4) Donner les noms des fournisseurs qui approvisionnent l'usine no 1 en produit no 3.

```
SELECT NomF  
FROM Fournisseur, Livraison  
WHERE Fournisseur.NF = Livraison.NF  
AND Livraison.NU=1 AND Livraison.NP=3;
```

ou

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

5) Donner le nom et la couleur des produits livrés par le fournisseur no 12.

```
SELECT distinct NomP, Couleur  
FROM Produit, Livraison  
WHERE Livraison.NP = Produit.NP AND NF = 12;
```

ou

```
SELECT NomP, Couleur FROM Produit  
WHERE NP IN ( SELECT NP FROM Livraison WHERE NF =  
12);
```

6) Donner les numéros des fournisseurs qui approvisionnent l'usine no 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;
```

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

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

ou

```

SELECT 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'));

```

- 8) 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;

```

- 9) Donner les NUMéros des produits livrés à une usine de Paris par un fournisseur de Paris.

```

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

```

ou

```

SELECT DISTINCT NP
FROM Livraison
WHERE NF IN (SELECT NF FROM Fournisseur WHERE Ville
= 'Paris') AND NU IN (SELECT NU FROM Usine WHERE Ville
= 'Paris');

```

- 10) 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;

```

- 11) Donner les numéros des fournisseurs qui approvisionnent à la fois les usines no 1 et no 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;

```

- 12) Donner les numéros des usines qui utilisent au moins un produit disponible chez le fournisseur no 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);

```

- 13) Donner le numéro du produit le plus léger (ou les numéros des produits les plus légers si plusieurs produits ont ce même poids). Cette requête peut s'écrire en cherchant le complément du résultat.

```

SELECT NP
FROM Produit
WHERE poids IN (SELECT min(poids) FROM Produit);

```

ou

```

SELECT NP
FROM Produit p1
WHERE NOT EXISTS (SELECT * FROM Produit WHERE
p1.poids > poids);

```

- 14) Donner les NUMéros des usines qui ne reçoivent aucun produit rouge d'un fournisseur parisien.

```

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

```

- 15) 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')));
```

- 16) Donner tous les triplets (VilleF, NP, VilleU) tels qu'un fournisseur de la première ville approvisionne une usine de la deuxième Ville.

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

- 17) Même question qu'en 15) mais sans les triplets où les deux Villes sont identiques.

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

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

```
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) ));
```

ou

```
SELECT DISTINCT NP
FROM Livraison
GROUP BY NP
HAVING SET NU contains (SELECT NU FROM Usine WHERE
Ville = 'Paris');
```

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

```
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) ));
```

ou

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

- 20) Donner les numéros des usines qui achètent au fournisseur no 4 tous les produits qu'il fournit.

```
SELECT DISTINCT NU
FROM Livraison
WHERE NF = 4 GROUP BY NU
HAVING SET NP contains (SELECT DISTINCT NP FROM
Livraison WHERE NF = 4);
```

- 21) Donner les numéros des usines qui s'approvisionnent uniquement chez le fournisseur no 3.

```
SELECT DISTINCT NU
FROM Livraison
GROUP BY NU HAVING SET NF = (3);
```

ou

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

- 22) Ajouter un nouveau fournisseur : < 45, Dupont, sous-traitant, Saint-Etienne >.

```
insert into Fournisseur values(45, Alfred, sous-traitant, Saint-Etienne);
```

- 23) Supprimer tous les produits de Couleur noire et de numéro compris entre 100 et 199.

```
delete Produit WHERE NP ≥ 100 AND NP ≤ 199 AND Couleur
= 'noire';
```

- 24) Changer la ville du fournisseur no1 : il a déménagé à Lyon.

```
UPDATE Fournisseur SET Ville = 'Lyon'; WHERE NF = 1
```