

CORRECTION REQUETES SQL

1.1. Exercices : Projection

- | | |
|--|--|
| 1. Toutes les informations sur les éditeurs
<pre>SELECT *
FROM editeur;</pre> | 2. Noms et prénoms des employés
<pre>SELECT nom_emp, pn_emp
FROM employe;</pre> |
| 3. Noms des villes où on trouve des magasins
<pre>SELECT ville
FROM magasins
where ville is not null;
ou
SELECT ville
FROM magasins;</pre> <p>En effet si le magasin existe il est obligatoirement dans un lieu</p> | 4. Liste des types de livres
<pre>SELECT type
FROM titres;</pre> <p>Ou</p> <pre>SELECT distinct type
FROM titres;</pre> <p>N'affiche pas les doublons</p> |
| 5. Titres et prix des livres TTC (multiplier par 1.19)
<pre>SELECT titre, prix * 1.19 as prixTTC
FROM titres;</pre> | 6. Titres et prix des livres TTC (multiplier par 1.19) seulement si le livre à un prix
<pre>SELECT titre, prix * 1.19 as prixTTC
FROM titres
where prix is not null;</pre> |
| 7. Combien y a-t-il d'éditeurs ?
<pre>SELECT distinct count(*) as nb_editeur
FROM editeur;</pre> | 8. Combien y a-t-il de types de titre différents ?
<pre>SELECT count(distinct type)
FROM titres</pre> |

1.2. Exercices : Sélection ou restriction

- | | |
|--|--|
| 1. Nom, prénom et l'état des auteurs qui habitent dans l'état CA
<pre>SELECT nom_auteur, pn_auteur
FROM auteurs
where pays ='CA';</pre> | 2. Quel est le prix du titre 'Sushi, Anyone?'
<pre>SELECT prix
FROM titres
where titre = 'Sushi, Anyone?';</pre> |
| 3. Nom et prénom des employés embauchés en 1994
<pre>SELECT nom_emp, pn_emp
FROM employe
where date_embauche like '%1994%';</pre> <p>ou</p> <pre>SELECT nom_emp, pn_emp
FROM employe
where date_embauche like '%1994%';</pre> | 4. Liste des noms d'éditeurs dont on ne connaît pas la région.
<pre>SELECT nom_editeur
FROM editeur
where region is null;</pre> |
| 5. Liste des titres publiés en 1991 et dont le cumul annuel des ventes est comprise entre 4000 et 7000 avec une avec une avance de 5000
<pre>select titre
from titres
where YEAR(date_edition) =1991
and avance = 5000
and cumulannuel_vente between 4000 and 7000</pre> | 6. Liste des numéros de commandes passées en 94 et payées au comptant (ON invoice) ou qui ont eu lieu dans le magasin numéro 7131.
<pre>select num_cmd, date_cmd, id_mag, modepaiement
from ventes
where YEAR(date_cmd) =1994
and (modepaiement = 'ON invoice' or id_mag ='7131')</pre> |
| 7. le nom et le prénom des employés qui n'ont pas de centre de rattachement (init_centrale) et dont l'échelon (position_emp) est supérieur à 100
<pre>select nom_emp, pn_emp, position_emp, init_centrale
from employe
where init_centrale=""
and position_emp > 100</pre> | |

1.3. Exercices : tris

1. Liste des titres classés par prix seulement si ils ont un prix

```
select prix ,titre
from titres
where prix is not null
order by prix
```

ou

```
select prix ,titre
from titres
where prix is not null
order by 1
```

2. Liste des ventes classées par modes de paiement puis par quantités décroissantes

```
select modepaiement,qte,num_cmd
from ventes
order by 1, 2 desc
```

ou

```
select modepaiement, qte,num_cmd
from ventes
order by modepaiement, qte desc
```

3. Liste des titres publiés par années croissantes puis pas prix TTC décroissant dans une année

```
select year(date_edition) as annee,prix*1.19 as TTC ,titre
from titres
order by 1 asc, 2 desc
```

1.4. Exercices : jointures

1. Liste des noms d'employés avec la description de leur emploi

```
select nom_emp, desc_emploi
from employe, emplois
where employe.id_emploi=emplois.id_emploi
```

2. Liste des titres des livres avec le ou les noms de leurs auteurs

```
select titre, nom_auteur
from titres, titreauteur, auteurs
where auteurs.id_auteur=titreauteur.id_auteur
and titreauteur.id_titre=titres.id_titre
```

3. Liste des noms d'éditeurs ayant prévus des droits de plus de 20%

```
select nom_editeur,Ro.droits
from editeur as Pu, titres as Ti, droits_prevus as Ro
where Pu.id_editeur=Ti.id_editeur
and Ti.id_titre=Ro.id_titre
and Ro.droits>20
```

4. Liste des noms des employés embauchés avant Howard Snyder.

```
select emp2.nom_emp,emp2.date_embauche
from employe as emp1,employe as emp2
where emp2.date_embauche<emp1.date_embauche
and emp1.nom_emp='snyder'
and emp1.pn_emp='howard'
```

5. Liste des noms d'employés dont la position correspond au niveau maximum de leur emploi

```
select pn_emp, nom_emp ,position_emp
from employe as e, emplois as j
where e.position_emp =j.niv_max
and e.id_emploi=j.id_emploi
```

```
select nom_emp, position_emp, niv_max
from employe, emplois
where employe.id_emploi =emplois.id_emploi
and employe.position_emp =emplois.niv_max
```

6. Liste des titres et nom d'éditeur qui ont le même éditeur que le titre 'Life Without Fear'.

```
select Ti1.titre, nom_editeur, Ti2.titre
from editeur as Pu, titres as Ti1, titres as Ti2
where Pu.id_editeur=Ti2.id_editeur
and Ti1.id_editeur =Ti2.id_editeur
and Ti2.titre='Life Without Fear'
```

7. Liste des magasins qui ont vendu le titre 'Emotional Security: A New Algorithm'

```
select nom_mag
from magasins, ventes, titres
where magasins.id_mag= ventes.id_mag
and ventes.id_titre = titres.id_titre
and titre='Emotional Security: A New Algorithm'
```

1.5. Exercices : regroupement

1. Numéros des livres avec le nombre d'auteurs de chaque titre

```
select titres.id_titre, count(id_auteur) as "nb author par titre"
from titres, titreauteur
where titres.id_titre = titreauteur.id_titre
group by titres.id_titre
```

CORRECTION REQUETES SQL

2. Numéros des magasins avec le nombre de ventes (de commandes : num_cmd) par magasin.

```
select magasins.id_mag, count(num_cmd) as "nb vente par magasins"
from ventes, magasins
where magasins.id_mag= ventes.id_mag
group by magasins.id_mag
```
3. Nombres de titres par type de titre, trier par ordre décroissant de nombres de titres

```
select type, count(id_titre) as "nb titres par type titre"
from titres
group by type
order by 2 desc
```
4. Nombre d'employés par éditeur

```
select nom_editeur, count(id_emp) as "nb d'employés par éditeur"
from editeur, employe
where editeur.id_editeur=employe.id_editeur
group by nom_editeur
```
5. Nom des éditeurs et nombre de titres qu'ils ont édités

```
select nom_editeur, count(id_titre) as "nombre de titres par editeur"
from titres, editeur
where titres.id_editeur = editeur.id_editeur
group by nom_editeur
```
6. Cumul des quantités commandées par numéro de magasin

```
select magasins.id_mag, sum(qte) as "Cumul des qtés cdé par numéro de magasin"
from magasins, ventes
where magasins.id_mag= ventes.id_mag
group by magasins.id_mag
```
7. idem pour le titre 'Is Anger the Enemy?'

```
select magasins.id_mag, sum(qte) as "Cumul des qtés cdé par numéro de magasin", titre
from magasins, ventes, titres
where magasins.id_mag= ventes.id_mag
and titres.id_titre= ventes.id_titre
and titre='Is Anger the Enemy?'
group by magasins.id_mag
```
8. Position moyenne des employés des Editions Lucerne Publishing

```
select nom_editeur, avg(position_emp) as "Position moyenne des employés"
from editeur, employe
where editeur.id_editeur=employe.id_editeur
group by nom_editeur
having nom_editeur ='Lucerne Publishing'
```

ou

```
select avg(position_emp) as "Position moyenne des employés"
from editeur, employe
where editeur.id_editeur=employe.id_editeur
and nom_editeur ='Lucerne Publishing'
```
9. Montant total des droits versés par nom d'éditeur

```
Select nom_editeur, sum(prix*(droits/100)*qte) as "somme des droits versés"
From editeur, titres, ventes
Where editeur.id_editeur=titres.id_editeur
and ventes.id_titre=titres.id_titre
group by nom_editeur
```
10. Nom des éditeurs et nombre d'employés des éditeurs qui emploient exactement 10 salariés

```
select nom_editeur, count(id_emp) as "nb d'employés par éditeur"
from editeur, employe
where editeur.id_editeur=employe.id_editeur
group by nom_editeur
having count(id_emp)=10
```

1.6. Exercices : sous-requêtes

1. Liste des noms d'employés qui travaillent chez un éditeur français.

```
select nom_emp
from employe
where id_editeur = ( select id_editeur from editeur
                    where pays ='France')
```

SOUS MYSQL AVANT LA VERSION 4.1

```
select nom_emp
from employe, editeur
where employe.id_editeur = editeur.id_editeur
and pays ='France'
```

2. Liste des noms, date d'embauche et type d'emploi des employés qui ont le même type d'emploi que Sommer

```
select nom_emp, date_embauche, desc_emploi
from employe, emplois
where employe.id_emploi=emplois.id_emploi
and emplois.id_emploi in (select id_emploi
                          from employe
                          where nom_emp='Sommer')
```

SOUS MYSQL AVANT LA VERSION 4.1

```
select e1.nom_emp, e1.date_embauche, desc_emploi
from employe as e1, employe as e2, emplois
where e1.id_emploi=emplois.id_emploi
and e1.id_emploi= e2.id_emploi
and e2.nom_emp='Sommer'
```

Permet de ne sortir que l'identifiant de l'emploi de l'employé nommé 'Sommer'

3. Liste des noms et type d'emploi des employés qui travaillent chez un éditeur de Boston et qui ont le même type d'emploi que Labrune.

```
select nom_emp, desc_emploi
from employe, emplois, editeur
where employe.id_emploi=emplois.id_emploi
and employe.id_editeur=editeur.id_editeur
and ville like 'Boston'
and emplois.id_emploi in (select id_emploi
                          from employe
                          where nom_emp = 'Labrune')
```

SOUS MYSQL AVANT LA VERSION 4.1

```
select e1.nom_emp, desc_emploi
from employe as e1, employe as e2, emplois, editeur
where e1.id_emploi=emplois.id_emploi
and e1.id_editeur=editeur.id_editeur
and ville like 'Boston'
and e1.id_emploi=e2.id_emploi
and e2.nom_emp = 'Labrune'
```

Permet de ne sortir que l'identifiant de l'emploi de l'employé nommé Labrune

4. Liste des noms des employés qui travaillent pour un éditeur qui a édité au moins un ouvrage de type 'business'.

```
select nom_emp
from employe
where employe.id_editeur in (select titres.id_editeur
                             from titres
                             where titres.type = 'business')
```

SOUS MYSQL AVANT LA VERSION 4.1

```
select distinct nom_emp
from employe , editeur, titres
where employe.id_editeur= editeur.id_editeur
and editeur.id_editeur= titres.id_editeur
and titres.type = 'business'
```

5. Noms des éditeurs qui emploient le plus grand nombre de salariés ?

```
Select nom_editeur, count(id_emp) as tot
from editeur, employe
where editeur.id_editeur=employe.id_editeur
group by nom_editeur
having count(id_emp) >= all ( select count(*)
                             from employe
                             group by id_editeur )
```

```
CREATE TABLE NB_EMP
SELECT COUNT(*) AS NB
FROM EMPLOYE
GROUP BY ID_EDITEUR
```

1.7. Exercice : division

1. Quels sont les numéros des magasins dans lesquels tous les titres ont déjà été commandés ?

```
select id_mag
from magasins
group by id_mag
having count(*) =(select count(*) from titres);
```

VOUS N'AVEZ AUCUNE DE RESULTAT CAR AUCUN MAGASINS N'A VENDU TOUS LES TITRES