Une image contenant jaune, conception

Description générée automatiquement

**Année 2023/2024**

**Syllabus Exercices SQL: Correction**

***Exercice 2.a.ii (Page 12) :***

1. *Quels sont les noms des auteurs habitant la ville de Oakland ?*

**SELECT DISTINCT** a.au\_lname **AS** Nom\_auteurs  
**FROM** authors a  
**WHERE** city = 'Oakland';

1. *Donnez les noms et adresses des auteurs dont le prénom commence par la lettre "A".*

**SELECT** a.au\_id, a.au\_lname, a.address  
**FROM** authors a  
**WHERE** a.au\_fname **LIKE** 'A%';

1. *Donnez les noms et adresses complètes des auteurs qui n'ont pas de numéro de téléphone.*

**SELECT** a.au\_id,a.au\_fname, a.address,a.city,a.state,a.country  
**FROM** authors a  
**WHERE** a.phone **IS NULL**;

1. *Y a-t-il des auteurs californiens dont le numéro de téléphone ne commence pas par "415" ?*

**SELECT** a.au\_id, a.au\_lname, a.au\_fname, a.phone, a.address, a.city, a.state, a.country  
**FROM** authors a  
**WHERE** a.state = 'CA'  
**AND** a.phone **NOT LIKE** '415%';

1. *Quels sont les auteurs habitant au Bénélux ?*

**SELECT** a.au\_id, a.au\_lname, a.au\_fname, a.phone, a.address, a.city, a.state, a.country  
**FROM** authors a  
**WHERE** a.country = 'BEL'  
**OR** a.country = 'LUX'  
**OR** a.country = 'NED';

1. *Donnez les identifiants des éditeurs ayant publié au moins un livre de type "psychologie" ?*

**SELECT DISTINCT** ti.pub\_id  
**FROM** titles ti  
**WHERE** ti.type = 'psychology';

1. *Donnez les identifiants des éditeurs ayant publié au moins un livre de type "psychologie", si l'on omet tous les livres dont le prix est compris entre 10 et 25 $ ?*

**SELECT DISTINCT** ti.pub\_id  
**FROM** titles ti  
**WHERE** ti.type = 'psychology'  
**AND** ((ti.price > 10.00)  
**AND** (ti.price < 25.00));

1. *Donnez la liste des villes de Californie où l'on peut trouver un (ou plusieurs) auteur(s) dont le prénom est Albert ou dont le nom finit par "er".*

**SELECT DISTINCT** au.city  
**FROM** authors au  
**WHERE** au.state = 'CA'  
**AND** ((*upper*(au.au\_fname) **LIKE** 'ALBERT')  
**OR** (*upper*(au.au\_lname) **LIKE** '%ER'))  
**AND** au.city **IS NOT NULL**;

1. *Donnez tous les couples Etat-pays ("state" - "country") de la table des auteurs, pour lesquels l'Etat est fourni, mais le pays est autre que "USA".*

**SELECT DISTINCT** au.state,au.country  
**FROM** authors au  
**WHERE** au.country != 'USA'  
**AND** au.state **IS NOT NULL**;

1. *Pour quels types de livres peut-on trouver des livres de prix inférieur à 15 $ ?*

**SELECT DISTINCT** ti.type  
**FROM** titles ti  
**WHERE** ti.price < 15.00  
**AND** ti.type **IS NOT NULL**;

***Exercice 2.b.iv (Page 17) :***

1. Affichez la liste de tous les livres, en indiquant pour chacun son titre, son prix et le nom de son éditeur.

**SELECT** l.title\_id, l.title, l.price, p.pub\_name  
**FROM** titles l, publishers p  
**WHERE** l.pub\_id = p.pub\_id;

1. Affichez la liste de tous les livres de psychologie, en indiquant pour chacun son titre, son prix et le nom de son éditeur.

**SELECT** t.title,t.price, p.pub\_name  
**FROM** titles t, publishers p  
**WHERE** t.pub\_id = p.pub\_id  
**AND** t.type = 'psychology';

1. Quels sont les auteurs qui ont effectivement écrit un (des) livre(s) présent(s) dans la DB ? Donnez leurs noms et prénoms.

**SELECT DISTINCT** au.au\_id,au.au\_lname, au.au\_fname, au.au\_id  
**FROM** authors au, titleauthor ta  
**WHERE** au.au\_id = ta.au\_id;

1. Dans quels Etats y a-t-il des auteurs qui ont effectivement écrit un (des) livre(s) présent(s) dans la DB ?

**SELECT DISTINCT** au.state  
**FROM** authors au, titleauthor ta  
**WHERE** au.au\_id = ta.au\_id  
**AND** au.state **IS NOT NULL**;

1. Donnez les noms et adresses des magasins qui ont commandé des livres en novembre 1991.

**SELECT DISTINCT** st.stor\_id, st.stor\_name, st address  
**FROM** stores st, sales sa  
**WHERE** sa.stor\_id = st.stor\_id  
**AND** *date\_part*('YEAR',sa.date) = 1991  
**AND** *date\_part*('MONTH',sa.date) = 11;

1. Quels sont les livres de psychologie de moins de 20 $ édités par des éditeurs dont le nom ne commence pas par "Algo" ?

**SELECT** t.title\_id, t.title  
**FROM** titles t, publishers p  
**WHERE** t.pub\_id = p.pub\_id  
**AND** t.type = 'psychology'  
**AND** p.pub\_name **NOT LIKE** 'Algo%'  
**AND** t.price < 20;

1. Donnez les titres des livres écrits par (au moins) un auteur californien (state = "CA").

**SELECT DISTINCT** t.title\_id, t.title  
**FROM** titles t, titleauthor ta, authors au  
**WHERE** t.title\_id = ta.title\_id  
**AND** ta.au\_id = au.au\_id  
**AND** au.state = 'CA';

1. Quels sont les auteurs qui ont écrit un livre (au moins) publié par un éditeur californien ?

**SELECT DISTINCT** au.au\_id, au.au\_lname, au.au\_fname  
**FROM** authors au, titleauthor ta, titles t, publishers p  
**WHERE** au.au\_id = ta.au\_id  
**AND** ta.title\_id = t.title\_id  
**AND** t.pub\_id = p.pub\_id  
**AND** p.state = 'CA';

1. Quels sont les auteurs qui ont écrit un livre (au moins) publié par un éditeur localisé dans leur Etat ?

**SELECT DISTINCT** au.au\_id, au.au\_lname, au.au\_fname  
**FROM** authors au, titleauthor ta, titles t, publishers p  
**WHERE** au.au\_id = ta.au\_id  
**AND** ta.title\_id = t.title\_id  
**AND** t.pub\_id = p.pub\_id  
**AND** au.state = p.state;

1. Quels sont les éditeurs dont on a vendu des livres entre le 1/11/1990 et le 1/3/1991 ?

**SELECT DISTINCT** p.pub\_id, p.pub\_id, p.pub\_name, p.city, p.state  
**FROM** publishers p , titles t, salesdetail sd, sales sa  
**WHERE** t.pub\_id = p.pub\_id  
**AND** t.title\_id = sd.title\_id  
**AND** sd.stor\_id = sa.stor\_id  
**AND** sd.ord\_num = sa.ord\_num  
**AND** sa.date **BETWEEN** '1990-11-1' **AND** '1991-3-1';

1. Quels magasins ont vendu des livres contenant le mot "cook" (ou "Cook") dans leur titre ?

**SELECT DISTINCT** st.stor\_id, st.stor\_name, st.stor\_address  
**FROM** stores st, salesdetail sd, titles t  
**WHERE** st.stor\_id = sd.stor\_id  
**AND** sd.title\_id = t.title\_id  
**AND** (t.title **LIKE** '%Cook%' **OR** *lower*(t.title) **LIKE** '%cook%');  
  
/\* OR \*/  
  
**SELECT DISTINCT** st.stor\_id, st.stor\_name, st.stor\_address  
**FROM** stores st, salesdetail sd, titles t  
**WHERE** st.stor\_id = sd.stor\_id  
**AND** sd.title\_id = t.title\_id  
**AND** t.title\_id **SIMILAR TO** '%[cC]ook%';

1. Y a-t-il des paires de livres publiés par le même éditeur à la même date ?

**SELECT** t1.title, t2.title, t1.title\_id, t2.title\_id  
**FROM** titles t1, titles t2  
**WHERE** t1.pub\_id = t2.pub\_id  
**AND** t1.title\_id < t2.title\_id  
**AND** t1.pubdate = t2.pubdate;

1. Y a-t-il des auteurs n'ayant pas publié tous leurs livres chez le même éditeur ?

**SELECT** au.au\_id, au.au\_fname, au.au\_lname  
**FROM** authors au, titleauthor ta, titles t  
**WHERE** au.au\_id = ta.au\_id  
**AND** ta.title\_id = t.title\_id  
**GROUP BY** au.au\_id, au.au\_fname, au.au\_lname  
**HAVING** *count* (**Distinct** t.pub\_id) > 1;

1. Y a-t-il des livres qui ont été vendus avant leur date de parution ?

**SELECT DISTINCT** t.title\_id, t.title  
**FROM** titles t, salesdetail sd, sales sa  
**WHERE** t.title\_id = sd.title\_id  
**AND** sd.stor\_id = sa.stor\_id  
**AND** sd.ord\_num = sa.ord\_num  
**AND** t.pubdate > sa.date;

1. Quels sont les magasins où l'on a vendu des livres écrits par Anne Ringer ?

**SELECT DISTINCT** st.stor\_id, st.stor\_name  
**FROM** stores st, salesdetail sd, titles t, titleauthor ta, authors au  
**WHERE** au.au\_id = ta.au\_id  
**AND** ta.title\_id = t.title\_id  
**AND** st.stor\_id = sd.stor\_id  
**AND** sd.title\_id = t.title\_id  
**AND** au.au\_fname = 'Anne'  
**AND** au.au\_lname = 'Ringer';

1. Quels sont les Etats où habite au moins un auteur dont on a vendu des livres en Californie en février 1991 ?

**SELECT** au.state  
**FROM** titleauthor ta, authors au, salesdetail sd, sales sa, stores st  
**WHERE** au.au\_id = ta.au\_id  
**AND** ta.title\_id = sd.title\_id  
**AND** sd.ord\_num = sa.ord\_num  
**AND** sd.stor\_id = sa.stor\_id  
**AND** sa.stor\_id = st.stor\_id  
**AND** st.state = 'CA'  
**AND** sa.date **BETWEEN** '1991-02-1' **AND** '1991-02-28'  
**AND** au.state **IS NOT NULL**;

1. Y a-t-il des paires de magasins situés dans le même Etat, où l'on a vendu des livres du même auteur ?

**SELECT** st.stor\_id, st.stor\_name, st2.stor\_id, st2.stor\_name  
**FROM** stores st, stores st2, salesdetail sd,salesdetail sd2, titleauthor ta, titleauthor ta2  
**WHERE** st.stor\_id = sd.stor\_id  
**AND** st2.stor\_id = sd2.stor\_id  
**AND** sd.title\_id = ta.title\_id  
**AND** sd2.title\_id = ta2.title\_id  
**AND** st.stor\_id < st2.stor\_id  
**AND** ta.au\_id = ta2.au\_id  
**AND** st.state = st2.state;

1. Trouvez les paires de co-auteurs.

**SELECT DISTINCT** au1.au\_id, au1.au\_lname, au2.au\_id, au2.au\_lname  
**FROM** authors au1, authors au2, titleauthor ta, titleauthor ta2  
**WHERE** au1.au\_id = ta.au\_id  
**AND** au2.au\_id = ta2.au\_id  
**AND** au1.au\_id < au2.au\_id;

1. Pour chaque détail de vente, donnez le titre du livre, le nom du magasin, le prix unitaire, le nombre d'exemplaires vendus, le montant total et le montant de l'éco-taxe totale (qui s'élève à 2% du chiffre d'affaire)

**SELECT** sd.stor\_id, sd.ord\_num, sd.title\_id, t.title, st.stor\_name, t.price, sd.qty, t.price\*sd.qty **AS** montantTotal,  
 t.price\*sd.qty\*0.02 **AS** montantEcoTaxe  
**FROM** titles t, stores st, salesdetail sd  
**WHERE** sd.title\_id = t.title\_id  
**AND** sd.stor\_id = st.stor\_id;

***Exercice 2.d.ii (Page 23) :***

/\* 1. Quel est le livre le plus cher publié par l'éditeur "Algodata Infosystems" ?\*/  
SELECT t.title\_id,t.title,t.price  
FROM titles t, publishers pu  
WHERE t.pub\_id = pu.pub\_id  
AND pu.pub\_name = 'Algodata Infosystems'  
AND t.price = (SELECT *max*(t2.price)  
 FROM titles t2  
 WHERE t2.pub\_id = pu.pub\_id);  
  
/\* 2. Quels sont les livres qui ont été vendus dans plusieurs magasins ? \*/  
SELECT DISTINCT t.title\_id, t.title  
FROM titles t, salesdetail sd  
WHERE t.title\_id = sd.title\_id  
AND t.title\_id IN (SELECT sd2.title\_id  
 FROM salesdetail sd2  
 WHERE sd2.stor\_id != sd.stor\_id  
 GROUP BY sd2.title\_id);  
  
/\* OR \*/  
  
SELECT DISTINCT t.title\_id, t.title  
FROM titles t, salesdetail sd, salesdetail sd2  
WHERE t.title\_id = sd.title\_id  
AND t.title\_id = sd2.title\_id;  
  
  
/\* 3. Quels sont les livres dont le prix est supérieur à une fois et demi le prix moyen des livres du  
même type \*/  
SELECT t1.title, t1.title\_id  
 FROM titles t1  
 WHERE t1.price > (SELECT *avg*(t2.price) \* 1.5  
 FROM titles t2  
 WHERE t1.type = t2.type);  
  
/\* 4. Quels sont les auteurs qui ont écrit un livre (au moins), publié par un éditeur localisé dans le  
même état ?\*/  
SELECT a.au\_id, a.au\_fname, a.au\_lname  
FROM authors a  
WHERE a.au\_id IN (SELECT ta.au\_id  
 FROM publishers p,titleauthor ta,titles t  
 WHERE p.pub\_id = t.pub\_id  
 AND t.title\_id = ta.title\_id  
 AND a.state = p.state);  
  
  
/\* 5 - Quels sont les éditeurs qui n'ont rien édité ?\*/  
SELECT p.pub\_id, p.pub\_name  
FROM publishers p  
WHERE NOT *EXISTS*(SELECT t.title\_id  
 FROM titles t);  
  
/\* 6 - Quel est l'éditeur qui a édité le plus grand nombre de livres ?\*/  
SELECT p.pub\_id,p.pub\_name  
FROM publishers p  
WHERE (SELECT *count*(t.title\_id)  
 FROM titles t  
 WHERE t.pub\_id = p.pub\_id) >= ALL (SELECT *count*(t2.title\_id)  
 FROM titles t2  
 GROUP BY t2.pub\_id);  
  
/\* OR \*/  
SELECT p.pub\_id,p.pub\_name  
FROM publishers p, titles t  
WHERE p.pub\_id = t.pub\_id  
GROUP BY p.pub\_id,p.pub\_name  
HAVING *count*(t.title\_id) >= ALL (SELECT *count*(t2.title\_id)  
 FROM titles t2  
 GROUP BY t2.pub\_id);  
  
  
/\* 7 - Quels sont les éditeurs dont on n'a vendu aucun livre ?\*/  
SELECT p.pub\_id, p.pub\_name  
FROM publishers p  
WHERE p.pub\_id NOT IN (SELECT t.pub\_id  
 FROM salesdetail sd,titles t  
 WHERE sd.title\_id = t.title\_id);  
  
/\* 8. Quels sont les différents livres écrits par des auteurs californiens, publiés par des éditeurs  
californiens, et qui n'ont été vendus que dans des magasins californiens ? avec un sub select \*/  
SELECT DISTINCT t.title\_id, t.title  
FROM titles t,publishers p,titleauthor ta, authors a  
WHERE t.pub\_id = p.pub\_id  
AND t.title\_id = ta.title\_id  
AND ta.au\_id = a.au\_id  
AND p.state = 'CA'  
AND a.state = 'CA'  
AND t.title\_id NOT IN (SELECT sd.title\_id  
 FROM salesdetail sd, stores s  
 WHERE sd.stor\_id = s.stor\_id  
 AND s.state != 'CA');  
  
/\* 9. Quel est le titre du livre vendu le plus récemment ? (S'il a des ex-aequo, donnez-les tous.\*/  
SELECT DISTINCT t.title\_id, t.title  
FROM titles t, salesdetail sd, sales s  
WHERE t.title\_id = sd.title\_id  
AND sd.ord\_num = s.ord\_num  
AND sd.stor\_id = s.stor\_id  
AND s.date = (SELECT *max*(s.date) FROM sales s);  
  
/\* OR \*/  
  
SELECT DISTINCT t.title\_id, t.title  
FROM titles t, salesdetail sd LEFT OUTER JOIN sales s on sd.stor\_id = s.stor\_id AND sd.ord\_num = s.ord\_num  
WHERE t.title\_id = sd.title\_id  
AND s.date >= ALL (SELECT *max*(s.date) FROM sales s);  
  
  
/\* 10. Quels sont les magasins où l'on a vendu (au moins) tous les livres vendus par le magasin  
"Bookbeat" ? \*/  
SELECT DISTINCT s.stor\_id, s.stor\_name  
FROM stores s, salesdetail sd  
WHERE s.stor\_id = sd.stor\_id  
AND sd.title\_id IN (SELECT sd2.title\_id  
 FROM salesdetail sd2, stores s2  
 WHERE s2.stor\_name = 'Bookbeat'  
 AND s2.stor\_id = sd2.stor\_id)  
GROUP BY s.stor\_id, s.stor\_name;  
  
  
/\* 11. Quelles sont les villes de Californie où l'on peut trouver un auteur, mais aucun magasin \*/  
SELECT DISTINCT a.city  
FROM authors a  
WHERE a.state ='CA'  
AND a.city IS NOT NULL  
AND a.city NOT IN(SELECT st.city FROM stores st WHERE st.state ='CA');  
  
/\* arrêter ici \*/  
/\* 12. Quels sont les éditeurs localisés dans la ville où il y a le plus d'auteurs ? \*/  
SELECT DISTINCT p.pub\_id, p.pub\_name  
FROM publishers p, authors a  
WHERE p.city = a.city  
AND p.state = a.state  
AND a.city IN (SELECT a2.city  
 FROM authors a2  
 GROUP BY a2.city  
 HAVING *count*(a2.au\_id) >= ALL(SELECT *count*(a3.au\_id)  
 FROM authors a3  
 GROUP BY a3.city));  
  
/\* 13. Donnez les titres des livres dont tous les auteurs sont californiens \*/  
SELECT DISTINCT t.title\_id,t.title  
FROM titles t,titleauthor ta  
WHERE t.title\_id = ta.title\_id  
AND ta.au\_id IN (SELECT a.au\_id  
 FROM authors a  
 WHERE a.state = 'CA');  
  
/\* 14. Quels sont les livres qui n'ont été écrits par aucun auteur californien ? \*/  
SELECT DISTINCT t.title\_id,t.title  
FROM titles t,titleauthor ta  
WHERE t.title\_id = ta.title\_id  
AND ta.au\_id NOT IN (SELECT a.au\_id  
 FROM authors a  
 WHERE a.state = 'CA');  
  
/\* 15. Quels sont les livres qui n'ont été écrits que par un seul auteur ? \*/  
SELECT t.title\_id,t.title  
FROM titles t  
WHERE 1 = (SELECT *count*(ta.title\_id)  
 FROM titleauthor ta  
 WHERE ta.title\_id = t.title\_id);  
  
  
/\* 16. Quels sont les livres qui n'ont qu'un auteur, et tels que cet auteur soit californien ?\*/  
SELECT DISTINCT t.title\_id,t.title  
FROM titles t,titleauthor ta  
WHERE t.title\_id = ta.title\_id  
AND ta.au\_id IN (SELECT ta2.au\_id  
 FROM titleauthor ta2, authors a  
 WHERE ta2.au\_id = a.au\_id  
 AND a.state = 'CA'  
 GROUP BY ta2.au\_id  
 HAVING *count*(ta2.title\_id) = 1);  
  
/\* 2.G.ii \*/  
  
/\* 1 Donnez la liste des magasins, en ordre alphabétique, en mentionnant pour chacun le chiffre  
d'affaire total.\*/  
SELECT s.stor\_id, s.stor\_name, *sum*(sd.qty \* t.price ) as chiffre\_affaire  
FROM stores s LEFT OUTER JOIN salesdetail sd on s.stor\_id = sd.stor\_id LEFT OUTER JOIN titles t on sd.title\_id = t.title\_id  
GROUP BY s.stor\_id, s.stor\_name  
ORDER BY s.stor\_name;  
  
/\* 2 Donnez la liste des magasins, en mentionnant pour chacun le chiffre d'affaire total. Classez  
cette liste par ordre décroissant de chiffre d'affaire \*/  
SELECT s.stor\_id, s.stor\_name, *sum*(sd.qty \* t.price ) as chiffre\_affaire  
FROM stores s LEFT OUTER JOIN salesdetail sd on s.stor\_id = sd.stor\_id LEFT OUTER JOIN titles t on sd.title\_id = t.title\_id  
GROUP BY s.stor\_id, s.stor\_name  
ORDER BY chiffre\_affaire DESC;  
  
/\* 3 Donnez la liste des livres de plus de $20, classés par type, en donnant pour chacun son type,  
son titre, le nom de son éditeur et son prix \*/  
SELECT t.type, t.title,t.price,p.pub\_name  
FROM titles t LEFT OUTER JOIN publishers p on t.pub\_id = p.pub\_id  
WHERE t.price > 20  
ORDER BY t.type;  
  
/\* 4 Donnez la liste des livres de plus de $20, classés par type, en donnant pour chacun son type,  
son titre, les noms de ses auteurs et son prix. \*/  
SELECT t.type,t.title,t.price,a.au\_lname, a.au\_fname  
FROM titles t LEFT OUTER JOIN titleauthor ta on t.title\_id = ta.title\_id LEFT OUTER JOIN authors a on ta.au\_id = a.au\_id  
WHERE t.price > 20  
ORDER BY t.type;  
  
/\* 5 Quelles sont les villes de Californie où l'on peut trouver un auteur et/ou un éditeur, mais  
aucun magasin ? \*/  
WITH californiaStores AS  
(SELECT st.\* FROM stores  
st WHERE st.state = 'CA')  
SELECT a.city  
FROM authors a  
WHERE a.state = 'CA' AND a.city IS NOT NULL  
UNION (SELECT p.city FROM publishers p WHERE p.state = 'CA'AND p.city IS NOT NULL)  
EXCEPT (SELECT st.city FROM californiaStores st WHERE st.state = 'CA'AND st.city IS NOT NULL);  
  
/\* OR \*/  
SELECT a.city  
FROM authors a  
WHERE a.state = 'CA' AND a.city IS NOT NULL  
UNION (SELECT p.city FROM publishers p WHERE p.state = 'CA'AND p.city IS NOT NULL)  
EXCEPT (SELECT st.city FROM stores st WHERE st.state = 'CA'AND st.city IS NOT NULL);  
  
/\* 6. Donnez la liste des auteurs en indiquant pour chacun, outre son nom et son prénom, le  
nombre de livres de plus de 20 $ qu'il a écrits. Classez cela par ordre décroissant de nombre de livres,  
et, en cas d'ex aequo, par ordre alphabétique. N'oubliez pas les auteurs qui n'ont écrit aucun livre de  
plus de 20 $ !\*/  
  
/\* ------ Compliqué ------ \*/  
  
/\* bonne response \*/  
SELECT a.au\_id,a.au\_lname,a.au\_fname,*count*(ta.title\_id)  
FROM authors a LEFT OUTER JOIN titleauthor ta on a.au\_id = ta.au\_id  
LEFT JOIN titles t on t.title\_id = ta.title\_id  
AND t.price>20  
GROUP BY a.au\_id, a.au\_lname, a.au\_fname  
ORDER BY *count*(ta.title\_id) DESC ,a.au\_lname ASC , a.au\_fname;  
  
/\* OR \*/  
  
SELECT a.au\_lname, a.au\_fname, *count*(t.title\_id) as nb\_livres  
FROM authors a LEFT OUTER JOIN titleauthor ta on a.au\_id = ta.au\_id LEFT OUTER JOIN titles t on ta.title\_id = t.title\_id  
WHERE t.price > 20  
GROUP BY a.au\_lname, a.au\_fname  
ORDER BY nb\_livres DESC, a.au\_lname, a.au\_fname;  
  
/\* 2.H.ii \*/  
  
/\* 1. Donnez la liste complète des couples auteur-livre avec, pour chaque couple, le nom et le  
prénom de l'auteur, son adresse complète, ainsi que l'identifiant et le titre du livre. Les auteurs qui  
n'ont rien écrit doivent aussi figurer dans le résultat (avec "aucun livre" comme titre de livre). A  
l'affichage, les tuples relatifs au même livre doivent se suivre. \*/  
SELECT t.title\_id, t.title, au.au\_id, au.au\_lname, au.au\_fname, au.city, au.state, au.country, *COALESCE*(t.title, 'Aucun livre') AS nom\_livre  
FROM titles t LEFT OUTER JOIN titleauthor ta ON t.title\_id = ta.title\_id LEFT OUTER JOIN authors au ON ta.au\_id = au.au\_id  
ORDER BY title\_id;  
  
/\* Interro 2021 \*/  
/\* 1. Listez les auteurs (identifiant, nom, prénom) n’ayant écrit que des  
livres de type business. Il ne faut prendre les auteurs n’ayant pas  
écrit de livres, ni ceux ayant écrit un livre dont on ne connaît pas  
le type. \*/  
SELECT a.au\_id, a.au\_lname, a.au\_fname  
FROM authors a  
WHERE a.au\_id NOT IN (SELECT ta.au\_id  
 FROM titleauthor ta, titles t  
 WHERE ta.title\_id = t.title\_id  
 AND t.type != 'business');  
  
/\* 2. Donnez, pour chaque auteur, son identifiant, son nom, son prénom et le  
nombre de livres de type psychology ou de type sociology qu’il a écrits.  
Veillez bien à ce que tous les auteurs de la DB apparaissent. \*/  
SELECT a.au\_id, a.au\_lname, a.au\_fname, *count*(t.title\_id)  
FROM authors a LEFT OUTER JOIN titleauthor ta on a.au\_id = ta.au\_id LEFT OUTER JOIN titles t on ta.title\_id = t.title\_id  
WHERE t.type = 'psychology' OR t.type = 'sociology'  
GROUP BY a.au\_id, a.au\_lname, a.au\_fname;  
  
/\* 3. Donnez toutes les paires d’auteurs (identifiant + nom de chaque auteur)  
ayant été coauteurs et qui habitent dans le même état mais pas dans la même  
ville. \*/  
SELECT DISTINCT a1.au\_id, a1.au\_lname,a2.au\_id, a2.au\_lname  
FROM authors a1, authors a2, titleauthor ta1, titleauthor ta2  
WHERE a1.au\_id = ta1.au\_id;  
  
/\* 4. Donnez la liste des auteurs (identifiant, nom et prénom) dont on a vendu  
au moins 10000 exemplaires de livre en septembre 2021.\*/  
SELECT a.au\_id, a.au\_lname, a.au\_fname  
FROM authors a, titleauthor ta, titles t, salesdetail sd, sales s  
WHERE a.au\_id = ta.au\_id  
AND ta.title\_id = t.title\_id  
AND t.title\_id = sd.title\_id  
AND sd.ord\_num = s.ord\_num  
AND sd.stor\_id = s.stor\_id  
AND s.date >= '2021-09-01'  
AND s.date <= '2021-09-30'  
GROUP BY a.au\_id, a.au\_lname, a.au\_fname  
HAVING *SUM*(sd.qty) >= 10000;  
  
/\* 5. Donnez le(s) livre(s) (identifiant, titre et type) publié(s) le plus  
récemment parmi ceux écrits par Anne Ringer (Remarque : on suppose qu’il  
n’y a qu’un auteur nommé Anne Ringer)\*/  
SELECT t.title\_id, t.title, t.type  
FROM titles t LEFT OUTER JOIN titleauthor ta on t.title\_id = ta.title\_id LEFT OUTER JOIN authors a on ta.au\_id = a.au\_id  
WHERE a.au\_fname = 'Anne'  
AND a.au\_lname = 'Ringer'  
AND t.pubdate = (SELECT *max*(t2.pubdate)  
 FROM titles t2 LEFT OUTER JOIN titleauthor ta2 on t2.title\_id = ta2.title\_id LEFT OUTER JOIN authors a2 on ta2.au\_id = a2.au\_id  
 WHERE a2.au\_fname = 'Anne'  
 AND a2.au\_lname = 'Ringer');  
  
/\* 6. Pour chaque éditeur californien (CA en Db), donnez son identifiant, son  
nom et le nombre d’auteurs dont ils ont publié au moins un livre. Classezles par ordre décroissant du nombre d’auteurs dont ils ont publié au moins  
un livre. En cas d’ex aequo, classez par ordre d’identifiant. Veillez bien  
à ce que tous les éditeurs californiens de la DB apparaissent. \*/  
SELECT p.pub\_id, p.pub\_name, *count*(ta.au\_id)  
FROM publishers p LEFT OUTER JOIN titles t on p.pub\_id = t.pub\_id LEFT OUTER JOIN titleauthor ta on t.title\_id = ta.title\_id LEFT OUTER JOIN authors a on ta.au\_id = a.au\_id  
WHERE p.state = 'CA'  
GROUP BY p.pub\_id, p.pub\_name  
ORDER BY *count*(ta.au\_id) DESC, p.pub\_id;  
  
/\* 7. Donnez la liste des auteurs (identifiant, nom, prénom) ayant écrit au  
moins un livre seul (sans coauteur).\*/  
SELECT a.au\_id, a.au\_lname, a.au\_fname  
FROM authors a LEFT OUTER JOIN titleauthor ta on a.au\_id = ta.au\_id LEFT OUTER JOIN titles t on ta.title\_id = t.title\_id  
WHERE t.title\_id NOT IN (SELECT ta2.title\_id  
 FROM titleauthor ta2  
 WHERE ta2.au\_id != a.au\_id);  
  
/\* 8. Donnez tous les auteurs (identifiant, nom et prénom) ayant écrit au  
moins un livre publié en Californie (CA en Db) en 2020. Triez-les par nom  
et, en cas de même nom, par prénom. \*/  
SELECT a.au\_id, a.au\_lname, a.au\_fname  
FROM authors a LEFT OUTER JOIN titleauthor ta on a.au\_id = ta.au\_id LEFT OUTER JOIN titles t on ta.title\_id = t.title\_id LEFT OUTER JOIN publishers p on t.pub\_id = p.pub\_id  
WHERE p.state = 'CA'  
AND t.pubdate >= '2020-01-01'  
AND t.pubdate <= '2020-12-31'  
GROUP BY a.au\_id, a.au\_lname, a.au\_fname  
ORDER BY a.au\_lname, a.au\_fname;  
  
/\* Partie 3 \*/  
/\* 3.a.ii \*/  
/\* 1. Ecrivez la commande permettant de créer la table salesdetail \*/  
CREATE TABLE salesdetail(  
 qty INTEGER NOT NULL,  
 stor\_id INTEGER NOT NULL,  
 ord\_num VARCHAR(20) NOT NULL,  
 title\_id INTEGER NOT NULL,  
 FOREIGN KEY (stor\_id) REFERENCES stores(stor\_id),  
 FOREIGN KEY (ord\_num) REFERENCES sales(ord\_num),  
 FOREIGN KEY (title\_id) REFERENCES titles(title\_id),  
 PRIMARY KEY (qty)  
);  
  
/\* 2.Ecrivez la commande permettant de créer la table authors \*/  
CREATE TABLE authors(  
 au\_id INTEGER NOT NULL,  
 au\_lname VARCHAR(40) NOT NULL,  
 au\_fname VARCHAR(20) NOT NULL,  
 phone VARCHAR(12) NOT NULL,  
 address VARCHAR(40) NOT NULL,  
 city VARCHAR(20) NOT NULL,  
 state VARCHAR(5) NOT NULL,  
 country VARCHAR(20) NOT NULL,  
 PRIMARY KEY (au\_id)  
);  
  
/\* 3.Ecrivez la commande permettant de créer la table stores \*/  
CREATE TABLE stores(  
 stor\_id INTEGER NOT NULL,  
 stor\_name VARCHAR(40) NOT NULL,  
 stor\_address VARCHAR(40) NOT NULL,  
 city VARCHAR(20) NOT NULL,  
 state VARCHAR(5) NOT NULL,  
 country VARCHAR(20) NOT NULL,  
 PRIMARY KEY (stor\_id)  
);  
/\* 3.a.iv \*/  
/\* 1.Ajouter un ISBN à la table titles, de type varchar(15). \*/  
ALTER TABLE titles  
ADD isbn VARCHAR(15);  
  
/\* 2.Modifier le champ ISBN pour qu’il soit unique. \*/  
ALTER TABLE titles  
ADD CONSTRAINT isbn\_unique UNIQUE (isbn);  
  
/\* 3. Finalement supprimer ce champ. \*/  
ALTER TABLE titles  
DROP COLUMN isbn;  
  
/\* interro 2022 \*/  
/\* 1. Donnez la liste des auteurs (identifiant, nom et prénom) ayant écrit au  
moins 10 livres entre le 1/1/2010 et le 31/12/2019.\*/  
SELECT au.au\_id,au.au\_fname,au.au\_lname  
FROM authors au, titleauthor ta, titles t  
WHERE au.au\_id = ta.au\_id AND ta.title\_id = t.title\_id  
AND t.pubdate BETWEEN '2010/01/01' AND '2019/12/31'  
GROUP BY au.au\_id  
HAVING *COUNT*(t.title\_id) >= 10;  
  
/\*3. Donnez toutes les paires de livres ayant le même titre et éditeur et au  
moins un auteur en commun. Pour chaque paire, affichez les identifiants et  
le titre\*/  
SELECT DISTINCT t1.title\_id as "id livre 1", t2.title\_id as "id livre 2",  
t1.title  
FROM titles t1, titles t2, titleauthor ta1, titleauthor ta2  
WHERE t1.title\_id =ta1.title\_id AND t2.title\_id = ta2.title\_id  
AND ta1.au\_id = ta2.au\_id AND t1.pub\_id = t2.pub\_id  
AND t1.title = t2.title AND t1.title\_id < t2.title\_id;  
  
/\* 4. Listez les magasin (identifiant, nom et pays) qui n’ont vendu que des  
livres de type 'psychology'. Il ne faut pas prendre les magasins n’ayant  
rien vendu, ni les magasins ayant vendu des livres dont on ne connaît pas  
le type.\*/  
SELECT DISTINCT st.stor\_id,st.stor\_name,st.country  
FROM sales s, stores st  
WHERE s.stor\_id = st.stor\_id  
AND s.stor\_id NOT IN(SELECT sd.stor\_id  
 FROM titles t,salesdetail sd  
 WHERE t.title\_id = sd.title\_id  
 AND t.type = 'psychology'  
 AND t.type IS NOT NULL);