

INF3180 : Fichiers et bases de données

Solutions aux exercices

Représentation UML du schéma *VentesPleinDeFoin* :

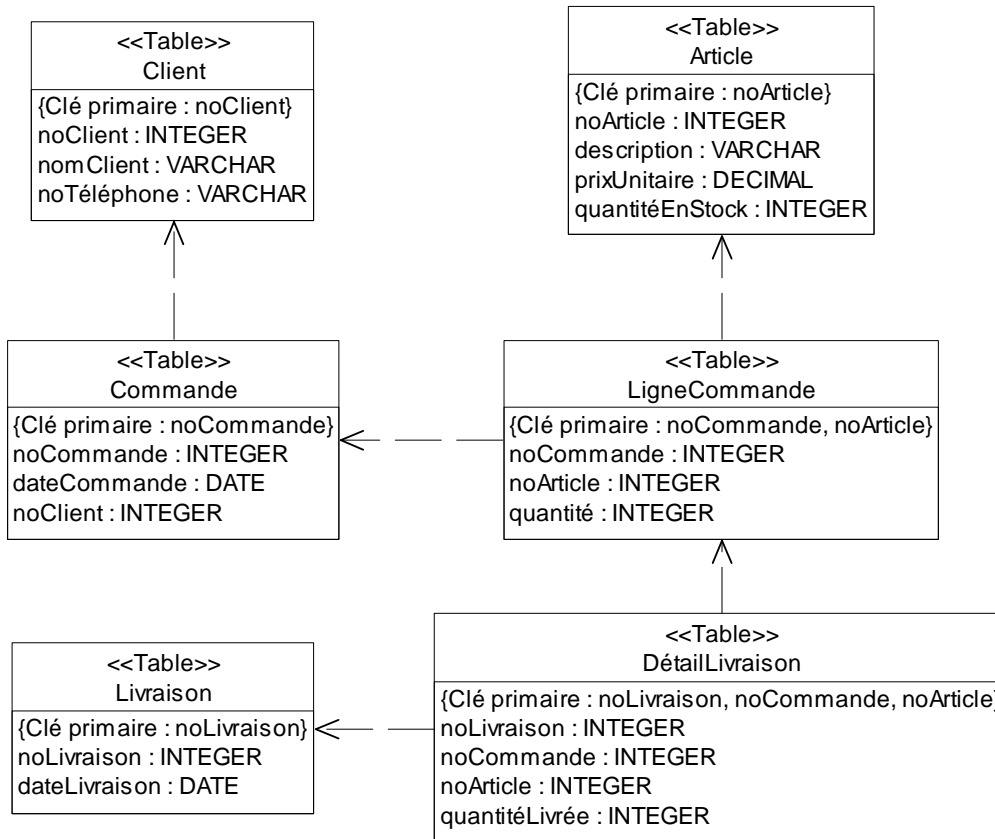


Schéma SQL *VentesPleinDeFoin* :

```
CREATE TABLE Client
(noClient      INTEGER          NOT NULL,
 nomClient     VARCHAR(20)      NOT NULL,
 noTéléphone   VARCHAR(15)      NOT NULL,
 PRIMARY KEY   (noClient)
)
CREATE TABLE Article
(noArticle     INTEGER          NOT NULL,
 description    VARCHAR(20),
 prixUnitaire  DECIMAL(10,2)    NOT NULL,
 quantitéEnStock INTEGER        DEFAULT 0 NOT NULL
    CHECK (quantitéEnStock >= 0),
 PRIMARY KEY   (noArticle))
CREATE TABLE Commande
(noCommande    INTEGER          NOT NULL,
 dateCommande  DATE             NOT NULL,
 noClient      INTEGER          NOT NULL,
 PRIMARY KEY   (noCommande),
 FOREIGN KEY   (noClient) REFERENCES Client
)
CREATE TABLE LigneCommande
(noCommande    INTEGER          NOT NULL,
 noArticle     INTEGER          NOT NULL,
 quantité      INTEGER          NOT NULL
    CHECK (quantité > 0),
 PRIMARY KEY   (noCommande, noArticle),
 FOREIGN KEY   (noCommande) REFERENCES Commande,
 FOREIGN KEY   (noArticle) REFERENCES Article
)
CREATE TABLE Livraison
(noLivraison   INTEGER          NOT NULL,
 dateLivraison DATE             NOT NULL,
 PRIMARY KEY   (noLivraison)
)
CREATE TABLE DétaillLivraison
(noLivraison   INTEGER          NOT NULL,
 noCommande    INTEGER          NOT NULL,
 noArticle     INTEGER          NOT NULL,
 quantitéLivrée INTEGER          NOT NULL
    CHECK (quantitéLivrée > 0),
 PRIMARY KEY   (noLivraison, noCommande, noArticle),
 FOREIGN KEY   (noLivraison) REFERENCES Livraison,
 FOREIGN KEY   (noCommande, noArticle) REFERENCES LigneCommande
)
```

1) Formulez en SQL les requêtes suivantes sur le schéma de la BD de la pépinière *PleinDeFoin* :

a) Les *Clients* dont le *noTéléphone* = (999)999-9999

```
SELECT      *
FROM        Client
WHERE       noTéléphone = '(999)999-9999'
```

b) Le *noCommande* et la *dateCommande* des *Commandes* du *Client* #10 dont le *noCommande* est supérieur à 5.

```
SELECT      noCommande, dateCommande
FROM        Commande
WHERE       noClient = 10 AND noCommande > 5
```

c) Les *noArticle* et *description* des *Articles* dont le *prixUnitaire* est entre \$10 et \$20.

```
SELECT      noArticle, description
FROM        Article
WHERE       prixUnitaire BETWEEN 10 AND 20
```

```
SELECT      noArticle, description
FROM        Article
WHERE       prixUnitaire >= 10 AND prixUnitaire <= 20
```

d) Le *noClient*, *noTéléphone* du *Client* et *noCommande* pour les *Commandes* faites le 4/06/2000.

```
SELECT      Client.noClient, noTéléphone, noCommande
FROM        Client, Commande
WHERE       Client.noClient = Commande.noClient AND
           DateCommande = '4/06/2000'
```

e) Les *noArticles* commandés au moins une fois par le *Client* #10 après le 01/06/2000.

```
SELECT      DISTINCT noArticle
FROM        Commande, LigneCommande
WHERE       Commande.noCommande = LigneCommande.noCommande AND
           noClient = 10 AND
           DateCommande > '1/06/2000'
```

Solution avec SELECT imbriqué :

```

SELECT    DISTINCT noArticle
FROM      LigneCommande
WHERE     noCommande IN
          (SELECT    noCommande
           FROM      Commande
           WHERE     noClient = 10 AND
                    DateCommande > '1/06/2000' )

```

f) Les *noLivraisons* correspondant aux *Commandes* faites par le *Client* #10.

```

SELECT    DISTINCT noLivraison
FROM      Commande C, DétailLivraison D
WHERE     C.noCommande = D.noCommande AND
          noClient = 10

```

Solution avec SELECT imbriqué :

```

SELECT    DISTINCT noLivraison
FROM      DétailLivraison
WHERE     noCommande IN
          (SELECT    noCommande
           FROM      Commande
           WHERE     noClient = 10)

```

g) Les *noCommandes* des *Commandes* qui ont été placées à la même date que la *Commande* #2.

```

SELECT    Commande.noCommande
FROM      Commande, Commande C2
WHERE     Commande.dateCommande = C2.dateCommande AND
          C2.noCommande = 2

```

h) Les *noLivraison* faites à la même date qu'une des *Commandes* correspondant à la *Livraison*.

```

SELECT    DISTINCT V.noLivraison
FROM      Commande C, DétailLivraison D, Livraison V
WHERE     C.noCommande = D.noCommande AND
          D.noLivraison = V.noLivraison AND
          C.dateCommande = V.dateLivraison

```

i) La liste des *noCommande* avec les *noLivraisons* associées incluant les *noCommandes* sans livraison.

```

SELECT    DISTINCT noCommande, noLivraison
FROM      Commande NATURAL LEFT OUTER JOIN DétailLivraison

```

Solution avec le dialecte Oracle

```

SELECT    DISTINCT C.noCommande, D.noLivraison

```

```
FROM      Commande C,DétailLivraison D
WHERE     C.noCommande = D.noCommande (+)
```

j) Les *noClient*, *nomClient* des *Clients* qui n'ont pas placé de *Commande* au mois de mars de l'année 2000.

```
SELECT      noClient, nomClient
FROM        Client
WHERE       NOT EXISTS
            (SELECT      *
             FROM        Commande
             WHERE       noClient = Client.noClient AND
                        dateCommande BETWEEN '01/03/2000' AND '31/03/2000')
```

Solution avec MINUS (N.B. Oracle utilise MINUS plutôt que EXCEPT)

```
(SELECT      noClient, nomClient
FROM        Client)
MINUS
(SELECT      noClient, nomClient
FROM        Client, Commande
WHERE       Commande.noClient = Client.noClient AND
           dateCommande BETWEEN '01/03/2000' AND '31/03/2000')
```

k) Les *noCommandes* qui ne contiennent pas l'*Article* # 10.

```
SELECT      noCommande
FROM        Commande
WHERE       NOT EXISTS
            (SELECT      *
             FROM        LigneCommande
             WHERE       Commande.noCommande = noCommande AND
                        noArticle =10)
```

1) Les *noArticle* qui apparaissent dans toutes les *Commandes*.

[illegible]

ou :

```
SELECT      noArticle
FROM        Article
WHERE       NOT EXISTS
            ((SELECT      noCommande
               FROM        Commande
              )
             EXCEPT
            (SELECT      noCommande
               FROM        LigneCommande
               WHERE       noArticle = Article.noArticle
              )
            )
```

(N.B. Oracle utilise MINUS plutôt que EXCEPT)

ou en utilisant l'équivalence $T_1(X) \div T_2(Y) = \pi_{XY}(T_1) - \pi_{XY}((\pi_{XY}(T_1) \times T_2) - T_1)$:

```
(SELECT DISTINCT noArticle
FROM LigneCommande)
EXCEPT
(SELECT DISTINCT noArticle FROM
 ((SELECT DISTINCT LigneCommande.noArticle, Commande.noCommande
    FROM LigneCommande, Commande)
    EXCEPT
 (SELECT DISTINCT noArticle, noCommande
    FROM LigneCommande)))
```

m) Les *noArticles* qui apparaissent dans toutes les *Commandes* du *Client* #10.

```
SELECT      noArticle
FROM        Article
WHERE       NOT EXISTS
            (SELECT      noCommande
               FROM        Commande
               WHERE       noClient = 10 AND NOT EXISTS
                        (SELECT      *
                           FROM        LigneCommande
                           WHERE       noArticle = Article.noArticle AND
                                       noCommande = Commande.noCommande))
```

ou :

```
SELECT      noArticle
FROM        Article
WHERE       NOT EXISTS
            ((SELECT      noCommande
               FROM        Commande
```

```

WHERE      noClient = 10
)
EXCEPT
(SELECT     C.noCommande
FROM        Commande C, LigneCommande L
WHERE       noArticle = Article.noArticle AND
            C.noCommande = L.noCommande AND
            noClient = 10))

```

(N.B. Oracle utilise MINUS plutôt que EXCEPT)

- n) Les *Articles* dont la *description* débute par la lettre « C ».

```

SELECT      *
FROM        Article
WHERE       description LIKE 'C%'

```

- o) Le *Clients* dont le *noTéléphone* n'est pas NULL.

```

SELECT      *
FROM        Client
WHERE       noTéléphone IS NOT NULL

```

- p) Les *Articles* dont le prix est supérieur à la moyenne.

```

SELECT      *
FROM        Article
WHERE       prixUnitaire >
            (SELECT AVG(prixUnitaire)
             FROM Article)

```

- q) Le montant total de la *Commande* #1 avant et après la taxe de 15%.

```

SELECT      SUM(quantité*prixUnitaire)AS totalCommande,
            SUM(quantité*prixUnitaire*1.15)AS totalPlusTaxe
FROM        LigneCommande L, Article A
WHERE       L.noArticle = A.noArticle AND
            noCommande = 1

```

- r) Le montant total de la *Livraison* #1 avant et après la taxe de 15%

```

SELECT      SUM(quantitéLivrée*prixUnitaire)AS totalLivraison,
            SUM(quantitéLivrée*prixUnitaire*1.15)AS
totalPlusTaxe

```

```
FROM      DétailLivraison D, Article A
WHERE     D.noArticle = A.noArticle AND
          noLivraison = 1
```