

[ACCEUIL](#)[ALGORITHME](#)[EXAMENS](#)[PROGRAMMATION »](#)[WEB »](#)[BASES DE DONNÉES »](#)[BUREAUTIQUE »](#)[CONCEPTION »](#)[RÉSEAUX](#)

## SOCIAL PROFILES

[Home](#) » [SQL](#) » Série Exercices SQL

## Série Exercices SQL

0

By [X-Proof](#) 06:40

### Série Exercices SQL



#### Exercice 1

Faire les requêtes [SQL](#) permettant de répondre aux demandes sur la base de données suivante:

**Servir** (Café, Boisson)

**Fréquenter** (Client, Café)

**Apprécier** (Client, Boisson)

Exprimer les requêtes suivantes en [SQL](#).

Les Cafés qui servent une Boisson Apprécier par 'Ahmed'.

Les Clients qui vont dans les mêmes Cafés que Ahmed.

Les Clients qui fréquentent au moins un Café où l'on sert une Boisson qu'ils aiment.

Les Clients qui ne fréquentent aucun Café où l'ont sert une Boisson qu'ils aiment.

Les Clients qui fréquentent tous les Cafés.

Les Clients qui fréquentent tous les Cafés qui servent au moins une Boisson qu'ils aiment.

Les Clients qui ne fréquentent que les Cafés qui servent une Boisson qu'ils aiment.

Donner pour chaque Client, le nombre de Cafés servant une Boisson qu'ils aiment.

Les Clients qui fréquentent au moins 2 Cafés où l'on sert une Boisson qu'ils aiment.

## Exercice 2

### Cliniques

**A - Modèle relationnel sous-jacents à ce texte.**

Implantation (ville, dpt, nbhab)

Clinique (noclinique, nom, nblits, ville)

Spécialité (nospécialité, libellé)

Service (nospécialité, noclinique)

**B - Donnez les instructions SQL de création des différentes tables du modèle.**

**C - Instruction SQL permettant l'insertion dans la base de la première**

**D - Donner la requête SQL et le résultat de chacune des requêtes suivantes :**

Noms des cliniques ayant une capacité inférieure à 500.

Numéros de clinique ayant la spécialité 100.

Noms des cliniques ayant la spécialité "chirurgie cardiaque".

Noms des cliniques des villes de plus de 500 000 habitants du département du Rhône, qui ont plus de 100 lits.

Pour chaque ville, le nom de la clinique qui a le plus grand nombre de lits.

Noms des spécialités de la clinique Clairval à Marseille

Numéros des cliniques qui disposent des spécialités 100 et/ou 600.

Numéros des cliniques qui disposent à la fois des spécialités 100, 600 et 700.

## Exercice 3

Sur la base de données exemple du cours (CLIENT-COMMANDE-PRODUIT-FOURNISSEUR), formuler avec le langage SQL\*Plus les requêtes suivantes.

[Popular](#)

[Tags](#)

[Blog Archives](#)

## Série 2 Exercices UML



Série 2 Exercices UML Exercice 1 :  
DAB Le système à étudier est celui  
d'un distributeur de billets (DAB). •  
Un client peut effe...

$$3) = \overline{A \cdot B}$$

A

$$A + \overline{B} + \overline{A \cdot C} =$$

### Exercices Algèbre de boole

Exercice 1 – Algèbre de Boole

Exercice 2 – Analyse et synthèse de  
circuits L'implémentation d'une  
fonction logique Z relativement ...



### Série 2 Exercices Langage C

Série 2 Exercices Langage C

Exercice1 On appelle palindrome  
une suite de caractères qui se lit de  
la même façon dans les deux sen...

### Exercices algèbre de boole et circuits logiques



Sachant que A, B, C et D sont des  
variables booléennes. a) En  
utilisant exclusivement l'algèbre

booléenne, démontrez que : b) En u...



### Exercice Corrigé base de données relationnel SGBD Access

Énoncé 1 Un fanatique de cinéma  
et de festivals de films décide de  
réaliser sa base de données personnelle en y  
stockant les diffé...

Fourni par Blogger.

**FOLLOW BY EMAIL**

- 1) Désignation et prix unitaire de tous les produits.
- 2) Désignation des produits de prix inférieur à 100 F.
- 3) Nom des clients qui ont commandé le produit n° 1.
- 4) Nom des clients qui ont commandé au moins un produit de prix supérieur à 500 F.
- 5) Nom des clients qui n'ont pas commandé le produit n° 1.
- 6) Numéro des clients qui ont commandé tous les produits.
- 7) Numéro des clients qui ont commandé tous les produits commandés par le client n° 2.

### Exercice 4

Soit le schéma relationnel de la base FABRICATION.

CLIENT (NOC, NOM, ADRESSE)

SERVICE (NOS, INTITULE, LOCALISATION)

PIECE (NOP, DESIGNATION, COULEUR, POIDS) clés primaires

COMMANDE (*NOP, NOS, NOC*, QUANTITE) *clés étrangères*

Formuler en SQL\*Plus les commandes de création de la structure de cette base, puis exprimer les requêtes  
suivantes.

- 1) Donner pour chaque service le poids de la pièce commandée de couleur bleue la plus pesante.
- 2) Donner le poids moyen des pièces commandées pour chacun des services "Promotion".
- 3) Donner les pièces de couleur bleue qui sont commandées par plus de trois services différents.
- 4) Donner le maximum parmi les totaux des quantités des pièces commandées par les différents services.

### Exercice 5

Soit le schéma relationnel de la base de données « pilotes-avions-vols ».

PILOTE (PLNUM, PLNOM, PLPRENOM, VILLE, SALAIRE)

AVION (AVNUM, AVNOM, CAPACITE, LOCALISATION)

VOL (VOLNUM, *PLNUM, AVNUM*, VILLEDEP, VILLEARR, HEUREDEP, HEUREARR)

Exprimer les requêtes suivantes en SQL\*Plus.

- 1) Liste de tous les vols.
- 2) Nom, prénom et ville de tous les pilotes, par ordre alphabétique.
- 3) Nom, prénom et salaire des pilotes dont le salaire est supérieur à 20 000 F.

Email address...

Submit

- 4) Numéro et nom des avions localisés à Paris.
- 5) Caractéristiques (AVNUM, AVNOM, CAPACITE, LOCALISATION) des avions localisés dans la même ville que le pilote Tanguy.
- 6) Caractéristiques (VOLNUM, VILLEDEP, VILLEARR, HEUREDEP, HEUREARR, AVNOM, PLNOM) du vol numéro 714.
- 7) Nom, prénom et numéro de vol des pilotes affectés à un vol.
- 8) Numéro et nom des avions affectés à des vols.
- 9) Nombre total de vols.
- 10) Somme des capacités par type (nom) d'avion.
- 11) Moyenne des durées des voyage

## Correction

<p><b>Exercice 1</b></p> <p>1- Les Cafés qui servent une Boisson Apprécié par 'Ahmed'.</p> <pre>SELECT Café FROM Apprécier A, Servir S WHERE S.Boisson = A.Boisson AND Client ='Ahmed';</pre> <p>2- Les Clients qui vont dans les mêmes Cafés que Ahmed.</p> <pre>SELECT Fréquenter.Client FROM Fréquenter F1, Fréquenter F2 WHERE F1.Café = F2.Café AND F2.Client = 'Ahmed';</pre>	<p><b>Exercice2</b></p> <p><b>Create Table</b> Implantation</p> <pre>(     ville <b>Char</b> (30) <b>Primary Key</b>     département <b>Char</b> (30),     nbhab <b>Number</b> );</pre> <p><b>Create Table</b> Clinique</p> <pre>(     noclinique <b>Number Primary Key</b>     nom <b>Char</b> (40) <b>Not Null</b>,     ville <b>Char</b> (30) <b>Foreign</b> );</pre> <p><b>Key References</b> Implantation (ville), nblits <b>Number</b></p> <p><b>Create Table</b> Specialite</p> <pre>(</pre>	<p><b>Exercice3</b></p> <pre>1) SELECT Desi, PrixUni FROM Client ;</pre> <pre>2) SELECT Desi FROM Client WHERE PrixUni &lt; 100 ;</pre> <pre>3) SELECT DISTINCT Nom FROM Client C1, Commande C2 WHERE C1.NumCli = C2.NumCli AND NumProd = 1 ;</pre>
---	---	---

3- Les Clients qui fréquentent au moins un Café où l'on sert une Boisson qu'ils aiment.

```
SELECT Fréquenter.Client
FROM Fréquenter F,
Apprécier A, Servir S
WHERE A.Boisson =
S.Boisson AND S.Café =
F.Café AND A.Client =
F.Client ;
```

4- Les Clients qui ne fréquentent aucun Café où l'ont sert une Boisson qu'ils aiment.

```
SELECT Client
FROM Fréquenter F
WHERE Client NOT IN
(SELECT
Client
FROM
Apprécier A, Servir S
WHERE
S.Café = F.Café
AND
S.Boisson = A.Boisson );
```

5- Les Clients qui fréquentent tous les Cafés.

```
SELECT Client
FROM Fréquenter
WHERE Client NOT IN
(SELECT
```

nospecialite **Number Primary Key**

nom **Char (20) Not Null**

);

**Create Table** Service

(

noclinique **Number foreign**

**key References** Clinique (noclinique),

nospecialite **Number foreign**

**key References** Specialite (nospecialite)

**constraint** cle **primary key** (noclinique, nospecialite)

);

**occurrence de chaque table.**

**Insert Into** Implantation Values ('Marseille', 'Bouches du Rhône', 150 000);

**Insert Into** Clinique Values (10, 'Clairval', 'Marseille', 70);

**Insert Into** Specialite Values (100, 'Chirurgie cardiaque');

**Insert Into** Service Values (10, 400);

**Select** nom **From** Clinique **Where** nblits < 500;

nom
Clairval
Notre-Dame
Tonkin
Charpennes
Point du jour
Lapulus petite

4) SELECT DISTINCT

Nom

FROM Client C1,

Commande C2,

Produit P

WHERE C1.NumCli =

C2.NumCli

AND C2.NumProd =

P.NumProd

AND PrixUni > 500 ;

5) SELECT NumCli

FROM Client C1

WHERE NOT EXISTS (

SELECT \*

FROM Commande C2

WHERE C2.NumCli =

C1.NumCli

AND NumProd = 1) ;

6) SELECT NumCli

FROM Client C1

WHERE NOT EXISTS (

SELECT \*FROM

Produit P

WHERE NOT EXISTS (

SELECT \*

FROM Commande C2

WHERE C2.NumCli =

F1.Client FROM Fréquenter  
 F1, Fréquenter F2  
 WHERE  
 F1.Client NOT IN  
 (SELECT Client  
 FROM Fréquenter  
 WHERE  
 Café=F2.Café)  
 );

6- Les Clients qui  
 fréquentent tous les Cafés  
 qui servent au moins une  
 Boisson qu'ils aiment.

SELECT Client  
 FROM Fréquenter F  
 WHERE Client NOT IN  
 (SELECT  
 Apprécier.Client  
 FROM  
 Servir S, Apprécier A  
 WHERE  
 S.Boisson=A.Boisson  
 AND  
 A.Client NOT IN  
 (SELECT Client  
 FROM Fréquenter

**Select** noclinique **From** service **Where** nospecialite=100;

noclinique
20
30
50
60
70
80
90

c)

**Select** Clinique.nom  
**From** Clinique, Specialite, Service  
**Where** Clinique.noclinique = Service.noclinique  
**And** Service.nospecialite = Specialite.nospecialite  
**And** specialite.nom='Chirurgie cardiaque';

Une autre formulation :

**Select** nom **From** Clinique **Where** noclinique **In**

(**Select** noclinique **From** Service **Where** nospecialite **In**  
 (**Select** nospecialite **From** Specialite  
**Where** nom='Chirurgie cardiaque'));

Clinique.nom
Notre-Dame
Tonkin
Point du Jour

C1.NumCli  
 AND C2.NumProd =  
 P.NumProd)) ;  
 7) SELECT Nom  
 FROM Client C0  
 WHERE NOT EXISTS (  
 SELECT \*  
 FROM Commande C1  
 WHERE NumCli = 2  
 AND NOT EXISTS (  
 SELECT \*  
 FROM Commande C2  
 WHERE C2.NumCli =  
 C0.NumCli  
 AND C2.NumProd =  
 C1.NumProd)) ;

WHERE Café=S.Café)	Charmettes
	Massues
7- Les Clients qui ne fréquentent que les Cafés qui servent une Boisson qu'ils aiment	La plus grande
	La moyenne
SELECT Client FROM Fréquenter WHERE Client NOT IN (SELECT Client FROM Fréquenter F WHERE Client NOT IN  (SELECT Client  FROM Servir S, Apprécier A  WHERE S.Café = F.Café  And S.Boisson = A.Boisson) );	d) <b>Select</b> nom <b>From</b> Clinique, Implantation <b>Where</b> nblits >100 <b>And</b> Clinique.ville=Implantation.ville <b>And</b> departement='Rhône' <b>And</b> nbhab>500 000;  Une autre formulation : <b>Select</b> nom <b>From</b> Clinique <b>Where</b> nblits >100 <b>And</b> Ville <b>In</b> ( <b>Select</b> ville <b>From</b> Implantation <b>Where</b> departement = 'Rhône' <b>And</b> nbhab >500 000);
	nom
	Charmettes
	Massues
8- Donner pour chaque Client, le nombre de Cafés servant une Boisson qu'ils aiment.	e)  <b>Select</b> nom <b>From</b> Clinique <b>Where</b> (ville, nblits) <b>In</b> ( <b>Select</b> ville, MAX(nblits) <b>From</b> Clinique <b>Group</b> <b>By</b> ville);

```
SELECT Client,
COUNT(DISTINCT Café)
FROM Servir S, Apprécier A
WHERE S.Boisson =
A.Boisson
GROUP BY Client
```

9- Les Clients qui  
fréquentent au moins 2  
Cafés où l'on sert une  
Boisson qu'ils aiment.

```
SELECT Client
FROM Servir S, Apprécier A,
Fréquenter F
WHERE S.Boisson =
A.Boisson AND A.Client =
F.Client AND F.Café =
S.Café
GROUP BY Client
HAVING COUNT(Café)>2;
```

nom
Charmettes
Massues
Notre-Dame
La plus grande
Charpennes

f)

```
Select Specialite.nom
From Specialite, Service, Clinique
Where Specialite.nospecialite = Service.nospecialite
And Service.noclinique=Clinique.noclinique
And Clinique.nom='Clairval'
And Clinique.ville = 'Marseille';
```

Autre formulation :

```
Select nom From Specialite Where nospecialite In
(Select nospecialite From service Where noclinique In
(Select noclinique From clinique
Where nom='Clairval' And ville='Marseille'));
```

nom
obstétrique
urgence

g)

```
Select Distinct noclinique From Service
```



**Where** Service.nospecialite = 100

**Or** Service.nospecialite=600;

noclinique
------------

20
----

30
----

50
----

60
----

70
----

80
----

90
----

h)

**Select** noclinique **From** Service

**Where** Service.nospecialite = 100

**Intersect**

**Select** noclinique **From** Service

**Where** Service.nospecialite = 600

**Intersect**

**Select** noclinique **From** Service

**Where** Service.nospecialite = 700;

noclinique
------------

30
----

80
----

90
----

<p><b>Exercice4</b></p> <pre> CREATE TABLE CLIENT (NOC NUMBER(3), NOM VARCHAR(40), ADRESSE VARCHAR(100), CONSTRAINT PRICLI PRIMARY KEY (NOC)); CREATE TABLE SERVICE (NOS NUMBER(3), INTITULE VARCHAR(30), LOCALISATION VARCHAR(100), CONSTRAINT PRISER PRIMARY KEY (NOS)); CREATE TABLE PIECE (NOP NUMBER(3), DESIGNATION VARCHAR(30), COULEUR VARCHAR(20), POIDS NUMBER(5,2), CONSTRAINT PRIPIE PRIMARY KEY (NOP)); CREATE TABLE COMMANDE (NOP NUMBER(3), NOS NUMBER(3), NOC NUMBER(3), QUANTITE NUMBER(3), CONSTRAINT PRICOM </pre>	<p><b>Exercice5</b></p> <pre> 1) select * from vol; 2) select plnom, plprenom, ville from pilote order by plnom, plprenom; 3) select plnom, plprenom, salaire from pilote where salaire&gt;20000; 4) select avnum, avnom from avion where localisation='Paris'; 5) select avnum, avnom, capacite, localisation from avion a, pilote p where a.localisation=p.ville and plnom='Tanguy'; 6) select volnum, villedep, villearr, heureddep, heurearr, avnom, plnom from avion a, pilote p, vol v where p.plnum=v.plnum and a.avnum=v.avnum and volnum=714; 7) select plnom, plprenom, avnum from pilote p, vol v </pre>	
---	---	--

```
PRIMARY KEY (NOP, NOS,  
NOC),  
CONSTRAINT ETRPIE  
FOREIGN KEY (NOP)  
REFERENCES PIECE(NOP),  
CONSTRAINT ETRSER  
FOREIGN KEY (NOS)  
REFERENCES  
SERVICE(NOS)  
CONSTRAINT ETRCLI  
FOREIGN KEY (NOC)  
REFERENCES  
CLIENT(NOC));
```

```
1) SELECT INTITULE,  
MAX(POIDS)  
FROM SERVICE S,  
COMMANDE C, PRODUIT P  
WHERE S.NOS=C.NOS  
AND C.NOP=P.NOP  
AND COULEUR='bleu'  
GROUP BY INTITULE ;
```

```
2) SELECT AVG(POIDS)  
FROM SERVICE S,  
COMMANDE C, PRODUIT P  
WHERE S.NOS=C.NOS  
AND C.NOP=P.NOP  
AND INTITULE='Promotion'  
GROUP BY S.NOS ;
```

<pre>3) SELECT P.NOP FROM PRODUIT P WHERE COULEUR='bleu' AND 3 &lt; (SELECT COUNT(DISTINCT NOS) FROM COMMANDE C WHERE C.NOP=P.NOP);  4) SELECT MAX(SUM(QUANTITE)) FROM COMMANDE GROUP BY NOS;</pre>		
---	--	--

### Related Posts:



#### [Série 2 Exercices SQL](#)

Série Exercices SQL Exercice 1 On suppose qu'une bibliothèque gère une base de données dont le schéma est le suivant (les clés primaires des relations sont soulignées) Emprunt (Personne, Livre, DateEmprunt, Date... [Read More](#)



#### [Série Exercices SQL](#)

Série Exercices SQL Exercice 1 Faire les requêtes SQL permettant de répondre aux demandes sur la base de données suivante: Servir (Café, Boisson) Fréquenter (Client, Café) Apprécier (Client, Boisson) Exprimer l... [Read More](#)

[← Article plus récent](#)[Accueil](#)[Article plus ancien →](#)

0 commentaires :

Enregistrer un commentaire

Saisissez votre commentaire...

Commentaire : Sélectionner le pr

Publier

Aperçu

## CONTACT

Votre nom

Votre adresse email

Votre message

## À PROPOS

### X-PROOF

Puisque les langages de programmation ont devenu un outil nécessaire dans le développement informatique, j'ai opté la réalisation d'un petit Blog .. L'objectif de ce Blog est de devenir un outil familier , utile pour les newbies du développement et aussi pour ceux préparant le concours Il rassemble de nombreux

## LABELS

Algorithmique  
Bureautique

Système de numération  
Réseau  
Visual Basic (VB)

Envoyer

exercices, quelques problèmes ,sujets ,et des applications J'espère que vous avez trouvé au moins ce que vous cherchiez :)

AFFICHER MON PROFIL COMPLET

Copyright © 2015 - Tous droits réservés | <http://devinfos.blogspot.com>

Theme by Lasantha - Politique de confidentialité