

Exercices SQL

Requête Select, jointure, opérateur et fonctions SQL, modèle relationnel

E. Thirion - 01/08/2015

Liens utiles:

- [Accéder au cours associé à ce document](#)
- [Installation des dossiers et logiciels nécessaires.](#)
- [Conditions d'accès au téléchargement.](#)
- [Voir tous les cours du même auteur.](#)

I - Exécution de requêtes sous Php-Myadmin

I-1 : Selection de colonnes

Afficher une table contenant les colonnes **NumeroV**, **Jour** et **PlacesLibres** de la table **Vols**.

I-2 : Selection de lignes

Afficher une table contenant les lignes de la table **Vols** dont le jour de la semaine est **Lundi**.

I-3 : Selection multicritère

Afficher les lignes de la table **Vols** dont le jour de la semaine est Mardi, le nombre de places libres strictement supérieur à 150 et la date de départ antérieure au 11 septembre 2001.

Remarques :

- en MySQL les dates se notent entre guillemets dans le format aaaa-mm-jj. Par exemple "1789-07-14" désigne le 14 Juillet 1789.
- pour comparer des dates, on peut utiliser les symboles habituellement utilisés pour comparer des nombres (<, <=, >, >=, =, <>)

I-4 : Selection multi-tables

Exercice 1

Afficher la jointure des trois tables **Vols**, **Avions** et **horaires**, en sélectionnant uniquement (dans cet ordre) la ville de départ, la ville d'arrivée, le jour de la semaine, la date du départ, l'heure de départ, l'heure d'arrivée, la compagnie de transport, le constructeur et le modèle de l'avion.

Exercice2

Ecrire une requête qui affiche la jointure des tables **Vols**, **Horaires**, **Avions** en sélectionnant uniquement les vols Paris-Londre avec au moins 20 places libres, partant avant le 11 septembre 2001, au plus tard à dix heures, un lundi ou un mercredi. (utiliser JourSem **IN** ('Lundi','Mercredi')).

La table affichée ne doit contenir que les colonnes suivantes : le numéro du vol, le jour de la semaine, la date du vol, le nombre de places libres, la ville de départ, la ville d'arrivée, l'heure de départ, l'heure d'arrivée et la compagnie de transport.

I-5 : Elimination de doublons

Ecrire une requête qui affiche tous les types d'avions (Modèle+Constructeur) de la base par une requête **SELECT** appliquée à la table **Avions**.

I-6 : Tri d'une table

Ecrire une requête qui affiche les vols triés par la ville de départ, puis par la ville d'arrivée, puis par la date de départ, puis par l'heure de départ (ordre ascendant dans tous les cas).

I-7 - Colonnes calculées

Ecrire une requêtes donnant le nombre de places occupées de chaque vol. Le nom de cette nouvelle colonne sera **PlacesOccup**. Les autres colonnes à afficher sont les colonnes existantes **NumeroV**, **Avion**, **Capacite** et **PlacesLibres**.

Exercices de la partie I : vérification

| Exercice | Nombre de lignes de la table |
|----------------|------------------------------|
| 1, 4-Ex1, 6, 8 | 840 |
| 2 | 84 |
| 3 et 4 Ex2 | aléatoire |
| I-5 | 3 |

II- Exercices théoriques**II- 1 Jointures****II-1-A : Jointure sur deux tables**

La table **T1** fait référence à la table **T2** (clé primaire **C3**) , via sa clé étrangère **C2** :

| <u>C1</u> | C2 |
|-----------|----|
| 1 | C |
| 2 | A |
| 3 | D |
| 4 | A |

| <u>C3</u> | C4 | C5 |
|-----------|----|----|
| A | 10 | 15 |
| B | 0 | 15 |
| C | 30 | 20 |
| D | 50 | 10 |
| E | 20 | 60 |

Donnez le résultat de la requête suivante: **SELECT C1, C4, C5 FROM T1, T2 WHERE C2 = C3**

Quelles seraient les dimensions de la table obtenue en omettant la jointure C2 = C3 ?

II-1-B : Jointure sur trois tables

La table **T1** fait référence à la table **T2** (clé primaire **C4**) via sa clé étrangère **C2** et à la table **T3** (clé primaire **C7**) via sa clé étrangère **C3**.

| T1 | | | T2 | | | T3 | |
|-----------|----|----|-----------|----|----|-----------|------|
| <u>C1</u> | C2 | C3 | <u>C4</u> | C5 | C6 | <u>C7</u> | C8 |
| 1 | C | b | A | 10 | 15 | a | 1020 |
| 2 | A | a | B | 0 | 15 | b | 304 |
| 3 | D | a | C | 30 | 20 | c | 1961 |
| 4 | A | a | D | 50 | 10 | | |
| | | | E | 20 | 60 | | |

Donnez le résultat de la requête suivante:

SELECT C1, C5, C6, C8 FROM T1, T2, T3 WHERE C2 = C4 AND C3 = C7

Quelle est le nombre de lignes de la table obtenue via les requêtes suivantes ? :

SELECT C1, C5, C6, C8 FROM T1, T2, T3

SELECT C1, C5, C6, C8 FROM T1, T2, T3 WHERE C3 = C7

SELECT C1, C5, C6, C8 FROM T1, T2, T3 WHERE C2 = C4

II-1-C Création des tables sous PhpMyAdmin

Démo (création de la table T1 du A)

Créer une nouvelle base de données

Créer une nouvelle table: nom T1, nombre de champs = 2

Entrer les noms et types des colonnes : INT pour C1 , VARCHAR (1) pour C2, cliquez sur Sauvegarder

Remplir la table: onglet Insérer

Votre travail

Vérifier les résultats des deux exercices précédents en créant les tables sous PhpMyAdmin.

II-2 Fonctions et opérateurs

Le système de gestion d'une bibliothèque utilise une base de donnée contenant une table **EMPRUNT** définit comme suit:

EMPRUNT (Numero, DateEmprunt, Par)

Numero: clé primaire

Par: clé étrangère vers la table lecteur: lecteur ayant effectué cet emprunt.

DOCUMENT (Numero, Titre, Type, Emprunt)

Numero: clé primaire

Type: Livre, CD, DVD

Emprunt: clé étrangère vers la table emprunt. Contient la valeur NULL si le document n'est pas emprunté.

LECTEUR (Numero, Nom, Prenom)

Numero: clé primaire

- A)** Ecrire la requête SQL donnant la liste des livres (titres) empruntés aujourd'hui. Indication: **CURDATE** () donne la date du jour courant.
- B)** Ecrire la requête SQL donnant la liste des CDs ou DVDs (titre, type, date d'emprunt) emprunté par Eric Thirion et trié par date d'emprunt.
- C)** Ecrire la requête SQL donnant la liste des livres (titres) emprunté par Eric Thirion ou Jimi Hendrix.
- D)** Ecrire la requête SQL donnant la liste des livres (titres) empruntés depuis plus d'une semaine. Indication: on pourra utiliser les fonctions **CURDATE** et **TO_DAYS**.
- E)** Ecrire la requête SQL donnant la liste des lecteurs (nom et prénom) ayant emprunté un CD entre le 4 octobre 2012 et le 11 octobre 2012. Indication : les dates SQL peuvent être notées sous la forme "AAAA-MM-JJ".
- F)** Ecrire la requête SQL donnant les documents (numéro, titre, type) emprunté un mardi, jeudi ou vendredi entre le mois de Juin et le mois d'Octobre de l'année en cours. On supposera que les noms des jours sont donnés en français.
- G)** Sachant qu'un livre ou un CD ne peut être emprunté que pour 21 jours et qu'un DVD ne peut être emprunté que pour 14 jours, écrire la requête SQL donnant les lecteurs (nom et prénom) ne respectant pas cette règle.

III - Extraits d'études de cas

Toner-Express - Option D - 1999 - Dossier2

Pour gérer les informations concernant les types de cartouches neuves vendues par l'entreprise, Monsieur Clermont, responsable technique, utilise une base de données dont le schéma est le suivant :

CARTOUCHE (RéfCartouche, DesCartouche, PrixCartoucheNeuve)

IMPRIMANTE (RéfImprimante, DesImprimante, CodeType#)

TYPE(CodeType, LibelléType)

ACCEPTER(RéfImprimante#, RéfCartouche#)

Un attribut (ou un ensemble d'attributs) souligné est une clé primaire. Un attribut dont le nom est suivi du symbole # (croisillon) est une clé étrangère.

- La table CARTOUCHE renseigne sur tous les modèles de cartouches vendues par TONER EXPRESS.
- La table IMPRIMANTE renseigne sur tous les modèles d'imprimantes existants.
- La table TYPE contient trois lignes (imprimantes matricielles, laser et à jet d'encre).
- La table ACCEPTER permet de connaître toutes les imprimantes acceptant un modèle de cartouche et, réciproquement, toutes les cartouches compatibles avec un modèle d'imprimante.

Monsieur Clermont souhaite extraire des données de cette base .

Exprimer en langage SQL la requête permettant d'obtenir la liste des imprimantes (référence et désignation) de type "matricielle".

CNC - Option D - 1999 - Dossier 2

Le schéma relationnel présenté ci-dessous est utilisé pour les enquêtes de satisfaction des clients :

CLIENT (NumeroCli, NomCli, AdresseCli, Numdept)

REPRESENTANT (NumeroRep, NomRep, Numdept)

CRU (CodeInseeCru, NomCru, Classement)

PREFERER (NumeroCli#, CodeInseeCru#, Note)

Les clés primaires sont soulignées, les clés étrangères sont repérées par le caractère #.

Rédiger en langage SQL les requêtes permettant d'obtenir :

- a) La liste des crus (*nom du cru uniquement*) ayant un classement de 100. Cette liste devra être triée sur le nom du cru.
- b) La liste des clients (*nom et adresse*) ayant donné une note égale à 20 à un cru de nom "Ay".

VISTE - Option R - 2004 - Dossier 3

Le Domaine VISTE dispose d'un stand dans un salon parisien où trois personnes sont présentes : le responsable du stand et deux commerciaux pour répondre aux besoins des visiteurs. En vous aidant de l'extrait de la base de données du Domaine VISTE :

BOUTEILLE(TypBouteille, Contenance)

TypBouteille : clé primaire

VIN(CodeVin, DesignationVin, Millesime)

CodeVin : clé primaire

EMBOUTEILLER(CodeVin, TypBouteille, PrixBase, QteStock)

CodeVin, TypBouteille : clé primaire

CodeVin : clé étrangère en référence à VIN

TypBouteille : clé étrangère en référence à BOUTEILLES

CEPAGE(CodeCepage, LibelleCepage)

CodeCepage : clé primaire

COMPOSER(CodeVin, CodeCepage, Proportion, Commentaires)

CodeVin , CodeCepage : clé primaire

CodeVin : clé étrangère en référence à VIN

CodeCepage : clé étrangère en référence à CEPAGE

Rédiger la requête SQL permettant aux commerciaux du stand d'afficher, pour chaque bouteille, la désignation du vin, son millésime, sa contenance et son prix salon (réduction de 10 % du prix de base).