

Développement Web et bases de données

Requête SQL

Site du cours : <https://defelice.up8.site/devwebbdd.html>

Les exercices marqués d'un @ sont optionnels et à faire dans un second temps.

Exercice 1. *Order*

Quel est le résultat de la requête `SELECT * FROM "client" ORDER BY "nom" ASC, "prenom" DESC;` sur la table `client` suivante :

nom	prenom	dateEnregistrement
Robbe	Jean	01/02/2003
Dupont	Anne	04/05/2006
Cash	Jonny	09/07/2013
Dupont	Pierre	01/11/2011
Dentz	Charles	01/08/2007

Exercice 1bis *Selectionner*

Quel est le résultat de la requête `SELECT * FROM "personne" WHERE "age"<37;` sur la table `personne` suivante :

nom	prenom	age
Guyot	Jacqueline	17
Fontaine	Nicole	37
Petit	Véronique	52
Dubois	Chantal	26
Durand	Sylvie	4

Exercice 2. *Choisir*

Quel est le résultat de la requête `SELECT "dateConst","modele" FROM "Navire";` sur la table `"Navire"` suivante :

modele	matricule	dateConstr
Ovni37	SA-104	01/02/2003
Antares25	PO-32	14/08/1996
Mojito650	CH-435	31/07/1990
Aventura37	NA-23	21/12/2001
First29	BR-4567	01/08/2007

Exercice 3. *Jointure*

Quel est le résultat de la requête `SELECT "figure"."forme", x FROM "position","figure";` Voici les tables `position` et `figure` :

position

x	y
1	3
1	2
2	4

figure

forme	couleur
carre	rouge
cercle	jaune

Exercice 4. *Ecrasement*

Quel est le résultat de la requête `SELECT AVG(note),matiere FROM Note GROUP BY matiere;`

Voici la table `Note` :

note	matiere
14	sport
16	math
10	philo
15	philo
18	math
5	philo

Compléments

1. entre guillemet anglais (") : identifiant (nom de table, nom de colonne, vue,...)
2. entre apostrophes (') : chaîne de caractère
3. pour utiliser un apostrophe entre apostrophes : il faut le doubler exemple 'voici l''apostrophe'. Pareil pour placer un guillemet entre guillemets ""les enfants"".
4. Schéma général de select WITH ... SELECT ... FROM ... WHERE ... GROUP BY ... HAVING ... ORDER BY ... LIMIT seule la clause select est obligatoire. (les clauses sont traitées dans cet ordre : with,from,where,group by, having, order by, limit,select)
5. Les éléments de la clause from ou with peuvent aussi être des requêtes select (requêtes imbriquées).
6. Les types sqlite sont : **text**, **integer**, **real**, **blob**.
7. Pour plus d'information sur sqlite voir <https://sqlite.org/lang.html>.
8. Pour ajouter ou supprimer des tables : CREATE TABLE, DROP TABLE.
9. Pour modifier les valeurs d'une table UPDATE, ajouter une ligne INSERT INTO, supprimer des lignes DELETE FROM.
10. On peut joindre des résultats de requête avec UNION exemple : SELECT UNION SELECT
11. On peut utiliser des sous-requête comme table. SELECT FROM table, (SELECT) AS tempTable WHERE ... ;
12. On peut demander de supprimer les doublons SELECT DISTINCT

Exercice 5. *Creation Requête*

L'école Anormalsup possède une base de données pour gérer les notes de ses élèves. Voici la liste des tables avec leurs champs :

- **Etudiant(nEtu,nom,pren,age)** liste des étudiants. **nEtu** représente le numéro d'étudiant. Seul **age** est un champs numérique.
- **Groupe(groupe,nEtu)** relation entre les étudiants et les classes (ou groupes). Chaque étudiant n'appartient qu'à une seule classe.
- **Mail (nEtu,mailP,mails)** Liste des adresses mails des étudiants. Un mail principal, un mail secondaire.
- **Cours (nom,matiere)** Liste des cours chaque cours est associé, à une matière.
- **Coeff (cours,classe,coef)** Qui indique quelle classe (groupe) suit quel cours. Pour chaque classe et pour chaque cours **coef** (un nombre) correspond au coefficient du cours dans la note global de chaque étudiant de la classe.
- **Examen (examen,cours,pointsMax)** La table **Examen** liste des examens donnés. Chaque examen n'est associé qu'à un seul cours. Chaque examen a été donné à l'ensemble des classes qui suivent le cours. **pointMax** est le nombre de points qu'il fallait obtenir à l'examen pour avoir 20/20.
- **Note (nEtu,examen,point)** : Chaque ligne indique le nombre de point qu'à eu l'étudiant à l'examen, il est forcément inférieur ou égal à **pointsMax**.

Seul les champs **age**, **coef** **pointMax** et **point** sont des champs numériques. Tous les autres sont des champs de texte.

Pour chaque demande suivante écrire la requête **sql** qui permet d'obtenir le résultat demandé.

1. Afficher la liste des élèves (nom,prenom)
2. Afficher la liste des étudiants qui ont moins de 18 ans.

3. Afficher la liste de toutes les matières, une seule ligne par matière.
4. Afficher (sur une colonne) la liste de toutes les adresses mails.
5. Afficher la table Note en remplaçant la note par un nombre entre 0 et 1 (0/34-> 0 12/24->0.5 20/20-> 1).
6. Afficher, pour chaque classe , le nombre d'élèves.
7. Afficher, pour chaque couple (classe,cours) la moyenne de la classe à ce cours.
8. Afficher une liste qui pour chaque étudiant (nom,prenom) donne la liste des cours qu'il suit.
9. Affiche pour chaque étudiant (nom,prenom) sa moyenne totale. (pour simplifier la requete vous pouvez créer et utiliser des tables intermédiaires/temporaires).

Illustration :

Voici une requête qui affiche la liste des noms des étudiant associé à leur adresse courriel.

```
SELECT "nom","mailP" FROM Etudiant,Mail WHERE Mail.nEtu=Etudiant.nEtu;
```