

Aide-mémoire SQL

Requêtes simples

`SELECT colonnes FROM table WHERE condition`

SELECT ...	
<code>SELECT titre</code>	sélectionne la colonne <code>titre</code>
<code>SELECT titre, isbn</code>	sélectionne les colonnes <code>titre</code> et <code>isbn</code>
<code>SELECT *</code>	sélectionne toutes les colonnes
<code>SELECT titre AS le_titre</code>	sélectionne la colonne <code>titre</code> et la renomme
Fonctions d'agrégation	
<code>SELECT COUNT(*)</code>	nombre de résultats
<code>SELECT AVG(note)</code>	moyenne
<code>SELECT SUM(note)</code>	somme
<code>SELECT MIN(annee)</code>	valeur minimale
<code>SELECT MAX(annee)</code>	valeur maximale
WHERE ...	
<code>editeur = 'Dargaud'</code>	égalité
<code>annee <> 1980</code>	différence
<code>annee <= 1980</code>	inégalité (de même <code><</code> , <code>></code> et <code>>=</code>)
<code>annee >= 1970 AND annee < 1980</code>	conjonction (de même <code>OR</code> et <code>NOT</code>)
<code>2 * note + 1 > 20</code>	arithmétique (de même <code>-</code> , <code>/</code> et <code>%</code>)
<code>titre LIKE '%Asterix%'</code>	contient le mot Asterix la chaîne après <code>LIKE</code> est un motif, où <ul style="list-style-type: none"> • <code>%</code> désigne n'importe quelle chaîne • <code>_</code> désigne n'importe quel caractère
<code>annee >= 1980 AND (editeur = 'Dargaud' OR editeur = 'Plon')</code>	parenthèses
Tri et suppression de doublons	
<code>SELECT ... ORDER BY titre ASC</code>	trie les résultats selon la colonne <code>titre</code> , dans l'ordre croissant (<code>ASC</code> pour <i>ascending</i>); de même pour l'ordre décroissant avec <code>DESC</code> (pour <i>descending</i>)
<code>SELECT DISTINCT annee</code>	supprime les doublons
<code>SELECT COUNT(DISTINCT annee)</code>	compte les valeurs distinctes
Commentaires	
<code>-- livres récents</code> <code>SELECT titre -- seuls les titres</code> <code>FROM livre WHERE annee >= 2010</code>	un commentaire commence par <code>--</code> et s'étend jusqu'à la fin de la ligne

Requêtes avec jointures

SELECT ... FROM *table*₁ JOIN *table*₂ ON *condition* WHERE ...

Notation	
SELECT livre.titre ...	quand il y a plusieurs tables, on peut préfixer le nom d'une colonne par le nom d'une table
Exemples	
SELECT * FROM emprunt JOIN livre ON emprunt.isbn = livre.isbn	on fait la jointure des tables emprunt et livre en réunissant les lignes ayant le même isbn
SELECT livre.titre, emprunt.retour FROM emprunt JOIN livre ON emprunt.isbn = livre.isbn	même chose, mais en sélectionnant deux colonnes seulement ; on préfixe les noms des colonnes par les noms des tables
SELECT l.titre, u.nom FROM emprunt AS e JOIN livre AS l ON e.isbn = l.isbn JOIN usager AS u ON u.code_barre = e.code_barre	on peut combiner plusieurs jointures on se sert de AS pour donner des noms plus courts aux trois tables

Requêtes imbriquées

Le résultat d'une requête est une table, sur laquelle on peut effectuer une requête :

SELECT ... FROM (*requête*) AS *nom* WHERE ...

Exemple	
SELECT t.titre, t.annee FROM (SELECT * FROM livre WHERE annee >= 2000) AS t WHERE t.titre LIKE 'Les %'	une première requête renvoie les livres édités à partir de 2000, puis on extrait ceux dont les titres commencent par Les (on pouvait aussi le faire avec AND)

Si le résultat d'une requête est une table contenant une seule valeur, on peut l'utiliser dans une condition.

Exemple	
SELECT titre FROM livre WHERE annee = (SELECT MIN(annee) FROM livre)	tous les titres des livres édités la même année que le livre dont la date d'édition est la plus ancienne