

Filtrer et analyser ses données

Une table remplie ne sert à rien si on ne sait pas l'interroger. Ce niveau apprend à poser des questions précises à une table : trier, filtrer, chercher, compter et regrouper.

4TTR

5TTR

6TTR



Découverte

Afficher toute une table, c'est facile. Répondre à une question précise, comme « quels élèves de Jodoigne ont plus de trois absences ? » ou « combien d'élèves habitent chaque commune ? », demande d'autres outils. Ce niveau te les donne un par un, toujours sur la même table.



Objectifs du niveau

À la fin de ce niveau, tu seras capable de :

1. trier le résultat d'une requête sur une ou plusieurs colonnes ;
2. filtrer les lignes avec des conditions, combinées avec `AND` et `OR` ;
3. chercher un motif dans du texte ;
4. repérer et tester les valeurs absentes (`NULL`) ;
5. limiter le nombre de lignes d'un résultat ;
6. utiliser `DISTINCT` pour contrôler la qualité des données ;
7. compter, additionner, calculer des moyennes, et obtenir un résultat par groupe ;
8. traduire une question en français en requête SQL, clauses dans le bon ordre.

Les articles du niveau

Chaque article se lit seul, mais l'ordre proposé construit les requêtes pas à pas.

Article	Ce que tu y apprends
Trier les résultats avec ORDER BY	ranger les lignes, en ordre croissant ou décroissant, sur plusieurs colonnes ; comment SQLite trie nombres, dates et chaînes
Filtrer les lignes avec WHERE	écrire une condition, les opérateurs de comparaison, les apostrophes, <code>AND</code> et <code>OR</code> , <code>BETWEEN</code> , <code>IN</code>
Chercher un motif avec LIKE	les jokers <code>%</code> et <code>_</code> , <code>NOT LIKE</code> , majuscules et accents
Les valeurs absentes : NULL	ce que signifie <code>NULL</code> , le piège de <code>= NULL</code> , <code>IS NULL</code> et <code>IS NOT NULL</code>
Limiter le nombre de lignes avec LIMIT	garder les premières lignes, l'ordre imposé des clauses, traduire une question en requête
Éliminer les doublons avec DISTINCT	afficher chaque valeur une fois, et s'en servir pour repérer des fautes de frappe
Compter et regrouper	<code>COUNT</code> , <code>MIN</code> , <code>MAX</code> , <code>SUM</code> , <code>AVG</code> , <code>ROUND</code> , et un résultat par groupe avec <code>GROUP BY</code>

La base utilisée

Tous les articles utilisent la base `mon-ecole-eleves.db`, à télécharger dans la sidebar de chaque article et à ouvrir dans DB Browser for SQLite. Elle contient une seule table, `eleve`, avec douze élèves :

Colonne	Contenu
<code>id</code>	numéro de l'élève, sa clé primaire
<code>nom</code> , <code>prenom</code>	obligatoires
<code>date_naissance</code>	texte au format <code>AAAA-MM-JJ</code>
<code>commune</code>	commune de domicile
<code>email</code>	adresse email, inconnue (<code>NULL</code>) pour trois élèves
<code>absences</code>	nombre de demi-jours d'absence

⚠ Une faute de frappe a été laissée volontairement dans la base : la commune de Sarah Lambert est écrite `jodoigne`, en minuscules. Ne la corrige pas. Elle sert de fil conducteur : chaque article montre comment elle trompe une requête, ou comment la repérer.



La fiche à étudier

[La fiche à étudier](#) rassemble tout ce qu'il faut savoir **par cœur** à la fin du niveau : le vocabulaire, la syntaxe de chaque clause, l'ordre des clauses, les pièges à connaître, et une série de questions pour te tester. Elle s'imprime seule.

Suite

Tu sais maintenant interroger une table sans la modifier. Le niveau suivant, [Définir et modifier ses données \(DDL et DML\)](#), apprend à créer des tables, à ajouter des lignes, à les modifier... et enfin à corriger la faute sur `jodoigne`.