

Définir et modifier ses données (DDL et DML)

Jusqu'ici, on lisait une base déjà construite. Dans ce niveau, tu écris toi-même la structure d'une table, puis tu y ajoutes, modifies et supprimes des lignes, sans rien casser.

4TTR

5TTR

6TTR

 Découverte

Lire une base, c'est utile. Mais une base vivante change tous les jours : un professeur arrive, un élève déménage, une erreur de saisie doit être corrigée. Dans ce niveau, tu passes de lecteur à constructeur : tu écris la structure d'une table, puis tu écris dans ses lignes.



Objectifs du niveau

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

1. **Distinguer** les instructions qui définissent la structure (DDL) de celles qui manipulent les lignes (DML).
2. **Écrire** un `CREATE TABLE` complet, avec des types adaptés et des contraintes (`PRIMARY KEY` , `NOT NULL` , `UNIQUE` , `DEFAULT`).
3. **Modifier** une table existante avec `ALTER TABLE` , et enregistrer un script SQL qu'on peut rejouer.
4. **Ajouter** des lignes avec `INSERT` , y compris depuis un fichier CSV.
5. **Modifier** et **supprimer** des lignes avec `UPDATE` et `DELETE` , en vérifiant d'abord par un `SELECT` quelles lignes seront touchées.
6. **Sauvegarder** une base dans un fichier `.sql` et la restaurer.

Les articles du niveau

Article	Ce que tu y apprends
Créer et modifier une table : CREATE et ALTER	DDL et DML, <code>CREATE TABLE</code> , types, contraintes, <code>ALTER TABLE</code> , <code>DROP TABLE</code> , script SQL
Ajouter des lignes : INSERT	<code>INSERT INTO ... VALUES</code> , colonnes omises, refus de la base, import d'un fichier CSV
Modifier des lignes : UPDATE	<code>UPDATE ... SET ... WHERE</code> , la méthode « <code>SELECT</code> d'abord », le <code>WHERE</code> oublié
Supprimer des lignes : DELETE	Sauvegarde et restauration, <code>DELETE FROM ... WHERE</code> , <code>DELETE</code> ou <code>DROP TABLE</code>

Les articles se suivent dans cet ordre : la table créée dans le premier est remplie dans le deuxième, corrigée dans le troisième, puis nettoyée dans le quatrième.

Les bases utilisées

Tout le niveau travaille sur la base d'une école : une table `eleve` (12 élèves), à laquelle on ajoute une table `professeur` .

Article	Base de départ	Ce qu'elle contient
CREATE et ALTER	<code>mon-ecole-eleves.db</code>	la table <code>eleve</code> seule
INSERT	<code>mon-ecole-eleves.db</code>	la table <code>eleve</code> seule (le script de création de <code>professeur</code> est donné)
UPDATE	<code>mon-ecole-professeurs.db</code>	<code>eleve</code> + <code>professeur</code> remplie (6 professeurs)
DELETE	<code>mon-ecole-relier.db</code>	la même base, après les corrections de l'article UPDATE

💡 POURQUOI CHAQUE ARTICLE FOURNIT SA BASE

Dans ce niveau, tu modifies réellement les données. Une manipulation ratée ou une séance manquée, et ta base ne ressemble plus à celle de l'article. En téléchargeant la base de départ dans la barre latérale, tu repars toujours d'un état identique à celui du texte : les résultats que tu obtiens sont ceux que l'article affiche.

⚠️ Travaille toujours sur une **COPIE** du fichier téléchargé. Si tout tourne mal, il suffit de le télécharger à nouveau.



La fiche à étudier

Tout ce qui s'apprend par cœur dans ce niveau (vocabulaire, syntaxe, contraintes, bons réflexes, pièges) est rassemblé dans une fiche imprimable : [Fiche à étudier — Définir et modifier ses données](#).

Suite

Ta base contient maintenant deux tables solides, mais elles s'ignorent. Le niveau suivant les relie : [Relier deux entités \(1-N\)](#).