

Objectifs pédagogiques :

- Écrire les requêtes SQL de définition des données sur une table (CREATE, ALTER, DROP) ;

Situation problème :

Vous travaillez dans une entreprise de gestion scolaire et êtes chargé(e) de créer une base de données pour stocker les informations des élèves, des enseignants et des cours. Actuellement, il n'existe aucun système structurant ces informations.

Consignes :

Vous devez :

1. Créer des tables pour organiser les données (ex. élèves, enseignants, cours).
2. Définir les colonnes pour chaque table (nom, prénom, identifiant, etc.).
3. Permettre des modifications futures si les besoins changent.

Resumé

1. Notion Du langage SQL

SQL (Structured Query Language ou langage de requêtes structuré) est un langage de définition, de manipulation, d'interrogation, et de contrôle les données dans une base de données. **Il a été conçu en 1970 par la société IBM.**

Une requête est une instruction permettant d'interagir avec une base de données. Elle peut être utilisée pour récupérer, insérer, modifier ou supprimer des données, ainsi que pour gérer la structure des tables.

2. Création et suppression d'une base de données

a. Création d'une base de données

CREATE DATABASE Nom ;

b. Suppression d'une base de données

DROP DATABASE Nom ;

3. Création et suppression d'une table

Une table est une collection de données organisées sous formes de tableau à deux dimensions et relatives à un domaine bien défini. Il contient des tuples ou lignes ou enregistrements et des colonnes ou champs de données ou attributs.

a. Création de table

La commande CREATE TABLE permet de créer une nouvelle table.

- Sa syntaxe est la suivante :

CREATE TABLE nom_de_la_table
 (nom_de_la_colonne1 Type_de_donnée [Paramètre_colonne],
 nom_de_la_colonne2 Type_de_donnée [Paramètre_colonne],
 :
 :
 nom_de_la_colonnen Type_de_donnée [Paramètre_colonne]) ;

- Les principaux types de données sont présentés dans le tableau ci -après

Catégorie	Type	Description	Exemple
Numériques	INT	Entier standard	23, -56
	SMALLINT	Petit entier	1200
	DECIMAL(p,s)	Nombre décimal précis (p = chiffres, s = décimales)	DECIMAL(10,2) → 1234.56
	FLOAT	Nombre à virgule flottante (moins précis)	3.14159
Textuels (Chaînes de caractères)	DOUBLE	Nombre à virgule flottante plus précis	12345.6789
	CHAR(n)	Chaîne de longueur fixe n caractères	CHAR(5) → 'SQL '
Temporels (Dates et heures)	VARCHAR(n)	Chaîne de longueur variable (jusqu'à n caractères)	VARCHAR(50) → 'Bonjour'
	TEXT	Texte long (jusqu'à plusieurs Mo)	"Ceci est une description longue."
	DATE	Date au format YYYY-MM-DD	'2024-02-04'
Booléens (Vrai/Faux)	TIME	Heure au format HH:MI:SS	'14:30:00'
	DATETIME	Date et heure combinées	'2024-02-04 14:30:00'
	TIMESTAMP	Date/heure avec fuseau horaire	'2024-02-04 14:30:00'
	BOOLEAN	Valeur vraie/faux	TRUE, FALSE
Spéciaux	BIT(1)	Bit unique (0 ou 1)	0, 1
	ENUM	Liste de valeurs prédéfinies	ENUM('Homme', 'Femme')
	SET	Liste de valeurs multiples	SET('Lecture', 'Musique', 'Sport')

- Une **contrainte d'intégrité** en SQL est une règle qui garantit la cohérence, la validité et la fiabilité des données dans une base de données. Elle empêche l'insertion de données incorrectes, incohérentes ou redondantes.

Contrainte	Description	Exemple
PRIMARY KEY	Définit une clé primaire, qui identifie chaque enregistrement de manière unique.	ID INT PRIMARY KEY
FOREIGN KEY	Crée une relation entre deux tables en reliant une colonne à la clé primaire d'une autre table.	FOREIGN KEY (Client_ID) REFERENCES Clients(ID)
NOT NULL	Empêche une colonne d'accepter des valeurs NULL (obligatoire).	Nom VARCHAR(100) NOT NULL
UNIQUE	Assure que toutes les valeurs d'une colonne sont uniques.	Email VARCHAR(255) UNIQUE
DEFAULT	Définit une valeur par défaut si aucune valeur n'est spécifiée.	Statut VARCHAR(20) DEFAULT 'Actif'
CHECK	Impose une condition sur les valeurs d'une colonne.	Age INT CHECK (Age >= 18)
AUTO_INCREMENT	Génère automatiquement une valeur unique pour une colonne (souvent utilisé avec ↓ PRIMARY KEY).	ID INT PRIMARY KEY AUTO_INCREMENT

NB :

- Une **Clé primaire (PRIMARY KEY)** est une colonne (ou un ensemble de colonnes) qui identifie de manière unique chaque enregistrement d'une table.

- Une **Clé étrangère (FOREIGN KEY)** est une colonne qui établit une relation entre deux tables en faisant référence à la clé primaire d'une autre table.

Exemple :

```
CREATE TABLE Enseignants (
  id INT PRIMARY KEY,
  nom VARCHAR(50) NOT NULL,
  prenom VARCHAR(50) NOT NULL,
  matiere VARCHAR(50)
);
```

```
CREATE TABLE Cours (
  id_cours INT PRIMARY KEY,
  intitule VARCHAR(100) NOT NULL,
  enseignant_id INT,
  FOREIGN KEY (enseignant_id) REFERENCES Enseignants(id)
);
```

b. Suppression d'une table

```
DROP TABLE Nom_table ;
```

4. Modification des tables

Type de modification	Syntaxe	Commentaire
Ajouter un champ	ALTER TABLE Nom_Table ADD Champ Type ;	Permet d'ajouter un champ à une table
Supprimer un champ	ALTER TABLE Nom_Table DROP COLUMN Champ;	Permet de supprimer un champ dans une table
Renommer un champ	ALTER TABLE Nom_Table CHANGE Ancien_champ Nouveau_Champ Nouveau_Type;	Permet de changer le nom d'un champ. Elle combine la suppression et l'ajout d'un champ
Modifier un champ	ALTER TABLE Nom_Table MODIFY Champ Nouveau_Type ;	Permet de modifier le type d'un champ

EXERCICE D'APPLICATION : Mise en Pratique

Contexte : Vous devez créer et modifier des tables pour une base de données scolaire.

Consignes :

1. Créez une table Enseignants avec les colonnes suivantes :
 - id : Identifiant unique (entier, clé primaire).
 - nom : Nom de l'enseignant (chaîne de caractères, obligatoire).
 - prenom : Prénom de l'enseignant (chaîne de caractères, obligatoire).
 - matiere : Matière enseignée (chaîne de caractères).
2. Ajoutez une colonne email à la table Enseignants (chaîne de caractères).
3. Supprimez la colonne matiere de la table Enseignants.
4. Supprimez la table Enseignants.