

# FORMATION EN DEVELOPPEMENT WEB

100/100 ACADEMY

# Requêtes SQL Classées par Catégories

Toutes les requetes

# 1. Introduction

- ▶ SQL (Structured Query Language) : Un langage utilisé pour interagir avec les bases de données relationnelles.
- ▶ **Catégories de Requêtes SQL :**
  - ▶ Requêtes DDL (Data Definition Language)
  - ▶ Requêtes DML (Data Manipulation Language)
  - ▶ Requêtes DQL (Data Query Language)
  - ▶ Requêtes DCL (Data Control Language)
  - ▶ Requêtes TCL (Transaction Control Language)

# Requêtes DDL (Data Definition Language)

- ▶ Les requêtes DDL sont utilisées pour définir la structure de la base de données.
  - ▶ **CREATE** : Créer une nouvelle table, base de données ou index.

```
CREATE TABLE users (  
    id INT PRIMARY KEY,  
    name VARCHAR(100),  
    email VARCHAR(100)  
);
```

# Requêtes DDL (Data Definition Language)

- ▶ **ALTER** : Modifier la structure d'une table existante.

```
ALTER TABLE users ADD COLUMN age INT;
```

# Requêtes DDL (Data Definition Language)

- ▶ DROP : Supprimer une table ou une base de données.

```
DROP TABLE users;
```

# Requêtes DDL (Data Definition Language)

- ▶ TRUNCATE : Vider une table de toutes ses données sans supprimer la structure.

```
TRUNCATE TABLE users;
```

# 3. Requêtes DML (Data Manipulation Language)

- ▶ Les requêtes DML sont utilisées pour manipuler les données dans les tables.
  - ▶ INSERT : Insérer des données dans une table.

```
INSERT INTO users (id, name, email) VALUES (1, 'John Doe', 'john@example.com');
```



### 3. Requêtes DML (Data Manipulation Language)

- ▶ UPDATE : Mettre à jour des données existantes.

```
UPDATE users SET email = 'john.doe@example.com' WHERE id = 1;
```

### 3. Requêtes DML (Data Manipulation Language)

- ▶ DELETE : Supprimer des données spécifiques.

```
DELETE FROM users WHERE id = 1;
```

## 4. Requêtes DQL (Data Query Language)

- ▶ Les requêtes DQL sont utilisées pour interroger les données dans une table.
- ▶ SELECT : Récupérer des données.

```
SELECT name, email FROM users;
```

## 4. Requêtes DQL (Data Query Language)

- ▶ WHERE : Appliquer des filtres aux résultats.

```
SELECT * FROM users WHERE age > 25;
```

## 4. Requêtes DQL (Data Query Language)

- ▶ ORDER BY : Trier les résultats.

```
SELECT * FROM users ORDER BY name ASC;
```

## 4. Requêtes DQL (Data Query Language)

- ▶ GROUP BY : Grouper les résultats par une colonne spécifique.

```
SELECT age, COUNT(*) FROM users GROUP BY age;
```

## 4. Requêtes DQL (Data Query Language)

- ▶ HAVING : Appliquer des filtres après un GROUP BY.

```
SELECT age, COUNT(*) FROM users GROUP BY age HAVING COUNT(*) > 1;
```

## 4. Requêtes DQL (Data Query Language)

- ▶ **JOIN** : Combiner des données de plusieurs tables.
- ▶ **INNER JOIN** : Retourne uniquement les lignes ayant des correspondances dans les deux tables.

```
SELECT users.name, orders.amount  
FROM users  
INNER JOIN orders ON users.id = orders.user_id;
```



## 4. Requêtes DQL (Data Query Language)

- ▶ **JOIN** : Combiner des données de plusieurs tables.
  - ▶ **LEFT JOIN** : Retourne toutes les lignes de la première table, même sans correspondance dans la seconde table.

```
SELECT users.name, orders.amount  
FROM users  
LEFT JOIN orders ON users.id = orders.user_id;
```

## 5. Requêtes DCL (Data Control Language)

- ▶ Les requêtes DCL sont utilisées pour contrôler l'accès aux données.
  - ▶ GRANT : Accorder des privilèges à un utilisateur.

```
GRANT SELECT, INSERT ON database_name TO 'username';
```

## 5. Requêtes DCL (Data Control Language)

- ▶ REVOKE : Retirer des privilèges d'un utilisateur.

```
REVOKE INSERT ON database_name FROM 'username';
```

## 6. Requêtes TCL (Transaction Control Language)

- ▶ Les requêtes TCL sont utilisées pour gérer les transactions.
- ▶ BEGIN TRANSACTION : Commencer une transaction.

```
BEGIN;
```

## 6. Requêtes TCL (Transaction Control Language)

- ▶ COMMIT : Valider les changements dans la base de données.

```
COMMIT;
```

## 6. Requêtes TCL (Transaction Control Language)

- ▶ ROLLBACK : Annuler les changements dans une transaction.

```
ROLLBACK;
```

## 6. Requêtes TCL (Transaction Control Language)

- ▶ SAVEPOINT : Définir un point de sauvegarde dans une transaction.

```
SAVEPOINT sp1;
```

## 6. Requêtes TCL (Transaction Control Language)

- ▶ RELEASE SAVEPOINT : Supprimer un point de sauvegarde.

```
RELEASE SAVEPOINT sp1;
```



# 7. Requêtes Avancées

- ▶ UNION : Combiner les résultats de plusieurs requêtes SELECT.

```
SELECT name FROM users  
UNION  
SELECT name FROM admins;
```

# 7. Requêtes Avancées

- ▶ EXISTS : Vérifier l'existence d'une sous-requête.

```
SELECT * FROM users WHERE EXISTS (  
    SELECT 1 FROM orders WHERE orders.user_id = users.id  
);
```

# 7. Requêtes Avancées

- ▶ IN : Vérifier si une valeur est dans une liste.

```
SELECT * FROM users WHERE id IN (1, 2, 3);
```

# 7. Requêtes Avancées

- ▶ LIKE : Rechercher un modèle dans une colonne.

```
SELECT * FROM users WHERE name LIKE 'J%';
```

## 8. Conclusion

- ▶ SQL : Un langage puissant pour interagir avec les bases de données.
- ▶ Catégories de requêtes : DDL, DML, DQL, DCL, TCL.
- ▶ Utilisation : Connaître ces commandes permet de gérer efficacement la structure, les données et les autorisations dans une base de données,