

# SQL: Structured Query Language

---

INES.BENFREDJ@ESPRIT.TN

2LBC-BIS-BI

ESPRIT SCHOOL OF BUSINESS

# SQL (Structured Query Language)

---

- Le **SQL**: Structured Query Language (*en français langage de requête structurée*) est un langage permettant de communiquer avec une base de données relationnelle.
- Les instructions principales de manipulation du contenu de la base de données sont: `SELECT`, `UPDATE`, `INSERT` et `DELETE`. Elles correspondent respectivement aux opérations de recherche de contenu, modification, ajout et suppression.

# SELECT: recherche de contenu (1)

---

- Syntaxe: `SELECT nom_du_champ FROM nom_du_table`
- Exemple: soit la table « client » suivante:

identifiant	prenom	nom	ville
1	Pierre	Dupond	Paris
2	Sabrina	Durand	Nantes
3	Julien	Martin	Lyon

Si l'ont veut avoir la liste de toutes les villes des clients, il suffit d'effectuer la requête SQL ci-dessous :

**SELECT ville FROM client**

On obtient alors le résultat suivant:

ville
Paris
Nantes
Lyon

# SELECT: recherche de contenu (2)

---

- On peut sélectionner plusieurs champs: *SELECT prenom, nom FROM client*
- Ou encore sélectionner tous les champs: *SELECT \* FROM client*

## Sélection conditionnée:

extraire les lignes d'une base de données qui respectent une condition avec la commande WHERE en complément avec SELECT.

**Syntaxe:** *SELECT nom\_des\_champs FROM nom\_table WHERE condition*

**Exemple:** *SELECT \* FROM client WHERE ville = 'Paris'*

Résultat:

identifiant	prenom	nom	ville
1	Pierre	Dupond	Paris

# UPDATE: mise à jour de contenu (1)

---

- La commande UPDATE permet d'effectuer des modifications sur des lignes existantes. Très souvent cette commande est utilisée avec WHERE pour spécifier sur quelles lignes doivent porter la ou les modifications.

## Syntaxe:

UPDATE *table*

SET *nom\_colonne\_1* = 'nouvelle valeur1', *nom\_colonne\_2* = 'nouvelle valeur2',...

WHERE condition

## Exemple:

UPDATE *client*

SET *prenom*= 'Patrick', *ville*= 'Grenoble'

WHERE *identifiant* = 2

# UPDATE: mise à jour de contenu (2)

---

Résultat:

identifiant	prenom	nom	ville
1	Pierre	Dupond	Paris
2	<b>Patrick</b>	Durand	<b>Grenoble</b>
3	Julien	Martin	Lyon

# INSERT: Insertion d'un nouvel enregistrement

---

## Syntaxe:

INSERT INTO table (nom\_colonne\_1, nom\_colonne\_2, ...) VALUES ('valeur 1', 'valeur 2', ...)

## Exemple (1): Insérer un seul enregistrement à la fois

*INSERT INTO client (prenom, nom, ville) VALUES ('Joseph', 'Ferrain', 'Reims')*

## Exemple (2): Insérer plusieurs enregistrements à la fois

*INSERT INTO client (prenom, nom, ville) VALUES ('Joseph', 'Ferrain', 'Reims'), ('Paul', 'Dasy', 'Paris'), ('Jack', 'Dudot', 'Lyon')*

*NB: Généralement, l'identifiant dans une table est une clé primaire. Il est auto-incrément. On ne touche pas à sa valeur dans la base de données. Il s'incrémente automatiquement. C'est pour cette raison, on ne l'a pas spécifié dans la requête précédente.*

# DELETE: suppression de contenu

---

- La commande DELETE permet de supprimer des lignes dans une table. En utilisant cette commande associée à WHERE il est possible de sélectionner les lignes concernées qui seront supprimées.

**Syntaxe:**           DELETE FROM `table`  
                          WHERE condition

**Exemple:**           *DELETE FROM client*  
                          *WHERE identifiant = 2*

Résultat:

identifiant	prenom	nom	ville
1	Pierre	Dupond	Paris
3	Julien	Martin	Lyon



---

Merci de votre attention