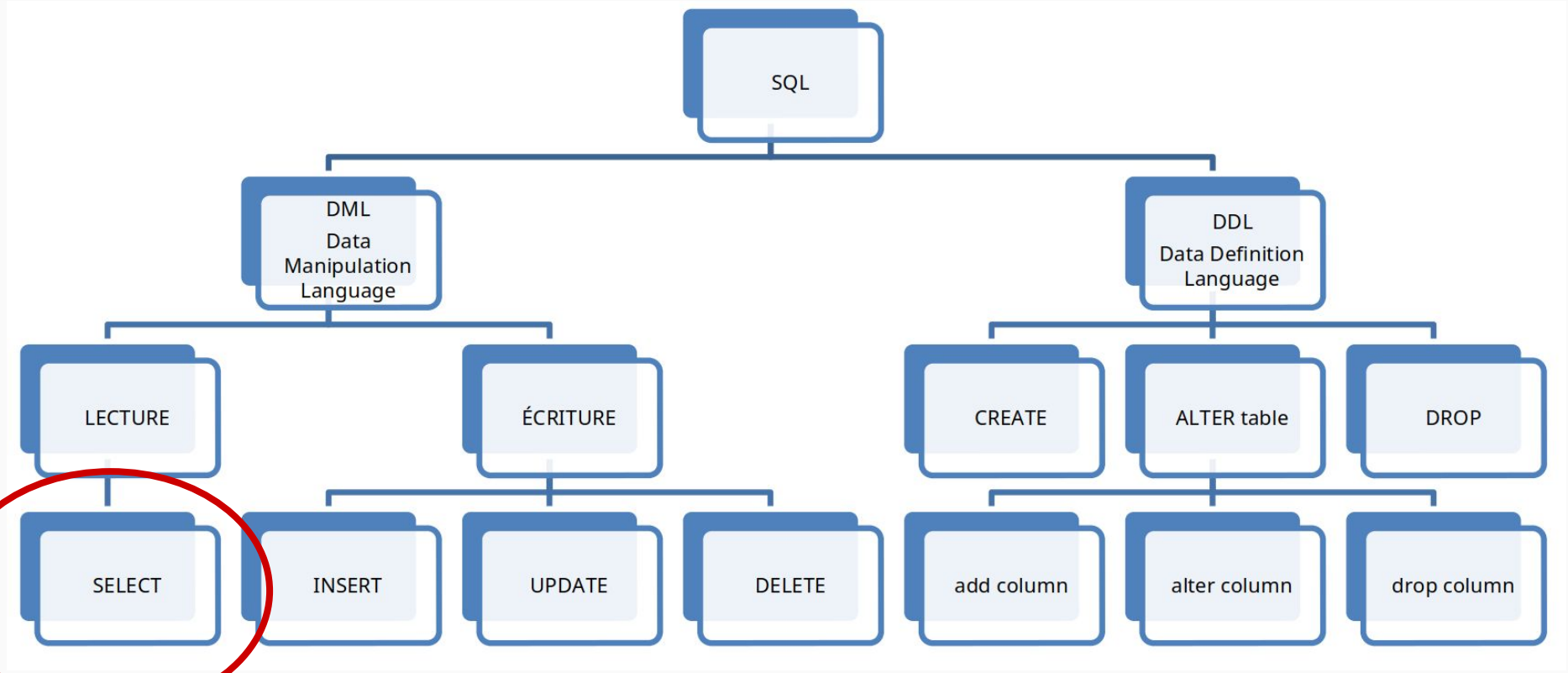


RECHERCHE en SQL

Comment trouver les données que l'on cherche

La RECHERCHE dans tout le langage SQL est une utilisation avancée du SELECT



Types de recherches de données

- RECHERCHE avec des conditions précises
  - Combinaison de conditions avec **and** et **or** et **not**
  - Intervalles (range) avec **between** et éventail avec **in**
  - Élimination des nulles **not** ou répétitions avec **distinct**
- RECHERCHE avec un pattern de texte
  - Utilisation du mot-clé **like**

RECHERCHE avec des conditions précises

...

CRUD

Combinaison de conditions AND et OR

★ Si on combine deux conditions avec AND, chaque rangée devra répondre aux deux conditions pour être sélectionnées

```
SELECT * FROM film
WHERE annee > 2018 AND duree > 120
```

★ Si on combine deux conditions avec OR, chaque rangée peut seulement satisfaire une seule des deux conditions

```
SELECT * FROM film
WHERE annee > 2018 OR duree > 120
```

CRUD

Intervalles de valeurs et éventails de valeurs

★ C'est équivalent à combiner une condition plus grande et plus petite

```
SELECT * FROM film
WHERE annee BETWEEN 2015 AND 2020
```

★ Si on imagine que les films cultes sont toujours produits dans des années multiples de 5 (voir sql). Aucune formule n'est acceptée par le SQL

```
SELECT * FROM film
WHERE annee IN (2010, 2015, 2020)
```

CRUD

Élimination des valeurs invalides (nulles ou répétées)

★ On peut exclure les films dont le titre serait null

```
SELECT * FROM film
WHERE titre NOT NULL
```

★ On peut exclure des films dont le titre serait en double

```
SELECT DISTINCT(titre) FROM film
WHERE annee > 2017
```

RECHERCHE avec un PATTERN de texte

...

CRUD

Recherche avec un pattern de texte et like et %

★ On peut rechercher les films dont le nom commence par 'Les'

```
SELECT * FROM film
WHERE titre LIKE 'Les%'
```

★ On peut exclure des films dont le titre contient le mot 'amour' n'importe ou

```
SELECT * FROM film
WHERE titre not LIKE '%amour%'
```

CRUD

Recherche avec un pattern de texte et like et %

'abc' LIKE 'abc' true

'abc' LIKE 'a%' true

'abc' LIKE '_b_' true

'abc' LIKE 'c' false

CRUD

Recherche avec un pattern de texte et similar

★ On peut rechercher les gens qui sont des etudiant(s) ou etudiante(s)

```
SELECT * FROM personne
WHERE occupation SIMILAR TO 'etudian(te|ts|tes|t)'
```

★ Même chose avec un multiplicateur

```
SELECT * FROM personne
WHERE occupation SIMILAR TO 'etudiante?s?'
```

CRUD

Recherche avec un pattern de texte et similar

'abc' SIMILAR TO 'abc' true

'abc' SIMILAR TO 'a' false

'abc' SIMILAR TO '%(b|d)%' true

'abc' SIMILAR TO '(b|c)%' false

'-abc-' SIMILAR TO '%\mabc\M%' true

'xabcy' SIMILAR TO '%\mabc\M%' false

CRUD

Substring

★ SUBSTRING - On peut extraire des sous-chaines dans la requête

https://www.postgresql.org/docs/current/functions-matching.html#FUNCTIONS-POSIX-REGEXP

★ SPLIT_PART

https://www.postgresqltutorial.com/postgresql-string-functions/postgresql-split_part/

★ REGEX_MATCH

https://www.postgresql.org/docs/current/functions-matching.html#FUNCTIONS-SIMILAR-TO-REGEXP

CRUD

Substring

★ CONCAT - pour concaténer des valeurs et des constantes

https://www.postgresqltutorial.com/postgresql-string-functions/postgresql-concat-function/

★ string_agg - pour concaténer des valeurs et des constantes

https://www.postgresql.org/docs/9.0/functions-aggregate.html