

MÉMO SYNTAXE SQL

requête simple

SELECT *
col1
col1, col3, col1) - ici on peut faire des
col1+col2) calculs ligne à ligne

FROM table

+ on peut utiliser **AS** pour renommer une colonne ou une table

+ avec le mot clé **DISTINCT** seules des lignes 2 à 2 différentes sur les colonnes indiquées sont conservées.

requête avec filtrage

SELECT ... **FROM** ...
WHERE une condition

→ cette condition s'écrit à partir des opérateurs $>$, $=$, $>=$, $<$, $<=$ et des opérateurs booléens **AND**, **OR** et **NOT**.

→ on peut utiliser **IS NULL** pour tester si la valeur d'un champ est NULL (notamment avec les jointures gauches)

→ on peut utiliser **IN** (sous-requête)

classement et extrait

SELECT ... **FROM** ...
(**WHERE** ...)
ORDER BY col **ASC/DESC**

+ **LIMIT** n

peut choisir ordre croissant ou décroissant, ASC par défaut

on peut mettre plusieurs

permet de ne garder que les n premiers enregistrements (optionnel)

f. d'agrega,
sum, max, min
average...

agrégation

SELECT (f(col1, col2, ...)) **FROM** table

(+ **WHERE** ... envenhullim t)

GROUP BY condition c1

+ **HAVING** condition c2

si omis les lignes sont toutes agrégées

optionnel, filtre les lignes après agrégation contrairement au where qui les filtre avant.

jointure

SELECT t1... t2... **FROM** table1 AS t1

JOIN table2 AS t2 **ON** condition
par exemple t1.col1 = t2.col3

NB: sans condition le résultat est le produit cartésien des tables

SELECT ... **FROM** table1 AS t1
LEFT JOIN table2 AS t2 **ON** cond.

ex t1 =

a
b

t2 =

c	d
e	f
g	h

SELECT * **FROM** t1 **JOIN** t2
ON t1.col1 = t2.col3

a	c	d
a	e	f

SELECT * **FROM** t1 **LEFT JOIN** t2
ON t1.col1 = t2.col3

a	c	d	
a	e	f	
b	NULL	NULL	NULL