

SIG

QGIS version 3.6

OS : Elementary

LES OPÉRATEURS

Dans la calculatrice de champs, dans les sélections par expressions et toutes les fois que vous voyez



Les exemples sont donnés sur la base du fichier GEOFLA®

=, <, >, >=, <=

Ne devraient pas soulever de questions ardues. Sauf peut-être pour les lettres, oui, ces symboles mathématiques peuvent fonctionner sur des chaînes alphanumériques (des lettres, quoi)

- " NOM_COM " > 'B' ou " NOM_COM " >= 'B' élimine toutes les communes qui commencent par A.
- " NOM_COM " < 'BE' va sortir toutes les communes qui commencent par A et BA (mais aussi BB, BD, si y'en avait.....).

En fait, imaginer votre liste triée par ordre alphabétique et faites un trait au niveau du B, si vous garder le haut de la liste, vous avez fait > 'B'... et inversement pour le bas < 'B'. Je vous laisse méditer là-dessus et faire des essais aux résultats parfois surprenants.

!= ou <>

signifie différent de, il fonctionne aussi sous sa forme <>

LIKE et ILIKE

Rien à voir avec bookFace. C'est 2 mots permettent de faire une recherche sur des « modèles ». Avant les exemples, il est important de préciser que LIKE est sensible à la casse (MAJUSCULE-minuscules) alors ILIKE ne l'est pas.

Sauf recherches particulières, j'ai donc tendance à utiliser que ILIKE

- " NOM_DEPT " LIKE 'FINISTERE' ⇒ tester ⇒ 283 lignes
- " NOM_DEPT " LIKE 'finistere' ⇒ tester ⇒ 0 lignes
- " NOM_DEPT " ILIKE 'FINISTERE' ⇒ tester ⇒ 283 lignes
- " NOM_DEPT " ILIKE 'finistere' ⇒ tester ⇒ 283 lignes

Vous remarquerez qu'il faut le contenu exact et entier pour qu'ils fonctionnent, aussi LIKE et ILIKE sont quasi indissociables des caractères « jokers » % et _.

% et _

Ces caractères dits JOKER vont permettre de remplacer une partie du contenu recherché.

% remplace une chaîne entière (plusieurs caractères)

- " NOM_DEPT (I)LIKE 'FINIS%' ⇒ tester ⇒ va sortir tout ce qui commence par FINIS (FINISTERE, FINIS, FINISBLABLA, FINIS123.....)
- " NOM_DEPT " (I)LIKE '%ERE' ⇒ tester ⇒ va sortir tout ce qui finit par ERE (FINISTERE, ISERE, LOZERE, BLABLAERE, 2323ERE.....)
- " NOM_DEPT " (I)LIKE '%ER%' ⇒ tester ⇒ va donc sortir tout ce qui contient ER quelque part (ALLIER, CHER, FINISTERE, GERS, HERAULT, ISERE, LOZERE, TERRITOIRE DE BELFORT...)

Quizz : devinez ce que sort 'A%E' ?

_ (underscore), qui n'est pas dans les boutons, c'est mon astuce du chef à moi, ne remplace que 1 caractère, c'est donc comme la case blanche du scrabble®

- " NOM_DEPT " (I)LIKE 'A_B' va sortir ABB, ACB, A2B mais pas AB BB, A123B ou AaaalalalacB

AND, OR

Deux autres boutons AND (ET) et OR (ou), ces expressions permettent de compiler plusieurs recherches. Un peu d'Algèbre de Boole (non, restez, cela va être simple) avant de montrer les exemples. Donc pour Boole et ses potes

condition 1	AND ou ET ou.	condition 2	resultat
vraie	AND	vraie	= vrai
vraie	AND	fausse	= faux
fausse	AND	vraie	= faux
fausse	AND	fausse	= faux

AND (ET)

Le ET fait en sorte que les 2 conditions doivent être remplies pour que le résultat soit positif (vrai ou 1). C'est donc Fromage ET dessert, vous mangez les deux ou rien.

OR (OU)

le OU fait en sorte que l'une des deux conditions doit être vraie pour que le résultat soit positif (vrai ou 1). C'est donc Fromage OU dessert, vous mangez forcément l'un ou l'autre, voire les deux.

condition 1	OR ou OU ou +	condition 2	résultats
vraie	OR	vraie	vraie
vraie	OR	fausse	vraie

condition 1 OR ou OU ou + condition 2 résultats

fausse	OR	vraie	vraie
--------	----	-------	-------

fausse	OR	fausse	faux
--------	----	--------	------

Comment cela intervient dans nos filtres, on peut s'en servir pour cumuler 2 requêtes

- " CODE_DEPT " = '29' AND " NOM_COM " ILIKE 'A%' ⇒ Ici, seules 3 communes remplissent les 2 critères : département 29 et commune commençant par A (Audierne, Argol, Arzano)
- " CODE_DEPT " = '29' OR " NOM_COM " ILIKE 'A%' ⇒ Je vais avoir toutes les communes du département 29 (la condition 1 est vraie pour ces communes) et toutes les communes commençant par " A " (la condition 2 est vraie pour ces communes) = 2207 résultats

Ça fait mal au crâne au début mais cela vient vite. Et puis rien n'empêche de se tromper et de taper OR à la place AND ou inversement, pour essayer.

IN, NOT IN

In or not in, that is the question : whether 'tis nobler.....

Ces 2 petits mots permettent d'enchaîner plusieurs valeurs uniques dans un élément, c'est un élément de sous-requête. Cela ne vous parle pas, moi non plus alors exemple concret.

Au lieu de faire " CODE_DEPT " = '22' OR " CODE_DEPT " = '29' OR " CODE_DEPT " = '56' qui me permet d'afficher les départements 22,29 et 56.

Je peux faire " CODE_DEPT " IN ('22','29','56'), ce qui est plus simple lorsque l'on a plein de valeurs.

Attention, prenez bien soin à respecter la syntaxe avec les (et , à taper vous-même.

NOT IN fait l'inverse de IN, donc " CODE_DEPT " NOT IN ('22','29','56') va afficher tous les départements sauf les 22,29,56.

NOT

Pour vous embrouiller, le NOT permet de sortir le résultat inverse de celui demandé. Donc quelle que soit ma requête, si je mets NOT devant, j'obtiens l'inverse de ma demande.

NOT " NOM_DEPT " (I)LIKE '%ER%' va me sortir toutes les communes qui ne contiennent pas ER dedans,

NOT " CODE_DEPT " = '29' va afficher tous les départements sauf le 29, j'aurais pu faire aussi " CODE_DEPT " != '29' ou " CODE_DEPT " NOT IN ('29')

Quizz : Que me sort NOT " CODE_DEPT " NOT IN ('22','29','56') ?

En fait c'est pratique lorsque vous avez une requête super costaud et que vous voulez inverser son résultat sans tout retaper ou modifier.

TOUJOURS PASSER EN MODE ÉDITION POUR POUVOIR CRÉER DES DESSINS