



Rapports SQL

- *initiation* -

KohaLa - 29 Septembre 2025 Emmanuel Bétemps

SQL ?



Structured Query Language

langage de **gestion** d'une base de données relationnelle

*création,
modification,
interrogation,
suppression*

Intérêts du SQL



= obtenir n'importe quelles données avec n'importe quels critères

données statistiques (compter)

données bibliographiques (lister, pour vérifier la base par exemple)

Risques d'une requête mal formulée



pas de résultat (c'est visible)

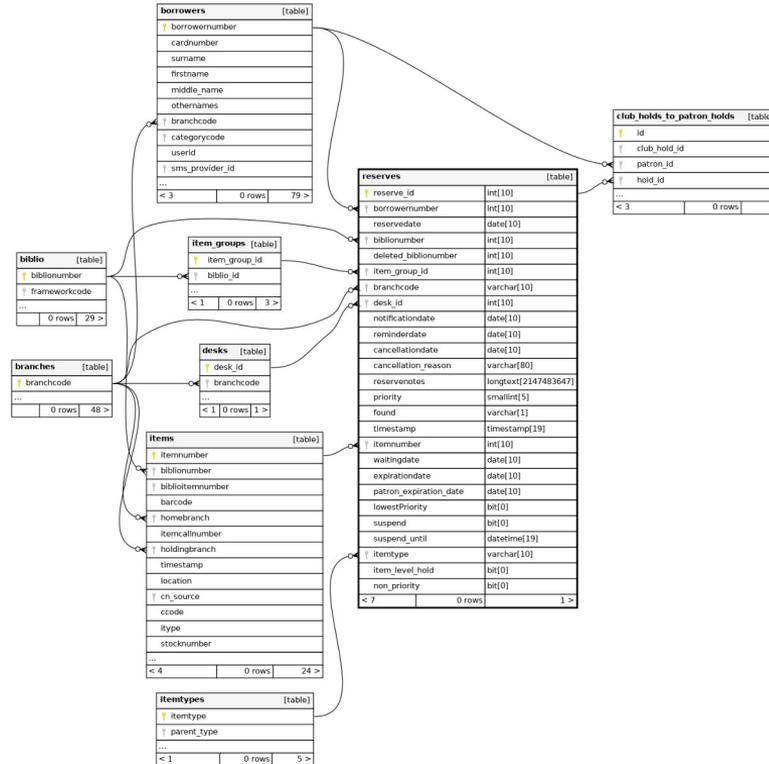
pas les bons résultats ou pas tous les résultats (c'est moins visible)

plantage de la base (la requête tourne sans fin, consomme trop de ressources et ralentit les autres actions -circulation, catalogage...-)

il n'y a aucun risque d'effacer des données ni d'en mélanger :

On (= les bibliothécaires) ne fait qu'interroger la base

Tables, colonnes, liaisons entre les tables



Permissions requises



Permet d'accéder au module Rapports (*reports*)

Cacher les détails

Créer des rapports SQL (*create_reports*)

Supprimer des rapports SQL (*delete_reports*)

Lancer les rapports SQL (*execute_reports*)

Interface du module Rapports

Rapports

Rapports

🔗 Créer un nouveau rapport guidé

+ Créer à partir de SQL

☰ Rapports sauvegardés

🔍 Rechercher par mot-clé

Recherche

Dictionnaire des rapports

- Voir dictionnaire

Assistant statistiques

- Acquisitions
- Adhérents
- Catalogue
- Circulation
- Périodiques
- Caisse
- Réservations

Plugins de rapport

- Plugins de rapport

Les palmarès

- Adhérents empruntant le plus
- Exemplaires les plus empruntés

Inactif

- Adhérents qui n'ont jamais emprunté
- Documents jamais prêtés

Autres

- Exemplaires perdus
- Commandes par poste budgétaire
- Catalogue par type de document
- Durées moyennes de prêt
- Schéma de la base de données Koha
- Bibliothèque de rapports Koha

Créer à partir de SQL

Créer un rapport à partir d'une requête SQL

Nom du rapport : *Obligatoire*

Groupe de rapports : Sélectionner (Aucun) ou créer : à Code à Nom

Rapport public : Non (défaut)

Expiration du cache : 300 Secondes (par défaut)

Notes :

SQL :

1

** Pour obtenir une saisie automatique pour les colonnes, merci de faire précéder le nom de la colonne du nom de la table, suivi d'un point. Par exemple : 'borrowers.surname'*
Obligatoire

Sauvegarder le rapport

Annuler

Classement des rapports sauvegardés

les onglets (Acquisition, Adhérents, Catalogue...) correspondent aux groupes du tableau
Groupes et sous-groupes sont des valeurs autorisées (REPORT_GROUP et REPORT_SUBGROUP)

Rapports sauvegardés

Tous Acquisitions Adhérents Autorités Bac à sable Bandeau Koha Bibliographies Bokeh Catalogue Circulation Encodage Journal de Caisse Politique documentaire Périodiques Recolement
Robot Réservations Statistiques Vérifications diverses

Sous-groupe : Tous

Affichage de 1 à 30 sur 464 entrées

Afficher entrées << Premier < Précédent Suivant > Dernier >> Rechercher: X Supprimer le filtre Colones Exporter Lien de copie partageable Configurer

	ID	Nom du rapport	Type	Groupes	Sous-groupe	Notes	Auteur	Date de création	Dernière modification	Dernière exécution	Public	URL JSON	Expiration du cache (en secondes)	Résultats sauvegardés	Mettre à jour	Actions
<input type="checkbox"/>	1	BIB - Nombre de notices bibliographiques					(1)	17/07/2017	17/07/2017 09:39	09/09/2025 12:20	Non	https://bm-pro.ville-caluire.fr/cgi-bin/koha/svc/report?id=1	300			▶ Exécuter -
<input type="checkbox"/>	2	BIB - Nombre de notices bibliographiques par type de document (0995t)					(1)	17/07/2017	24/01/2025 11:49	16/04/2025 16:21	Non	https://bm-pro.ville-caluire.fr/cgi-bin/koha/svc/report?id=2	0			▶ Exécuter -
<input type="checkbox"/>	3	BIB-Nombre de notices bibliographiques sans exemplaire					(1)	17/07/2017	17/07/2017 09:53	16/04/2025 16:23	Non	https://bm-pro.ville-caluire.fr/cgi-bin/koha/svc/report?id=3	300			▶ Exécuter -

Schéma de la base



Koha Community Database Schemas

- Koha version 25.05.x
 - Koha version 24.11.x
 - Koha version 24.05.x
 - Koha version 23.11.x
 - Koha version 23.05.x
 - Koha version 22.11.x
 - Koha version 22.05.x
 - Koha version 21.11.x
 - Koha version 21.05.x
 - Koha version 20.11.x
 - Koha version 20.05.x
 - Koha version 19.11.x
 - Koha version 19.05.x
 - Koha version 18.11.x
 - Koha version 18.05.x
 - Koha version 17.11.x
 - Koha version 17.05.x
 - Koha version 16.11.x
 - Koha version 16.05.x
 - Koha version 3.22.x
 - Koha version 3.20.x
 - Koha version 3.18.x
- Koha development version

Bibliothèque de rapports

Notices Sent

- **Developer:** Nicole C. Baratta, ByWater Solutions
- **Module:** Notices
- **Purpose:** Count of overdue notices sent in a specific time frame (by type). Uses the following codes for overdue messages: ODU1, ODU2, ODU3. Edit notice names as necessary. See [Notices Available](#) below.
- **Status:** Complete

```
SELECT monthname(message_queue.time_queued) AS month, year(message_queue.time_queued) AS year,
       message_queue.letter_code AS notice, count(message_queue.borrowernumber) AS count
FROM message_queue
WHERE message_queue.time_queued between <<Sent between (yyyy-mm-dd)|date>> and <<and (yyyy-mm-dd)|date>>
      AND message_queue.letter_code in ('ODUE1', 'ODUE2', 'ODUE3' )
      AND status = 'sent'
GROUP BY year(message_queue.time_queued), month(message_queue.time_queued), message_queue.letter_code
```

Overdue Notices Available

- **Developer:** Barton Chittenden, ByWater Solutions
- **Module:** Notices
- **Purpose:** Print a list of letter codes used in trigger table.
- **Status:** Complete

```
select
  distinct code
from
  letter
  inner join (
    select distinct letter1 as code from overduerules
    UNION select distinct letter2 as code from overduerules
    UNION select distinct letter3 as code from overduerules
  ) as overdue_trigger using (code)
```

Ressources



sur le SQL en général

[sql.sh](#) (*en français*)

<https://www.w3schools.com/sql/> (*en anglais*)

sur le SQL dans Koha : bibliothèques de rapports et aides

bibliothèque de rapports : https://wiki.koha-community.org/wiki/SQL_Reports_Library

site de l'association KohaLa : <https://koha-fr.org/category/sql-astuces/>

la liste de diffusion Koha fr

ManaKB

sur la structure de la base : le schéma (selon la version utilisée)

<https://schema.koha-community.org/>

Quelques tables



`biblio` et `biblioitems` : notices bibliographiques

`deletedbiblio` et `deletedbiblioitems` : notices bibliographiques supprimées

`items` : exemplaires

`deleteditems` : exemplaires supprimés

`auth_header` : notices autorités

`biblio_metadata` : contenu non mappé

`deletedbiblio_metadata` : contenu non mappé et supprimé

`issues` : prêts en cours

`reserves` : réservations en cours

`subscription` : notices d'abonnements des revues



borrowers : **fiches adhérents**

deletedborrowers : **fiches adhérents supprimées**

borrowers_attribute : **adhérents avec codes CSP, quartier..etc**

message_queue : **historique des relances et notifications envoyées**

accountlines : **factures et encaissements**

branches : **les différents sites**

authorised_values : **les valeurs autorisées**

virtualshelves, virtualshelfcontents **et** virtualshelfshares : **les listes**

Les champs mappés

Administration

Préférences système

Gestion des préférences système principales comme le format MARC, le format de la date, le courriel de l'administrateur, et les templates.

Rechercher

Conseil :

Configurer ces paramètres dans leur ordre d'apparition

Paramètres de base

[Tous les sites](#)

Catalogue

Grilles de catalogage MARC

Page définissant les caractéristiques de vos notices bibliographiques MARC (dénomination, sous-champs), ainsi que l'organisation de la grille

Liens Koha => MARC

Définition des correspondances entre la base de données relationnelle (SQL) de Koha et les notices bibliographiques MARC

Validation grilles de catalogage

Vérification de la structure MARC. Si vous modifiez les grilles de catalogage MARC, il est recommandé d'utiliser cet outil pour vérifier qu'il n'y a pas d'erreurs dans vos grilles

Types d'autorités

Champ Koha	Champ	Sous-champ	Lib
items.withdrawn_on			
items.withdrawn	995	3	Retiré de la circulation ?
items.uri	995	7	URL
items.timestamp			
items.stocknumber	995	j	Numéro inventaire
items.stack			
items.restricted	995	0	Exclu du prêt
items.reserves	995	x	Nombre de réservations

Liste les sous-champs mappés dans les tables biblio, biblioitems et items

Syntaxe élémentaire

SELECT ...

*= ce que je veux afficher
(des colonnes, des calculs sur le contenu des colonnes...)*

FROM ...

*= la source des données
(des tables)*

Interroger une table (afficher toutes les colonnes)

rédaction "classique", suffisante (mais moins lisible) :

```
select * from items limit 15
```

rédaction plus claire :

```
SELECT  
*  
  
FROM  
items  
  
LIMIT 15
```

lister
toutes les
colonnes
de la table

n'afficher que les 15 premiers résultats
(pour ne pas ralentir ni encombrer
l'affichage des résultats)

itemnumber	biblionumber	biblioitemnumber	barcode	dateaccessioned	booksellerid	homebranch	price	replacementprice	replaceme
2	2	2	1204227	2011-06-30	07	B	5.50	5.50	2017-12-1
3	2	2	1228421	2013-02-12	01	B	5.50	5.50	2017-12-1
4	3	3	1180147	2006-03-10	07	B	21.00	21.00	2017-12-1
6	5	5	1204283	2011-09-09	18	B	23.00	23.00	2017-12-1
7	6	6	1223740	2006-01-25	01	B	19.50	19.50	2017-12-1
9	8	8	1159308	2006-01-25	01	B	16.00	16.00	2017-12-1
10	9	9	1160087	2006-02-22	07	B	17.00	17.00	
11	10	10	1206642	2011-07-01	18	B	29.50	29.50	2017-12-1
14	13	13	1206683	2011-07-01	01	B	18.50	18.50	2017-12-1
17	16	16	1157589	2006-02-28	07	B	19.90	19.90	2017-12-1
19	18	18	1180076	2006-02-22	07	B	21.50	21.50	2017-12-1
22	21	21	1160565	2006-03-29	01	B	8.00	8.00	2017-12-1
24	23	23	1204212	2011-07-02	01	B	19.90	19.90	2017-12-1
25	24	24	1159356	2006-02-11	01	B	5.30	5.30	
26	25	25	1232319	2013-07-11	07	B	13.95	13.95	2017-12-1

Interroger une table (n'afficher que quelques colonnes)

- une `,` sépare les colonnes
- syntaxe de la colonne demandée : `table.colonne`

```
SELECT
items.barcode,
items.itemcallnumber

FROM
items

LIMIT 15
```

barcode	itemcallnumber
1204227	525.5 BEA
1228421	525.5 BEA
1160147	LV R CHA
1204283	180 JES
1223740	759.05 ING
1159308	759.05 DEG
1160087	P TAI
1206642	b3 HAE 35
1206683	321.07 REG
1157589	P BOU
1160076	P BRO
1160565	414 VAI U
1204212	292.08 SZA
1159356	155.24 GAL
1232319	BD D 7

Interroger une table (utiliser des alias)

déclarer un alias (dans le FROM) pour remplacer le nom de la table partout ailleurs dans la requête

```
SELECT
i.barcode,
i.itemcallnumber

FROM
items i

LIMIT 15
```

barcode	itemcallnumber
1204227	525.5 BEA
1228421	525.5 BEA
1160147	LV R CHA
1204283	180 JES
1223740	759.05 ING
1159308	759.05 DEG
1160087	P TAI
1206642	b3 HAE 35
1206683	321.07 REG
1157589	P BOU
1160076	P BRO
1160565	414 VAI U
1204212	292.08 SZA
1159356	155.24 GAL
1232319	BD D 7

Interroger une table (renommer des colonnes)

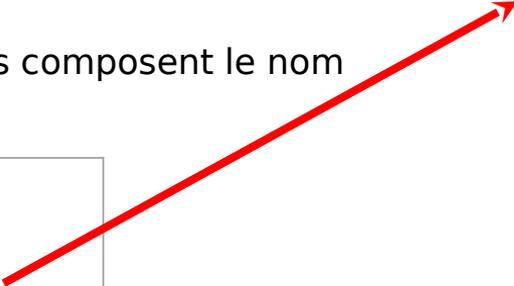
donner un nom aux colonnes affichées

les " " sont obligatoires si plusieurs mots composent le nom

```
SELECT
i.barcode AS "Code à barre",
i.itemcallnumber AS Cote

FROM
items i

LIMIT 15
```



Code à barre	Cote
1204227	525.5 BEA
1228421	525.5 BEA
1160147	LV R CHA
1204283	180 JES
1223740	759.05 ING
1159308	759.05 DEG
1160087	P TAI
1206642	b3 HAE 35
1206683	321.07 REG
1157589	P BOU
1160076	P BRO
1160565	414 VAI U
1204212	292.08 SZA
1159356	155.24 GAL
1232319	BD D 7

Associer plusieurs tables : les jointures



Si une table ne contient pas toutes les données nécessaires (à afficher ou à traiter), il faut utiliser plusieurs tables et les joindre.

Il existe plusieurs rédactions possible des jointures, selon les résultats qu'on souhaite obtenir.

Le principe de base est le même pour toutes ces formulations : il faut connaître le point commun entre les tables à interroger.

Ce point commun est tout simplement une des colonnes de chacune des tables à lier

Comment trouver cette colonne commune ?

en consultant le schéma de la base

biblio

1

Columns [↗](#)

Related columns Constraint Comments

Column	Type	Size	Nulls	Auto	Default	Children	Parents	Comments
bibliumber	3INT	10		√	null	aqorders article_requests biblio_metadata biblioimages biblioitems club_holds club_items club_reserves club_reviews club_suggestions club_tags_all club_tags_index club_virtualshelfcontents		unique identifier assigned to each bibliographic record

2

1. la table qui est présentée sur la page du schéma
2. la table qu'on souhaite lui lier
3. la colonne commune à ces deux tables

Comment rédiger une jointure ?

la considérer comme une condition de la requête et la déclarer dans le WHERE

```
SELECT
i.barcode AS "Code à barre",
i.itemcallnumber AS Cote,
b.title AS "Titre"

FROM
items i,
biblio b

WHERE
i.biblionumber = b.biblionumber

LIMIT 15
```

Code à barre	Cote	Titre
1204227	525.5 BEA	Saisons (Les)
1228421	525.5 BEA	Saisons (Les)
1160147	LV R CHA	Noces avec la vie
1204283	180 JES	Who you are
1223740	759.05 ING	Ingres
1159308	759.05 DEG	Degas
1160087	P TAI	Des morts qui dérangent
1206642	b3 HAE 35	Belshazzar, opéra
1206683	321.07 REG	Regards sur l'utopie
1157589	P BOU	Traces
1160076	P BRO	Jaloux
1160565	414 VAI U	Phonétique (La)
1204212	292.08 SZA	Feuilleton de Thésée (Le)
1159356	155.24 GAL	Psychologie de l'argent
1232319	BD D 7	Chasseurs de chimères

Affiner les résultats



Définir des critères pour affiner les résultats (et non limiter le nombre de résultats à l'affichage)

- opérateurs de comparaison
- concordance de caractères
- intervalles (BETWEEN)
- NULL
- listes

Combiner ces critères

AND, OR

Saisir les critères

entre guillemets si ce sont des chaînes de caractères

entre guillemets pour des dates (attention au format des dates)

sans guillemet pour des nombres

Critères de délimitation : opérateurs de comparaison



utilisation des opérateurs mathématiques

!= et <> fonctionnent avec des chaînes de caractères

Opérateur	Signification
=	égal à
>	strictement supérieur à
<	strictement inférieur à
>=	supérieur ou égal à
<=	inférieur ou égal à
!=	différent de
<>	différent de
!>	pas supérieur à
!<	pas inférieur à

Critères de délimitation : concordance de caractères



`LIKE "x"`

`NOT LIKE "x"`

```
WHERE items.itemcallnumber LIKE "84%"
```

utiliser les troncatures :

`_` : remplace un seul caractère

`%` : remplace un nombre indéfini de caractères (même aucun)

`LIKE "%A"`

`LIKE "A_"`

`LIKE "A%"`

`LIKE "A_B%"`

Critères de délimitation : intervalles



BETWEEN x AND y

NOT BETWEEN "x" AND "y"

```
WHERE items.price NOT BETWEEN 20 AND 30
```

les deux bornes sont comprises

fonctionne avec des caractères, des chiffres ou des dates !

Critères de délimitation : listes



`IN ("x", "y" ...)`

`NOT IN ("x", "y" ...)`

```
WHERE items.location IN ("ADU", "JEU", "AR")
```

équivalent à `("ADU" OR "JEU" OR "AR")`

ne permet pas d'utiliser les troncatures

Critères de délimitation : NULL



IS NULL

IS NOT NULL

```
WHERE items.itemcallnumber IS NULL
```

signifie : aucune valeur/ information disponible (i.e un sous-champ non renseigné)
ne signifie pas : valeur 0

Déclarer une variable

```
table.colonne = <<texte à afficher>>
```

```
table.colonne = <<texte à afficher|SOURCE>>
```

```
WHERE items.location = <<Localisation ?|LOC>>
```

Localisation ?:

Adulte



les sources ?

- une table de valeur autorisée (par son code) : |code>>
- une date : |date>>
- un type de prêt : |itemtypes>>
- un type d'abonnement : |categorycode>>

Déclarer une variable et “autoriser toutes les options”

```
table.colonne LIKE <<texte à afficher|SOURCE:all>>
```

dans le menu déroulant proposant les différents choix possibles, l’option “Tous” est ajoutée

```
WHERE items.location LIKE <<Localisation|LOC:all>>
```



A screenshot of a user interface element. It features a label 'Localisation:' followed by a dropdown menu. The dropdown menu is currently open, showing the selected option 'Tous' and a downward-pointing chevron icon on the right side of the menu box.

veiller à utiliser `LIKE` et taper `:all` en minuscules

Déclarer une variable : insérer un paramètre d'exécution

Créer à partir d'une requête SQL

Créer un rapport à partir d'une requête SQL

Nom du rapport : *Obligatoire*

Groupe de rapports : Sélectionner (Aucun) ou créer :

Rapport public : Non (défaut)

Expiration du cache : Secondes (par défaut)

Notes :

SQL :

1

Combiner les critères



```
WHERE
items.itemcallnumber LIKE "84%"

AND items.price BETWEEN "15" AND "20"

AND items.dateaccessioned <= <<Date de création de l'exemplaire|date>>

AND (items.location = "ADU" OR items.ccode = "IRP")
```

Je recherche les exemplaires :

ayant une cote commençant par 84

et ayant un prix d'achat entre 15 et 20

et ayant été créés avant la date à saisir lors de l'exécution de la requête

et ayant soit une localisation "ADU" soit la valeur "IRP" dans le champ mappé sous `ccode`

Troncature



`SUBSTRING_INDEX(colonne,"séparateur",1)`

n'affiche, des données de la colonne, que ce qui précède le 1er séparateur

`SUBSTRING(colonne,n)`

n'affiche les données de la colonne qu'à partir du n^{ième} caractère

```
SELECT
i.itemcallnumber AS "Cote",
substring_index(i.itemcallnumber,' ',1) AS "Cote tronquée au 1er espace",
substring_index(i.itemcallnumber,' ',2) AS "Cote tronquée au 2ème espace",
substring(i.itemcallnumber,1,1) AS "Seulement le 1er caractère de la cote",
substring(i.itemcallnumber,1,2) AS "Seulement les 2 premiers caractères de la cote"
```

Afficher les libellés des valeurs autorisées (plutôt que les codes)



possible par la rédaction d'une sous-requête

```
SELECT
i.barcode AS CB,
i.location AS Codes,
(SELECT
auval.lib
FROM
authorised_values auval
WHERE
auval.category='LOC'
and auval.authorised_value=i.location)
AS "Libellés"

FROM
items i
```

CB	Codes	Libellés
1204227	JEU	Jeunesse
1228421	JEU	Jeunesse
1160147	ADU	Adulte
1204283	MAS	Musique, Cinéma et Arts du spectacle
1223740	ADU	Adulte
1159308	ADU	Adulte
1160087	AR	Réserve Adulte
1206642	MR	Réserve Musique, Cinéma et Arts du spectacle
1206683	ADU	Adulte

Insérer un commentaire



```
/* texte du  
commentaire */
```

Permet de rédiger du texte pour expliquer ou commenter une requête

Trier ses résultats

ORDER BY

permet de trier les résultats affichés selon un ou plusieurs critères, croissant ou décroissant

```
ORDER BY Cote DESC, Localisation, CB DESC
```

triera les résultats par cote (ordre décroissant),
puis par localisation (ordre croissant),
et enfin par CB (ordre décroissant)

SELECT gère l'ordre d'affichage des colonnes
ORDER BY gère l'ordre d'affichage dans les colonnes du table

CB	Localisation	Cote
1235525	MAS	XLv1 A.
1230963	MAS	XL v307 PRO
1229129	MAS	XL v307 MIN
1179607	MR	XL v307 JER
1230946	MAS	XL v307 ESP
1221005	MR	XL v307 EDI
1162162	MAS	XL v307 CEN
1108789	MR	XL v307 25Y
1204079	MAS	XL v3 XEN 10
1116488	MR	XI v3 WFR

Comme LIMIT, ORDER BY se rédige en fin de requête et n'est pas classée dans le SELECT

Conseils



Quand on se retrouve face à une interrogation compliquée à *traduire* en SQL, on peut tester les difficultés une à une et vérifier la bonne marche de la requête à chaque ajout.

Pour vérifier que le rendu est juste, ne pas hésiter à utiliser les filtres d'un tableur ou une recherche documentaire équivalente et comparer les résultats.

Faire travailler son imagination et sa curiosité.

(ne pas forcément attendre qu'un besoin se présente pour imaginer comment y répondre)