

MySQL

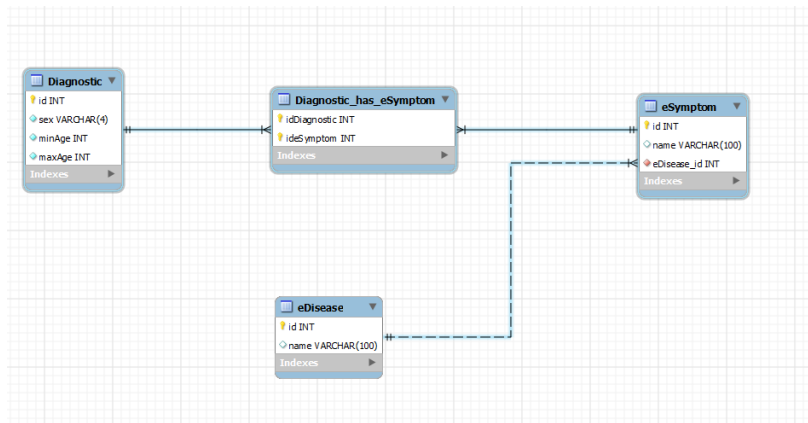
MySQL est un SGBDR apparu en 1995, développé par David Axmark et Michael Widenius. Le système de gestion est passé en licence GPL en 2000. Depuis, il a été racheté par SUN en 2008, qui fut racheté par Oracle en 2009. A l'heure actuelle, c'est Oracle qui distribue MySQL, sous une double licence GPL et Propriétaire, ainsi que son propre SGBDR. MySQL est l'un des logiciels SGBDR les plus utilisés au monde, aussi bien parmi les particuliers que parmi les entreprises.

Pour le moment, MySQL est toujours développé par Oracle, qui y voit un produit d'appel vers son Data Base, certes payant, mais proposant plus de fonctionnalités... et un très bon outil d'apprentissage du SQL.

Base de données

La base de données, ou BDD, est un moyen simple d'organiser des données. Les informations sont enregistrées sous forme structurée et organisée. Les bases de données sauvegardent les données utiles à une application, ce que ne peuvent pas faire les variables. De plus, ces données sont manipulables à travers des requêtes, alors qu'un fichier ne peut être que simplement lu. Dans la pratique la majorité des applications utilisent des BDD pour la persistance des données.

Voici un exemple de base de données :



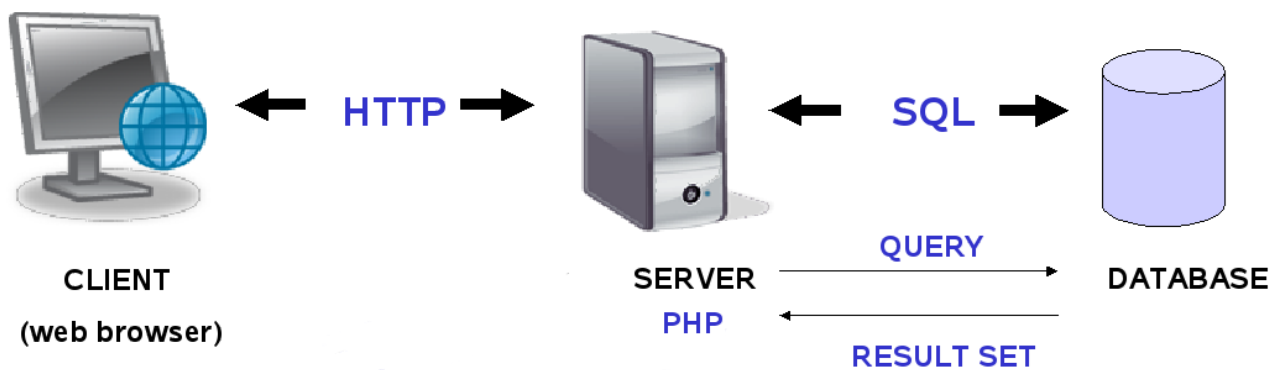
SQL

La base de données est modifiable, à condition d'utiliser le bon langage, SQL en général. Le SQL, ou Structured Query Language, est un langage informatique normalisé, depuis 1986, dédié aux bases de données. C'est le langage le plus utilisé par les SGBD, car il permet la définition des données et des bases, leurs manipulations, ainsi que le contrôle des transactions et des données. Enfin, le SQL intègre l'algèbre relationnel.

Donc, à chaque fois qu'il faut manipuler une BDD, il faut écrire une requête en SQL, comme dans cet exemple :

```
INSERT INTO Utilisateur VALUES (null, 'Lannister', 'Tyrion', '1991');
```

La plupart des langages de programmation, comme le PHP, sont capables de générer de telles requêtes.



SGBD

Un système de gestion de base de données, ou SGBD, est un logiciel utilisé pour gérer des bases de données. Il permet non seulement la création et la suppression de BDD, mais aussi la manipulation et la mise en forme des données ainsi que le déploiement et l'exploitation de la base. En général, ces logiciels proposent une surcouche et des interfaces simplifiant toutes ces opérations. Oracle, MySQL, SQLite, SQL Server sont des SGBD.

Un SGBDR est un système de gestion de base de données qui implémente le modèle relationnel, qui consiste à présenter les données sous forme de tableaux, avec en entête des attributs, et des enregistrements comme ligne. Ces tables, ou relations, sont manipulées grâce à un algèbre relationnel, qui permet, entre autre, des jointures, des sélections, des intersections...

Les SGBD intègrent des mécanismes pour sécuriser leurs bases. La plupart d'entre-eux propose une gestion des droits des utilisateurs sur les BDD, afin de contrôler son accès. De plus, ces logiciels sont garants de la cohérence des données, respectant les règles élémentaires de manipulation de bases de données (comme la règle d'unicité ou les règles d'intégrité référentielle).

Structure d'une base de données

Une base de données contient plusieurs tables. Chaque table contient des informations sous forme d'enregistrements (les lignes de la tables). Ces enregistrements sont décomposés en différentes données, ou champ, qui sont les colonnes de la table.

Par exemple, une table qui contient des enregistrements sur des clients :

id	nom	prenom	adresse
1	Lannister	Tyrion	Port-Réal
2	Daenerys	Targaryen	Essos
3	Jon	Snow	Winterfell

Il est recommandé de toujours ajouter un identifiant aux enregistrements pour les rendre uniques.

Concrètement, les données d'une base de données sont enregistrées dans des fichiers. MySQL les stocke, par exemple, dans un répertoire `\data`. On ne manipule jamais ces fichiers directement, c'est le travail du SGBD. Toutes les données sont bien manipulées grâce aux requêtes SQL. Les tables sont une représentation de ces données assez proche de la façon de les manipuler.

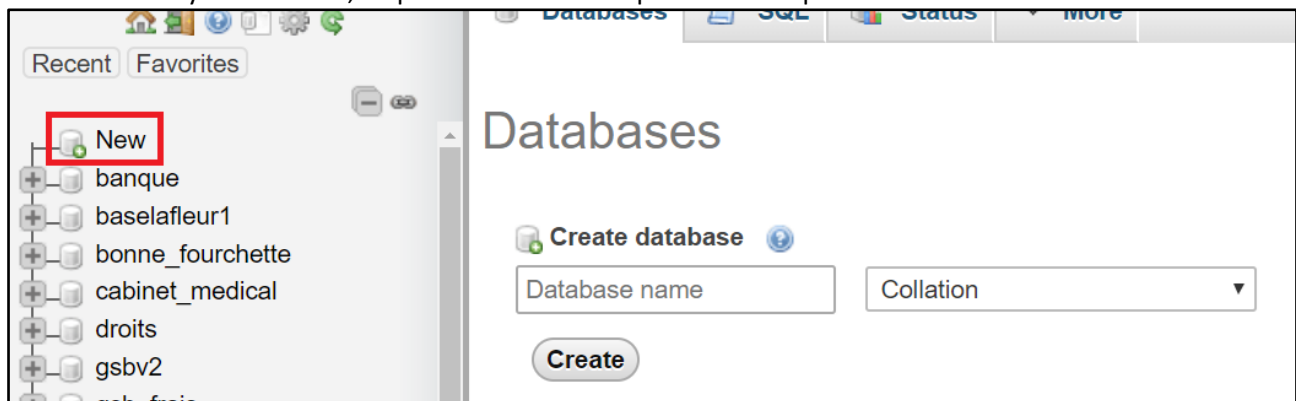
PHPMyAdmin

PHPMyAdmin est un outil de manipulation de base de donnée MySQL, développé en PHP, et utilisant directement le SQL pour gérer les données. Il existe d'autres façons de travailler avec des bases de données, comme les lignes de commandes ou le langage PHP. Mais PHPMyAdmin à l'avantage de proposer une interface graphique pour visualiser les données et permet de nombreuses manipulations sans passer par les requêtes.

PHPMyAdmin est livré avec la plupart des serveurs LAMP.

Créer une base de données

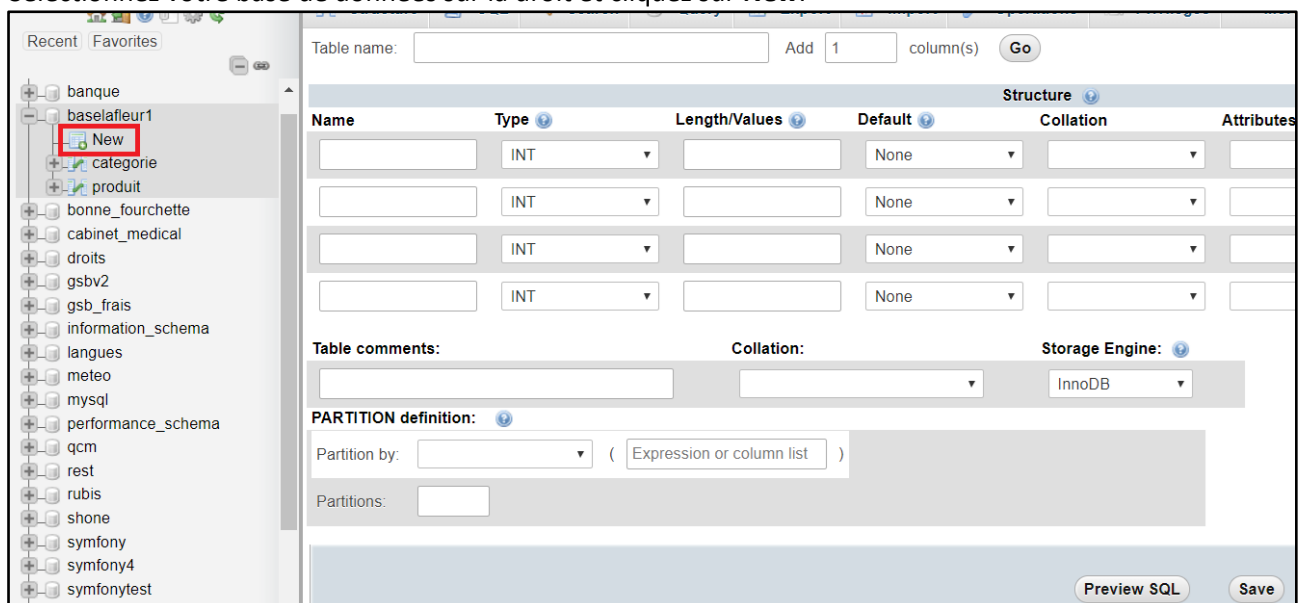
Une fois PHPMyAdmin lancé, cliquez sur **New** et remplissez le champ **Database name**.



A noter que le nom d'une base de donnée est non case sensitive. En général, pour le nom d'une base, on se contente des caractères alphanumériques. La **Collation** vous permet de choisir l'encodage de la base. En Europe, *utf8_general_ci* est recommandé. Puis bouton **Create**.

Ajouter une table

Sélectionnez votre base de données sur la droite et cliquez sur **New**.



Donnez un nom à votre table et définissez les colonnes en les nommant et en les typant. Les types les plus courants sont :

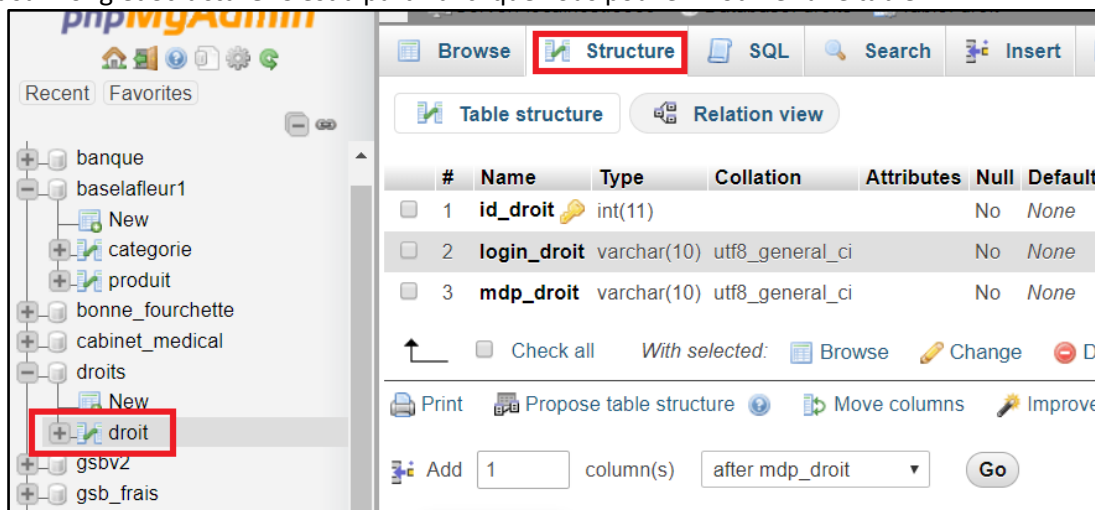
- **INT** : nombre entier,
- **VARCHAR** : texte court (entre 1 et 255 caractères),
- **TEXT** : texte long,
- **DATE**: pour des dates (peut comprendre des heures).

Il est recommandé de toujours commencé par la colonne *id*, qui va rendre vos enregistrements uniques, ce qui va simplifier de nombreuses recherches. En général, cette première colonne est définie comme clé primaire (**PRIMARY** dans **Index**) et auto-incrémentée (coche **I_A**) pour que MySQL se charge de calculer sa valeur.

A noter qu'un **VARCHAR** doit forcément avoir une taille (**Length/Values**).

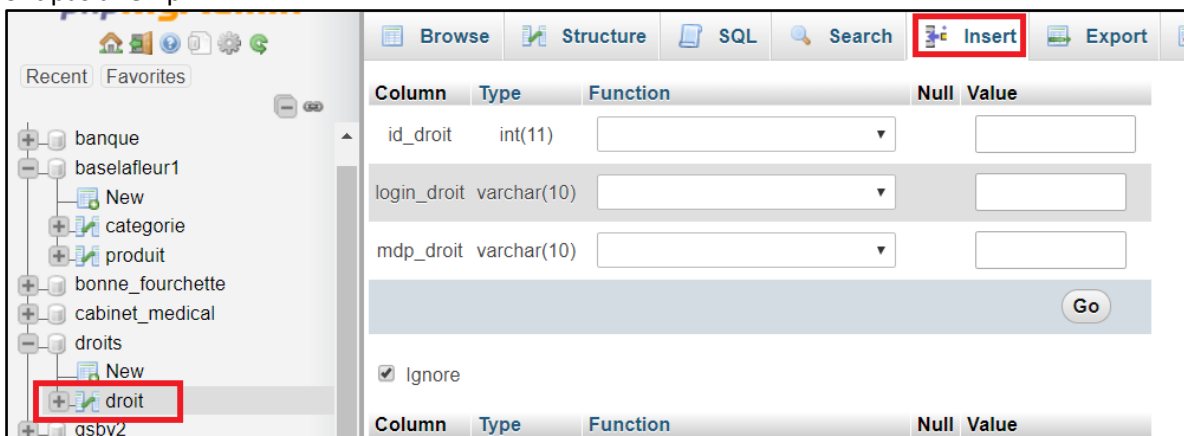
Une fois toutes les colonnes paramétrées, vous pouvez enregistrer la table (**Save**).

La table créée apparaît sous la base et vous pouvez visualiser son paramétrage en la sélectionnant et en cliquant sur l'onglet **Structure**. C'est à partir d'ici que vous pouvez modifier une table.



Remplir une table

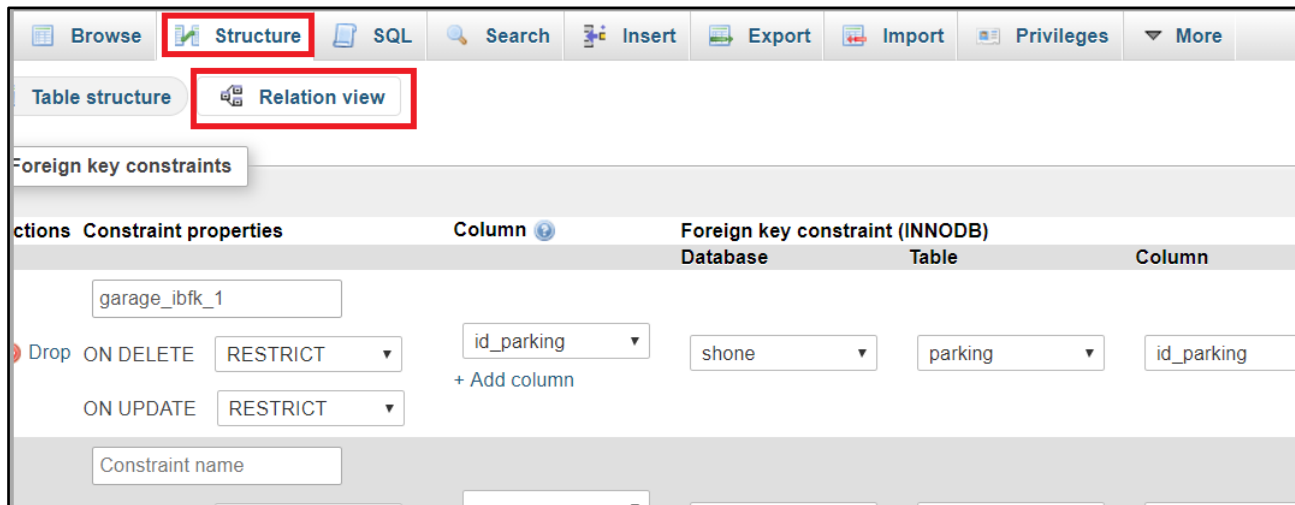
Il faut maintenant ajouter des enregistrements. Pour ça, sélectionnez la table à remplir, sur la droite, et cliquez sur l'onglet **Insert**. Saisissez les valeurs à insérer dans les champs et **Go**. Les champs auto-incrémentés ne sont pas à remplir.



Une fois la table remplie, vous pouvez tout visualiser à partir de l'onglet **Browse**.

Ajouter une clé étrangère

Il est possible de lier des tables entre-elles, grâce aux clés étrangères, si la table est gérée par un moteur InnoDB. Pour ajouter une clé étrangère, retournez sur l'onglet **Structure** de votre table et cliquez sur le bouton **Relation view**.



Donnez le nom de la colonne devant contenir la clé étrangère, la table sur laquelle il faut la lier et son id, puis **Save**.

Moteurs

MySQL propose plusieurs moteurs de stockage. Les deux plus connus sont :

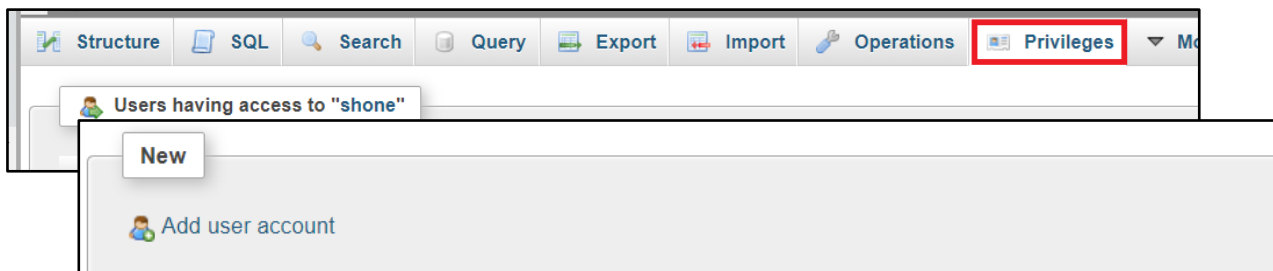
- **MyISAM** : moteur par défaut, dont les commandes d'insertion et de sélection sont rapides. Mais il ne gère pas certaines fonctionnalités comme les clés étrangères et les transactions.
- **InnoDB** : plus lent et plus gourmand que le précédent, il a l'avantage de gérer les clés étrangères. De plus en cas de crash du server, le moteur possède un système de récupération des données.

Gestion de droits

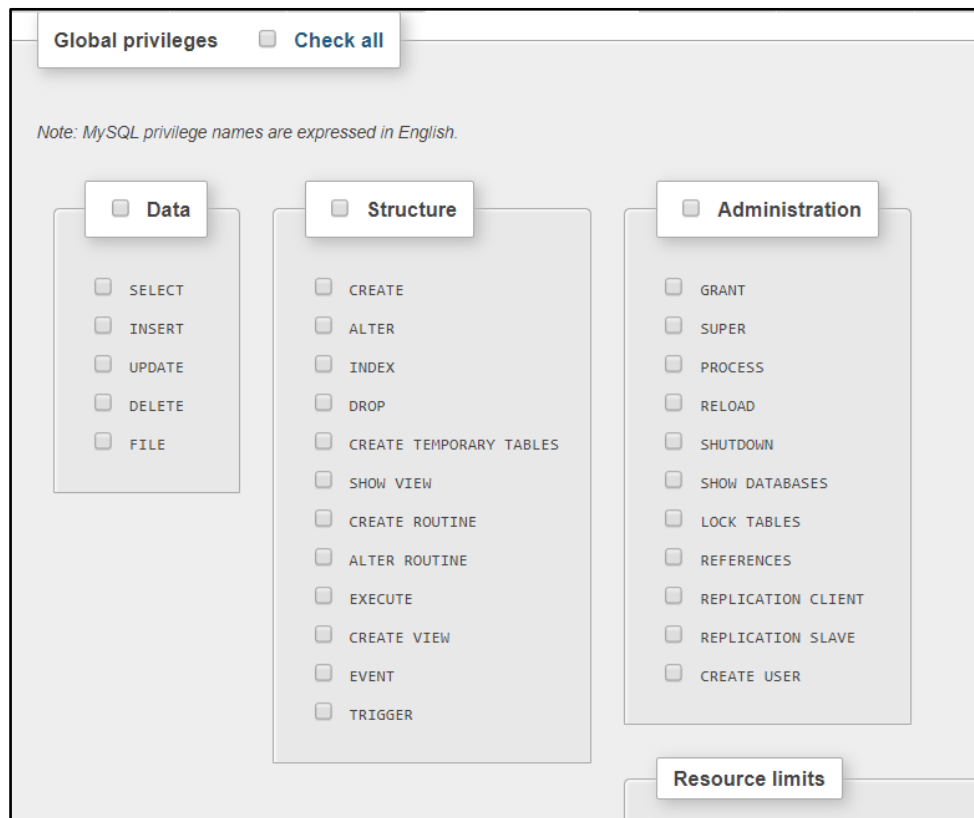
Par défaut, vous êtes connecté, à travers PHPMyAdmin, avec le compte *Root*. Ce profil vous donne tous les droits sur toutes les bases et même sur le SGBD, ce qui n'est pas très sécurisé.

Il est préférable, pour une base, de définir des droits plus restreints, surtout lorsqu'elle doit être exploitée à partir d'une appli. En effet, pour que l'appli puisse accéder à la BDD, il faut qu'elle se connecte avec un compte utilisateur.

Pour créer un nouveau compte, sélectionnez une base, et cliquez sur l'onglet **Privilege** et puis sur le lien **Add user account**.

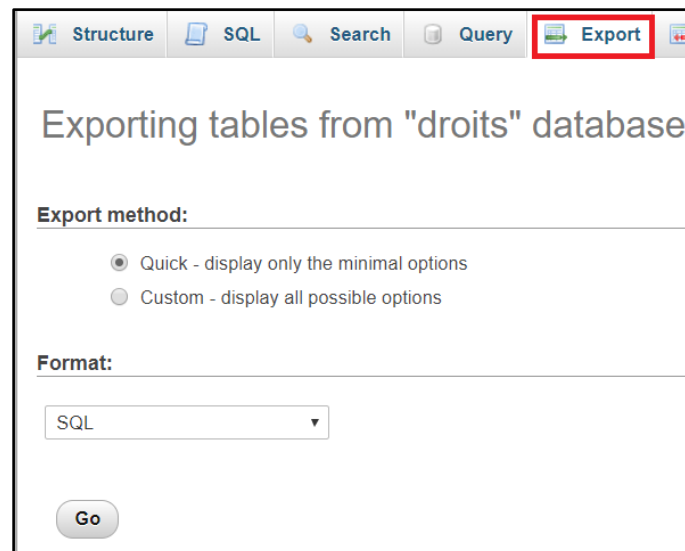


Dans la nouvelle fenêtre, saisir un **User name** et un **Password** puis ne donner que des droits sur la **Data**.



Exporter une base de données

A partir de l'onglet **Export**, vous pouvez exporter votre base sous forme d'un fichier SQL. Ce fichier contient toutes les requêtes pour recréer la même base et les remplir avec les données existantes. Ce fichier permet donc de reconstruire une base identique sur une autre machine (un serveur, par exemple) et de faire des sauvegardes.



Pensez à bien sélectionner la base à exporter avant de cliquer sur l'onglet **Export** et conservez le format SQL. Il ne vous reste plus qu'à cliquer sur **Go**.

Importer du SQL

Il est aussi possible d'importer un fichier SQL (voir d'autres formats compatibles avec la BDD comme le CSV).

Commencez par sélectionner la base sur laquelle exécuter les requêtes et onglet **Import**. **Choisir un fichier** et **Go**. Cet import vous permet d'exécuter un grand nombre de requêtes préparées dans un fichier ou de reconstruire une base à partir d'un fichier d'export.

The screenshot shows the 'Import' tab in PHPMyAdmin. The 'Import' tab is highlighted with a red box. Below the tabs, there is a text input field for the file name, followed by a 'Choisir un fichier' button, also highlighted with a red box. The text indicates that the file name must end in a specific format and provides an example: '.sql.zip'. Below this, there is a 'Browse your computer' button and a text field showing 'Aucun fichier choisi' with a maximum size of 2,048KiB. A note states: 'You may also drag and drop a file on any page.' Below this is a 'Character set of the file:' dropdown menu set to 'utf-8'. The 'Partial import:' section contains a checked checkbox for 'Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. (This is useful for large files, however it can break transactions.)' and a text input field for 'Skip this number of queries (for SQL) starting from the first one:' set to '0'. The 'Other options:' section has a checked checkbox for 'Enable foreign key checks'. The 'Format:' section has a dropdown menu set to 'SQL'. The 'Format-specific options:' section has a 'SQL compatibility mode:' dropdown menu set to 'NONE'.

Visualisation de la base

Il est possible de faire des vérifications sur la modélisation des données en base en affichant la BDD sous forme d'un schéma relationnel. Pour ça, choisissez votre base et cliquez sur le menu Designer.

