

## phpMyAdmin (1/X)

Il existe plusieurs façons d'accéder à sa base de données et d'y faire des modifications. On peut utiliser une ligne de commande (console), exécuter les requêtes en PHP ou faire appel à un programme qui nous permet d'avoir rapidement une vue d'ensemble.

phpMyAdmin, un des outils les plus connus permettant de manipuler une base de données MySQL.

Presque tous les hébergeurs permettent d'utiliser phpMyAdmin ; renseignez-vous auprès du vôtre pour savoir comment y accéder.

## phpMyAdmin (2/X)

### Ouvrir Wamp

Ouvrir phpMyAdmin sous wamp:

1. démarrez WAMP,
2. clic gauche sur l'icône de la barre des tâches et allez dans «phpMyAdmin ».

### Ouvrir Mamp

1. <http://localhost/MAMP/>

### Ouvrir Lamp

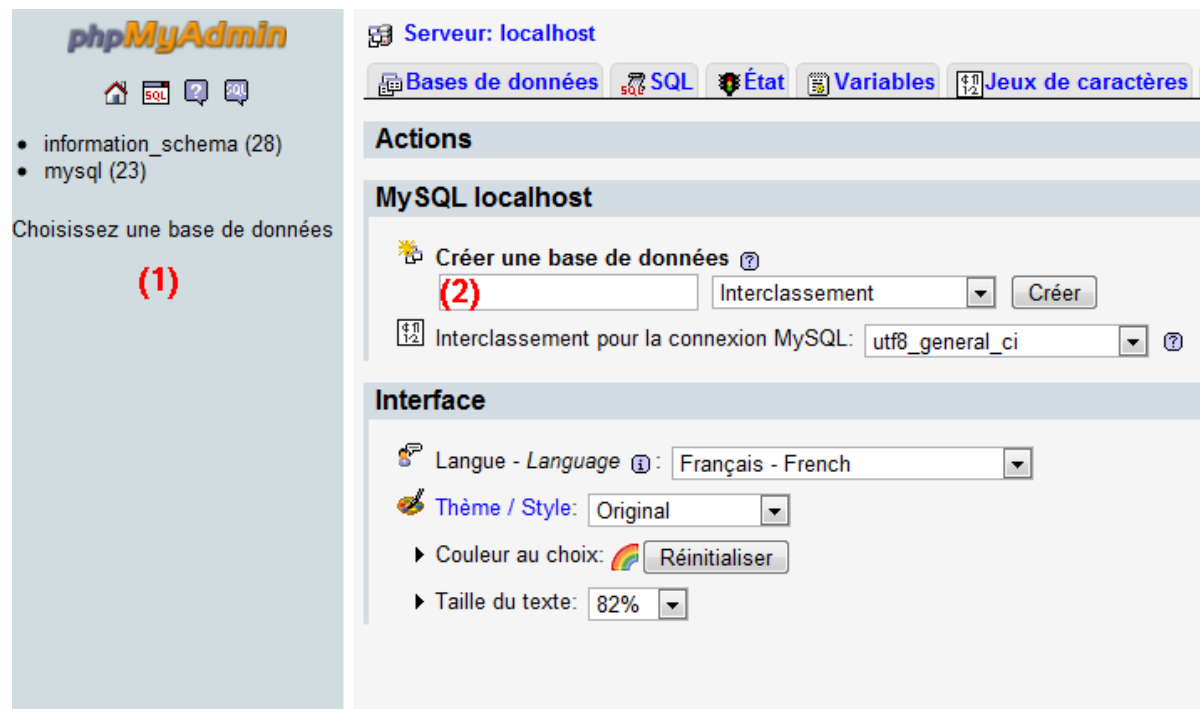
<http://localhost/phpmyadmin>

# phpMyAdmin (3/X)

## Créer une table

phpMyAdmin n'est pas un **programme** mais un **ensemble de pages PHP** toutes prêtes dont on se sert pour gagner du temps.

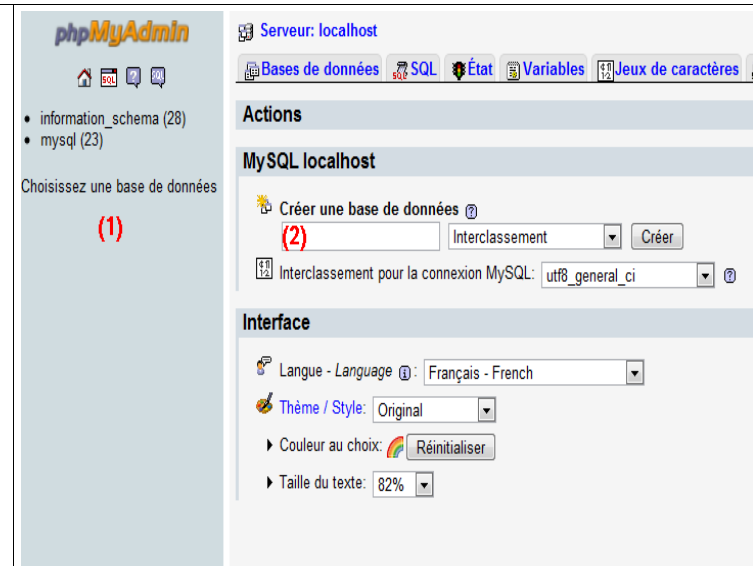
L'accueil de phpMyAdmin ressemble à la figure suivante.



**Liste des bases:** c'est la liste de vos bases de données. Le nombre entre parenthèses est le nombre de tables qu'il y a dans la base.

Sur la capture d'écran, on a donc deux bases :information\_schema, qui contient 28 tables, etmysql, qui en contient 23.

**Créer une base :** pour créer une nouvelle base de données, entrez un nom dans le champ de formulaire à droite, cliquez sur « Créer » et hop ! c'est fait.



**N.B:** Pour le moment, deux bases existent déjà :information\_schemaetmysql. N'y touchez pas, elles servent au fonctionnement interne de MySQL.

## phpMyAdmin (5/X)

### Créer une table

Créer une nouvelle base de données dans laquelle nous travaillerons tout le temps par la suite.

- ⑩ Utilisez le formulaire à droite pour créer cette base : entrez le nom `test` et cliquez sur le bouton `Créer`.
- ⑩ L'écran de la figure suivante devrait alors s'afficher si la base a bien été créée.

The screenshot displays the phpMyAdmin interface. On the left sidebar, the 'Base de données' (Databases) section shows a dropdown menu with 'test' selected. Below it, it indicates 'test (0)' and 'Aucune table n'a été trouvée dans cette base.' (No table was found in this database).

The main content area on the right shows the 'Base de données: test' (Database: test) header. Below the header, there is a green success message: '✓ La base de données test a été créée.' (The database test has been created). Below this message, the SQL command 'CREATE DATABASE `test` ;' is displayed, along with links for '[ Modifier ]' and '[ Créer source PHP ]'.

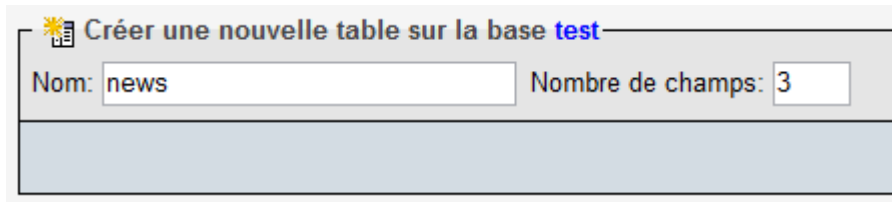
At the bottom of the main content area, there is a section titled 'Créer une nouvelle table sur la base test' (Create a new table on the database test). It includes a form with 'Nom:' (Name) and 'Nombre de champs:' (Number of fields) input fields, and an 'Exécuter' (Execute) button.

## phpMyAdmin (6/X)

On vous indique qu'aucune table n'a été trouvée dans la base.

### Créer une table

Dans le champ « Créer une nouvelle table sur la base test », entrez le nom `news` et le nombre de champs `3`, comme vous le montre la figure suivante.

The image shows a screenshot of the 'Créer une nouvelle table' (Create new table) form in phpMyAdmin. The form is titled 'Créer une nouvelle table sur la base test' with a yellow star icon. It contains two input fields: 'Nom:' with the value 'news' and 'Nombre de champs:' with the value '3'. Below these fields is a large, empty light blue rectangular area, likely for defining the table's structure or indexes.

Cliquez sur « Exécuter ».

La table n'est pas immédiatement créée :

➔ Pour cette table, on va créer les trois champs suivants.

- **id** : comme bien souvent, vous allez devoir créer un champ appelé id.
- **titre** : ce champ contiendra le titre de la news.
- **contenu** : enfin, ce champ contiendra la news elle-même.

Champ	id	titre	contenu
Type ?	INT	VARCHAR	TEXT
Taille/Valeurs*1		255	
Défaut <sup>2</sup>	Aucun	Aucun	Aucun
Interclassement			
Attributs			
Null	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Index	PRIMARY	---	---
AUTO_INCREMENT	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Commentaires			

phpMyAdmin vous demande beaucoup d'informations, il n'est pas nécessaire de tout remplir. La plupart du temps, les sections les plus intéressantes seront :

- **Champ** : permet de définir le nom du champ (très important !) ;
- **Type** : le type de données que va stocker le champ (nombre entier, texte, date...) ;
- **Taille/Valeurs** : permet d'indiquer la taille maximale du champ, utile pour le type VARCHAR notamment, afin de limiter le nombre de caractères autorisés ;
- **Index** : active l'indexation du champ. Ce mot barbare signifie dans les grandes lignes que votre champ sera adapté aux recherches. Le plus souvent, on utilise l'index PRIMARY sur les champs de type id ;
- **AUTO\_INCREMENT** : permet au champ de s'incrémenter tout seul à chaque nouvelle entrée. On l'utilise fréquemment sur les champs de type id. [Suivre](#)



On va remplir le formulaire comme sur la figure suivant. Veuillez à bien cocher **AUTO\_INCREMENT** et à définir un index **PRIMARY** sur le champ id.

Champ	id	titre	contenu
Type ?	INT	VARCHAR	TEXT
Taille/Valeurs <sup>*1</sup>		255	
Défaut <sup>2</sup>	Aucun	Aucun	Aucun
Interclassement			
Attributs			
Null	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Index	PRIMARY	---	---
AUTO_INCREMENT	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Commentaires			

Une fois que c'est fait, cliquez sur le bouton Sauvegarder. Votre table est créée !

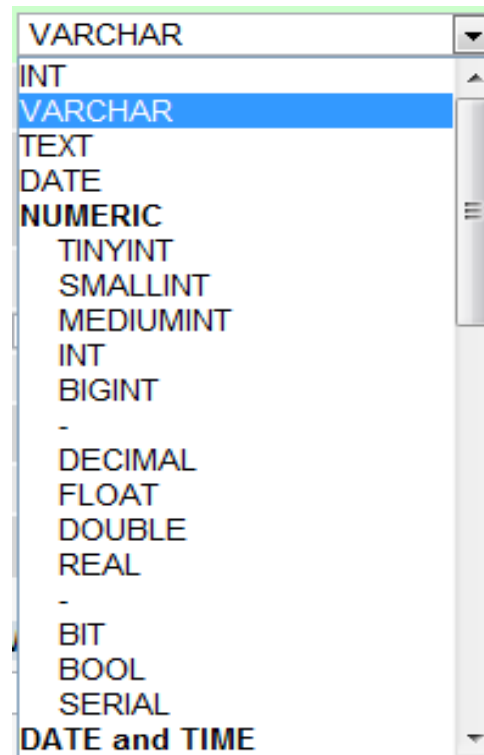
## phpMyAdmin (10/X)

Avant d'aller plus loin, on va revenir un peu plus en détail sur :

- les types de champs;
- les index, notamment l'index PRIMARY.

# phpMyAdmin (10/X)

## Les types de champs MySQL



Alors que PHP ne propose que quelques types de données que l'on connaît bien maintenant (int,string,bool...), MySQL propose une quantité très importante de types de données.

En fait, ceux-ci sont classés par catégories.

- **NUMERIC** : ce sont les nombres. On y trouve des types dédiés aux petits nombres entiers (TINYINT), aux gros nombres entiers (BIGINT), aux nombres décimaux, etc.
- **DATE and TIME** : ce sont les dates et les heures. De nombreux types différents permettent de stocker une date, une heure, ou les deux à la fois.
- **STRING** : ce sont les chaînes de caractères. Là encore, il y a des types adaptés à toutes les tailles.

## phpMyAdmin (12/X)

- . SPATIAL : cela concerne les bases de données spatiales, utiles pour ceux qui font de la cartographie. Ce ne sera pas notre cas, donc nous n'en parlerons pas ici.

En fait, phpMyAdmin propose au tout début de cette liste les quatre types de données les plus courants :

- . INT : nombre entier ;
- . VARCHAR : texte court (entre 1 et 255 caractères) ;
- . TEXT : long texte (on peut y stocker un roman sans problème) ;
- . DATE : date (jour, mois, année)

## Les clés primaires

Toute table doit posséder une clé primaire. La clé primaire permet d'identifier de manière unique une entrée dans la table.

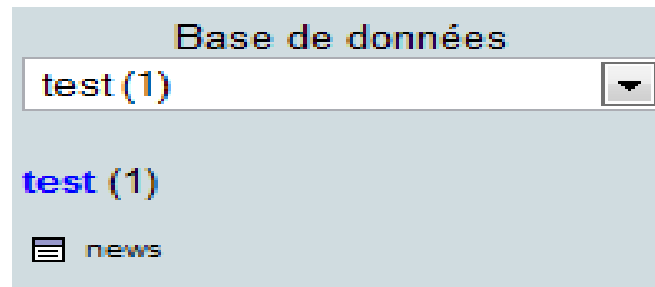
Dans un champ nommé « id ». **Il ne peut pas y avoir deux news avec le même id.**

**Il est vital** que chaque table possède sa clé primaire. On ne vous interdira pas de créer des tables sans clé primaire, mais leurs performances seront extrêmement réduites.

**Il faut prendre** le réflexe de créer à chaque fois ce champ « id » en lui donnant l'index PRIMARY, ce qui aura pour effet d'en faire une clé primaire. Vous en profiterez en général pour cocher la case `AUTO_INCREMENT` afin que ce champ gère lui-même les nouvelles valeurs automatiquement (1, 2, 3, 4...).

## Modifier une table

À gauche de votre écran, la table « news » que vous venez de créer devient visible, telle que vous la voyez sur la figure suivante.



- Si vous cliquez sur le mot « news », le contenu de la table s'affiche à droite de l'écran.
- Si vous cliquez sur la petite image de tableau à gauche, phpMyAdmin vous présentera la structure de la table.

Actuellement, comme la table est vide (elle ne contient aucune entrée), c'est la structure de la table (figure suivante) qui s'affichera.

Serveur: localhost ▶ Base de données: test ▶ Table: news

[Afficher](#) [Structure](#) [SQL](#) [Rechercher](#) [Insérer](#) [Exporter](#) [Importer](#) [Opérations](#) [Vider](#) [Supprimer](#)

	Champ	Type	Interclassement	Attributs	Null	Défaut	Extra	Action
<input type="checkbox"/>	<u>id</u>	int(11)			Non	Aucun	auto_increment	      
<input type="checkbox"/>	titre	varchar(255)	latin1_swedish_ci		Non	Aucun		      
<input type="checkbox"/>	contenu	text	latin1_swedish_ci		Non	Aucun		      

↑ [Tout cocher / Tout décocher](#) Pour la sélection :       

---

[Version imprimable](#) [Suggérer des optimisations quant à la structure de la table ?](#)

[Ajouter](#)  champ(s) ☒ En fin de table ☐ En début de table ☐ Après   [Exécuter](#)



Il est possible d'ajouter ou de supprimer des champs à tout moment. Vous avez des options pour renommer les champs, les supprimer, en ajouter, etc.










Onglets du haut : « Structure », « Afficher », « SQL », etc. Nous allons commencer par nous intéresser à l'onglet « Insérer », qui va nous permettre d'ajouter des entrées à la table.-> vous pouvez entrer des valeurs pour chacun des champs : c'est l'occasion d'insérer notre première news, comme le suggère la figure suivante.

Champ	Type	Fonction	Null	Valeur
id	int(11)	<input type="text"/>		<input type="text"/>
titre	varchar(255)	<input type="text"/>		<input type="text" value="Ma première news"/>
contenu	text	<input type="text"/>		<div><p>Vous êtes en train de lire ma première news. Bravo !</p></div>

☒ Ignorer

Champ	Type	Fonction	Null	Valeur
id	int(11)	<input type="text"/>		<input type="text"/>
titre	varchar(255)	<input type="text"/>		<input type="text"/>

Affichez le contenu de la table: vous pouvez cliquer soit sur l'onglet « Afficher », en haut, soit sur le nom de la table dans le menu à gauche. La figure suivante vous présente le contenu que vous devriez voir.

			id	titre	contenu
			1	Ma première news	Vous êtes en train de lire ma première news. Bravo...
			2	Autre news	Ceci est une autre news !
			3	Exclusif !	Ceci est une news !

Doit-on passer par phpMyAdmin à chaque fois que je veux ajouter ou supprimer un élément?

➔ La réponse est Non, phpMyAdmin est un outil d'administration. Il permet de voir rapidement la structure et le contenu de vos tables. Il est aussi possible d'ajouter ou de supprimer des éléments, comme on vient de le voir, mais on ne le fera que très rarement. Nous apprendrons à créer des pages en PHP qui insèrent ou suppriment des éléments directement depuis notre site web.

Par la suite nous découvrirons quelques-unes des fonctionnalités offertes par phpMyAdmin et nous aurons terminé notre tour d'horizon de cet outil.

## Autres opérations

Nous avons jusqu'ici découvert le rôle de trois onglets :

- *Afficher* : affiche le contenu de la table ;
- *Structure* : présente la structure de la table (liste des champs) ;
- *Insérer* : permet d'insérer de nouvelles entrées dans la table.

Par la suite nous présentons six autres onglets :

- *SQL* ;
- *Importer* ;
- *Exporter* ;
- *Opérations* ;
- *Vider* ;
- *Supprimer*.

Onglet SQL : Cliquez sur l'onglet « SQL »

Exécuter une ou des requêtes sur la base test: ?

```
SELECT * FROM `news` WHERE 1
```

Champs

- id
- titre
- contenu

<<

[ Délimiteur : ] ☒ Réafficher la requête après exécution Exécuter

Cela signifie: « **Afficher tout le contenu de la table 'news'** »

Onglet Importer: Dans la page qui s'affiche, vous pouvez envoyer un fichier de requêtes SQL (généralement un fichier.sql) à MySQL pour qu'il les exécute.

Fichier à importer

Emplacement du fichier texte   (Taille maximum: 2 048 Kio)  
Jeu de caractères du fichier:   
Ces modes de compression seront détectés automatiquement : aucune, gzip, zip

Importation partielle

☒ Permettre l'interruption de l'importation si la limite de temps est sur le point d'être atteinte. Ceci pourrait aider à importer des fichiers volumineux, au détriment du respect des transactions.  
Nombre d'enregistrements (requêtes) à ignorer à partir du début

Format du fichier d'importation

☐ CSV  
☐ CSV via LOAD DATA  
☒ SQL

Options

Mode de compatibilité SQL

Exécuter

## Onglet SQL - Suite-

➔ Seul le premier champ en haut devrait nous intéresser : il nous permet d'indiquer un fichier sur notre disque dur contenant des requêtes SQL à exécuter.

Cliquez ensuite sur le bouton « Exécuter » tout en bas sans vous préoccuper des autres champs.

➔ Quelle différence y a-t-il entre écrire la requête SQL (comme on vient de le voir juste avant) et envoyer un fichier contenant des requêtes SQL ? C'est la même chose, sauf que parfois quand on doit envoyer un très grand nombre de requêtes, il est plus pratique d'utiliser un fichier.

**Onglet Exporter:** c'est ici que vous allez pouvoir récupérer votre base de données sur le disque dur sous forme de fichier texte.sql (qui contiendra des tonnes de requêtes SQL).

Le rôle de ce fichier : c'est un fichier qui dit à MySQL **comment recréer votre base de données** (avec des requêtes en langage SQL). On peut s'en servir pour deux choses:

- **transmettre votre base de données sur Internet** : pour le moment, votre base de données se trouve sur votre disque dur. Mais lorsque vous voudrez héberger votre site sur Internet, il faudra utiliser la base de données en ligne de votre hébergeur !



Le fichier.sql que vous allez générer vous permettra de **reconstruire** la base de données grâce à l'outil d'importation de phpMyAdmin (en général, les hébergeurs proposent eux aussi phpMyAdmin pour que vous puissiez effectuer facilement des opérations sur votre base en ligne);

- **faire une copie de sauvegarde de la base de données** : on ne sait jamais, si vous faites une bêtise ou si quelqu'un réussit à détruire toutes les informations sur votre site (dont la base de données), il faut avoir une copie de secours sur votre disque dur !

**Onglet Opérations:** Ici, vous pouvez effectuer diverses opérations sur votre table.

Par exemple :

- **changer le nom de la table** : indiquez le nouveau nom pour cette table ;
- **déplacer la table vers** : si vous voulez placer cette table dans une autre base de données ;
- **copier la table** : faire une copie de la table, dans une autre base ou dans la même (attention : dans ce cas, il faudra qu'elle ait un nom différent) ;
- **optimiser la table** : à force d'utiliser une table, surtout si elle est grosse, on finit par avoir des « pertes » qui font que la table n'est plus bien organisée. Un clic là-dessus et hop ! c'est de nouveau arrangé.

## Onglet Vider

Vide tout le contenu de la table. Toutes les entrées vont disparaître, seule la structure de la table restera (c'est-à-dire les champs).

**Attention!** Il n'est pas possible d'annuler cette opération !

## Onglet Supprimer

Pour supprimer la totalité de la table (structure et données), cliquez sur cet onglet.

Là encore, réfléchissez-y à deux fois avant de tout supprimer, car vous ne pourrez rien récupérer par la suite, à moins d'avoir fait une sauvegarde au préalable avec l'outil d'exportation.

## En résumé

- phpMyAdmin est un outil qui nous permet de visualiser rapidement l'état de notre base de données ainsi que de la modifier, sans avoir à écrire de requêtes SQL.
- On crée généralement un champ nommé id qui sert à numéroter les entrées d'une table. Ce champ doit avoir un index PRIMARY (on dit qu'on crée une clé primaire) et l'option AUTO\_INCREMENT qui permet de laisser MySQL gérer la numérotation.
- MySQL gère différents types de données pour ses champs, à la manière de PHP. On trouve des types adaptés au stockage de nombres, de textes, de dates, etc.
- phpMyAdmin possède un outil d'importation et d'exportation des tables qui nous permettra notamment d'envoyer notre base de données sur Internet lorsque nous mettrons notre site en ligne.