

 **FREE eBook**

LEARNING phpmyadmin

Free unaffiliated eBook created from
Stack Overflow contributors.

#phpmyadm

in

Table of Contents

About.....	1
Chapter 1: Getting started with phpmyadmin.....	2
Remarks.....	2
Versions.....	2
Examples.....	3
Installation or Setup.....	3
How to create a database and grant privileges for database user.....	10
Credits.....	14

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [phpmyadmin](#)

It is an unofficial and free phpmyadmin ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official phpmyadmin.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with phpmyadmin

Remarks

phpMyAdmin is a free software used to manage MySQL databases via web interface. It is much easier than using MySQL console where the user must use commands to manage databases. By using phpMyAdmin, with a few mouse clicks people can create database and grant privileges for MySQL user for that database.

[Official documentation](#)
[Wiki](#)

Versions

Version	Release Date
1.1.0	1998-11-03
1.2.0	1998-11-29
1.3.0	1998-12-16
1.3.1	1998-12-27
1.4.0	1999-01-16
2.0.0	1999-04-11
2.1.0	2000-06-08
2.2.0	2001-08-31
2.3.0	2001-11-08
2.5.0	2003-11-05
2.6.0	2004-09-27
2.7.0	2005-12-04
2.8.0	2006-03-06
2.9.0	2006-09-20
2.10.0	2007-02-27
2.11.0	2007-08-22

Version	Release Date
3.0.0	2008-09-27
3.1.0	2008-11-28
3.2.0	2009-06-09
3.3.0	2010-03-07
3.4.0	2011-05-11
3.5.0	2012-04-07
4.0.0	2013-05-03
4.1.0	2013-12-12
4.2.0	2014-05-08
4.3.0	2014-12-05
4.4.0	2015-04-01
4.5.0	2015-09-23
4.6.0	2016-03-22

Examples

Installation or Setup

Description:

phpMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the Web. phpMyAdmin supports a wide range of operations on MySQL and MariaDB. Frequently used operations (managing databases, tables, columns, relations, indexes, users, permissions, etc) can be performed via the user interface, while you still have the ability to directly execute any SQL statement.

Note 1: Many operating systems already include a phpMyAdmin package and will automatically keep it updated, however these versions are sometimes slightly outdated and therefore may be missing the latest features.

Note 2: These method assumes that you have already setup a local web server with MySQL.

Windows:

- **Third Party Products:**

The easiest way to get phpMyAdmin on Windows is using third party products which include phpMyAdmin together with a database and web server such as [XAMPP](#)(cross-platform).

- **List of Cross-Platform third party products:** [XAMPP](#), [AMPPS](#), [Zend Server Community Edition](#)
- **List of Windows Specific third party products:** [WampServer](#), [EasyPHP Devserver](#), [Uniform Server](#), [Bitnami WAMP Stack](#), [WPN-XM Server Stack](#), [WTSerfer](#), [UwAmp](#), [Vertrigo](#)
- **Manual Installation** [phpadmin.net](#):

You can select any kit from given phpmyadmin kits.

Example:

For all languages choose some thing like **phpMyAdmin-4.6.4-all-languages.7z** or for language specific choose something like **phpMyAdmin-4.6.4-english.7z**

After Choosing a phpmyadmin kit extract the folder. Rename the extracted folder to *phpMyAdmin* it's easy to reference. Copy the extracted folder to your local web server's document root. Depending on how you have configured your web server, this could be the **www**, **htdocs**, **html**, or **public_html** folder. If you are using IIS, open IIS Manager. In the left navigation tree, right-click on "Default Web Site" and select "**Explore**". Copy and paste the extracted phpMyAdmin folder into the location that opens.

Linux distributions

Note: phpMyAdmin is included in most Linux distributions. It is recommended to use distribution packages when possible. They usually provide integration to your distribution and you will automatically get security updates from your distribution.

- **Third Party Products:** [XAMPP](#), [TurnKey Linux LAMP stack](#), [Bitnami LAMPStack](#),
- **Manual Installation:** [phpadmin.net](#):

You can select any kit from given phpmyadmin kits.

Example:

For all languages choose some thing like **phpMyAdmin-4.6.4-all-languages.7z** or for language specific choose something like **phpMyAdmin-4.6.4-english.7z**

After Choosing a phpmyadmin kit extract the folder. Rename the extracted folder to phpMyAdmin it's easy to reference. Copy the extracted folder to your local web server's document root. Depending on how you have configured your web server, this could be the **htdocs** or **public_html** folder.

- **Debian:**

Debian's package repositories include a phpMyAdmin package, but be aware that the

configuration file is maintained in `/etc/phpmyadmin` and may differ in some ways from the official phpMyAdmin documentation. Specifically it does:

1. Configuration of web server (works for Apache and lighttpd).
2. Creating of [phpMyAdmin configuration storage](#) using `dbconfig-common`.
3. Securing setup script, see [Setup script on Debian, Ubuntu and derivatives](#).

- **Terminal:** The easiest way to install phpmyadmin in **Debian** is through `apt-get`

```
sudo apt-get install phpmyadmin
```

During the installation, phpMyAdmin will walk you through a basic configuration. After the process starts follow these steps:

1. Select **Apache2** for the server
2. Choose **YES** when asked about whether to Configure the database for phpmyadmin with `dbconfig-common`
3. Enter your **MySQL password** when prompted
4. Enter the password that you want to use to log into phpmyadmin

After the installation has completed, add phpmyadmin to the apache configuration.

```
sudo nano /etc/apache2/apache2.conf
```

Add the phpmyadmin config to the file.

```
Include /etc/phpmyadmin/apache.conf
```

Restart apache:

```
sudo service apache2 restart
```

You can then access phpmyadmin by going to *your-ip-address/phpmyadmin*.

Security:

Unfortunately, older versions of phpMyAdmin have had serious security vulnerabilities, including allowing remote users to eventually exploit root on the underlying virtual private server. One can prevent a majority of these attacks through a simple process: locking down the entire directory with Apache's native user/password restrictions which will prevent these remote users from even attempting to exploit older versions of phpMyAdmin.

Set Up the .htaccess File:

To set this up, start off by allowing the `.htaccess` file to work within the phpmyadmin directory. You can accomplish this in the phpmyadmin configuration file:

```
sudo nano /etc/phpmyadmin/apache.conf
```

Under the directory section, add the line “AllowOverride All” under “Directory Index”, making the section look like this:

```
<Directory /usr/share/phpmyadmin>
    Options FollowSymLinks
    DirectoryIndex index.php
    AllowOverride All
    [...]
```

Configure the .htaccess file

With the .htaccess file allowed, we can proceed to set up a native user whose login would be required to even access the phpmyadmin login page.

Start by creating the .htaccess page in the phpmyadmin directory:

```
sudo nano /usr/share/phpmyadmin/.htaccess
```

Follow up by setting up the user authorization within .htaccess file. Copy and paste the following text in

```
AuthType Basic
AuthName "Restricted Files"
AuthUserFile /path/to/passwords/.htpasswd
Require valid-user
```

AuthType: This refers to the type of authentication that will be used to check the passwords. The passwords are checked via HTTP and the keyword Basic should not be changed.

AuthName: This is text that will be displayed at the password prompt. You can put anything here.

AuthUserFile: This line designates the server path to the password file (which we will create in the next step.) **Require valid-user:** This line tells the .htaccess file that only users defined in the password file can access the phpMyAdmin login screen.

Create the htpasswd file:

Now we will go ahead and create the valid user information.

Start by creating a htpasswd file. Use the htpasswd command, and place the file in a directory of your choice as long as it is not accessible from a browser. Although you can name the password file whatever you prefer, the convention is to name it .htpasswd.

```
sudo htpasswd -c /path/to/passwords/.htpasswd username
```

A prompt will ask you to provide and confirm your password.

Once the username and passwords pair are saved you can see that the password is encrypted in

the file.

Finish up by restarting apache:

```
sudo service apache2 restart
```

Accessing phpMyAdmin:

phpMyAdmin will now be much more secure since only authorized users will be able to reach the login page. Fill it in with the username and password that you generated. After you login you can access phpmyadmin with the MySQL username and password.

- **Ubuntu:**

To get started, we can simply install phpMyAdmin from the default Ubuntu repositories.

We can do this by updating our local package index and then using the apt packaging system to pull down the files and install them on our system:

```
sudo apt-get update
sudo apt-get install phpmyadmin
```

Warning:

When the first prompt appears, apache2 is highlighted, but not selected. If you do not hit "SPACE" to select Apache, the installer will not move the necessary files during installation. Hit "SPACE", "TAB", and then "ENTER" to select Apache.

1. For the server selection, choose apache2.
2. Select yes when asked whether to use `dbconfig-common` to set up the database
3. You will be prompted for your database administrator's password
4. You will then be asked to choose and confirm a password for the **phpMyAdmin** application itself

The installation process actually adds the phpMyAdmin Apache configuration file into the `/etc/apache2/conf-enabled/` directory, where it is automatically read.

The only thing we need to do is explicitly enable the `php5-mcrypt` extension, which we can do by typing:

```
sudo php5enmod mcrypt
```

Afterwards, you'll need to restart Apache for your changes to be recognized:

```
sudo service apache2 restart
```

You can then access phpmyadmin by going to *your-ip-address/phpmyadmin*.

- **Fedora:**

Fedora ships the phpMyAdmin package, but be aware that the configuration file is maintained in `/etc/phpMyAdmin/` and may differ in some ways from the official phpMyAdmin documentation.

Terminal:

First, you'll follow a simple best practice: ensuring the list of available packages is up to date before installing anything new.

```
dnf -y update
```

Then it's a matter of just running one command for installation via apt-get:

```
dnf -y install phpmyadmin
```

Security:

To secure phpMyAdmin we should lock down access to a specific IP address.

When the phpMyAdmin package is installed, an Apache Virtual Host file is added to configure web access. Let's edit that file:

```
vim /etc/httpd/conf.d/phpMyAdmin.conf
```

By default, the configuration for phpMyAdmin only allows access from the server on which it is installed.

Change each IP address to the one you found in [what is my ip address?](#) or another IP address that will be connecting to phpMyAdmin remotely:

```
Require ip 127.0.0.1
Allow from 127.0.0.1
Require ip 127.0.0.1
Allow from 127.0.0.1
```

Restart Apache:

```
systemctl restart httpd
```

Verify that phpMyAdmin is working by visiting `http://the_IP_of_your_server/phpmyadmin` For example: `http://1.2.3.4/phpmyadmin`

- **Composer:**

You can install phpMyAdmin using [Composer](#), however it's currently not available in the default

Packagist repository due to its technical limitations.

The installation is possible by adding our own repository

<https://www.phpmyadmin.net/packages.json>:

```
composer create-project phpmyadmin/phpmyadmin --repository-  
url=https://www.phpmyadmin.net/packages.json --no-dev
```

- **Installing from Git:**

You can clone current phpMyAdmin source from <https://github.com/phpmyadmin/phpmyadmin.git>:

```
git clone https://github.com/phpmyadmin/phpmyadmin.git
```

Additionally you need to install dependencies using Composer:

```
composer update
```

If you do not intend to develop, you can skip installation of developer tools by invoking:

```
composer update --no-dev
```

Mac:

- **Third Party Products** : [MAMP](#), [AMPPS](#), [Bitnami MAMPStack](#), [XAMPP](#)
- **Manually:**

Download [phpMyAdmin](#) , For all languages choose something like phpMyAdmin-4.6.4-all-languages.tar.bz2 or for language specific choose something like phpMyAdmin-4.6.4-english.tar.gz

After Choosing a phpmyadmin kit extract the folder. Rename the extracted folder to **phpMyAdmin** it's easy to reference. Copy the extracted folder to your local web server's document root. Depending on how you have configured your web server, this could be the htdocs or public_html folder.

Make the **config** folder

```
mkdir ~/Sites/phpmyadmin/config
```

Change the permissions

```
chmod o+w ~/Sites/phpmyadmin/config
```

Run the set up in the browser

<http://localhost/~username/phpmyadmin/setup/> or <http://localhost/phpmyadmin/setup/>

You need to create a new localhost mysql server connection, **click new server**.

Switch to the **Authentication tab** and set the local mysql root user and the password. Add in the username “root” (maybe already populated, add in the password that you set up earlier for the MySQL root user set up, click on save and you are returned to the previous screen. (This is not the OSX Admin or root password – it is the MySQL root user).

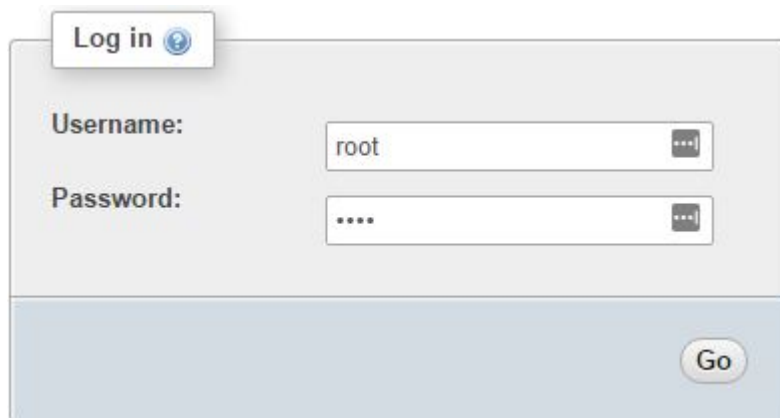
Make sure you click on save, then a config.inc.php is now in the /config directory of phpmyadmin directory, **move** this file to the root level of /phpmyadmin and then remove the now **empty** /config directory.

Now going to <http://localhost/~username/phpmyadmin/> will now allow you to interact with your MySQL databases.

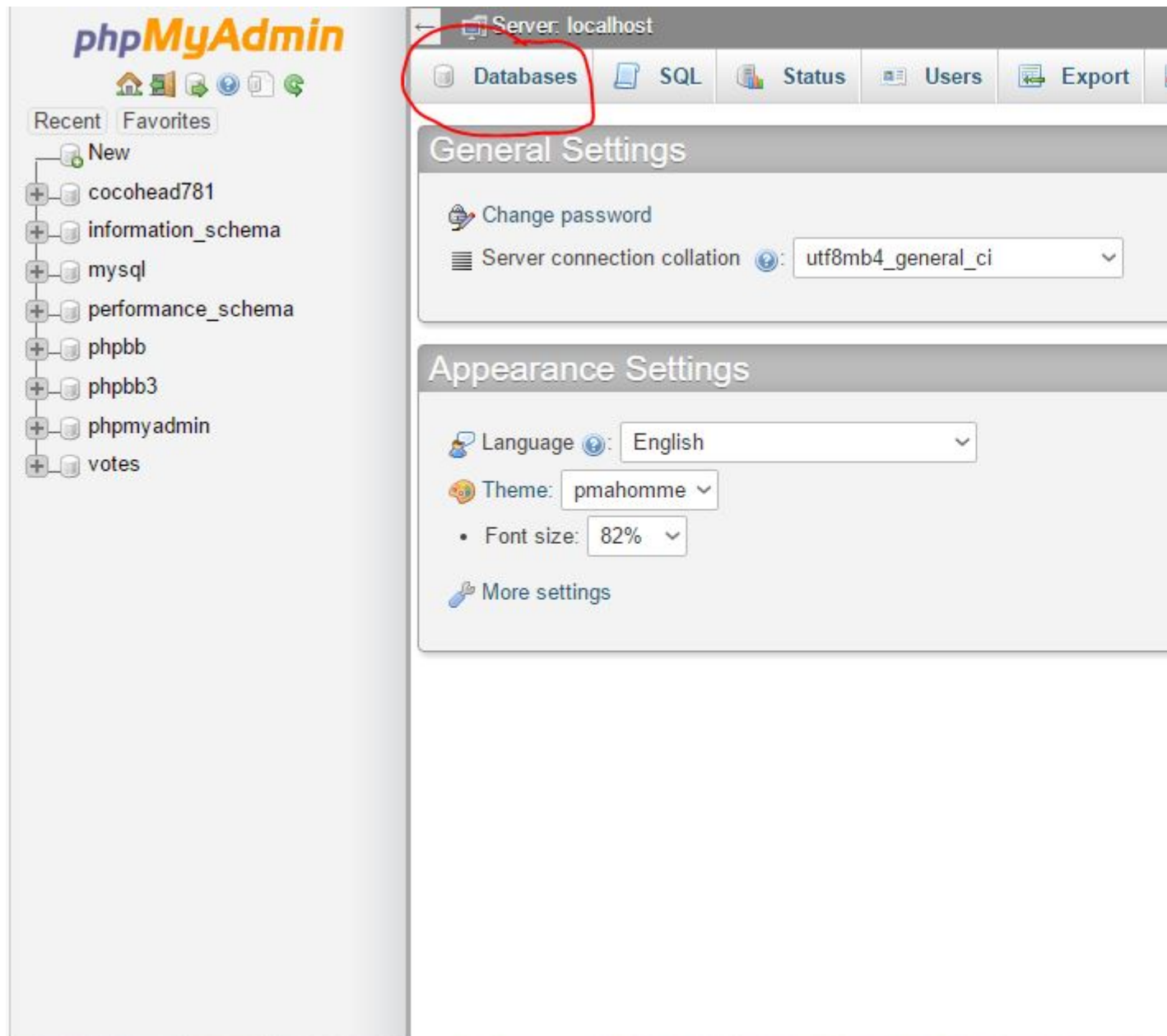
Note: Change localhost with **your-ip-address**

How to create a database and grant privileges for database user.

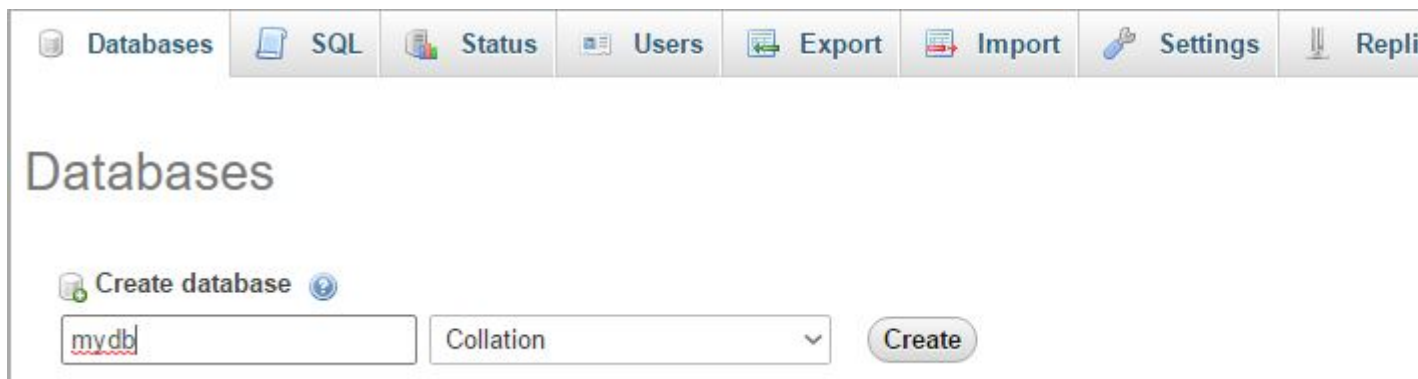
1. Navigate to phpMyAdmin by URL http://your_ip/phpmyadmin or <http://localhost/phpmyadmin>
2. Login using username root and root password.



3. Click on Databases tab.



4. Enter database name, select collation (you may leave it to default) and click create.



5. Click on Privileges tab and select "Add user account".

Users overview

User	Host	Password	Global privileges	User group	Grant	Action
<input type="checkbox"/> cocohead781	%	Yes	ALL PRIVILEGES		Yes	Edit Privileges Export
<input type="checkbox"/> root	localhost	Yes	ALL PRIVILEGES		Yes	Edit Privileges Export

☐ Check All With selected: Export

Add user

6. You can select localhost if the connection is made only form local. Select "any host" if you are planing to connect from different hosts.
7. Enter new username and password.

Add user

Login Information

User name:

Use text field: ▾

user

Host:

Any host ▾

%

Password:

Use text field: ▾

....

Re-type:

....

Generate password:

Generate

Database for user

☐ Create database with same name and grant all privileges.
 ☐ Grant all privileges on wildcard name (username_%).

Global privileges

☐ Check All

8. Select the privileges you need for that user.
9. Click on Database tab and select the database for which you need to give access(in our case the newly created database in step 4). Click Go.

Read [Getting started with phpmyadmin online](#):

<https://riptutorial.com/phpmyadmin/topic/5577/getting-started-with-phpmyadmin>

Credits

S. No	Chapters	Contributors
1	Getting started with phpmyadmin	cocohead781 , Community , Harikrishnan , ppovoski , Ray , Sheryar Ahmed