

docs.zendframework.com

Overview - tutorials

1. [Docs](#) »
2. MVC Tutorials »
3. Getting Started with Zend Framework »
4. Overview

This tutorial is intended to give an introduction to using Zend Framework by creating a simple database driven application using the Model-View-Controller paradigm. By the end you will have a working ZF application and you can then poke around the code to find out more about how it all works and fits together.

Some assumptions

This tutorial assumes that you are running at least PHP 5.6 with the Apache web server and MySQL, accessible via the PDO extension. Your Apache installation must have the `mod_rewrite` extension installed and configured.

You must also ensure that Apache is configured to support `.htaccess` files. This is usually done by changing the setting:

```
AllowOverride None
```

to

```
AllowOverride FileInfo
```

in your `httpd.conf` file. Check with your distribution's documentation for exact details. You will not be able to navigate to any page other than the home page in this tutorial if you have not configured `mod_rewrite` and `.htaccess` usage correctly.

Getting started faster

Alternatively, you can use any of the following as well:

- The built-in web server in PHP. Run `php -S 0.0.0.0:8080 -t public/public/index.php` in your application root to start a web server listening on port 8080.
- Use the shipped `Vagrantfile`, by executing `vagrant up` from the application root. This binds the host machine's port 8080 to the Apache server instance running on the Vagrant image.
- Use the shipped [docker-compose](#) integration, by executing `docker-compose up -d --build` from the application root. This binds the host machine's port 8080 to the Apache server instance running container.

The tutorial application

The application that we are going to build is a simple inventory system to display which albums we own. The main page will list our collection and allow us to add, edit and delete CDs. We are going to need four pages in our website:

Page	Description
List of albums	This will display the list of albums and provide links to edit and delete them. Also, a link to enable adding new albums will be provided.
Add new album	This page will provide a form for adding a new album.
Edit album	This page will provide a form for editing an album.
Delete album	This page will confirm that we want to delete an album and then delete it.

We will also need to store our data into a database. We will only need one table with these fields in it:

Field name	Type	Null?	Notes
id	integer	No	Primary key, auto-increment
artist	varchar(100)	No	
title	varchar(100)	No	
