

MAC TUTORIAL — BEGINNER'S GUIDE




# PHP Environment on MacBook

Step-by-step installation and usage guide  
macOS 14+ — Mac Intel & Apple Silicon (M1/M2/M3/M4/M5)

 Apache 2.4

 PHP 7.4

 MariaDB 10.6

 phpMyAdmin

 Portainer

 PHP Site

`http://localhost:8080`

 phpMyAdmin

`http://localhost:8081`

 Portainer (GUI)

`http://localhost:9000`

 Dashboard

`http://localhost:8082`

# Table of Contents — Mac

---

## 1 Install Docker Desktop on Mac p. 3

- Mac Intel or Apple Silicon — how to tell?
- Download and installation
- Verify that Docker is ready

## 2 First launch of the environment p. 4

- Prepare the scripts (one time only)
- Launch the environment with start.command

## 3 Access the web interfaces p. 5

- Dashboard (8082), PHP Site (8080), phpMyAdmin (8081)
- Portainer — create your account (important!)

## 4 Working with PHP p. 7

- The htdocs folder — your workspace
- Create and test a PHP file
- Connect to the database

## 5 Manage the database with phpMyAdmin p. 8

## 6 Stop and restart services p. 9

## 7 Mac Troubleshooting p. 10

- Portainer timeout — solution
- Common issues
- Credentials summary



### About this tutorial

This guide is designed for beginners. All operations are performed through graphical interfaces. No complex command line is required after the initial setup.

# 1

## CHAPTER 1

# Install Docker Desktop

To be done once on your MacBook

## What type of Mac do you have?

Before downloading, identify your processor. The two Mac types require different versions of Docker.



### How to tell?

Menu **Apple ()** → **"About This Mac"**

- You see **"Intel Processor"** → Mac Intel
- You see **"Apple M1/M2/M3/M4/M5 Chip"** → Mac Apple Silicon

## Download and installation



### Mac Intel

1. Go to **docker.com**
2. Click **"Download for Mac"**
3. Choose **"Intel Chip"**
4. Open the **.dmg** file
5. Drag Docker → Applications
6. Open Docker Desktop






### Mac Apple Silicon (M1/M2/M3/M4/M5)

1. Go to **docker.com**
2. Click **"Download for Mac"**
3. Choose **"Apple Chip"**
4. Open the **.dmg** file
5. Drag Docker → Applications
6. Open Docker Desktop

## Verify that Docker is ready

Docker Desktop places a **whale icon**  in the menu bar (top right of the screen).

### Color meanings

	<b>Green = Ready</b>	Docker is running — you can work
	<b>Yellow = Starting</b>	Wait a few seconds
	<b>Red = Error</b>	Restart Docker Desktop



### Docker must be green before continuing

Docker Desktop must remain open during your entire work session. If you close it, all services will stop.

## 2

### CHAPTER 2

## First launch

Launch the environment with a double-click

### 🔑 One-time step — Grant permissions

The scripts `start.command` and `stop.command` need execution permission. **This step only needs to be done once.**

1

#### Open Terminal

Launchpad → search for "Terminal" → click on it

2

#### Navigate to the project folder

Type and press **Enter**:

```
cd ~/Desktop/PHPenv
```

3

#### Grant permissions

Type and press **Enter**:

```
chmod +x start.command stop.command
```



#### Done!

You will never need the Terminal again. From now on, everything is done by double-clicking.



## Launch the environment

1

### Verify that Docker Desktop is open and green

Check the whale icon at the top right of the screen

2

### Double-click on `start.command`

This file is in the project folder (on your Desktop)

3

### A Terminal window opens automatically

Do not close it! Services are starting in the background

4

### Wait for the green confirmation message

✓ Environment started successfully!

!

### Go to Portainer immediately!

Open <http://localhost:9000> within the first 5 minutes to create your account. See Ch.3 for details.



### First time: wait 5 to 10 minutes

Docker is downloading the images (PHP, MariaDB...). This is slow **only the first time**. Subsequent launches: less than 10 seconds.





# 3

## CHAPTER 3

# Access the web interfaces

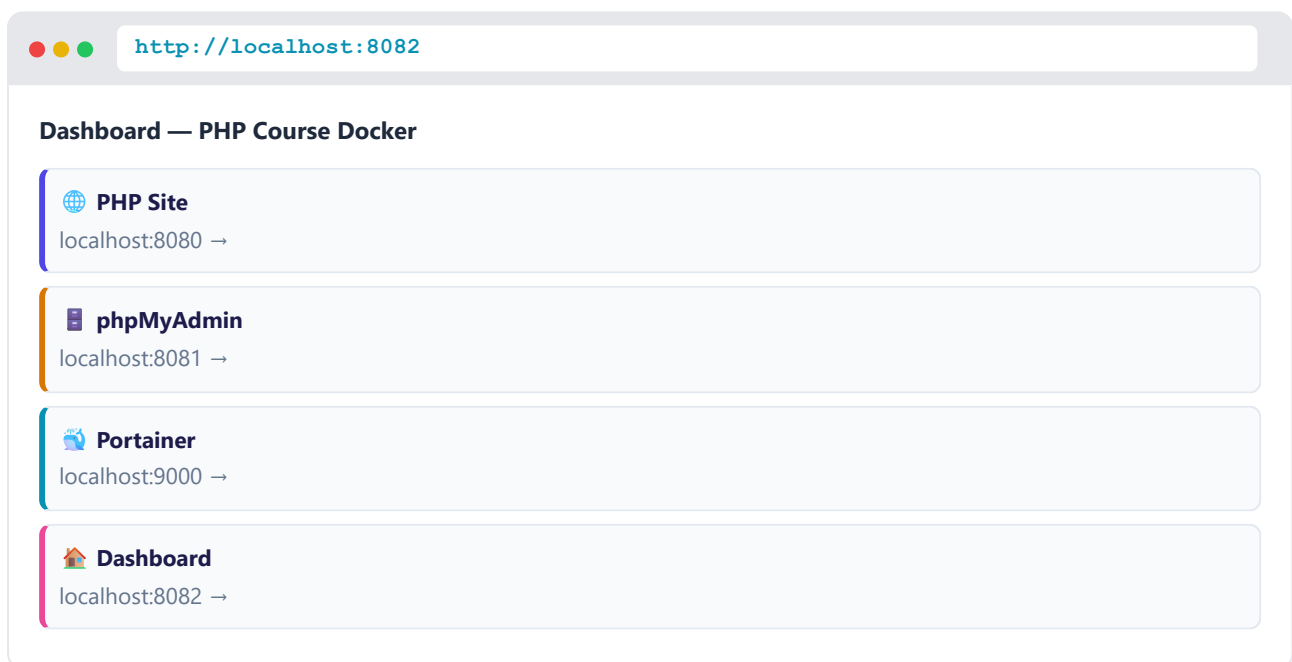
Everything happens in your browser

## Overview

SERVICE	ADDRESS	DESCRIPTION
 Dashboard	<code>localhost:8082</code>	Home page with all links
 PHP Site	<code>localhost:8080</code>	Your PHP files (htdocs/)
 phpMyAdmin	<code>localhost:8081</code>	Visual management of the DB
 Portainer	<code>localhost:9000</code>	Docker container management

## Dashboard — localhost:8082

Your daily **starting point**. It centralizes all links and your credentials.



## PHP Site — localhost:8080

Displays the contents of your `htdocs/` folder. The test page confirms that PHP and the database are working.



http://localhost:8080

# Hello PHP ! 🖐️





✅ MySQL connection successful! (MariaDB 10.6.x)

**PHP 7.4.x** — Linux 64-bit



## phpMyAdmin — localhost:8081

Graphical interface to manage your database. The connection is pre-configured (automatic).

 Address	<code>http://localhost:8081</code>
 Username	<code>root</code>
 Password	<code>root</code>
 Database	<code>app</code>



### phpMyAdmin not responding?

The DB takes 10 to 20 seconds to start. Wait and refresh with **Cmd+R**.



## Portainer — localhost:9000

Graphical interface to manage your Docker containers (start, stop, restart, view logs) — no command line needed.



### Act within the first 5 minutes!

Portainer expires its initialization for security reasons. After startup, go **immediately** to `http://localhost:9000` and create your account. If you see the *"timeout"* page → see the Troubleshooting chapter (p. 10).

### Create your account (first access)

1

#### Open `http://localhost:9000`

Portainer asks you to create an administrator account

2

#### Enter a username and a password

Example: `admin` — Password: 12 characters minimum













3





#### Click "Create user"

Then choose **"Get Started"** → **"local"**

## What you can do in Portainer

http://localhost:9000 → Containers

Container	Status	Available actions
cours_web	running	   
cours_db	running	   
cours_phpmyadmin	running	   

 Start  Stop  Restart  View logs

# 4

## CHAPTER 4

# Working with PHP

Create, edit and test your PHP scripts

## 📁 The htdocs folder — your workspace

All your PHP files must be in the `htdocs/` folder of the project. It is the equivalent of the `htdocs` folder in XAMPP.

### 📁 Project structure

```
PHPenv/  
├── 📁 htdocs/      ← YOUR PHP FILES HERE  
│   ├── index.php  ← test page (already created)  
│   └── hello.php   ← your next file  
└── docker-compose.yml ← do not modify
```

## 🔧 Create your first PHP file

### 1 Open the `htdocs/` folder in Finder

Double-click on the folder from your Desktop

### 2 Create a new file `hello.php`

In VS Code, TextEdit, or any text editor. The `.php` extension is required.

### 3 Write your PHP code

Minimal example below

### 4 Save and open in the browser

`http://localhost:8080/hello.php` — refresh with **Cmd+R**

```
<?php  
echo "Hello world!";  
?>
```



### No need to restart!

Every change to your PHP files is visible immediately after **Cmd+R** in the browser.



## Database connection (PDO)

```
// Connect to the database

$pdo = new PDO(
    "mysql:host=db;dbname=app;charset=utf8mb4",
    "app",    // username
    "app"     // password
);

// Retrieve data
$stmt = $pdo->query("SELECT * FROM my_table");
foreach ($stmt as $row) {
    echo $row['name'] . "<br>";
}
```



### The host is "db", not "localhost"

In Docker, services communicate via their service name. The database is called `db`.

# 5

## CHAPTER 5

# Manage the database

Create tables and manipulate data with phpMyAdmin

## 🔑 Log in to phpMyAdmin

Go to **http://localhost:8081**. The connection is pre-configured, you are automatically logged in as root.

🌐 Address	<code>http://localhost:8081</code>
👤 Username	<code>root</code>
🔑 Password	<code>root</code>
📁 Database	<code>app</code>

## 🔧 Create a table

### 1 In the left column, click on "app"

This is your database

### 2 "Structure" tab → "Create table"

Give it a name (e.g.: `users`) and choose the number of columns

### 3 Define your columns, then "Save"

Always add an `id` column as INT AUTO\_INCREMENT as the primary key

## 📊 Example "users" table

COLUMN	TYPE	LENGTH	OPTIONS
<code>id</code>	INT	11	PRIMARY KEY, AUTO_INCREMENT
<code>name</code>	VARCHAR	100	—
<code>email</code>	VARCHAR	150	—
<code>date_created</code>	DATETIME	—	DEFAULT CURRENT_TIMESTAMP

## Execute an SQL query

---

"SQL" tab in phpMyAdmin:

```
-- Insert data

INSERT INTO users (name, email)
VALUES ('Alice Martin', 'alice@example.com');

-- Read all data

SELECT * FROM users;
```

# 6

## CHAPTER 6

# Stop and restart

Manage the environment without the command line

## □ Stop the environment

### Option A — The simplest

Double-click on `stop.command` in the project folder. A window opens, everything stops cleanly.

### Option B — Via Portainer

1. Open `localhost:9000`
2. "Containers" tab
3. Check all containers
4. Click "Stop"



### Your data is preserved!

Stopping the services does not delete your PHP files or your database data. Everything is preserved for the next launch.

## ↺ Restart a specific service

1 Portainer → `http://localhost:9000` → "Containers"

2 Find the container to restart

`cours_web` = Apache+PHP | `cours_db` = database | `cours_phpmyadmin`

3 Click the ↺ (Restart) button

The service restarts in a few seconds



## View the logs

In Portainer → Containers → click on the **name** of a container → "Logs". Useful for diagnosing errors.



### Daily routine

**Morning:** Open Docker Desktop → double-click `start.command`

**Evening:** Double-click `stop.command`



# 7

## CHAPTER 7

# Mac Troubleshooting

Solutions to the most common issues

## Common issues

### ✗ Portainer — "timeout" page (/timeout.html)

**Cause:** admin account not created within the first 5 minutes.

**Solution:** Restart Portainer (preserves all data):

```
docker compose restart portainer
```

Then go **immediately** to <http://localhost:9000> to create your account.

**Full reset:** Docker Desktop → Volumes → delete `cours_portainer_data` → relaunch Portainer.

### ✗ Docker Desktop is not started

Open **Docker Desktop** from the Launchpad → wait for the green whale → relaunch `start.command`

### ✗ localhost:8080 is not loading

Wait 30 seconds (the DB starts last) → **Cmd+R** to refresh.

### ✗ "Permission denied" on start.command









Open Terminal and type:

```
chmod +x ~/Desktop/PHPenv/start.command stop.command
```

### ✗ phpMyAdmin — connection error

The DB takes 10 to 20 seconds to start → wait and refresh (**Cmd+R**).

## Credentials summary

 PHP Site	<code>http://localhost:8080</code>
 phpMyAdmin	<code>http://localhost:8081</code>
 Portainer	<code>http://localhost:9000</code>
 Dashboard	<code>http://localhost:8082</code>
 DB Host (in PHP)	<code>db</code>
 Database	<code>app</code>
 DB Username	<code>app</code>
 DB Password	<code>app</code>
 Root password	<code>root</code>

Mac Tutorial — Apache 2.4 + PHP 7.4 + MariaDB 10.6 + phpMyAdmin 5.2 + Portainer CE  
macOS 14+ — Mac Intel & Apple Silicon (M1/M2/M3/M4/M5)