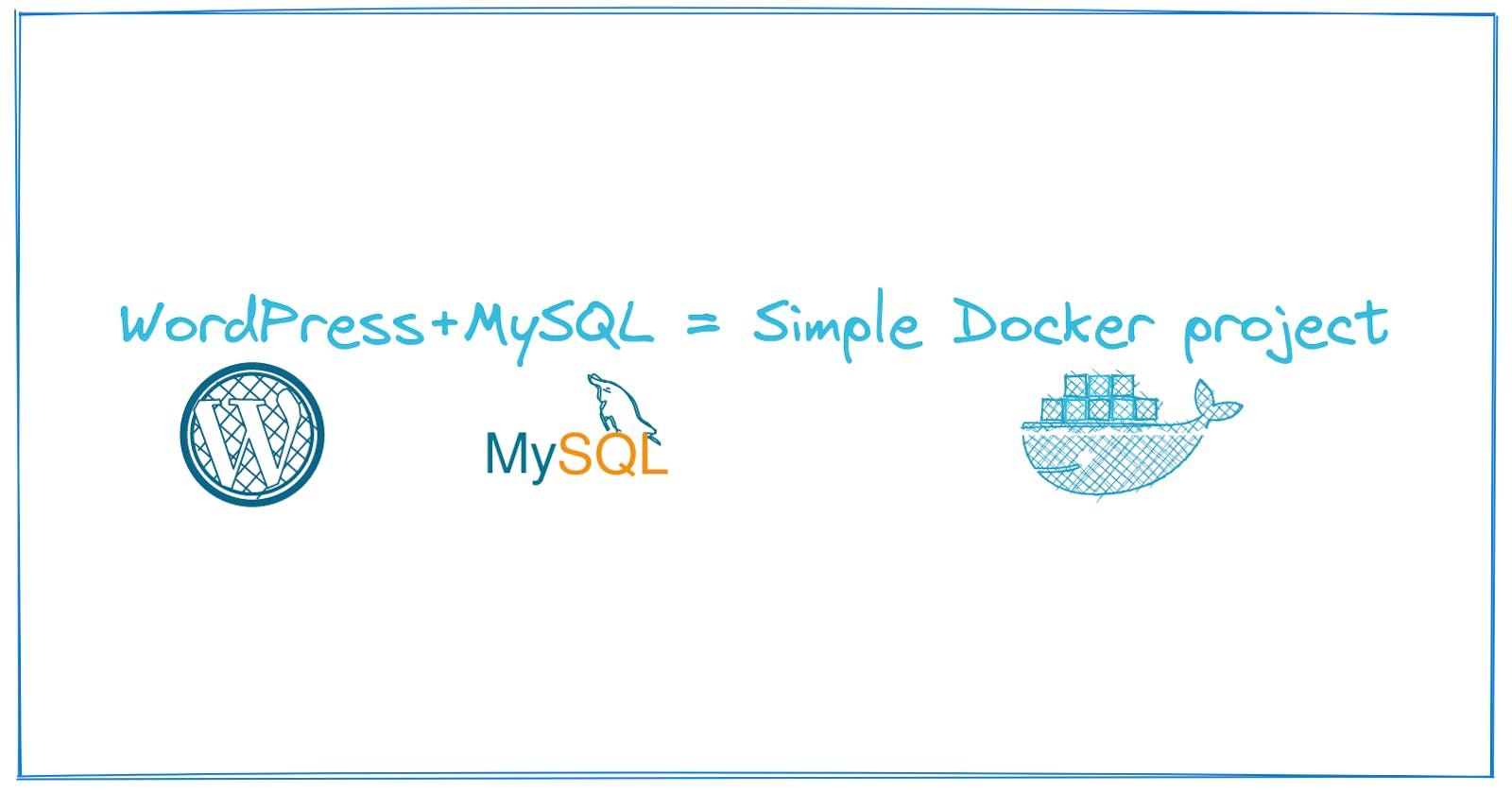
**WordPress + MySQL = Simple Docker Project**

**Let's practice Docker together with a simple project**



Practicing Docker can be hard, this simple project can help you practice a few Docker concepts. A very little understanding of Docker is needed, its fine even if you don't, all you need is just Docker installed and working.

**What will we use?**

* Docker
* WordPress
* MySQL
* Docker network

**Note**: This might seem complicated but it's not, just follow along and give it a try.

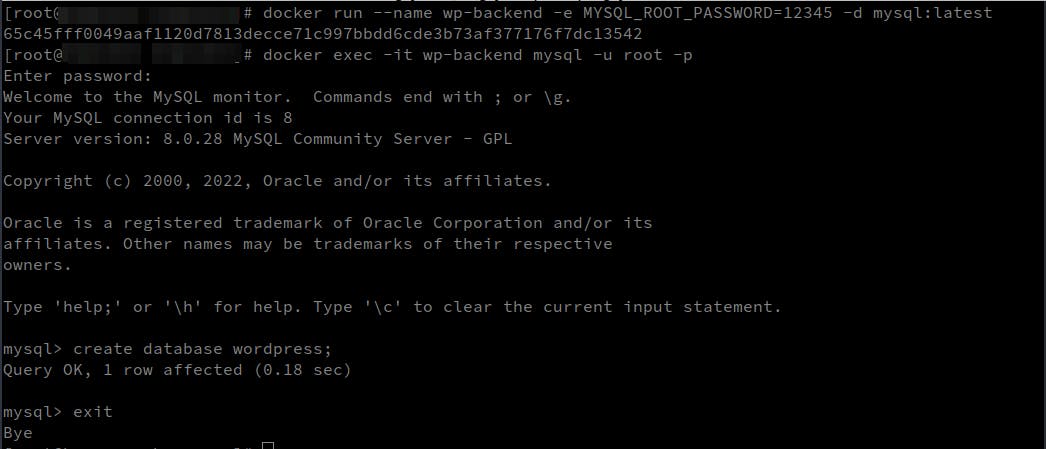
**Steps:**

1. We create a MySQL container and then exec into it to create a database.
2. Then we create a WordPress container and expose internal port 80 to external 8080.
3. WordPress cannot detect the MySQL database so we create a new network and add both the containers to the network.
4. We add the database details to the WordPress site and done!

**Step 1:**

First lets create the MySQL container, if you look at the MySQL docker hub page you will be told that you need a root password to create it. And thats how we know we should give it a environment variable and password.

Use docker run --name wp-backend -e MYSQL\_ROOT\_PASSWORD=12345 -d mysql:latest

* -e -> environment variables
* -d -> detached mode(runs in the background)

We have the container but we don't have a Database inside it. Wait for a few minutes for the container to be created, then run docker exec -it wp-backend mysql -u root -p

* exec -> Execute a command inside the container
* -it mysql-> using interactive terminal of *MySQL*
* -u root -p -> The command that you want to run inside the MySQL container
  + -u -> username
  + -p -> ask for a password(MYSQL\_ROOT\_PASSWORD)

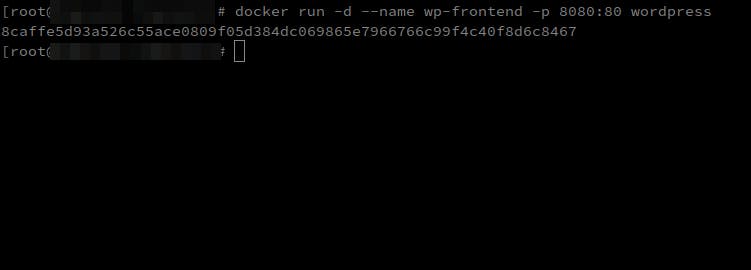
Once inside, run create database wordpress; to create a database for us to use.

Typeexit to exit the container.

**Step 2:**

Now that we have the **MySQL** backend lets create the **WordPress** frontend of our project.

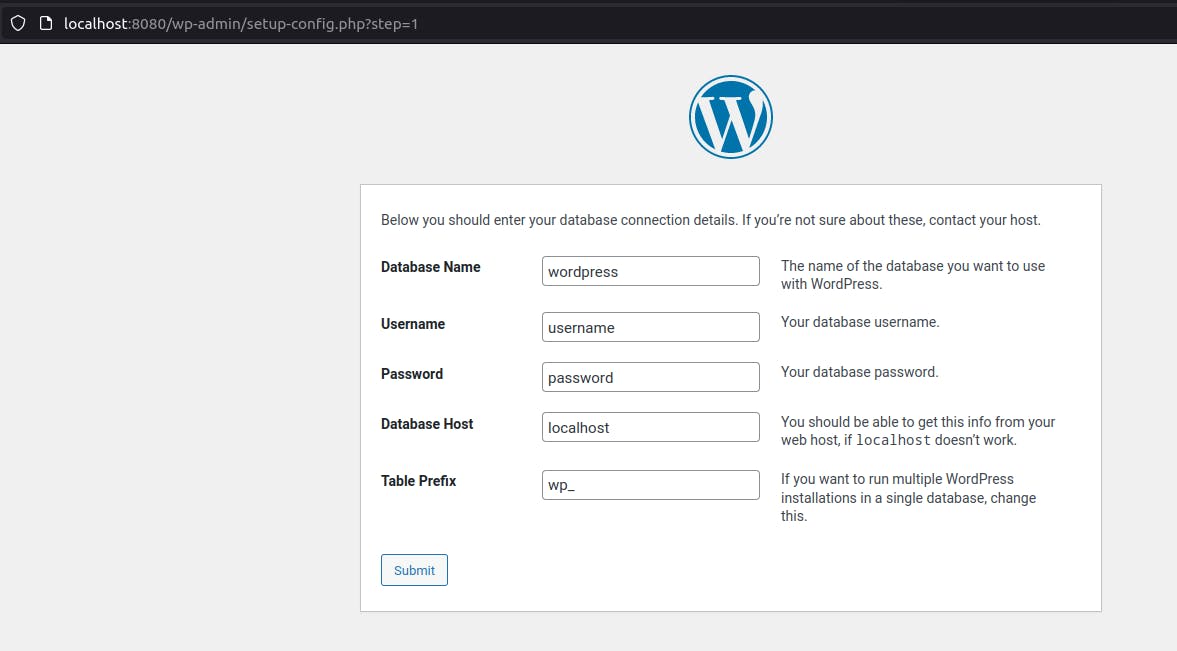
We can't directly reach the WordPress container, so we open the port 80 inside the container and tell docker to map it to port 8080 of our machine.



Run docker run -d --name wp-frontend -p 8080:80 wordpress

* -p -> ports
* 8080:80 -> Attach 8080 of host machine to 80 of WordPress container.

If you go to localhost:8080 you should see the WordPress site. Make a few clicks and you will land on the page below.



**Step 3:**

Our wp-frontend(WordPress container) cannot access the wp-backend(MySQL container) so we create a network and add both of them into it. It's like putting them in a same room so they can talk to each other.

Use docker network create --attachable wp-network to create a network

* --attachable -> Lets you add containers manually to this network.

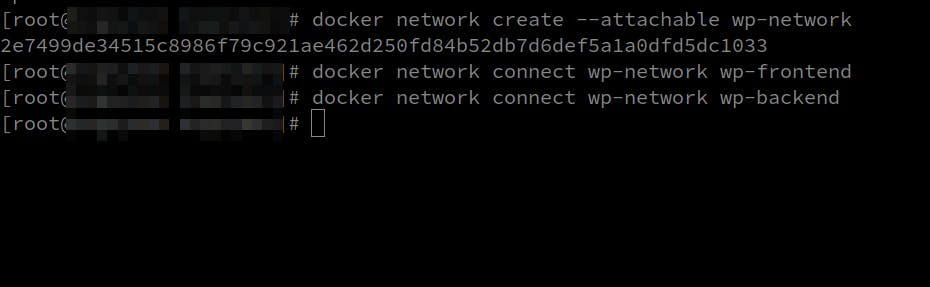
Now we should addboth **wp-frontend** and **wp-backend** containers into the **wp-network** network.

COPY

COPY

docker network connect wp-network wp-frontend

docker network connect wp-network wp-backend



**Step 4:**

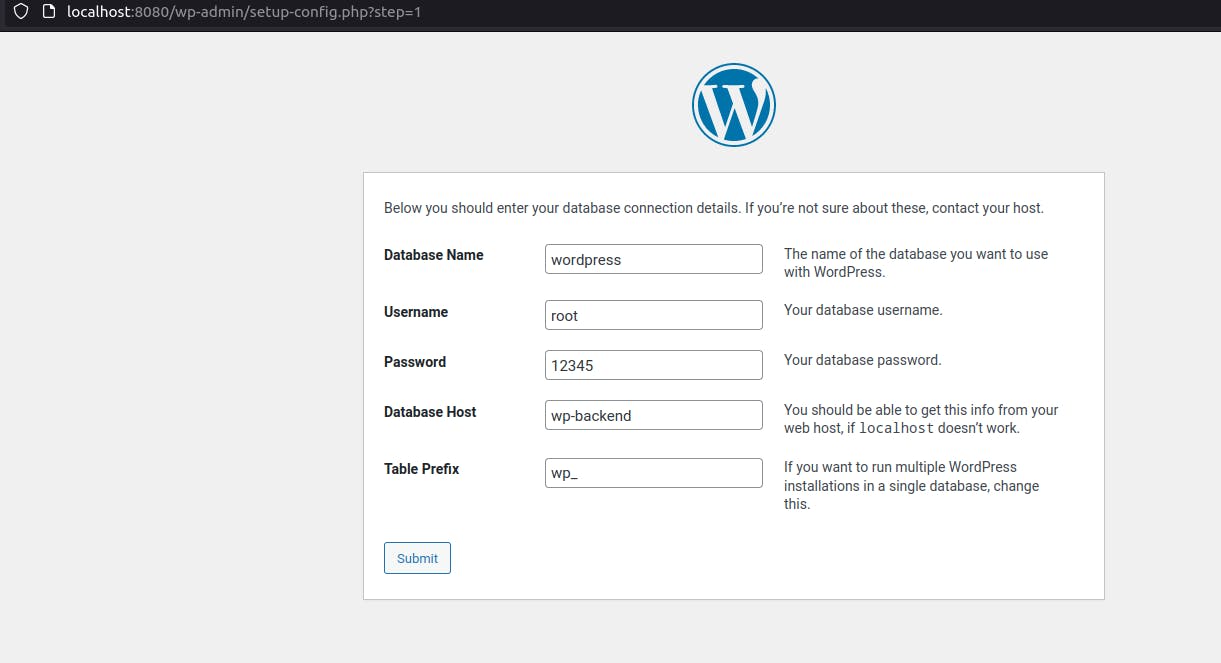
Now go back to localhost:8080 and add the details

Database Name -> wordpress

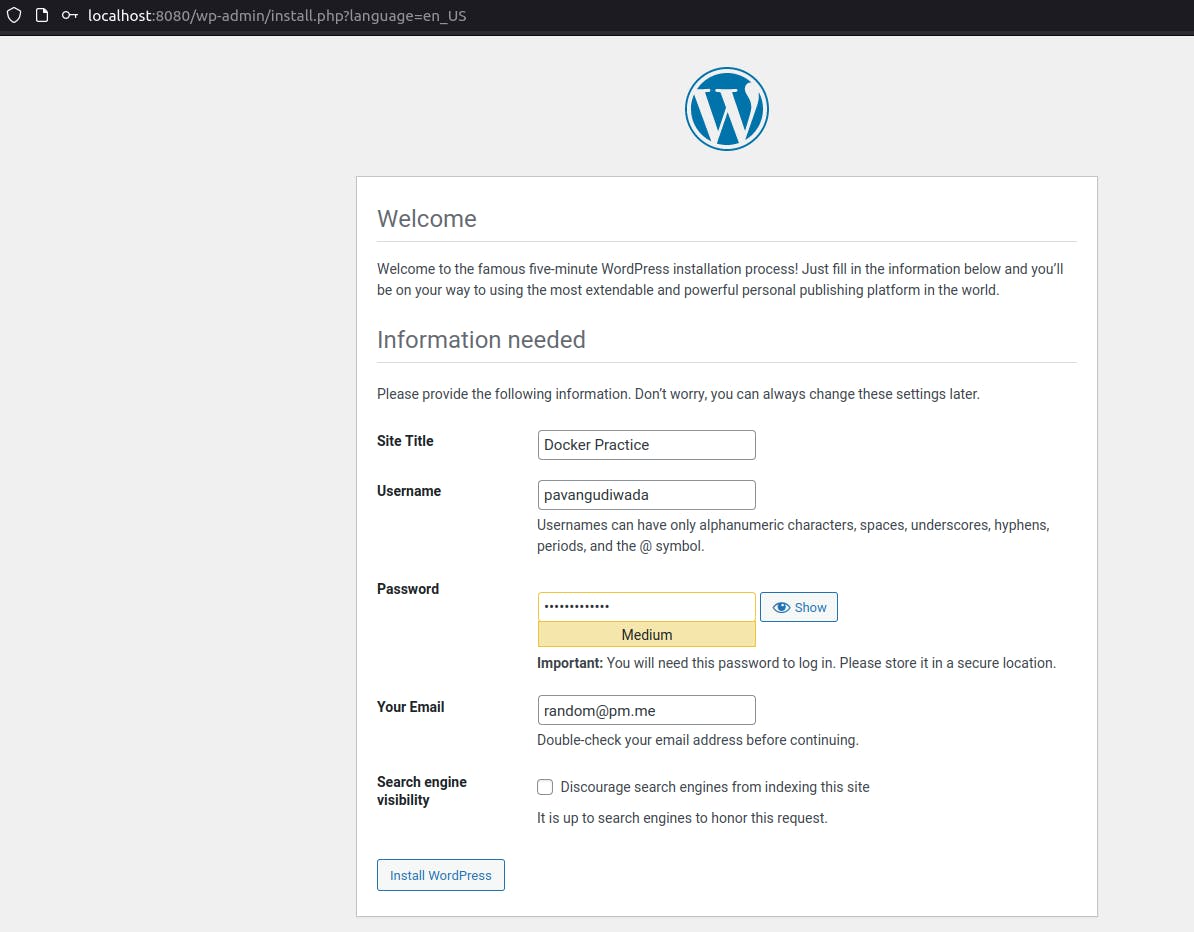
Username -> root

Password -> the password you set(MYSQL\_ROOT\_PASSWORD)

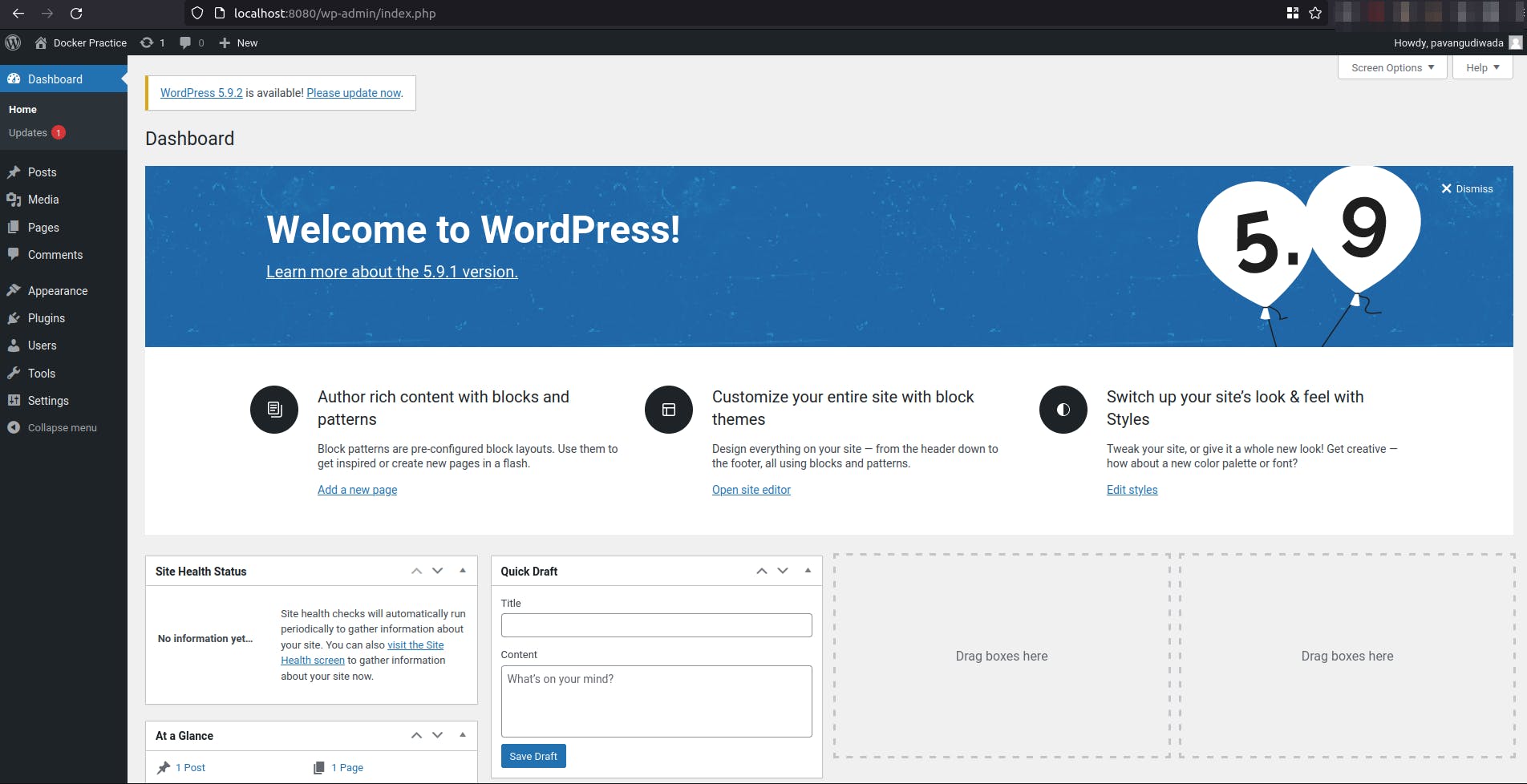
Database Host -> wp-backend



Click **Submit**, then give in the user details and **Install**.



Then **login** with your details and you will be in your WordPress homepage.



**Tips:**

* Use docker stop containerID to stop the container if you are unable to remove it.
* Thendocker rm ContainerID to remove the container.
* You can use docker rm ContainerID -f to force stop and remove the container.

**Resources used:**

[How to Easily Setup WordPress Using Docker](https://medium.com/@habibridho/how-to-easily-setup-wordpress-using-docker-a798081dc577)

[MySQL Docker hub](https://hub.docker.com/_/mysql)

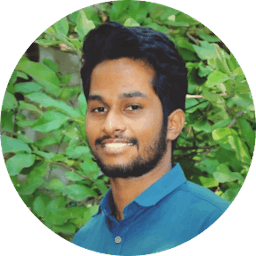
[WordPress Docker hub](https://hub.docker.com/_/wordpress)

Thank you for reading, you can find me on [Twitter](https://twitter.com/pavangudiwada_), read my blogs on [Hashnode](https://pavangudiwada.hashnode.dev/" \t "_blank) and [Medium](https://pavangudiwada.medium.com/). Have a great time

4

2

[Docker](https://pavangudiwada.hashnode.dev/tag/docker?source=tags_bottom_blogs)[WordPress](https://pavangudiwada.hashnode.dev/tag/wordpress?source=tags_bottom_blogs)[MySQL](https://pavangudiwada.hashnode.dev/tag/mysql?source=tags_bottom_blogs)[Devops](https://pavangudiwada.hashnode.dev/tag/devops?source=tags_bottom_blogs)[Beginner Developers](https://pavangudiwada.hashnode.dev/tag/beginners?source=tags_bottom_blogs)

[](https://hashnode.com/@pavangudiwada)

**WRITTEN BY**

[**Pavan Gudiwada**](https://hashnode.com/@pavangudiwada)

Follow

Azure DevOps engineer always interested in learning DevOps, OpenSource and Technical Writing.

I also tweet simplified DevOps guides. Find me on Twitter and my content on Hashnode and Medium.

**MORE ARTICLES**

[](https://hashnode.com/@pavangudiwada)

[**Pavan Gudiwada**](https://hashnode.com/@pavangudiwada)

[](https://pavangudiwada.hashnode.dev/simple-docker-project-part-2?source=more_articles_bottom_blogs)

[**Simple Docker Project - Part 2**](https://pavangudiwada.hashnode.dev/simple-docker-project-part-2?source=more_articles_bottom_blogs)

[In the last project we created a WordPress site with MySQL backend. The process was manual and is ca…](https://pavangudiwada.hashnode.dev/simple-docker-project-part-2?source=more_articles_bottom_blogs)

[](https://hashnode.com/@pavangudiwada)

[**Pavan Gudiwada**](https://hashnode.com/@pavangudiwada)

[](https://pavangudiwada.hashnode.dev/create-and-push-a-git-repo-to-github-in-minutes?source=more_articles_bottom_blogs)

[**Create and Push a Git Repo to Github in minutes!!**](https://pavangudiwada.hashnode.dev/create-and-push-a-git-repo-to-github-in-minutes?source=more_articles_bottom_blogs)

[Create your first Github repo Creating your first repo and pushing files into it can be a huge step …](https://pavangudiwada.hashnode.dev/create-and-push-a-git-repo-to-github-in-minutes?source=more_articles_bottom_blogs)

[](https://hashnode.com/@pavangudiwada)

[**Pavan Gudiwada**](https://hashnode.com/@pavangudiwada)

[](https://pavangudiwada.hashnode.dev/host-your-golang-app-on-azure-app-services-using-docker?source=more_articles_bottom_blogs)

[**Host your Golang app on Azure App Services using Docker**](https://pavangudiwada.hashnode.dev/host-your-golang-app-on-azure-app-services-using-docker?source=more_articles_bottom_blogs)

[This blog is a detailed breakdown of my learning's from Microsoft Reactor's How to use Azure to Kick…](https://pavangudiwada.hashnode.dev/host-your-golang-app-on-azure-app-services-using-docker?source=more_articles_bottom_blogs)

©2023 Tech Uncomplicated

[Archive](https://pavangudiwada.hashnode.dev/archive)**·**[Privacy policy](https://hashnode.com/privacy?source=blog-footer)**·**[Terms](https://hashnode.com/terms?source=blog-footer)

[**Publish with Hashnode**](https://hashnode.com/onboard?unlock-blog=true&source=blog-footer)

Powered by [Hashnode](https://hashnode.com/?source=blog-footer) - Home for tech writers and readers