**Docker Install:**

**Sudo apt updae**

**Sudo apt upgrade**

#type Y when it ask you to confirm

**Sudo apt install docker.io**

#type Y when it ask you to confirm

**Sudo service docker status**

**#**this command check whether the docker is running, if it is inactive use “sudo service docker start” to start it.

**sudo curl -L "https://github.com/docker/compose/releases/download/1.29.2/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose**

**#**this command download the docker-compose file from url github.com.........etc and download it to our local directory.

**sudo chmod +x /usr/local/bin/docker-compose**

#add the execute permission to the file we just download.

**Docker-compose –version**

**#**this command is used to check if the docker compose is installed successfully.

After install the docker and compose, we can go to next step which WordPress install

**WordPress install:**

**Cd**

**#** first of all , I like make project in user’s home directory. You can choose whatever directory you like.

**Sudo Mkdir word\_press**

# make a empty directory for project

**Sudo Cd word\_press**

**#**go to this folder

**Sudo Vi docker-compose.yml**

**#**create the file named “docker-compose.yml”

#Copy below code to the docker-compose.yml.

version: "3.9"

services:

db:

image: mysql:5.7

volumes:

- db\_data:/var/lib/mysql

restart: always

environment:

MYSQL\_ROOT\_PASSWORD: somewordpress

MYSQL\_DATABASE: wordpress

MYSQL\_USER: wordpress

MYSQL\_PASSWORD: wordpress

wordpress:

depends\_on:

- db

image: wordpress:latest

volumes:

- wordpress\_data:/var/www/html

ports:

- "8000:80"

restart: always

environment:

WORDPRESS\_DB\_HOST: db:3306

WORDPRESS\_DB\_USER: wordpress

WORDPRESS\_DB\_PASSWORD: wordpress

WORDPRESS\_DB\_NAME: wordpress

volumes:

db\_data: {}

wordpress\_data: {}

#this file contain 2 service. First one is DB(database) and second one is WordPress.

1. Image: means the image used for the service, for example. Db use mysql:5.7and WordPress use the newest WordPress version.
2. Volumes: means mount this service to a directory. Allowing you to modify the code on the fly, without having to rebuild the image
3. Restart: always means always restart.
4. Environment: means set the environment variable.(user, password, rootpassword etc…)
5. Depends\_on: in WordPress service means the order of service to start. WordPress depends on db.so the db. should start first because WordPress depends on it.
6. Ports means the port to visit this service, for example I use my “Ip address:8000” to visit my WordPress website

**Sudo docker-compose up -d**

when you see the below notification, you are good to go, wait for several minutes then go to the browser.

“Status: Downloaded newer image for wordpress:latest

Creating my\_wordpress\_db\_1 ... done

Creating my\_wordpress\_wordpress\_1 ... done”

电脑萤幕的截图

描述已自动生成

Use ip address:port to visit your wordpress website. The default port is 8000. Select a language and fill your information about your account.

图形用户界面, 应用程序

描述已自动生成

After type your information and password. You will direct to login page.

图形用户界面, 应用程序

描述已自动生成

Use the password you set and log in to the main page.

电脑软件截图

描述已自动生成

Write the first blog and published it.

图形用户界面, 文本

描述已自动生成