

Pratik Agrawal

500123601

Devops B2

Lab Exercise 6- Docker-Compose file

Objective:

Set up a WordPress environment using Docker Compose, including a MySQL database as the backend.

Prerequisites:

- Docker and Docker Compose installed on your system.

Step 1: Create a docker-compose.yml File

1. In the project directory, create a file named docker-compose.yml.
2. Add the following content to docker-compose.yml:

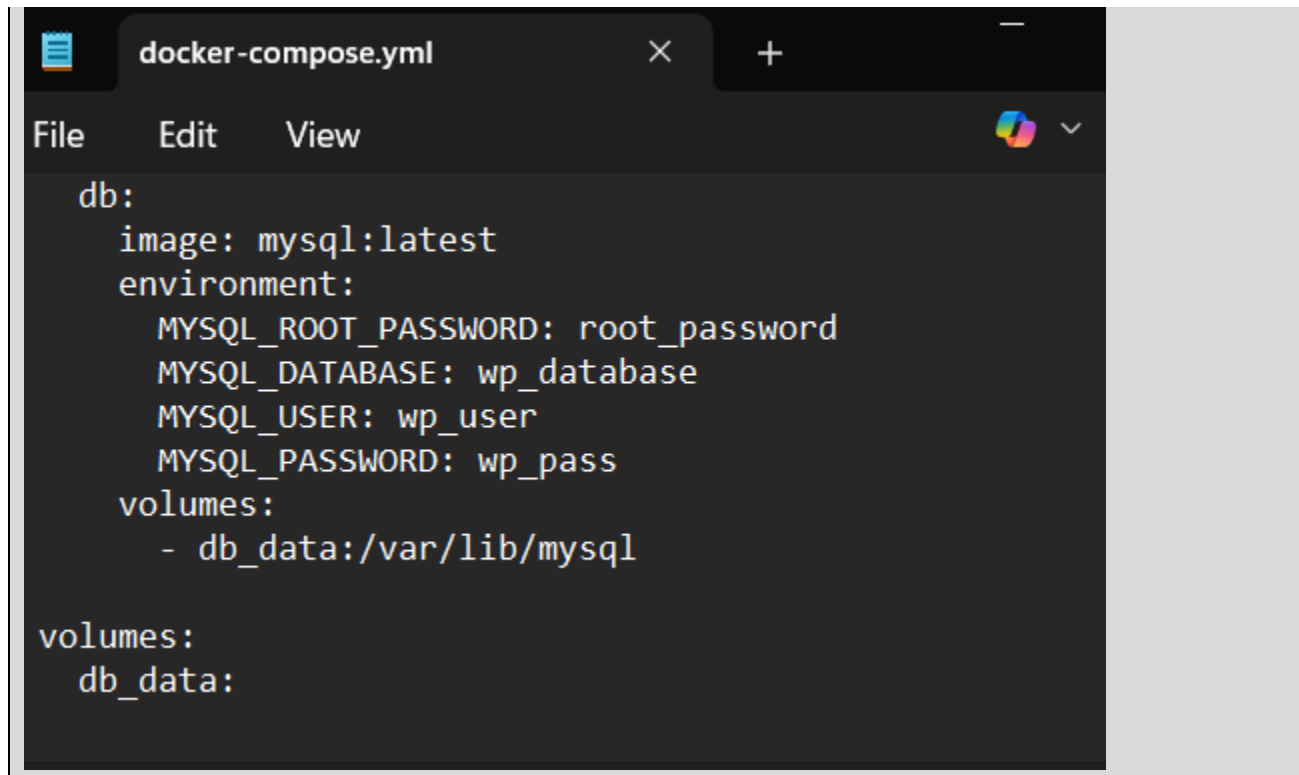
docker-compose.yml

```
version: '3.8'

services:
  wordpress:
    image: wordpress:latest
    ports:
      - "8002:80"
    environment:
      WORDPRESS_DB_HOST: db:3306
      WORDPRESS_DB_USER: wp_user
      WORDPRESS_DB_PASSWORD: wp_pass
      WORDPRESS_DB_NAME: wp_database
    depends_on:
      - db

  db:
    image: mysql:latest
    environment:
      MYSQL_ROOT_PASSWORD: root_password
      MYSQL_DATABASE: wp_database
      MYSQL_USER: wp_user
      MYSQL_PASSWORD: wp_pass
    volumes:
      - db_data:/var/lib/mysql

volumes:
  db_data:
```

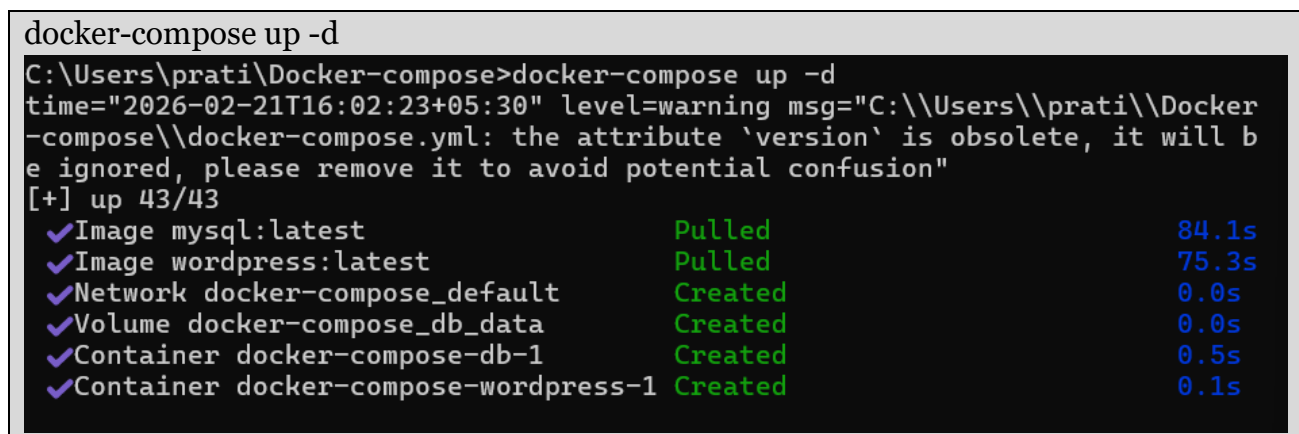
A screenshot of a code editor window titled 'docker-compose.yml'. The editor has a dark theme and shows the following YAML configuration:

```
db:
  image: mysql:latest
  environment:
    MYSQL_ROOT_PASSWORD: root_password
    MYSQL_DATABASE: wp_database
    MYSQL_USER: wp_user
    MYSQL_PASSWORD: wp_pass
  volumes:
    - db_data:/var/lib/mysql

volumes:
  db_data:
```

Step 2: Start the Containers

1. Run the following command to start the containers:

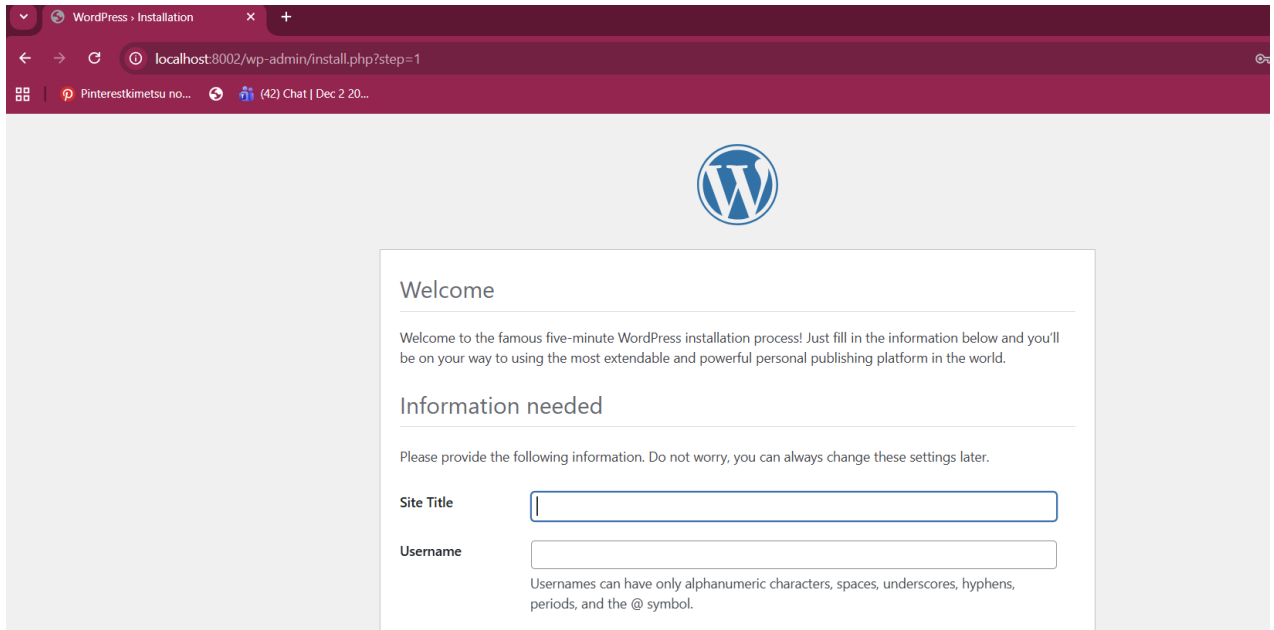
A screenshot of a terminal window showing the output of the 'docker-compose up -d' command. The output indicates that the containers were successfully started.

```
docker-compose up -d
C:\Users\prati\docker-compose>docker-compose up -d
time="2026-02-21T16:02:23+05:30" level=warning msg="C:\\Users\\prati\\Docker
-compose\\docker-compose.yml: the attribute 'version' is obsolete, it will b
e ignored, please remove it to avoid potential confusion"
[+] up 43/43
✓Image mysql:latest Pulled 84.1s
✓Image wordpress:latest Pulled 75.3s
✓Network docker-compose_default Created 0.0s
✓Volume docker-compose_db_data Created 0.0s
✓Container docker-compose-db-1 Created 0.5s
✓Container docker-compose-wordpress-1 Created 0.1s
```

2. Docker Compose will download the necessary images (WordPress and MySQL) and start both services.

Step 4: Access WordPress

1. Open your web browser and go to **http://localhost:8002**
2. Follow the WordPress installation steps to set up your site.



Step 5: Stop and Remove Containers

To stop the containers and remove the associated resources, run:

```
docker-compose down
```

```
C:\Users\prati\Docker-compose>docker-compose down
time="2026-02-21T16:05:59+05:30" level=warning msg="C:\\Users\\prati\\Docker
-compose\\docker-compose.yml: the attribute 'version' is obsolete, it will b
e ignored, please remove it to avoid potential confusion"
[+] down 3/3
✓Container docker-compose-wordpress-1 Removed 1.4s
✓Container docker-compose-db-1 Removed 1.0s
✓Network docker-compose_default Removed 0.3s
```

Explanation of docker-compose.yml:

- **wordpress:** Sets up the WordPress container, mapping port 80 inside the container to port 8002 on your local machine.
- **db:** Sets up the MySQL container with a volume (db_data) for persistent storage.

Additional Notes:

- Modify the environment variables as needed for different configurations.
- To view logs, use `docker-compose logs -f`.

This setup allows you to quickly start a WordPress site locally and experiment with configurations.