Lab Exercise 2- 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:

```
version: '3.8'

services:
  wordpress:
    image: wordpress:latest
    ports:
      - "8080:80"
    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:5.7
    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:

```
docker-compose up -d
```

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:8080.
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
```

Explanation of docker-compose.yml:

- **wordpress**: Sets up the WordPress container, mapping port 80 inside the container to port 8080 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.