DevOps Assignment 2: Docker (25 points)

Docker is a key technology in containerization, providing developers and DevOps engineers with an efficient way to package and distribute applications.

Theory Questions (10 points)

a) **Docker Fundamentals**

- What is a Docker container, and how is it different from a virtual machine (VM)?
- What is the purpose of a Dockerfile? Explain the significance of directives like FROM, COPY, RUN, and CMD.

b) **Image Management**

- Describe the layers of a Docker image. How does Docker optimize space and performance using these layers?
- What are the benefits of using Docker volumes? Give an example where data persistence is crucial in a Docker container.

c) **Networking in Docker**

- How does Docker handle networking? Explain the difference between bridge, host, and none network modes in Docker.
- Describe how you would configure container-to-container communication within a Docker network.

Practical Task (15 points)

a) **Dockerfile Creation**

- Write a Dockerfile that:
    - Uses Ubuntu as the base image.
    - Installs Nginx and serves a custom index.html file.
    - Exposes port 8080 and starts Nginx in the foreground.
    - The index.html file should display a custom message, such as "Welcome to DevOps World!"

- Build the Docker image and verify its functionality.

b) **Multi-Container Setup**

- Create a Docker Compose file to set up a simple web application stack:
    - One container running an Nginx web server.
    - One container running a PostgreSQL database.
    - Ensure that the web server can communicate with the database container, and the database is persistent using a Docker volume.

c) **Resource Limiting**

- Run your web server container with memory and CPU usage limits (e.g., limit the container to use a maximum of 512MB memory and 1 CPU core).
- Provide the command you used and explain how Docker resource limits can help in a production environment.