Lab 2: Building, Pushing, Pulling, and Running a Docker Image from Docker Hub

Step 1: Setting Up the Project Directory

1. **Create a project directory**:

   ```
   mkdir my-dockerhub-lab
   cd my-dockerhub-lab
   ```

Step 2: Creating the Dockerfile

2. **Create a file named `Dockerfile`**:

   ```
   touch Dockerfile
   ```

3. **Open the Dockerfile in a text editor** and add the following content:

   ```
   # Use an official Python runtime as a base image
   FROM python:3.9-slim

   # Set the working directory in the container
   WORKDIR /app

   # Copy the current directory contents into the container at /app
   COPY . .

   # Install any needed packages specified in requirements.txt
   RUN pip install --no-cache-dir -r requirements.txt

   # Make port 80 available to the world outside this container
   EXPOSE 80

   # Define the command to run the application
   CMD ["python", "-m", "http.server", "80"]
   ```

Step 3: Creating Content for the Container

4. **Create a simple `requirements.txt` file**:

   ```
   echo > requirements.txt
   ```

5. **Create a simple HTML file to serve**:

   ```
   touch index.html
   ```

6. **Open the `index.html` file in a text editor** and add the following content:

   ```
   <!DOCTYPE html>
   <html lang="en">
   <head>
      <meta charset="UTF-8">
      <meta name="viewport" content="width=device-width, initial-scale=1.0">
      <title>Welcome to Docker Hub Lab</title>
   ```

```
</head>
<body>
   <h1>Welcome to Docker Hub Lab</h1>
   <p>This is a simple web page served by a Docker container.</p>
</body>
</html>
```

Step 4: Building the Docker Image

7. **Build the Docker image**:

   ```
   docker build -t my-simple-webserver .
   ```

Step 5: Running and Testing the Docker Image Locally

8. **Run a container from the image**:

   ```
   docker run -d -p 80:80 --name my-webserver my-simple-webserver
   ```

9. **Open your web browser** and navigate to http://localhost or use curl to confirm. You should see the simple HTML page you created.
10. **Stop and remove the container**:

    ```
    docker stop my-webserver
    docker rm my-webserver
    ```

Step 6: Logging into Docker Hub

11. **Log in to Docker Hub**:

    ```
    docker login
    ```

    Enter your Docker Hub username and password when prompted.

Step 7: Tagging the Docker Image

12. **Tag the Docker image**:

    ```
    docker tag my-simple-webserver <your-dockerhub-username>/my-simple-webserver:latest
    ```

    Replace <your-dockerhub-username> with your Docker Hub username.

Step 8: Pushing the Docker Image to Docker Hub

13. **Push the Docker image to Docker Hub**:

    ```
    docker push <your-dockerhub-username>/my-simple-webserver:latest
    ```

Step 9: Pulling the Docker Image from Docker Hub

14. **Pull the Docker image from Docker Hub** (from any system with Docker installed):

    docker pull <your-dockerhub-username>/my-simple-webserver:latest

Step 10: Running the Pulled Docker Image

15. **Run a container from the pulled image**:

    docker run -d -p 80:80 --name my-webserver <your-dockerhub-username>/my-simple-webserver:latest

16. **Open your web browser or use curl** and navigate to http://localhost. You should see the simple HTML page you created.