

Práctica Docker y CLI

Parte 1: Manejo Básico de Contenedores y Imágenes

1. Crear y Ejecutar un Contenedor:

```
docker run -d -p 8080:80 --name mi-contenedor httpd
```

Esto descarga la imagen `httpd` y crea un contenedor llamado `mi-contenedor`.

2. Listar Contenedores:

```
docker ps
```

3. Detener un Contenedor:

```
docker stop mi-contenedor
```

4. Reiniciar un Contenedor:

```
docker start mi-contenedor
```

5. Eliminar un Contenedor:

```
docker rm mi-contenedor
```

6. Trabajar con Imágenes Docker:

- Descargar una imagen:

```
docker pull nginx
```

- Listar imágenes:

```
docker images
```

Parte 2: Dockerfile, Construcción y Registro

1. Crear el directorio del proyecto

En tu terminal, crea un nuevo directorio para tu proyecto y navega a él:

```
mkdir firstlab  
cd firstlab
```

2. Inicializar nuevo proyecto Node.JS

Inicializa el proyecto con **NPM** para crear un package.json

```
npm init -y
```

3. Crea el archivo de la aplicación NodeJS

Crea un archivo llamado `index.js` con el siguiente contenido:

```
const express = require('express');
const app = express();
const PORT = process.env.PORT || 3000;

app.get('/', (req, res) => {
  res.send('Hola Mundo');
});

app.listen(PORT, () => {
  console.log(`Servidor corriendo en el puerto ${PORT}`);
});
```

4. Agregar Express como dependencia

Añade Express, un framework de servidor web para NodeJS:

```
npm install express --save
```

5. Crear un Dockerfile para una Aplicación Node.js:

Escribe un Dockerfile básico para una aplicación Node.js.

```
FROM node:latest
WORKDIR /usr/src/app
COPY package*.json ./
RUN npm install
COPY . .
EXPOSE 3000
CMD ["node", "index.js"]
```

6. Construir la Imagen:

```
docker build -t mi-node-app .
```

7. Ejecutar la Aplicación Construida:

```
docker run -d -p 3000:3000 mi-node-app
```

8. Crea una cuenta Docker Hub

Crea una cuenta usando tu correo personal en

[Docker Hub](#)

9. Registrar en Docker Hub:

- Inicia sesión en Docker Hub: `docker login`
- Etiqueta tu imagen: `docker tag mi-node-app [TU USUARIO DOCKERHUB]/mi-node-app`
- Sube la imagen: `docker push [TU USUARIO DOCKERHUB]/mi-node-app`

10. Descargar y Ejecutar la Imagen desde Docker Hub:

```
docker run -d -p 3000:3000 [TU USUARIO DOCKERHUB]/mi-node-app
```