Docker container-Pushing images

What are Docker Images?

Docker images are templates used to create containers. They contain the application code, dependencies, and settings needed to run the application.

Pushing Images to Docker Hub

Docker Hub is a cloud-based repository where you can store and share your Docker images. To push an image to Docker Hub, follow these steps:

Step 1: Create a Docker Hub Account

If you haven't already, create a free account on Docker Hub at hub.docker.com.

Step 2: Login to Docker CLI

Run the following command in your terminal:

```
docker login
```

Enter your Docker Hub username and password when prompted.

Step 3: Tag Your Image

Tag your image with your Docker Hub username and the name of your repository. For example, if you want to push an image called `my-web-app` to your `john-doe` repository, run:

```
docker tag my-web-app john-doe/my-web-app
```

Step 4: Push Your Image

Push your tagged image to Docker Hub using the following command:

```
docker push john-doe/my-web-app
```

This will upload your image to Docker Hub.

Example Use Case

Let's say you have a simple web server application written in Node.js. You can create a Dockerfile to build an image for this app, then push it to Docker Hub.

Dockerfile

```
 FROM node:14

WORKDIR /app

COPY package*.json ./

RUN npm install

COPY . .

CMD ["npm", "start"]
```

This Dockerfile uses the official Node.js 14 image as a base and installs your app's dependencies. It then copies your code into the container and sets the default command to `npm start`.

Building and Pushing Your Image

Build your image using:

```
 docker build -t my-web-app .
```

Then, tag and push it to Docker Hub:

```
 docker tag my-web-app john-doe/my-web-app
docker push john-doe/my-web-app
```

Your `my-web-app` image is now available on Docker Hub!

Remember to replace `john-doe` with your actual Docker Hub username.

Curated by Brajesh Kumar