**﻿Docker**

Docker is an environment(software) where we can build, test and deploy our application. Docker have containers. Docker have their own Operating system.

**Q. Why docker when we can build our application locally (i.e in local-host)?**

A. If you build your app directly on your computer, you might run into problems later when you update your computer or its software. Docker solves this by creating a special environment for your app that stays the same no matter where it runs. This means your app will always work, even if you switch computers or update software**.**

For example:

Well, let's say you started building your React app in 2020 and everything was working fine with the packages you installed, like navigation and dropdown components. But then, a year later, you upgraded your computer or installed a new version of Node.js, and suddenly things stopped working because some packages didn't support the new version. With Docker, you create a special environment for your app called a container, and you build and run your app inside that container. Even if you update Node.js or anything else, it won't mess up your app because it's running in Docker's special environment. Plus, if you share your app with a friend, they won't have to worry about installing all the specific dependencies because Docker packages everything together neatly, making it easy for them to run your app without any fuss.

Docker lets you run an application on any operating system.

There was an example, once I built a webpage in next.js in my computer, and my computer was a window based/linux based. Now what happened was I built my webpage and then it was working completely fine on my system, then I sent the code to my brother, he was using mac, and then since mac has their own Operating system and while he was trying to build/run my webpage from code, he had some error such as some dependecy/package/liubrary I don’t know was not supported on mac. So what to do? There was a situation that some functionality could not be used or work in mac, so does It mean mac users cant run everything that runs on linux? Yes that may be case since operating system can support one thing on on system and may not on other. So what to do? I built my system in docker environment and then build and deploy in docker, and since docker has their own environment so I run my webpage in docker, and then I setup my code for docker and then share it to my brother, he run it inside his docker and it ran, because we were running our code in docker and our code was independent of any operating system we use, so we can code anything on any operating system freely

**Q. I built a webpage using Next.js on my Windows/Linux computer, and it worked fine. But when I sent the code to my brother who uses a Mac, he couldn't run it due to compatibility issues with Mac's operating system. What can be done in such situations?**

A. It's true that sometimes certain functionalities may not work across different operating systems. However, Docker comes to the rescue here. With Docker, I rebuilt my webpage in a Docker environment, which provides its own isolated system. So, when I shared the Dockerized code with my brother, he could easily run it on his Mac inside a Docker container. Since Docker creates a consistent environment regardless of the underlying operating system, our code became independent of any specific OS. This way, we could freely develop and share our code without worrying about compatibility issues between different systems.

Last updated at: 3/28/2024