

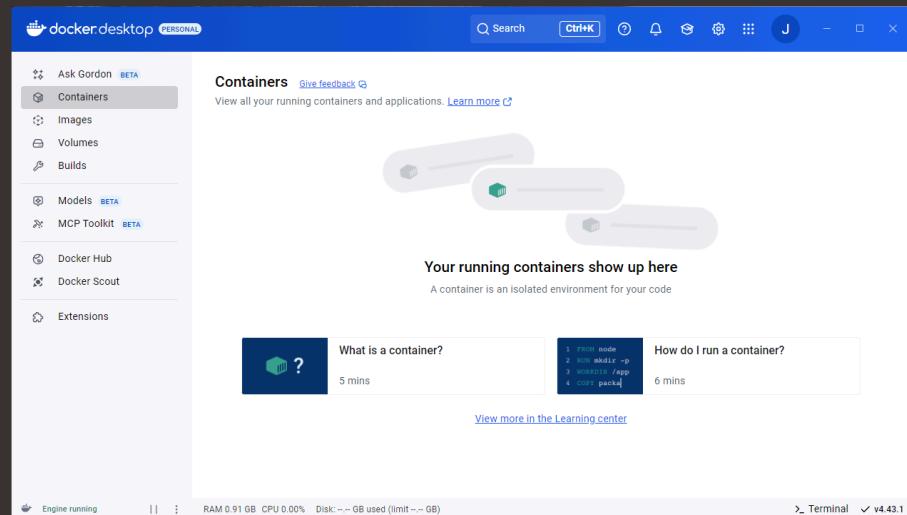


nextwork.org

Deploy an App with Docker

ME

Melvin J Bonner



Introducing Today's Project!

What is Docker?

Docker is a tool for creating and managing containers. Docker is the foundational tool for this project and is used for creating and managing containers.

One thing I didn't expect...

The one thing I didn't expect with this project was the ease of use of Elastic Beanstalk. The ability to just upload another file for an update was unexpected.

This project took me...

This project took me 3 hours.

Understanding Containers and Docker

Containers

Containers are a package of your application and everything it needs to run (dependencies) in one file. They are useful because they allow you run other developers' applications and software much faster, and containers are a tool for making software development and releases more efficient!

A container image is a blueprint or template for containers. It gives Docker instructions on what to include in a container, such as application code, libraries, dependencies, and necessary files.

Docker

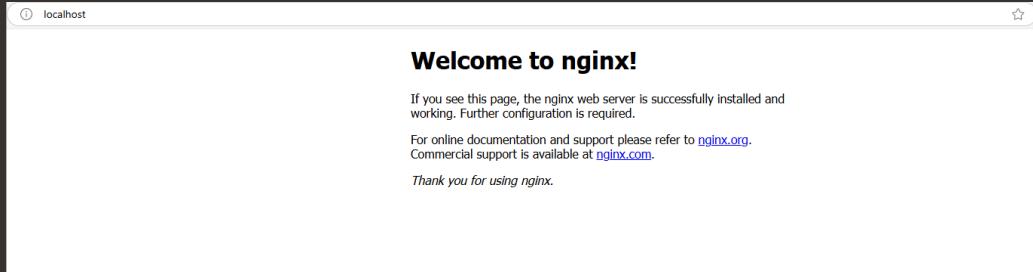
Docker is a tool for creating and managing containers. Docker Desktop is a tool that makes it easy to work with Docker.

The Docker deamon is the background process that manages the Docker containers on your computer.

Running an Nginx Image

Nginx is a web server, which means it's a program that serves web pages to people on the internet.

The command I ran to start a new container was docker run.



Creating a Custom Image

The Dockerfile is a document with all the instructions for building my Docker image. Docker would read a Dockerfile to understand how to set up your application's environment and which software packages it should install.

My Dockerfile tells Docker three things, the first line, FROM nginx:latest, means our image starts as a copy of the latest Nginx image, the next line, COPY index.html /usr/share/nginx/html/ replaces the default HTML file provided by Nginx with my own custom index.html file, the last line, EXPOSE 80 means I want the container to receive web traffic through port 80.

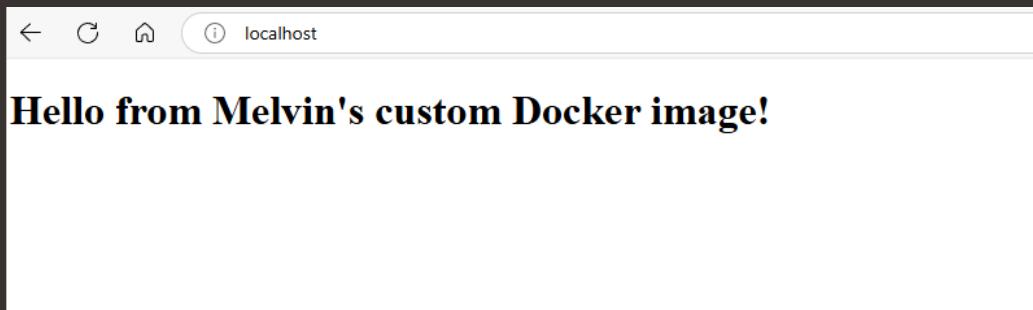
The command I used to build a custom image with my Dockerfile was 'docker build -t my-web-app .' The '.' at the end of the command tells Docker to find the Dockerfile in the current directory.

```
Dockerfile ✘ X
1  FROM nginx:latest
2  COPY index.html /usr/share/nginx/html/
3  EXPOSE 80
4
```

Running My Custom Image

There was an error when I ran my custom image because there was already a container using port 80. I resolved this by stopping the container that was using port 80.

In this example, the container image is the blueprint that tells Docker the application code, dependencies, and libraries that should go into a container. The container is the actual software that is created from this image and running the web server displaying my index.html.



Elastic Beanstalk

Elastik Beanstalk is a service that makes it easy to deploy cloud applications without worrying about the underlying infrastructure.

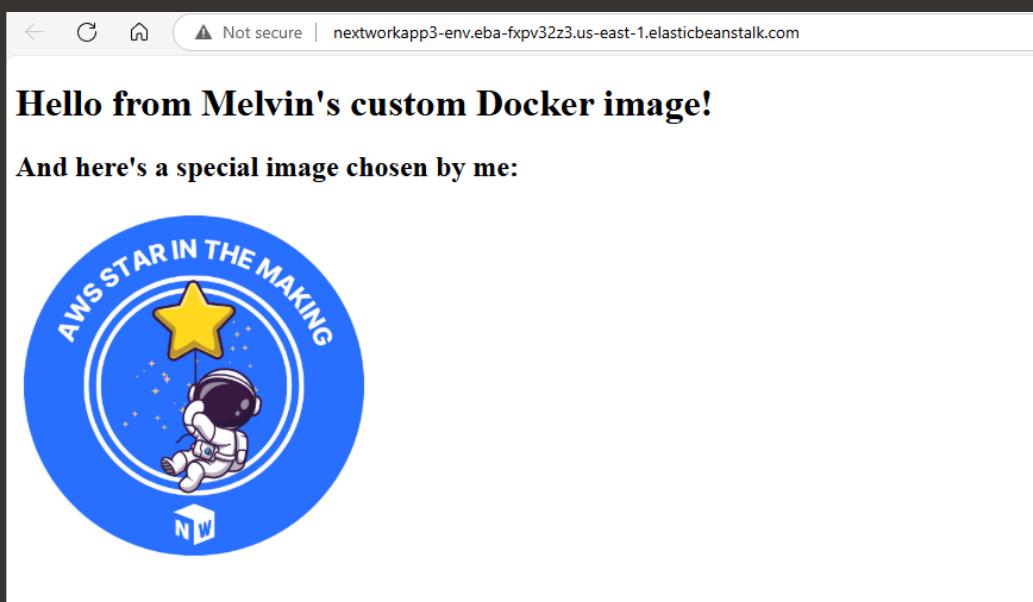
Deploying my custom image with Elastic Beanstalk took me 2 hours.



Deploying App Updates

To learn how to deploy app updates with Elastic Beanstalk, I updated my app by adding additional lines of text. I verified those changes by viewing the updated html file in my browser.

My app updates didn't show up in my live environment immediately because my updates were only done on my local computer. To deploy my changes, I only had to select upload and deploy and upload my latest html file.





nextwork.org

The place to learn & showcase your skills

Check out nextwork.org for more projects

