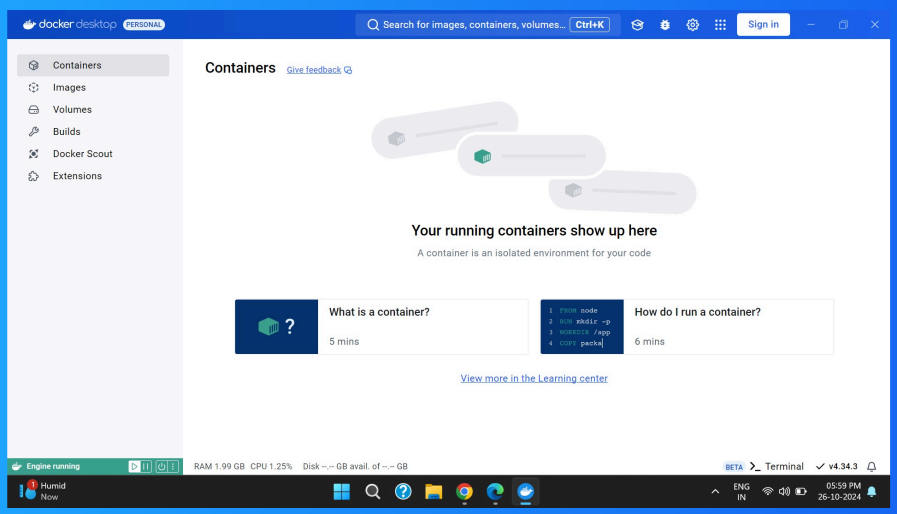
NextWork.org

# Containers on Elastic Beanstalk

Abhishek Bhagat



**Abhishek Bhagat**
NextWork Student



# Introducing Today's Project!

## What is Docker?

Docker is a software development platform to deploy apps on AWS. Today we built custom container image and ran the containerized application locally. Lastly deployed containerized application to AWS Elastic Beanstalk, and access it live on the web.

## One thing I didn't expect...

One thing I didn't expect was it being seamlessly managed by the Docker Desktop and its integration with Docker to host an Application live on the web.

## This project took me...

It took approximately 2 hrs to do this project as in between I ran into Terminal issues but was able to resolve it afterwards.

**Abhishek Bhagat**
NextWork Student



# Understanding Containers and Docker

## Containers

Containers are basically the package of an Application and everything it needs to run in one file. They are useful as developers/users can run this package instead of the application itself, which gets the application working much faster.

A container image is a template used to create containers. It contains the application code, libraries, dependencies, and other files needed to run the application.

## Docker

Docker is a software platform that allows developers to build, test, and deploy applications quickly on AWS. Docker Desktop lets you manage everything about your containers. You can create, adjust their settings, or monitor how they run.

Docker daemon is a background process that manages the Docker containers on your computer. It takes commands from the Docker client and does the heavy lifting of building, running, and distributing your containers.

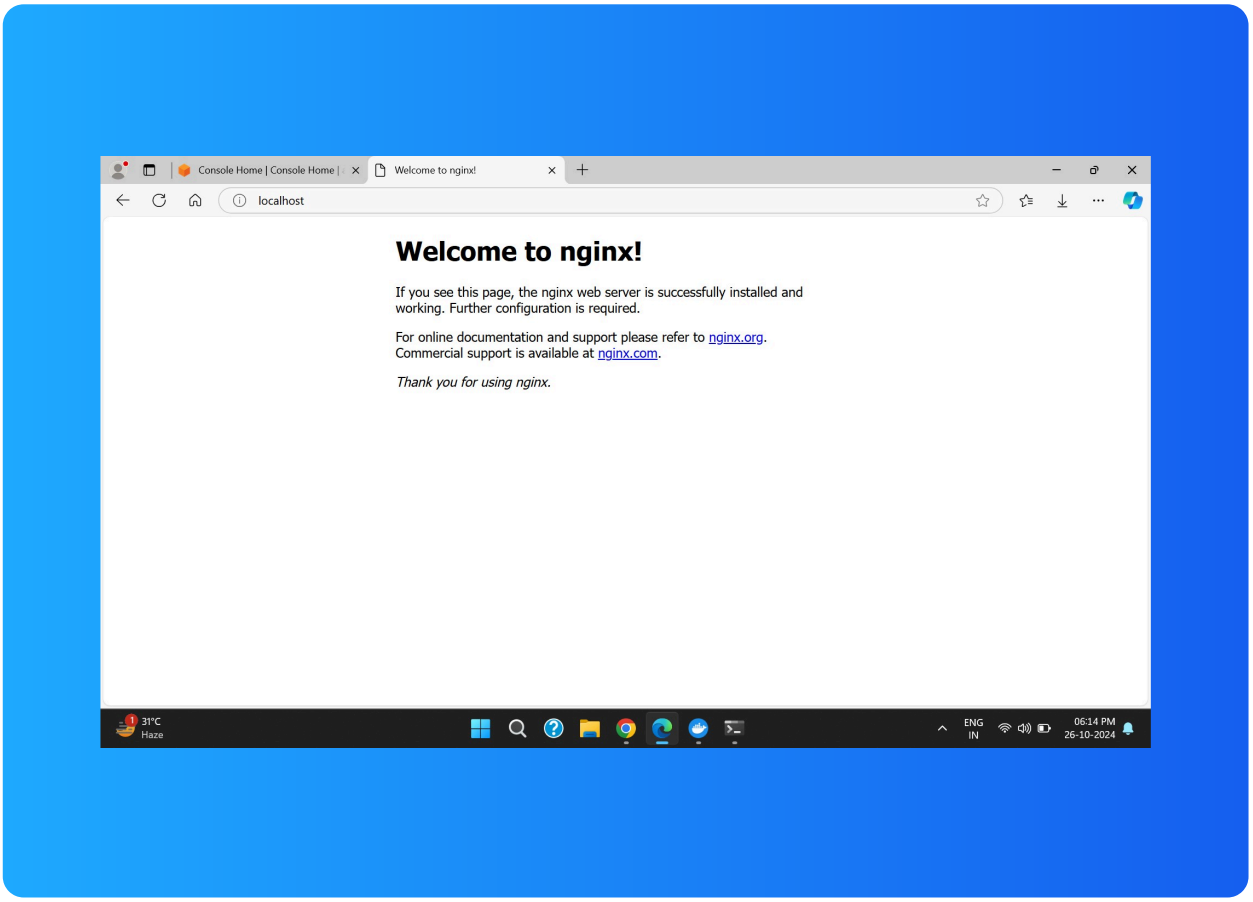
**Abhishek Bhagat**
NextWork Student



# Running an Nginx Image

Nginx (pronounced as "engine-x") is a web server, which means it's a program that serves web pages to people on the internet. Engineers use Nginx because it can handle lots of web traffic smoothly and efficiently.

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



**Abhishek Bhagat**
NextWork Student

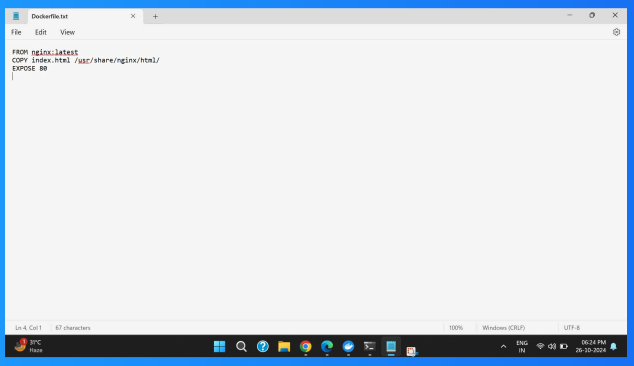


# Creating a Custom Image

The Dockerfile is a document with all the instructions for building your 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: FROM nginx:latest COPY index.html /usr/share/nginx/html/ EXPOSE 80

The command I used to build a custom image with my Dockerfile was the docker build -t my-web-app .



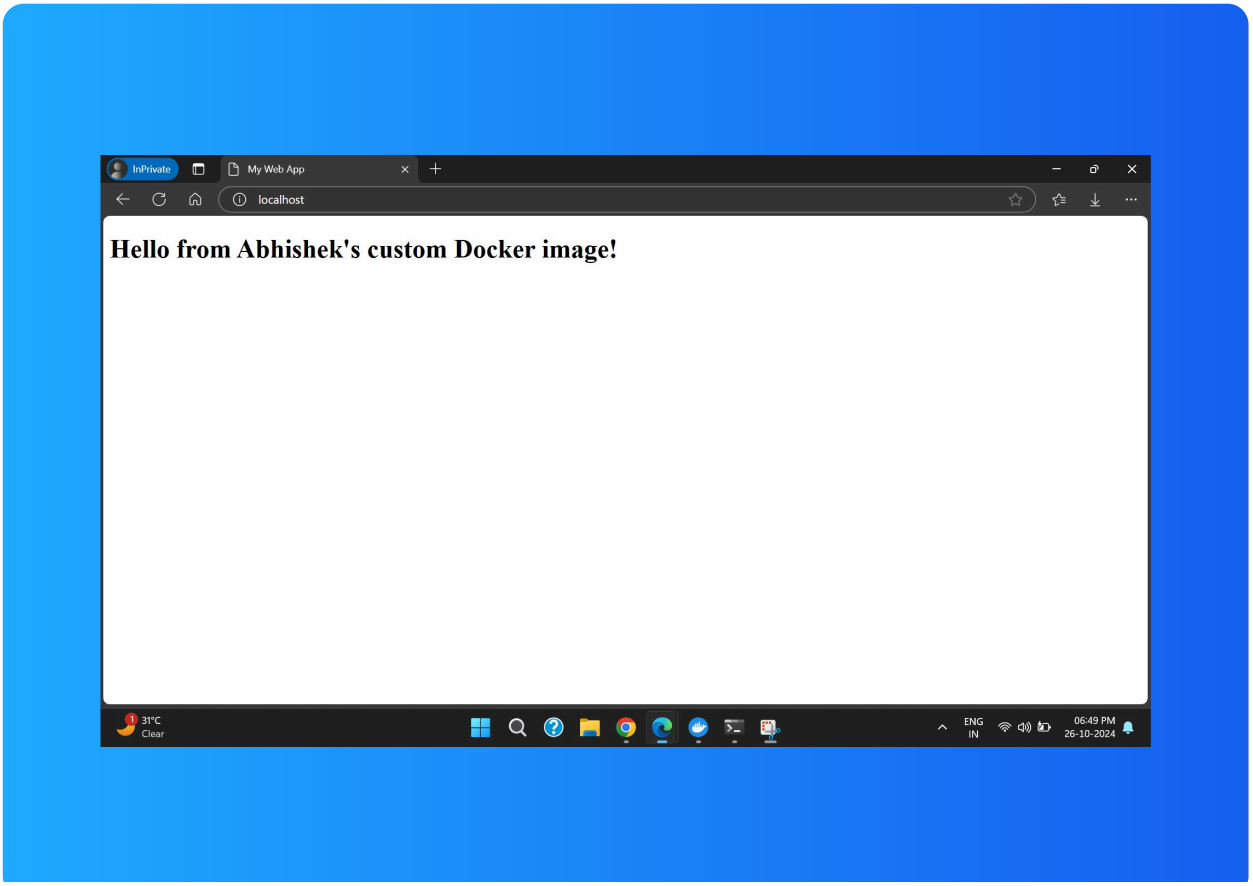
**Abhishek Bhagat**
NextWork Student



# Running My Custom Image

There was an error when I ran my custom image because there's already a container using port 80, so the new container I'm creating can't access it. I resolved this by going to Docker Desktop and stopped the first container which made port 80 free.

In this example, the container image is the blueprint that tells Docker the application code, dependencies, libraries etc that should go into a container while container is the actual software that's created from this image and running the web server



**Abhishek Bhagat**
NextWork Student

NextWork.org

# Elastic Beanstalk

AWS Elastic 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 5-7 mins after initial configuration.