Hands-on Lab: Write a Dockerfile for a Node app deployment

Estimate Time: 25 minutes

Introduction

In this lab, you will explore how generative AI can assist in creating a Dockerfile for a Node.js application. You will utilize IBM's generative AI lab, powered by advanced language models. The goal is to generate the necessary code for the Dockerfile, simplifying the development of the Node.js application in a Docker environment.

Learning Objectives

After completing this exercise, you will be able to:

- Understand and Utilize IBM's generative AI lab to generate appropriate responses for queries related to Dockerfile.
- Develop a Dockerfile using GenAI for a node.js application
- Ensure clarity and precision in providing details when seeking results from ChatGPT

Please note Generative AI is an evolving field. As you attempt the labs, your experience and output might differ from here.

Prologue

What is a Dockerfile, and why use it?

A Dockerfile is like a recipe for your application's environment. It contains instructions that Docker uses to build a Docker image. This image, in turn, is a lightweight, standalone, and executable package that includes everything needed to run a piece of software, including the code, runtime, libraries, and system tools.

Dockerfiles are crucial in containerization, offering a standardized and reproducible way to package and deploy applications. They allow developers to encapsulate their applications and dependencies, ensuring consistent behavior across different environments. Dockerfiles simplify the development process and facilitates seamless collaboration and deployment.

Start generating Dockerfile

In case you need familiarity with the Interface/classroom please see the This Lab for reference.

Once your AI Classroom is ready, let's proceed and ask Generative AI for assistance writing a Dockerfile.

1. Enter the following question in the message box: `What is Dockerfile ?`

Now, view the AI's response for an explanation of a Dockerfile.

▶ Click here to view the sample response generated

2. Enter the following question in the message box: `What are the requirements for a Dockerfile?`

Next, read the AI's response and proceed to ask questions about Dockerfile requirements.

▶ Click here to view the sample response generated

3. Enter the following question in the message box: `What's the first crucial step when creating a Dockerfile for a Node.js application?`

It is essential to understand the first step after selecting the base image for a Node.js Dockerfile.

▶ Click here to view the sample response generated

4. Enter the following instructions in the message box: `Following the base image selection, Set the working directory inside the container to /app`

Then, you will learn about the next step to define the working directory inside the container.

▶ Click here to view the sample response generated

5. Enter the following instructions in the message box: `As selected working directory /app, bring the package.json and package-lock.json to the working directory`

After that, you will understand the process of bringing package files into the working directory.

▶ Click here to view the sample response generated

6. Enter the following questions in the message box: `Now, how do I install dependencies? And copy the remaining application code to the working directory?`

You will then get guidance on installing dependencies and copying the remaining application code.

▶ Click here to view the sample response generated

7. Enter the following question in the message box: `Node.js app needs to be accessible on a specific port. What Dockerfile instruction do you include for this purpose?`

Then, you will learn the instructions to include for exposing a specific port.

▶ Click here to view the sample response generated

8. Enter the following question in the message box: `How to specify the command to start the Node.js application using CMD?`

After that, you will understand the final step in specifying the command to start the Node.js application.

▶ Click here to view the sample response generated

9. Enter the following instructions in the message box: `Combine all the steps to form a complete Dockerfile for a Node.js application`

Now, you have completed all of the steps so you can view the generated Dockerfile.

▶ Click here to view the sample response generated

Summary

In this lab, you leveraged generative AI to simplify the creation of a Dockerfile for a Node.js application. You did this by interacting with the AI in the designated classroom and addressing key questions and steps involved in Dockerfile development. The generated Dockerfile serves as a foundation for deploying Node.js applications in a containerized environment. This experience showcases the potential of generative AI in enhancing the efficiency of Docker-based development workflows. Through guided interactions, you and othere can gain valuable insights into the Dockerfile creation process, making it more accessible and practical for real-world application deployments.

Congratulations!

You have leveraged generative AI and developed a complete Dockerfile for a Node.js application.

Author(s)

Nikesh Kumar
Pallavi Rai