EXPERIMENT : 9

Date of Performance	
Date of Submission	

AIM

To learn Dockerfile instructions, build an image for a sample web application using Dockerfile.

PROBLEM DEFINITION

Master Dockerfile instructions to build an image for a sample web application.

THEORY

Jenkins is an open-source automation tool created with Java. It is extensively used as a Continuous Integration (CI) and Continuous Deployment (CD) tool.

Maven:

- Maven primarily intends to provide developers with:
  1. A comprehensive, reusable, easily maintainable model for projects.
  2. Plugins or tools to interact with and operate within this model.
- Maven is a POM (Project Object Model)-based build automation and project management tool written in Java. However, it is compatible with projects written in C#, Python, Ruby, etc.

A few Maven features worth mentioning are:

1. Maven can be used to build any number of projects into predefined output types like .jar, .war, metadata, etc.
2. Maven can automatically download necessary files from the repository when building a project.

Selenium using Maven in Jenkins: Selenium is a widely used test automation framework for validating web applications across different combinations of browsers, platforms, and devices (or emulators). It is widely used for testing areas such as functional testing, end-to-end testing, and more.

Why Jenkins and Selenium?

- Running Selenium tests in Jenkins allows you to run your tests every time your software changes and deploy the software to new environments when the tests pass.

Advantages of using Maven and Jenkins with Selenium:

- Whenever a change is made in the implementation, the changes are deployed on the test environment. Automation testing is performed continuously, and developers are kept informed about the build and test stage results.
- Test suites that comprise many test scenarios might take longer duration testing. A nightly build run can be scheduled for build and execution on the Jenkins server in such cases.

Traditionally, the Dockerfile is called Dockerfile and located in the root of the context. You use the -f flag with docker build to point to a Dockerfile anywhere in your file system.

$ docker build -f /path/to/a/Dockerfile .

You can specify a repository and tag at which to save the new image if the build succeeds:

$ docker build -t shykes/myapp .

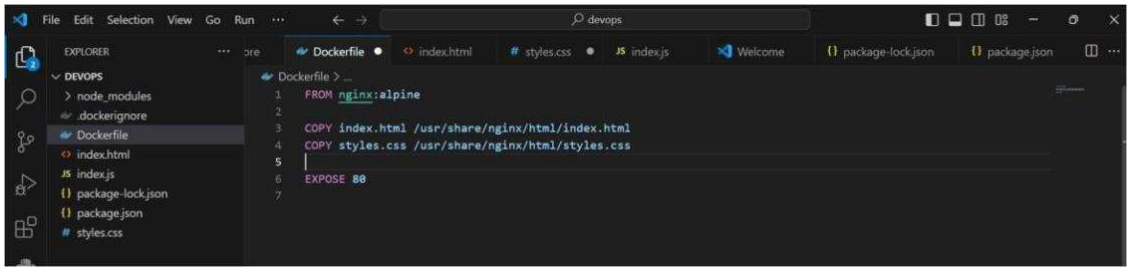
To tag the image into multiple repositories after the build, add multiple -t parameters when you run the build command:

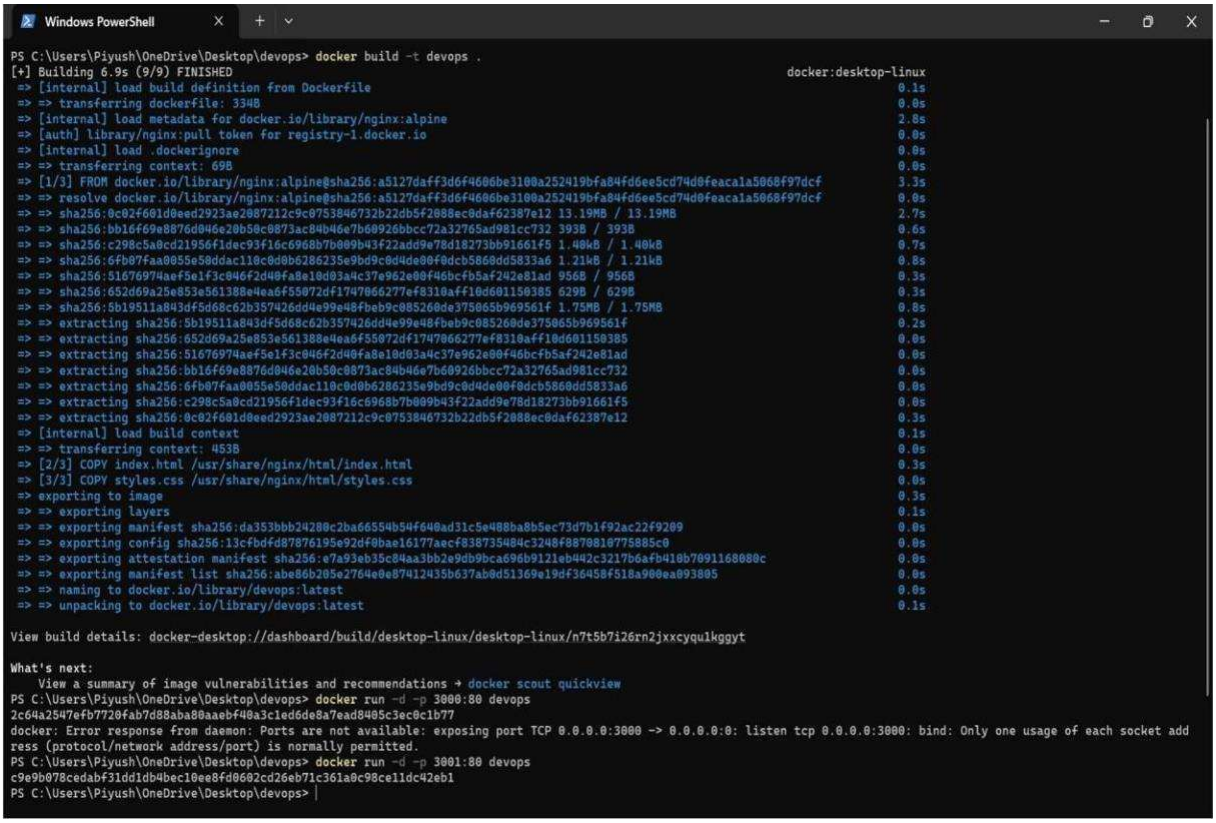
$ docker build -t shykes/myapp:1.0.2 -t shykes/myapp:latest .

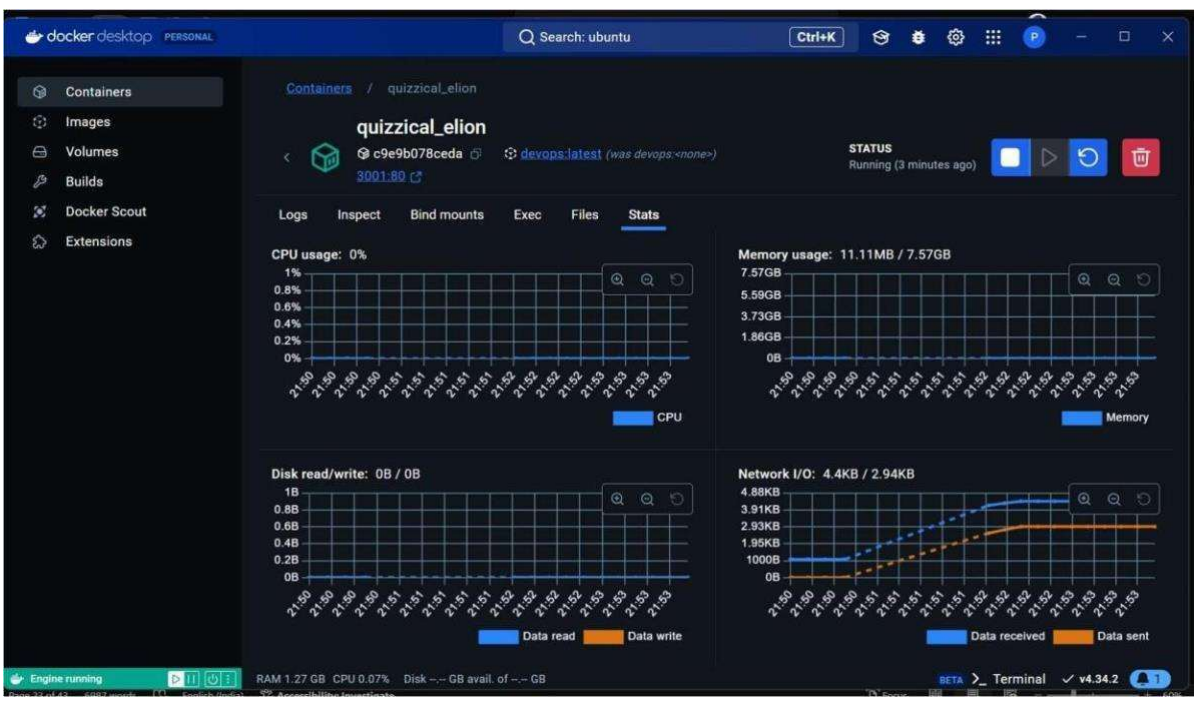
Before the Docker daemon runs the instructions in the Dockerfile, it performs a preliminary validation of the Dockerfile and returns an error if the syntax is incorrect:

$ docker build -t test/myapp . Syntax: syntax=[remote image reference]

OUTPUT

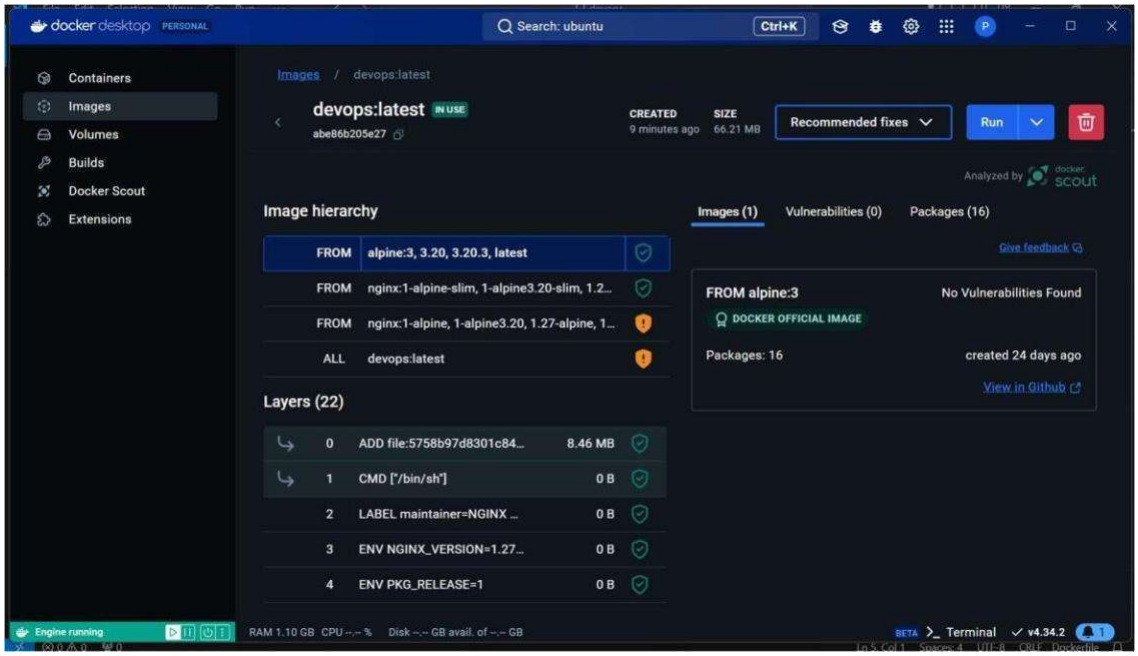






Welcome to My Sample Web Application

This is a simple web app running in a Docker container.



CONCLUSION:

Consequently, learning Dockerfile instructions and building an image for a sample web application provided practical experience in containerization and application deployment.