Here are 50 hands-on practice questions that will help you test your ability to write Dockerfiles and work with Docker:

1. **Write a Dockerfile to create an image that installs and runs Nginx.**

2. **Create a Dockerfile that installs Python, sets up a virtual environment, and runs a Flask app.**

3. **Write a Dockerfile to build a Node.js application and expose it on port 3000.**

4. **Create a Dockerfile that installs MongoDB and exposes port 27017.**

5. **Write a Dockerfile to install MySQL, configure a custom root password, and expose port 3306.**

6. **Create a Dockerfile to set up a basic Ruby on Rails application with PostgreSQL.**

7. **Write a Dockerfile for a Java application with Maven that builds a WAR file.**

8. **Create a Dockerfile that installs Apache, configures a custom document root, and exposes port 80.**

9. **Write a Dockerfile to install and configure Redis.**

10. **Create a Dockerfile for a simple PHP application that runs on Apache.**

11. **Write a Dockerfile to create an image with Ubuntu, installs `curl` and `git`, and sets a custom environment variable.**

12. **Create a Dockerfile to build a static website with an Nginx server.**

13. **Write a Dockerfile to install and configure PostgresQL.**

14. **Create a Dockerfile for a Python Django application with PostgreSQL.**

15. **Write a Dockerfile that uses multi-stage builds to reduce image size for a Go application.**

16. **Create a Dockerfile that installs JDK 11 and builds a Java application.**

17. **Write a Dockerfile that installs Node.js and npm and creates a simple app that serves “Hello World”.**

18. **Create a Dockerfile for a Spring Boot application.**

19. **Write a Dockerfile to set up a WordPress application with MySQL as the backend.**

20. **Create a Dockerfile for a simple Python application that interacts with a MongoDB database.**

21. **Write a Dockerfile that adds a file to a Docker image and runs a shell script during the build process.**

22. **Create a Dockerfile that installs `curl`, `wget`, and `vim` in an Alpine-based container.**

23. **Write a Dockerfile that uses an official image to create a container with a custom entry point.**

24. **Create a Dockerfile that builds a React application and serves it using Nginx.**

25. **Write a Dockerfile that installs Elasticsearch and exposes the appropriate port.**

26. **Create a Dockerfile that installs Redis and runs it as the container's main process.**

27. **Write a Dockerfile to install a package (e.g., `jq`) on top of a busybox image.**

28. **Create a Dockerfile for an application that installs Java 8 and sets environment variables for Java applications.**

29. **Write a Dockerfile that uses the `--no-install-recommends` option when installing packages to minimize the size of the image.**

30. **Create a Dockerfile that uses Alpine Linux as the base image for a lightweight container.**

31. **Write a Dockerfile to install and run Jenkins in a Docker container.**

32. **Create a Dockerfile that automatically generates a Docker image from a local directory containing a Node.js application.**

33. **Write a Dockerfile that runs tests in a Python project before copying the code into the image.**

34. **Create a Dockerfile for an application that runs a shell script upon container startup.**

35. **Write a Dockerfile to set up a development environment for a JavaScript application with Node.js and npm.**

36. **Create a Dockerfile that uses `docker-compose` to run multiple containers for a project.**

37. **Write a Dockerfile for an application that uses environment variables for configuration.**

38. **Create a Dockerfile that uses `COPY` and `ADD` commands and explains their differences.**

39. **Write a Dockerfile that performs optimizations like removing unnecessary files to reduce the image size.**

40. **Create a Dockerfile that sets up a Go application with dependencies.**

41. **Write a Dockerfile for a PHP application that runs a `composer install` command during build.**

42. **Create a Dockerfile for a container that runs an SSH server.**

43. **Write a Dockerfile to install Nginx, configure a custom virtual host, and set up SSL.**

44. **Create a Dockerfile that configures a proxy server inside a container.**

45. **Write a Dockerfile that installs and runs the Grafana dashboard.**

46. **Create a Dockerfile to install a Python web framework like Flask or Django, set up a web app, and run it.**

47. **Write a Dockerfile that includes multiple health checks for the application running inside the container.**

48. **Create a Dockerfile to install and run an FTP server.**

49. **Write a Dockerfile to configure Docker to run as a service inside the container (Docker-in-Docker).**

50. **Create a Dockerfile that builds an image for a microservices architecture with multiple dependencies.**