Docker

**PART A**

**1. With minimal possible layers to install WebServices and expose appropriate http port for accessing**

1. Pulling the official node image form docker hub.
2. We have a simple nodejs app in place with hello world

“<https://github.com/MaheedharTalluri/DockerAssessment/blob/main/microservices/hello.js>”

1. We have a dockerfile to copy the nodejs app and run it.

“<https://github.com/MaheedharTalluri/DockerAssessment/blob/main/microservices/Dockerfile.txt>”

1. When we build the image we use

“docker build –f Dockerfile.txt –t nodeapp .”

1. When we run the image , we map the port with 8080 in our local machine.

“docker run –d –p 8080:8080 –name nodeapp nodeapp:latest”

**2. Docker container is not starting**.

1. A container with same name already exists.
2. Not enough space on the device.
3. Port is already allocated to another service
4. Docker daemon stopped working.
5. User does not have enough permissions to the directory.
6. Proxy is not setup with Docker registry.
7. Container build cache.
8. Running too many services in the same container.
9. Authentication error while pulling the image form Docker registry.

**Part B**

**1.Using shell command search for a particular pattern (of your choice) to search Docker logs using awk command**

**2. Execute a shell command inside a container to search for a file.**

1. To execute find command inside a running container

“docker exec –it <container name> find / -name <filename>”

* Docker exec –it command is used to run a command in a running container without opening a bash interactive terminal.
* Here “/” after the find command implies that find the file from root directory.
* -name tag is used to specify filename / directory name

1. If we want to get an interactive terminal, run the same command as follows.

“docker exec –it <container name> find / -name <filename> /bin/bash”

**JENKINS**

**PART A**

1. **There is a Jenkins pipeline failure the failure error at the outset suggest that build has failed and one of the tool used for the build is running inside the container – what are the commands you will use for debugging the issue**

1. docker stats <container id> --> when we are using any metrics, and they show how the error occured

2. docker logs <container id> --> basic command which shows all i/p o/p and errors of a container

3. docker exec -it <container\_id> /bin/bash --> go inside a container and debug

4. docker commit <container\_id> my-broken-container && docker run -it my-broken-container /bin/bash --> if we cant really start a container

5. docker cp <container\_id>:/pathinserver /local-path --> to save the container log locally and debug the issue.

1. **There is a build failure due to connectivity issue with one of the tools – what actions you will take to narrow down the issue.**

1. Build failure analyzer to analyze the exact issue

2. Checking whether the credentials are properly configured

3. Checing whether the required plugins for the required tools are correctly installed

4. In case of master-slave configuration check whether master and slave are properly configured

5. Check whether the apis written are properly working.

6. Check whether the configured secret-token got expired.

1. **Image push failed**
2. Docker registry configuration error.
3. Proxy error (Not able to connect to Docker registry).
4. Authentication error.
5. Conflict with other image (When two builds are triggered at the same time with same image name , one will pass and the other will throw an error)
6. Improper tagging of image (Image has to be tagged while building and then only it can be pushed using that tag)
7. **Write a shell script to setup git repo, docker build, push image to docker repository.**
8. The shell script at “<https://github.com/MaheedharTalluri/DockerAssessment/blob/main/ShellScript/shell.sh>”
9. Clone the shell script and give executable permissions to the file.