DATE:7/08/2024

Module 4

1.Mention the actions of following comments: git remote add origin "http://github/a.git" Git pull origin master Git push origin dev

2. What are the functions of f

 Git Commands:

- **git remote add origin "http://github/a.git"** This command adds a new remote repository to your local Git repository with the name origin. The URL "http://github/a.git" points to the remote repository where you'll push your code. Essentially, this sets up a reference to the remote repository.
- **git pull origin master** This command fetches the changes from the master branch of the remote repository named origin and merges those changes into your current branch. It is used to update your local branch with the latest changes from the remote branch.
- **git push origin dev** This command pushes your local dev branch to the remote repository named origin. It updates the remote dev branch with your local changes. If the dev branch doesn't exist on the remote, it will be created.

Functions of f (Assuming you mean the f in a context such as git fetch):

- In Git commands like git fetch, the f stands for "fetch." The fetch command retrieves updates from a remote repository without merging them into your local branch. This allows you to see changes on the remote without applying them directly.
- In general programming contexts, f is often used as a shorthand for "function" or as a variable name, but without more context, it’s hard to give a specific function for f.

2. What are the functions of following Docker objects and key components: Dockerd: Dockerfile Docker-compose.yaml Docker Registries DockerHos

Key Components:

- **Services:** Define the individual containers that make up the application.
- **Networks:** Configure how the services communicate with each other.
- **Volumes:** Specify data storage and sharing between containers.
- **Configurations:** Includes settings for build context, environment variables, and ports.

3.What's the isolation in Docker container?

Filesystem Isolation:

Docker uses a layered filesystem to ensure that each container operates in its own isolated filesystem:

- **Union Filesystem:** Combines layers from Docker images and container layers, giving each container its own filesystem without affecting the base image or other containers.
- **Read-Only Layers:** Base images are typically read-only, and containers have their own writable layer on top. Changes made in a container are isolated to that container's writable layer.

4. Process Isolation:

Containers run their own processes isolated from the host and other containers. This is achieved through:

- **Process Space:** Each container has its own process namespace, ensuring that processes in one container are not visible or accessible to processes in another container or on the host.

5. Network Isolation:

Docker containers have isolated network environments:

- **Virtual Networks:** Containers can be connected to virtual networks, which can be configured to be isolated from other networks or containers.
- **Port Binding:** Containers can expose specific ports to the host, and by default, they are isolated from the host's network unless explicitly connected.