

1. Mention the actions of following comments:

```
git remote add origin "http://github/a.git"
Git pull origin master
Git push origin dev
```

Answer:-

1. **git remote add origin "http://github/a.git":**

- **Action:** This command sets up a new remote repository named origin with the URL `http://github/a.git`. In Git, origin is a conventional name used for the default remote repository. By adding this remote, you're telling Git where to fetch and push changes to and from.

2. **git pull origin master:**

- **Action:** This command fetches the changes from the master branch of the remote repository named origin and merges them into your current branch. Essentially, it updates your local branch with any new commits from the master branch of the remote repository.

3. **git push origin dev:**

- **Action:** This command pushes your local changes from the dev branch to the dev branch of the remote repository named origin. It updates the remote repository with the commits that are in your local dev branch but not in the remote dev branch.

2. What are the functions of following Docker objects and key components:

Dockerd: Dockerfile

Docker-compose.yaml

Docker Registries DockerHost

### **dockerd:**

- **Function:** dockerd is the Docker daemon, which is the core background service that manages Docker containers and images. It handles container lifecycle operations such as building, running, and managing containers. It also communicates with Docker clients and manages Docker images and containers.

### **Dockerfile:**

- **Function:** A Dockerfile is a text file containing a series of instructions that define how a Docker image should be built. It includes commands for setting up the image's environment, installing necessary packages, copying files, and configuring the image. The Dockerfile allows you to automate the creation of Docker images with a consistent setup.

### **docker-compose.yaml:**

- **Function:** The docker-compose.yaml (or docker-compose.yml) file is used with Docker Compose, a tool for defining and running multi-container Docker applications. This YAML file specifies the configuration

for your services, including their images, build contexts, networks, and volumes. It allows you to start and manage multiple containers as a single application stack with a single command.

### **Docker Registries:**

- **Function:** Docker Registries are repositories for storing Docker images. They can be public, like Docker Hub, or private, like a private Docker Registry or third-party services. Registries store and distribute Docker images, allowing you to pull images from the registry to your local system or push images to the registry from your local system.

### 3.What's the isolation in Docker container?

Isolation in Docker containers refers to the ability of containers to operate independently from each other and from the host system while sharing the same underlying OS kernel. This isolation ensures that containers are secure, reliable, and do not interfere with one another. Here's how Docker achieves isolation:

## **Docker Host:**

- **Function:** A Docker Host is a machine (physical or virtual) that runs the Docker daemon (dockerd). This host is responsible for running Docker containers. It can be a local development machine, a server in a data center, or a cloud-based virtual machine. The Docker Host provides the environment in which c