Name : Prathik Balaji N
Date : 07-08-24

Module 4

1.Mention the actions of following comments:
git remote add origin "http://github/a.git"

This will add a new remote repository using the name origin, with the URL "http://github/a.git".

-git pull origin master

This command fetches and integrates changes from the master branch of the remote repository named origin into the current branch in your local repository.

git push origin dev

This command pushes changes from your local 'dev' branch to the 'dev' branch on the remote repository named origin. If there isn't already a 'dev' branch on the remote repository, it will create one.

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

Dockerd (Docker Daemon) :

Dockerd is a Docker daemon running on some host machine. It manages Docker containers, images, networks, and storage volumes. It listens for Docker API requests and processes them.

Dockerfile:

A Dockerfile is a text file that contains a series of instructions on how to build a Docker image.

docker-compose.yaml:

This YAML file defines and manages multi-container Docker applications. It allows you to configure application services, networks, and volumes in a single file.

Docker Registries:

Docker registries are repositories where Docker images are stored and distributed. The most well-known public registry is Docker Hub.

Docker Host:

A Docker host is a physical or virtual machine where Docker is installed and running.

3.What's the isolation in Docker container?

Isolation in Docker containers refers to the separation of applications and their environments, ensuring that each container runs independently without affecting others.

Docker Example:

1. Create image for python application using docker (Gadgets)

1. Create a Python script (app.py).
2. Create a Dockerfile to define the image.
3. Build the Docker image with docker build.
4. Run the Docker container with docker run.

1. app.py

```
class Gadget:
    def __init__(self, name, category, price):
        self.name = name
        self.category = category
        self.price = price

    def display_info(self):
        return f"Gadget: {self.name}\nCategory: {self.category}\nPrice:
${self.price}"

if __name__ == "__main__":
    gadget1 = Gadget("Smartphone", "Electronics", 999)
    gadget2 = Gadget("Laptop", "Computers", 1299)
    print(gadget1.display_info())
    print(gadget2.display_info())
```

2.Dockerfile

```
# Use an official Python runtime as a parent image
FROM python:3.8-slim
# Set the working directory in the container
WORKDIR /app
# Copy the current directory contents into the container at /app
COPY . /app
# Run gadgets.py when the container launches
CMD ["python", "app.py"]
```

3. Build image

docker build -t gadgets .

```
D:\Python\gadgets>docker build -t gadgets .
[+] Building 0.3s (8/8) FINISHED                                                                                    docker:desk
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 330B
 => [internal] load metadata for docker.io/library/python:3.8-slim
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [internal] load build context
 => => transferring context: 879B
 => CACHED [1/3] FROM docker.io/library/python:3.8-slim
 => [2/3] WORKDIR /app
 => [3/3] COPY . /app
 => exporting to image
 => => exporting layers
 => => writing image sha256:eb401eb1a5b923c7844f6829ce087f5b29307fd43c79f28617d1cd731f28929d
 => => naming to docker.io/library/gadgets

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/gs8aq7npfvnp3ee32slr4pdi3
```

4. Run the image

docker run gadgets (or)

Can be directly started from docker desktop

```
D:\Python\gadgets>docker run gadgets
Gadget: Smartphone
Category: Electronics
Price: $999
Gadget: Laptop
Category: Computers
Price: $1299
```

5. Push the image to docker hub repository

After creating image login with the docker hub authentication and push the image to the hub.

```
D:\Python\gadgets>docker push prathik008/gadgets:latest
The push refers to repository [docker.io/prathik008/gadgets]
28879097acd1: Pushed
de07b781ac62: Pushed
0cfc6ead4554: Mounted from library/python
1109f5b27710: Mounted from library/python
cad3599a3016: Mounted from library/python
e7817ba7b646: Mounted from library/python
e0781bc8667f: Mounted from library/python
```