Introduction to Docker

#devops

Docker tool will be extensively used for training and deployment of your AI projects on DGX server almost every time.

1. What is Containerization and Docker?
    - By containerizing, developers bundle a program's code, runtime engine, tools, libraries and settings into a portable "container." That way, the software requires fewer resources to run and is much easier to deploy in new environments.
    - Docker is a popular implementation of containerization concept having the tag line - *"Develop faster. Run anywhere"*.
2. Installation of Docker:
    - On NVIDIA DGX:
        - All versions of DGX comes pre-installed with Docker
    - On Ubuntu 20.04
        - Reference - https://phoenixnap.com/kb/install-docker-on-ubuntu-20-04
        - Use the **second** method in the given tutorial for Docker installation
3. How to check whether Docker is installed and working fine?

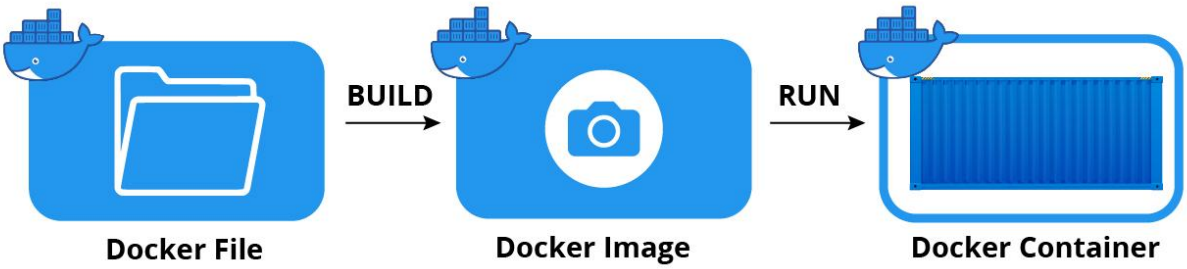
    ```
    service docker status
    ```

    Alternatively, we can use:

    ```
    docker run hello-world
    ```

4. What is a Docker Image?
    - A *Docker image* is a read-only template that contains a set of instructions for creating a container that can run on the Docker platform
5. What is a Docker Container?
    - Docker Containers are runtime instances of Docker images. Containers contain the whole kit required for an application, so the application can be run in an

isolated way



6. What is a Docker group and how to add users in a docker group?
    - A docker group is Linux group of users that have been given permission to execute docker commands without using *sudo* (super user permissions).
    - Use following link to add a user to the docker group:

7. What is Docker hub?
   Docker hub is the repository for docker images shared by developers all over the world for various popular frameworks.

8. How to create account on Docker Hub and download docker images from Docker hub to local system?
    - Not applicable while working on DGX Server

9. What is NGC?
   NGC is a cloud based service that provides NVIDIA GPU accelerated docker images for popular frameworks like PyTorch, TensorFlow, TAO, Merlin etc.

10. How to download images from NGC to NVIDIA DGX A100?
    - Can be performed only the system admin of DGX A100
    - Follow - Downloading Docker Images from NGC to NVIDIA DGX A100

11. Common commands related to docker images
    - Listing images: `docker images`
    - Searching images in **docker hub**: `docker search image_name`
    - Downloading **latest** images: `docker pull image_name`
        - Downloading specific version of an image: `docker pull image_name:tag`
    - Removing images: `docker image rm image_name`
    - Inspecting images: `docker image inspect image_name`

12. Common commands related to docker containers

- To run a container (and download the image if it is not present in local system):
  - `docker run image_name`
  - Popular Flags:
    - `-it` *:The -it flag tells docker that it should open an interactive container instance*
    - `--rm` *:The --rm flag tells docker that the container should automatically be removed when we exit it
    - `--name` *:This flag is used to provide a custom name to the running container*
    - `-v` *: This flag can be used to mount (connect) a directory on the system to the running docker container*
    - `-p x:y` *: This flag tells docker to expose port x on our local system to docker application running on port y*
  - An example:
    - `docker run -it --rm --name my_python_app python`
  - Another example:
    - `docker run -it --rm --name my_python_app -v /home/anubhav/Documents:/mnt python`
      - Here the command is same but we have also used `-v` flag
      - We are mounting (attaching a volume) our local directory `home/anubhav/Documents` with the container and it will be mounted (attached) at `/mnt` directory inside the container. Anything modified in the local directory will be reflected inside the mounted directory within the container and vice versa.
- List containers:
  - List only currently running containers: `docker ps`
  - List currently running and stopped containers: `docker ps -a`
- To create a container without starting it
  - `docker create image_name`
- To come out of a running container and stop it:
  - Type `exit` inside container
- To come out of a running container without stopping it press following key sequence:
  - *ctrl+p*
  - *ctrl+q*
  - The container will continue to run in background
- To start a stopped container:
  - `docker start container_name/container_id`

        - The container will start in the background
    - To go inside a container running in background
        - `docker attach container_name/container_id`
    - To stop a container running in background
        - `docker stop container_name/container_id`
    - Remove a stopped container: `docker rm my_container`
        - *Note: Running containers first need to be stopped*

13. How to update tag of an existing docker image without recreating a new one? [1]

    - We can use *tag* flag
    - Syntax - `docker tag OldName:tag NewName:tag`
    - For example, `docker tag pytorch:deepops-kubeflow-minimal pytorch:1.8-deepops`

14. How to create an image from an existing modified container

    - First stop the running container by typing `exit`
    - `docker commit container_id/container_name new_image_name`

15. How to build a new docker image and run it on DGX?

    - *Docker File → Docker Image → Docker Container*
    - An example of **Docker file** can be:

```
# syntax=docker/dockerfile:1

FROM nvcr.io/nvidia/tensorflow:22.06-tf2-py3

WORKDIR /multi_person_face_detection

#COPY requirements.txt requirements.txt
#RUN pip3 install -r requirements.txt

RUN alias pip=pip3

RUN pip3 install --upgrade pip

RUN cp /usr/local/bin/pip /usr/bin/

RUN pip3 install deepface

RUN pip3 install python-socketio==4.6.0

RUN pip3 install Flask-SocketIO==4.3.1
```

```
RUN pip3 install python-engineio==3.13.2

RUN pip3 install Flask==2.0.3

RUN pip3 install Werkzeug==2.0.3

#RUN pip3 install
git+https://github.com/untitaker/werkzeug.git@reloader-perf

# For setting time zone
#https://grigorkh.medium.com/fix-tzdata-hangs-docker-image-build-
cdb52cc3360d
ENV TZ=Asia/Kolkata
RUN ln -snf /usr/share/zoneinfo/$TZ /etc/localtime && echo $TZ >
/etc/timezone

RUN apt update -y && apt install ffmpeg libsm6 libxext6  -y

#RUN pip3 install Flask
#RUN pip3 install Flask-Bootstrap

COPY . .

RUN mkdir ~/.deepface/weights -p

RUN cp -r weights/. ~/.deepface/weights/

#RUN pip3 install dlib

CMD ["/bin/bash"]
```

- Create **Docker image** from Docker file
  `docker build --tag tensorflow_face_recognition .`

- Run Docker image as **Container** on DGX with GPUs
  - Running Docker Containers Directly via Terminal Access to DGX
  - `docker run --gpus '"device=7:1"' -p 9990:9999 -it --rm --name test_container tensorflow_face_recognition`

16. How to start a container and interact with it if the image was built to start a program at startup?
    `docker run -it --entrypoint sh <image_name>`

Assignment:

Download Docker image (Ubuntu), build a container from it, modify it, recreate an image and upload it to Docker hub

Resources for assignment:

- Docker Hub Login
- Pushing Docker Images to Docker Hub

References

---

1. https://www.janbasktraining.com/community/devops/how-to-rename-docker-images-without-rebuilding-it↩