1. Visual Code, plugin to write Dockerfile.
2. Base Images.
3. By default, the topmost directory (which is root directory) inside a container is the working directory.
4. By Default Image each name that we generate, a name is given.  
   We can given this command to run the image 🡺 docker run <image-name>
5. docker run -p <host-port>:<container-port> <image-name>
   1. -d : to run in detached as default is attached.
   2. -i: --interactive.
   3. -t: --tty : Allocate a pseudo-TTY
   4. --rm : To remove container automatically when it stops or exits.
   5. --name: To name a container.
6. docker ps: TO list running container.
   1. -a : to list the stopped ones too.
7. docker attach <container-name-or-id>
8. docker start <container-name\_or\_container-id>
   1. -a : to start in attached mode as default is detached mode.
9. docker attach <container\_id\_or\_container\_name>
10. docker container prune 🡸 To remove all stopped containers.
11. docker image
    1. prune : Removed only Dangling images (having no tag)
       1. -a: To remove dangling images too.
12. **docker rm** <container\_name> [ container\_name container\_name…]
13. docker rmi
14. docker inspect <image-id>
15. docker cp
    1. Host to Container :
       1. docker cp <host-file-or-folder-address> <container-id>:/test (Where \test is the location inside container: If not then will be created).
    2. Container to Host:
       1. docker cp <container-id>:/test <host-file-or-folder-address>
16. docker log : to print the log of the running container till now.
    1. -f : means follow.
17. Attached and Detached
    1. Attached: run Command
    2. Detached: start Command.
    3. Interactive mode: Requires attached mode.
18. Share Images:
    1. Docker Hub:
       1. Public Registry,
    2. Private Registry
19. Docker Hub:
    1. docker push username/<image\_name>
       1. To push on private 🡺 docker push <URL:Name> <image-name>
    2. docker pull <image\_name>
    3. docker login
    4. docker logout
20. d
21. AtGraphical user interface, application

    Description automatically generated

Docker Image

1. Text

   Description automatically generated
2. Command to build image 🡺 docker build . (dot).
   1. Tagging/naming an image 🡺 name : tag = Image Tag = Unique ID.  
      -t <name:tag> where tag is optional -t = --tag  
      Where <name> is repository/group and tag is a specific image in that group.  
      Like 🡺 node:4.9 where node is a group of images and 4.9 is specific version in that group.