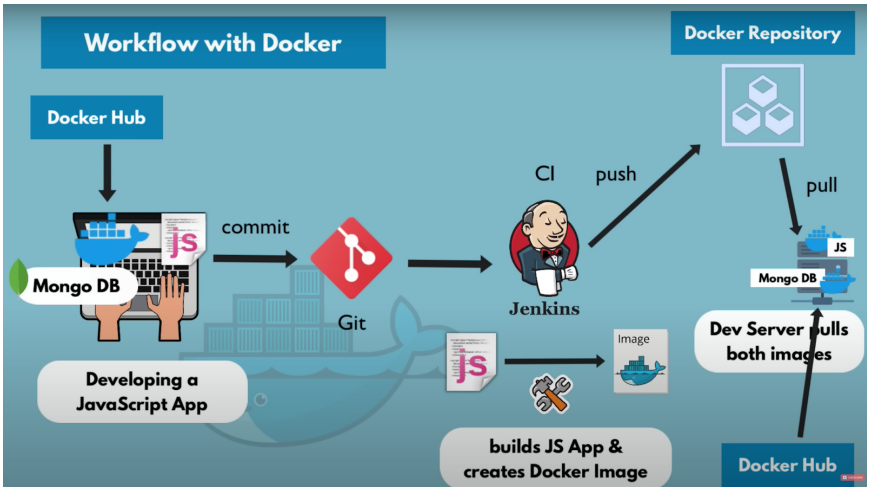
- **docker run -d -p[PC_PORT]:[Container_PORT] --name [Container_NAME] version_no**: Creates a docker container with custom name. eg.docker run -d -p6001:6379 --name redis-older redis:5.0
- **docker exec -it [container_id/container_name] /bin/bash** : can be used for debugging , we can get the terminal of running container i.e go inside file directory structure and all.




- **docker network ls**: list the docker networks in a isolated docker network.
- **docker create network [network_name]** : creates a new network in isolated docker network

- **docker run -d -p 27017:27017 --network mongo-network --name mongodb **
**-e MONGO_INITDB_ROOT_USERNAME=admin **
**-e MONGO_INITDB_ROOT_PASSWORD=password **
**mongo** : start mongodb

- **docker run -d -p 8081:8081 -e ME_CONFIG_MONGODB_ADMINUSERNAME=admin -e ME_CONFIG_MONGODB_ADMINPASSWORD=password --network mongo-network --name mongo-express -e ME_CONFIG_MONGODB_SERVER=mongodb mongo-express** : start mongo-express
- docker-compose -f .yaml up=> starts up the config mentioned in yaml file. The docker compose file would be used on server to deploy all the application/services.
- docker-compose -f .yaml down=> shutdowns all the config mentioned in yaml file.
- docker build -t image-name:tag-name .
- docker tag image-name:tag-name repo_name/image-name:tag-name
- docker push repo_name/image-name:tag-name

Docker Volumes: Need for data persistence i.e for database/stateful applications.