Lab1. Container vs Container Image

[실습1. ubuntu Base Image + git]

1. Ubuntu Base Image에 Git 설치하기

```
    $ docker images
    REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
    hello-world   latest    feb5d9fea6a5   19 months ago   13.3kB


    $ docker pull ubuntu:latest
    latest: Pulling from library/ubuntu
    2ab09b027e7f: Pull complete
    Digest: sha256:67211c14fa74f070d27cc59d69a7fa9aeff8e28ea118ef3babc295a0428a6d21
    Status: Downloaded newer image for ubuntu:latest
    docker.io/library/ubuntu:latest


    $ docker images
    REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
    ubuntu        latest    08d22c0ceb15   6 weeks ago     77.8MB
    hello-world   latest    feb5d9fea6a5   19 months ago   13.3kB


    $ docker run -it --name git ubuntu:latest bash
    root@9fb2faf09844:/# ls
    bin  boot  dev  etc  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var

    root@9fb2faf09844:/# git
    bash:git : command not found

    root@9fb2faf09844:/# apt update
    Get:1 http://archive.ubuntu.com/ubuntu jammy InRelease [270 kB]
    Get:2 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
    Get:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
    Get:4 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [23.2 kB]
    Get:5 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [108 kB]
    Get:6 http://archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [17.5 MB]
    Get:7 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1000 kB]
    Get:8 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [911 kB]
    Get:9 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [959 kB]
    Get:10 http://archive.ubuntu.com/ubuntu jammy/restricted amd64 Packages [164 kB]
    Get:11 http://archive.ubuntu.com/ubuntu jammy/main amd64 Packages [1792 kB]
    Get:12 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [266 kB]
    Get:13 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1301 kB]
    Get:14 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1054 kB]
    Get:15 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [28.6 kB]
    Get:16 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1154 kB]
    Get:17 http://archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [49.4 kB]
    Get:18 http://archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [25.6 kB]
    Fetched 26.8 MB in 6s (4624 kB/s)
    Reading package lists... Done
    Building dependency tree... Done
    Reading state information... Done
    4 packages can be upgraded. Run 'apt list --upgradable' to see them.


    root@9fb2faf09844:/# apt-get install -y git

    root@9fb2faf09844:/# git --version
    git version 2.34.1
```

2. 다른 세션에서, 즉 또 다른 커넥션으로 연결하여

```
        $ docker images | grep ubuntu
        Last login: Thu Apr 20 05:44:18 2023 from 182.208.131.42
        ubuntu@ip-10-0-10-23:~$ docker images | grep ubuntu
        ubuntu        latest    08d22c0ceb15   6 weeks ago     77.8MB

        $ docker commit git ubuntu:git
        sha256:ada81e9a54411b13b0602e6b64dfa7f1523fdf747e160ec800ed6f486c0f806d

        $ sudo docker images | grep ubuntu
        ubuntu        git       ada81e9a5441   41 seconds ago   196MB
        ubuntu        latest    08d22c0ceb15   6 weeks ago      77.8MB

        $ sudo docker run -it --name git2 ubuntu:git bash
        root@b51cafbbb61b:/# git --version
        git verion 2.25.1
```

[실습2. Container Image Layer 다루기]

1. Docker Image 정보 확인

```
$ sudo docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
26c5c85e47da: Pull complete
4f3256bdf66b: Pull complete
2019c71d5655: Pull complete
8c767bdbc9ae: Pull complete
78e14bb05fd3: Pull complete
75576236abf5: Pull complete
Digest: sha256:63b44e8ddb83d5dd8020327c1f40436e37a6fffd3ef2498a6204df23be6e7e94
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest


$ sudo docker inspect nginx
[
    {
        "Id": "sha256:6efc10a0510f143a90b69dc564a914574973223e88418d65c1f8809e08dc0a1f",
        "RepoTags": [
            "nginx:latest"
        ],
        "RepoDigests": [
            "nginx@sha256:63b44e8ddb83d5dd8020327c1f40436e37a6fffd3ef2498a6204df23be6e7e94"
        ],
        "Parent": "",
        "Comment": "",
        "Created": "2023-04-12T08:51:21.83677948Z",
        "Container": "6d1de575d4fe315e404a58482deb30ca38735c5a52309bf8b7f46b81fffc80d4",
        "ContainerConfig": {
            "Hostname": "6d1de575d4fe",
            "Domainname": "",
            "User": "",
            "AttachStdin": false,
            "AttachStdout": false,
            "AttachStderr": false,
            "ExposedPorts": {
                "80/tcp": {}
            },
            "Tty": false,
            "OpenStdin": false,
            "StdinOnce": false,
            "Env": [
                "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",
                "NGINX_VERSION=1.23.4",
                "NJS_VERSION=0.7.11",
                "PKG_RELEASE=1~bullseye"
            ],
            "Cmd": [
                "/bin/sh",
                "-c",
                "#(nop) ",
                "CMD [\"nginx\" \"-g\" \"daemon off;\"]"
            ],
            "Image": "sha256:20d4c097fb72a6a8c4d8c3989a40adcdcfbdba4da700bd82c594d24d4effcdba",
            "Volumes": null,
            "WorkingDir": "",
            "Entrypoint": [
                "/docker-entrypoint.sh"
            ],
            "OnBuild": null,
            "Labels": {
                "maintainer": "NGINX Docker Maintainers <docker-maint@nginx.com>"
            },
            "StopSignal": "SIGQUIT"
        },
        "DockerVersion": "20.10.23",
        "Author": "",
        "Config": {
            "Hostname": "",
            "Domainname": "",
            "User": "",
            "AttachStdin": false,
            "AttachStdout": false,
            "AttachStderr": false,
            "ExposedPorts": {
                "80/tcp": {}
            },
            "Tty": false,
            "OpenStdin": false,
            "StdinOnce": false,
            "Env": [
                "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",
                "NGINX_VERSION=1.23.4",
                "NJS_VERSION=0.7.11",
                "PKG_RELEASE=1~bullseye"
            ],
            "Cmd": [
                "nginx",
                "-g",
                "daemon off;"
            ],
            "Image": "sha256:20d4c097fb72a6a8c4d8c3989a40adcdcfbdba4da700bd82c594d24d4effcdba",
            "Volumes": null,
            "WorkingDir": "",
            "Entrypoint": [
                "/docker-entrypoint.sh"
            ],
            "OnBuild": null,
            "Labels": {
                "maintainer": "NGINX Docker Maintainers <docker-maint@nginx.com>"
            },
            "StopSignal": "SIGQUIT"
        },
        "Architecture": "amd64",
        "Os": "linux",
        "Size": 142151046,
        "VirtualSize": 142151046,
        "GraphDriver": {
            "Data": {
                "LowerDir":
                "/var/lib/docker/overlay2/9fbd542e2b00e1b058bfefe2e2db910fba9c4a7ae2629566bbbeb3129b7d0afe/diff:/var/lib/
                docker/overlay2/74457d7ff87c761a5d673265c09a985c7f267e4c30e1c7ddb5e86f514d9626fb/diff:/var/lib/docker/ov
                erlay2/dd4882652704815d75803cff66cdfef92bc962beac459db8c27da04a6c7bc9e9/diff:/var/lib/docker/overlay2/68
                18156a861295438c7a5bb66b9aa97fb8328f610636f3894bb475ed824749e8/diff:/var/lib/docker/overlay2/f6795d3e6
                de6123376df02197ec0e99ae7780f86773e14c18bf5c3f166d37c0e/diff",
                "MergedDir":
                "/var/lib/docker/overlay2/dd90ce838ca3f9f9a093701fd8250f4af050c138f7337d74fa23b5b0bf7d22ff/merged",
                "UpperDir":
                "/var/lib/docker/overlay2/dd90ce838ca3f9f9a093701fd8250f4af050c138f7337d74fa23b5b0bf7d22ff/diff",
                "WorkDir":
                "/var/lib/docker/overlay2/dd90ce838ca3f9f9a093701fd8250f4af050c138f7337d74fa23b5b0bf7d22ff/work"
            },
            "Name": "overlay2"
        },
        "RootFS": {
            "Type": "layers",
            "Layers": [
                "sha256:ed7b0ef3bf5bbec74379c3ae3d5339e666a314223e863c70644f7522a7527461",
                "sha256:fb6d57d46ad57cbb50dfe16eba53a31d8808aa43e3a4a3ddd5c04d2d6ee0ecc5",
                "sha256:935b5bd454e1973eb283fbc76d5613def62c190c8c4cc621576ef32f7c7dcb56",
                "sha256:f12d4345b7f35505fc42fd613ae2c6749ddba30d6c9025e1a5b070c97c1ad2bb",
                "sha256:79974a1a12aa3a3876d8dbcf16b2eda15ba5372574d38dd94bac68d339b6e124",
                "sha256:9d907f11dc742442217bda25fde38f11851c7d495cacc6fc4e3869f4cd2c9ad9"
            ]
        },
        "Metadata": {
            "LastTagTime": "0001-01-01T00:00:00Z"
        }
    }
]
```

2. Docker Image 저장소 위치 확인

```
$ sudo docker info
Client:
 Context:    default
 Debug Mode: false
 Plugins:
  buildx: Docker Buildx (Docker Inc.)
    Version:  v0.10.4
    Path:     /usr/libexec/docker/cli-plugins/docker-buildx
  compose: Docker Compose (Docker Inc.)
    Version:  v2.17.2
    Path:     /usr/libexec/docker/cli-plugins/docker-compose

Server:
 Containers: 3
  Running: 1
  Paused: 0
  Stopped: 2
 Images: 4
 Server Version: 23.0.4
 Storage Driver: overlay2
  Backing Filesystem: extfs
  Supports d_type: true
  Using metacopy: false
  Native Overlay Diff: true
  userxattr: false
 Logging Driver: json-file
 Cgroup Driver: systemd
 Cgroup Version: 2
 Plugins:
  Volume: local
  Network: bridge host ipvlan macvlan null overlay
  Log: awslogs fluentd gcplogs gelf journald json-file local logentries splunk syslog
 Swarm: inactive
 Runtimes: io.containerd.runc.v2 runc
 Default Runtime: runc
 Init Binary: docker-init
 containerd version: 2806fc1057397dbaeefbea0e4e17bddfbd388f38
 runc version: v1.1.5-0-gf19387a
 init version: de40ad0
 Security Options:
  apparmor
  seccomp
   Profile: builtin
  cgroupns
 Kernel Version: 5.15.0-1031-aws
 Operating System: Ubuntu 22.04.2 LTS
 OSType: linux
 Architecture: x86_64
 CPUs: 1
 Total Memory: 966.2MiB
 Name: ip-10-0-10-23
 ID: bdf0c5e6-005e-41fa-af85-2194ba9394ee
 Docker Root Dir: /var/lib/docker                    <------ 여기서 확인
 Debug Mode: false
 Registry: https://index.docker.io/v1/
 Experimental: false
 Insecure Registries:
  127.0.0.0/8
 Live Restore Enabled: false
```

```
$ sudo -i
# cd /var/lib/docker/overlay2
```

3. Disk size check

```
# du -sh image
root@ip-10-0-10-23:/var/lib/docker/overlay2# cd ..
root@ip-10-0-10-23:/var/lib/docker# ls
buildkit  containers  engine-id  image  network  overlay2  plugins  runtimes  swarm  tmp  volumes

root@ip-10-0-10-23:/var/lib/docker# du -sh image
1.2M    image


# du -sh overlay2
root@ip-10-0-10-23:/var/lib/docker# du -sh overlay2
681M    overlay2
```

4. Directory 정리

```
1)/var/lib/docker/image --> imagedb
                            --> layerdb : imagedb에 대한 정보 소유
2)/var/lib/docker/overlay2 --> layerdb에 대한 정보 소유
3)/var/lib/docker/overlay2/... --> Image에 대한 변경 사항들 저장
4)/var/lib/docker/overlay2/l(영문소문자 l) --> 실제 변경사항들 저장된 directory
5)/var/lib/docker/overlay2/l --> 실제 Container에 대한 파일 시스템 저장
6)해당 Image를 Download할 때마다 overlay2의 사이즈가 커짐.
```