

# DOCKER 01

## 1) Install docker on ec2.

→ Create one EC2 instance and connect then follow below commands

--yum install docker

--systemctl start docker

--systemctl status docker

```
[root@ip-172-31-7-238 ~]# systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; disabled; preset: disabled)
   Active: active (running) since Wed 2025-04-02 10:15:27 UTC; 2h 5min ago
 TriggeredBy: ● docker.socket
    Docs: https://docs.docker.com
   Process: 27381 ExecStartPre=/bin/mkdir -p /run/docker (code=exited, status=0/SUCCESS)
   Process: 27382 ExecStartPre=/usr/libexec/docker/docker-setup-runtimes.sh (code=exited,
 Main PID: 27383 (dockerd)
    Tasks: 26
   Memory: 1.3G
      CPU: 32.776s
   CGroup: /system.slice/docker.service
           └─27383 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
           └─30126 /usr/bin/docker-proxy -proto tcp -host-ip 0.0.0.0 -host-port 81 -conta
           └─30132 /usr/bin/docker-proxy -proto tcp -host-ip :: -host-port 81 -container-

Apr 02 10:15:27 ip-172-31-7-238.us-east-2.compute.internal dockerd[27383]: time="2025-04-02
Apr 02 10:15:27 ip-172-31-7-238.us-east-2.compute.internal systemd[1]: Started docker.servi
Apr 02 10:19:46 ip-172-31-7-238.us-east-2.compute.internal dockerd[27383]: time="2025-04-02
Apr 02 10:22:27 ip-172-31-7-238.us-east-2.compute.internal dockerd[27383]: time="2025-04-02
Apr 02 10:22:27 ip-172-31-7-238.us-east-2.compute.internal dockerd[27383]: time="2025-04-02
Apr 02 10:25:39 ip-172-31-7-238.us-east-2.compute.internal dockerd[27383]: time="2025-04-02
Apr 02 10:40:19 ip-172-31-7-238.us-east-2.compute.internal dockerd[27383]: time="2025-04-02
Apr 02 10:40:19 ip-172-31-7-238.us-east-2.compute.internal dockerd[27383]: time="2025-04-02
Apr 02 10:40:19 ip-172-31-7-238.us-east-2.compute.internal dockerd[27383]: time="2025-04-02
Apr 02 10:56:36 ip-172-31-7-238.us-east-2.compute.internal dockerd[27383]: time="2025-04-02
Apr 02 10:56:36 ip-172-31-7-238.us-east-2.compute.internal dockerd[27383]: time="2025-04-02
lines 1-26/26 (END)
```

i-0babd69af48e8196f (docker-server)

PublicIPs: 3.147.237.157 PrivateIPs: 172.31.7.238

## 2) Pull 5 docker images.

(nginx,apache tomcat,ubuntu,jenkins,sonarqube)

→ Follow these commands to pull above docker images:

--docker pull nginx/nginx-ingress

--docker pull tomcat

--docker pull ubuntu

--docker pull Jenkins/Jenkins

--docker pull sonarqube

→Then check status by using below command:

**--docker images**

```
[root@ip-172-31-7-238 ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
jenkins/jenkins	latest	d331b6bde0f8	45 minutes ago	467MB
tomcat	latest	88b0f1cee84c	3 weeks ago	519MB
sonarqube	latest	94a7de374354	4 weeks ago	1.17GB
nginx	latest	53a18edff809	7 weeks ago	192MB
ubuntu	latest	a04dc4851cbc	2 months ago	78.1MB

```
[root@ip-172-31-7-238 ~]#
```

**i-0babd69af48e8196f (docker-server)**

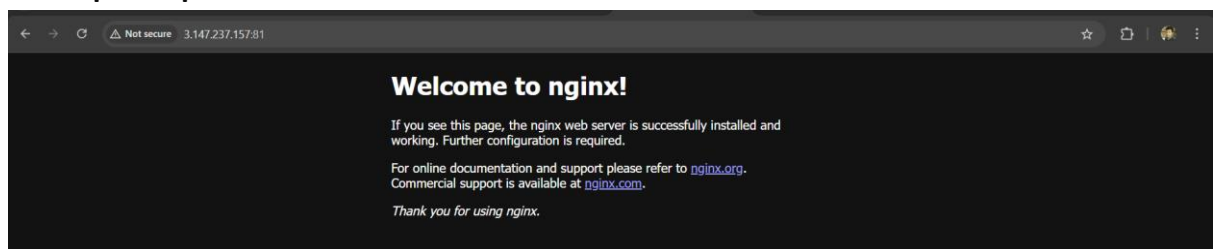
PublicIPs: 3.147.237.157 PrivateIPs: 172.31.7.238

### 3) Run nginx container and expose on port 81.

→run below command

```
[root@ip-172-31-7-238 ~]# docker container run -itd -p 81:80 --name test-container nginx
```

→run publicip of ec2:81 on browser



### 4) Delete the apache tomcat image from local.

→to delete tomcat image from local follow below command

**--docker image rmi tomcat (use image id or image name)**

```
[root@ip-172-31-7-238 ~]# docker image rmi 88b0f1cee84c
Untagged: tomcat:latest
Untagged: tomcat@sha256:1374a565d5122fdb42807f3a5f2d4fcc245a5e15420ff5bb5123afedc8ef769d
Deleted: sha256:88b0f1cee84c76bb84a450edacdc37fb3ee00a8706be9298dfe8ec69e5040cdb
Deleted: sha256:554b93e4efae63e670310d49458e7ca7901fc90c647f999797084c28e5a33164
Deleted: sha256:8b6e9da6ba7a87bac5b98abb1050ee3f7ce0d7f0c46e5d82a4c6c688199b80c7
Deleted: sha256:8a6df4884b902faae9e4d5036537d74435ebb00f3c70fedbf93d88bf3d11a4b4
Deleted: sha256:2c2c7a0cae5f13fb5b883bc20fa88a75212bcc15cf4adef5a9868bdb944ecf5b
Deleted: sha256:41eac632d049d1723c63f840862edbb9277b229bffa0771f5634216d4be532e1
Deleted: sha256:0502c25b5f5c805c4bdf5a87e08e7688d2bb500a7bedaf29e74762051e473591
Deleted: sha256:078db589602a51936a4c9d2d88398cf70fab89bdc539a46577280951579ab237
Deleted: sha256:81a845f1c33c725b3ad836c9d7d2a3fbae8e8781b8fc7979e8f86288821310cb
```

→checked available images

--docker images

```
[root@ip-172-31-7-238 ~]# docker images
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
jenkins/jenkins      latest          d331b6bde0f8   About an hour ago  467MB
sonarqube            latest          94a7de374354   4 weeks ago     1.17GB
nginx                latest          53a18edff809   7 weeks ago     192MB
ubuntu               latest          a04dc4851cbc   2 months ago    78.1MB
[root@ip-172-31-7-238 ~]#
```

i-0babd69af48e8196f (docker-server)

PublicIPs: 3.147.237.157 PrivateIPs: 172.31.7.238

## 5) Inspect the jenkins image,sonarqube image.

→inspect the Jenkins image

-- docker inspect jenkins/Jenkins

```
[root@ip-172-31-7-238 ~]# docker inspect jenkins/jenkins
[
  {
    "Id": "sha256:d331b6bde0f8ea00fb8969cb4860e7519ba531e647bf89b57adef6b7da2c92",
    "RepoTags": [
      "jenkins/jenkins:latest"
    ],
    "RepoDigests": [
      "jenkins/jenkins@sha256:93554ebe9d60f711c63d1f71ce50cfd05d03ff3a26ab8b0203988854c7b3f1e5"
    ],
    "Parent": "",
    "Comment": "buildkit.dockerfile.v0",
    "Created": "2025-04-02T11:50:38.953273657Z",
    "Container": "",

```

→It show about

- > Environment variables
- > Exposed ports (8080, 50000)
- > Volumes (for persistent data)
- > Image layers
- > metadata

```
    "Cmd": null,
    "Image": "",
    "Volumes": {
      "/var/jenkins_home": {}
    },
    "WorkingDir": "",
    "Entrypoint": [
      "/usr/bin/tini",
      "--",
      "/usr/local/bin/jenkins.sh"
    ]
  ]
}
```

```

    },
    "Architecture": "amd64",
    "Os": "linux",
    "Size": 467378925,
    "GraphDriver": {
      "Data": {
        "LowerDir": "/var/lib/docker/o
      },
      "AttachStdout": false,
      "AttachStderr": false,
      "ExposedPorts": {
        "50000/tcp": {},
        "8080/tcp": {}
      },
      "Tty": false,
      "OpenStdin": false,
      "StdinOnce": false,
      "Env": [
    ],
    "Metadata": {
      "LastTagTime": "0001-01-01T00:00:00Z"
    }
  }
}

```

→inspect the Jenkins image

--docker inspect sonarqube

```

[root@ip-172-31-7-238 ~]# docker inspect sonarqube
[
  {
    "Id": "sha256:94a7de374354b23c76d3c6296244b16ae46b6e1188aff06f81bea35b061ba4e1",
    "RepoTags": [
      "sonarqube:latest"
    ],
    "RepoDigests": [
      "sonarqube@sha256:c0a734bd7e62c1a35794e3a070b4945f5a081b0053201eb926bcb936b0e5d2e6"
    ],
    "Parent": "",
    "Comment": "buildkit.dockerfile.v0",
    "Created": "2025-03-04T14:26:37Z",
    "Container": "",
    "ContainerConfig": {
      "Hostname": "",
      "Domainname": "",
      "User": "",
      "AttachStdin": false,
      "AttachStderr": false,
      "ExposedPorts": {
        "9000/tcp": {}
      },
      "Tty": false,
      "OpenStdin": false,
      "StdinOnce": false,
      "Env": [
        "PATH=/opt/java/openjdk/bin:/usr/lo
        "JAVA_HOME=/opt/java/openjdk",
        "LANG=en_US.UTF-8"
      ]
    }
  }
]

```

## 6) Run jenkins container and run one sample job.

→run below command:

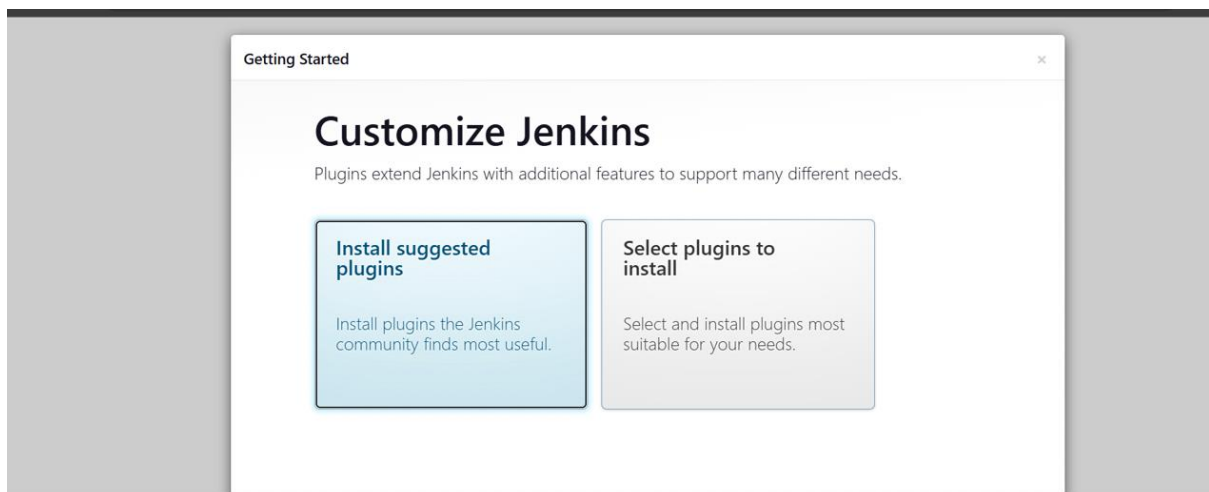
**--docker container run -itd -p 8081:8080 --name stage-container jenkins/Jenkins**

```
[root@ip-172-31-7-238 ~]# docker container run -itd -p 8081:8080 --name stage-container jenkins/jenkins  
aba488587547d8d85e33e76d4de61031699ae91fc61a99a36126c533fca1c7b8  
[root@ip-172-31-7-238 ~]#
```

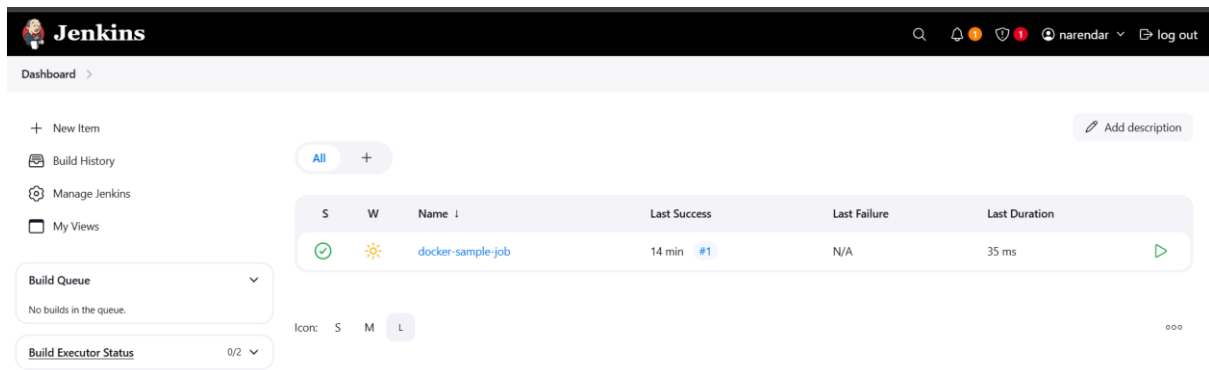
**i-0babd69af48e8196f (docker-server)**

PublicIPs: 3.147.237.157 PrivateIPs: 172.31.7.238

→run public ip of ec2:8081 on browser:

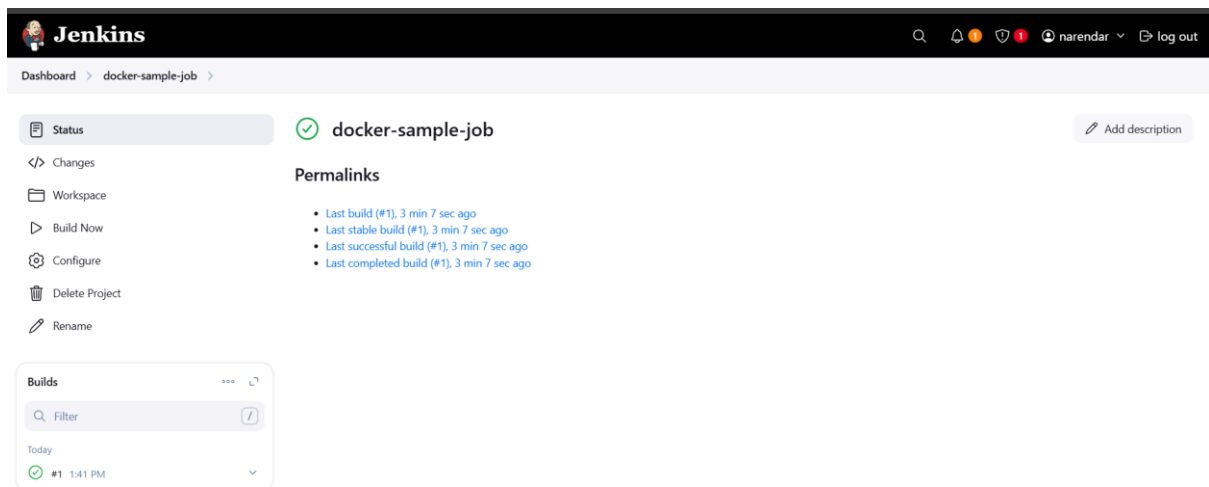


## → Created and run one sample job in Jenkins:



The screenshot shows the Jenkins Dashboard. The top navigation bar includes the Jenkins logo, a search icon, notification icons, a user profile for 'narendar', and a 'log out' link. The main content area features a sidebar on the left with links for 'New Item', 'Build History', 'Manage Jenkins', and 'My Views'. The central panel displays a table of build history for the 'docker-sample-job'. The table has columns for 'S' (Success), 'W' (Warning), 'Name', 'Last Success', 'Last Failure', and 'Last Duration'. A single build is listed with a success status, a warning icon, and a duration of 35 ms. Below the table, there are sections for 'Build Queue' (showing no builds) and 'Build Executor Status' (showing 0/2 executors).

S	W	Name	Last Success	Last Failure	Last Duration
✓	⚠	docker-sample-job	14 min #1	N/A	35 ms



The screenshot shows the Jenkins job detail page for 'docker-sample-job'. The top navigation bar is identical to the dashboard. The breadcrumb trail shows 'Dashboard > docker-sample-job'. The left sidebar contains links for 'Status', 'Changes', 'Workspace', 'Build Now', 'Configure', 'Delete Project', and 'Rename'. The main content area displays the job name 'docker-sample-job' with a success icon. Below this, there is a 'Permalinks' section with a list of links for the last build, last stable build, last successful build, and last completed build, all dated '3 min 7 sec ago'. At the bottom, there is a 'Builds' section with a search filter and a list of builds, showing a single build '#1' at '1:41 PM' with a success status.

**Permalinks**

- [Last build \(#1\), 3 min 7 sec ago](#)
- [Last stable build \(#1\), 3 min 7 sec ago](#)
- [Last successful build \(#1\), 3 min 7 sec ago](#)
- [Last completed build \(#1\), 3 min 7 sec ago](#)

**Builds**

Today

- [#1 1:41 PM](#)