

Step-1: List the images.

```
# docker images
or
# docker image ls
```

Step-2: Now, Start the ubuntu containers:

```
# docker container run -itd ubuntu bash

student@gitOps:~$ docker container ls
CONTAINER ID   IMAGE      COMMAND      CREATED          STATUS          PORTS          NAMES
6e5900f9ba43   ubuntu    "bash"       43 seconds ago   Up 42 seconds           stoic_eas
```

Step-2: Now, Commit the container and create new container image.

```
student@gitOps:~$ docker container commit 6e5900f9ba43 servera
sha256:6a746ab5415859c908baa53c58e408e46dcdb8c38f115459e2fa934990992d94

student@gitOps:~$ docker image ls
REPOSITORY    TAG       IMAGE ID       CREATED          SIZE
servera       latest    6a746ab54158   9 seconds ago   72.8MB
```

Step 3. Before push the container you have to sign-in with hub.docker.com.

```
student@gitOps:~$ docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a
Username: darwikdev11
Password:
WARNING! Your password will be stored unencrypted in /home/student/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
student@gitOps:~$`
```

Step-4: Now, Define the tag. And push the image to docker hub.

```
student@gitOps:~$ docker image tag servera darwikdev11/servera
student@gitOps:~$ docker image ls
REPOSITORY          TAG       IMAGE ID       CREATED          SIZE
darwikdev11/servera latest    6a746ab54158   11 minutes ago   72.8MB
servera             latest    6a746ab54158   11 minutes ago   72.8MB
```

Step-5: Now, pushing client machine.

```
student@git0ps:~$ docker push darwikdev11/servera
```

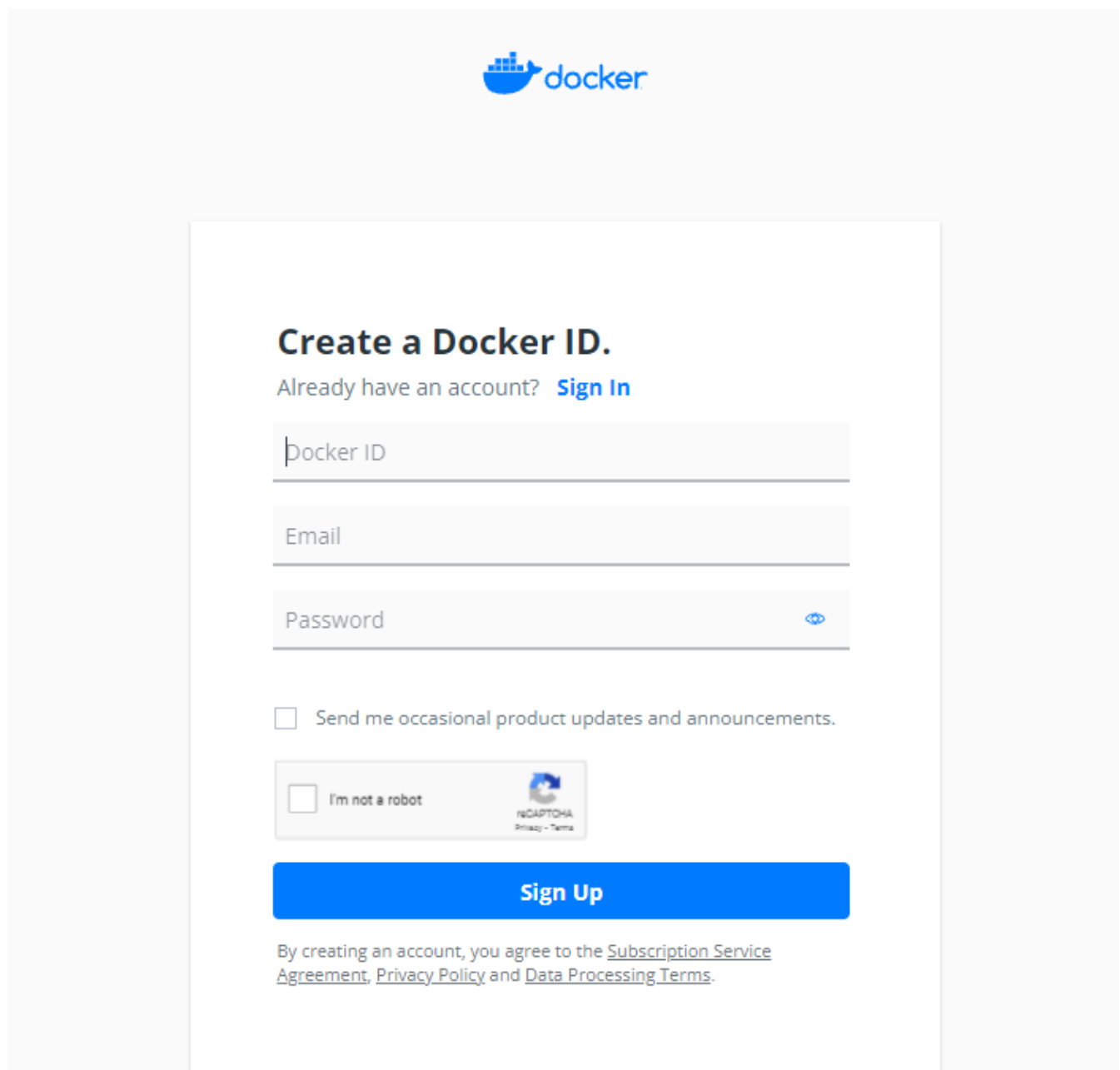
Search the image in private registry.

Step.6: After the pushed image, checkout the website

Login: <https://hub.docker.com>



NOTE: Follow these instructions if you don't have an account.

Step-7.0: If you don't have any account, Now, signup today. Now, go to website :
<http://hub.docker.com>



The screenshot shows the Docker Hub sign-up page. At the top is the Docker logo. Below it, the heading 'Create a Docker ID.' is displayed. A link 'Sign In' is provided for users who already have an account. The form contains three input fields: 'Docker ID', 'Email', and 'Password'. Below the 'Password' field is a checkbox for 'Send me occasional product updates and announcements.' At the bottom of the form is a reCAPTCHA widget with the text 'I'm not a robot' and a 'Sign Up' button. Below the button, there is a disclaimer: 'By creating an account, you agree to the [Subscription Service Agreement](#), [Privacy Policy](#) and [Data Processing Terms](#).'

Step-7.1: If you an account, Now, sign-in.

Welcome Back

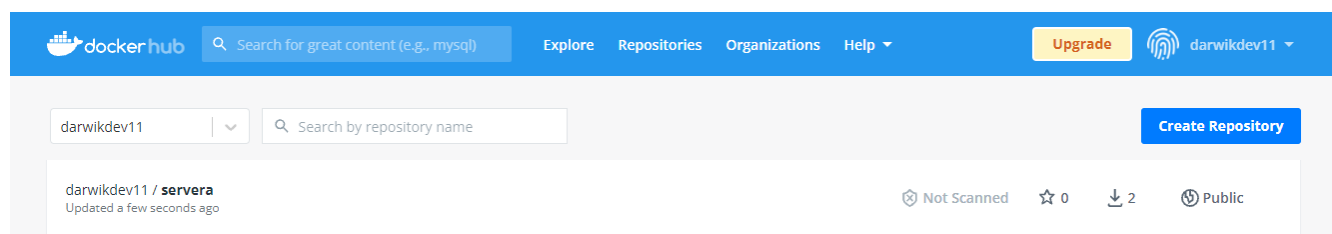
Sign in with your Docker ID

Docker ID is required.

[Sign In](#)

[Forgot Docker ID or Password?](#) | [Sign Up](#)

Step-7.2: After sign-in, you can check your repositories in your account.



The screenshot shows the Docker Hub interface for a user named 'darwikdev11'. The top navigation bar includes the Docker Hub logo, a search bar, and links for Explore, Repositories, Organizations, and Help. A yellow 'Upgrade' button and the user's profile icon are on the right. Below the navigation bar, there's a search bar with 'darwikdev11' selected. A 'Create Repository' button is visible. The main content area shows a repository named 'servera' with a status of 'Not Scanned', 0 stars, 2 downloads, and is public.

Step-7.3: Create your own repository.

Create Repository

Visibility

Using 0 of 1 private repositories. [Get more](#)

☒ Public

Appears in Docker Hub search results

☐ Private

Only visible to you

[Cancel](#)

[Create](#)

Pro tip

You can push a new image to this repository using the CLI

```
docker tag local-image:tagname new-repo:tagname
docker push new-repo:tagname
```

Make sure to change *tagname* with your desired image repository tag.

Step-7.4: tap to create a repository.

Create Repository

darwikdev11 | demo

demo docker |

Visibility

Using 0 of 1 private repositories. [Get more](#)

☒ Public

Appears in Docker Hub search results

☐ Private

Only visible to you

Cancel

Create

Pro tip

You can push a new image to this repository using the CLI

```
docker tag local-image:tagname new-repo:tagname
docker push new-repo:tagname
```

Make sure to change *tagname* with your desired image repository tag.

Step-7.5: after the created repository, look like this :

darwikdev11 > Repositories > demo

Using 0 of 1 private repositories. [Get more](#)

[General](#) Tags Builds Collaborators Webhooks Settings

Advanced Image Management

View all your images and tags in this repository, clean up unused content, recover untagged images. Available for Pro and Team accounts. [View preview](#)

darwikdev11 / demo

demo docker

Last pushed: never

Docker commands [Public View](#)

To push a new tag to this repository,

```
docker push darwikdev11/demo:tagname
```

Tags and Scans

VULNERABILITY SCANNING - DISABLED [Enable](#)

This repository is empty. When it's not empty, you'll see a list of the most recent tags here.

Automated Builds

Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.

Available on Pro and Team plans.


[Upgrade to Pro](#) [Learn more](#)

Step-7.6: For remove : go to settings, and delete the repository.

[General](#) [Tags](#) [Builds](#) [Collaborators](#) [Webhooks](#) [Settings](#)

Vulnerability Scanning

Enable vulnerability scanning to identify vulnerabilities in your container images. Images you push to Hub after enabling this option will be automatically scanned.

[Enable Image Scanning](#)  Vulnerability scanning is off

Visibility Settings

Using 0 of 1 private repositories. [Get more](#)

Make this repository private. Private repositories are only available to you or members of your organization.

[Make private](#)

Delete Repository

Deleting a repository will **destroy** all images stored within it! This action is **not reversible**.

[Delete repository](#)

Delete Repository: darwikdev11/demo

This deletes the repository, all the images it contains, and its build settings. This cannot be undone.

Please type the name of your repository to confirm deletion: **demo**

demo|

[Cancel](#)[Delete](#)

Step-8: Search a new image.

The screenshot shows the Docker Hub interface with a search for 'mysql'. The top navigation bar includes the Docker Hub logo, a search bar with 'mysql' entered, and links for Explore, Repositories, Organizations, and Help. A user profile for 'darwikdev11' is visible in the top right. Below the navigation bar, there are tabs for Docker, Containers (selected), and Plugins. The main content area displays search results for 'mysql', showing 1 - 25 of 29,021 results. A sidebar on the left contains filters for Images (Verified Publisher, Official Images) and Categories (Analytics, Application Frameworks, etc.). The search results list three images: 'mysql' (Official Image, 1B+ Downloads, 10K+ Stars), 'mariadb' (Official Image, 1B+ Downloads, 4.4K Stars), and 'percona' (Official Image, 100M+ Downloads, 558 Stars). Each image entry includes a description and a list of supported architectures.

Filters

1 - 25 of 29,021 results for **mysql**. [Clear search](#)

Most Popular

Images

- ☐ Verified Publisher
- ☐ Official Images

Official Images Published By Docker

Categories

- ☐ Analytics
- ☐ Application Frameworks
- ☐ Application Infrastructure
- ☐ Application Services
- ☐ Base Images
- ☐ Databases
- ☐ DevOps Tools
- ☐ Featured Images
- ☐ Messaging Services
- ☐ Monitoring
- ☐ Operating Systems
- ☐ Programming Languages
- ☐ Security
- ☐ Storage

mysql Official Image 1B+ 10K+
Downloads Stars
Updated 44 minutes ago
MySQL is a widely used, open-source relational database management system (RDBMS).
Container Linux x86-64 Databases

mariadb Official Image 1B+ 4.4K
Downloads Stars
Updated 44 minutes ago
MariaDB Server is a high performing open source relational database, forked from MySQL.
Container Linux PowerPC 64 LE ARM 64 386 IBM Z x86-64 Databases

percona Official Image 100M+ 558
Downloads Stars
Updated 41 minutes ago
Percona Server is a fork of the MySQL relational database management system created by Percona.
Container Linux x86-64 386 Application Infrastructure

Step-8.1: now go back to the terminal and pull the image.

```
# docker pull mysql
```