



[Course](#) > [Chapter 4: M...](#) > [Manipulatin...](#) > [Tagging Ima...](#)

## Tagging Images

A project with multiple images based on the same software could be distributed, creating individual projects for each image; however, this approach requires extra work to manage and deploy the images to the correct locations.

Container image registries support the tag concept so that you can distinguish multiple releases of the same project. For example, a customer might use a container image to run with a MySQL or PostgreSQL database, using a tag as a way to differentiate which database will be used by a container image.

Usually, the tags are used by container developers to distinguish between multiple versions of the same software, such as the one observed for MySQL container image documentation.

### Note

Multiple tags are provided to easily identify a release. On the official MySQL container image website, the version is used as the tag's name (5.5.16). In addition, the same image has a second tag with the minor version (5.5) to minimize the need to get the latest release for a certain version.



To tag an image, use the **docker tag** command:

```
# docker tag [OPTIONS] IMAGE[:TAG] [REGISTRYHOST/][USERNAME/]NAME[:TAG]
```

The **IMAGE** argument is the image name with an optional tag that was locally stored to the docker daemon. The following argument refers to alternative names for the image that are stored locally. If no tag is provided, the **latest** tag will be considered. For example, to tag an image, the following command may be used:

```
# docker tag mysql-custom devops/mysql
```

The **mysql-custom** option is the image name that is stored in the **docker** daemon's cache.

To use a different tag name, use the following command instead:

```
# docker tag mysql-custom devops/mysql:snapshot
```

## Removing Tags from the Image

To associate multiple tags with a single image, use the **docker tag** command. Tags can be removed using the **docker rmi** command mentioned previously. Therefore, to delete a specific image tag from the daemon, run the following command:

```
# docker rmi devops/mysql:snapshot
```

### Note



Because multiple tags can point to the same image, to remove an image referred to by multiple tags, each tag should be individually removed first. Alternatively, use the **docker rmi --force** command.

© All Rights Reserved

