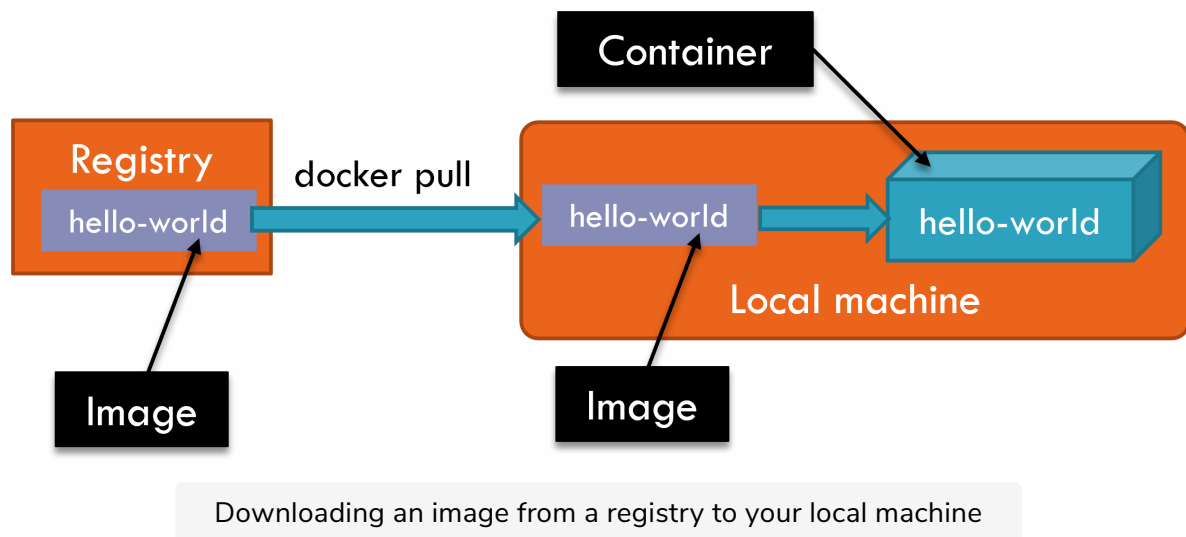


Where Do Images Come From?

Throughout this chapter, we have been using images to create containers; but where do images come from? We explore just that in this lesson.

Each container is created from an image. You provide the image name to the *docker run* command. Docker first looks for the image locally and uses it when present. When the image is not present locally, it is downloaded from a *registry*.



You can list the local images using the following command: *docker images ls*

When an image is published to a registry, its name must be:

```
<repository_name>/<name>:<tag>
```

- *tag* is optional; when missing, it is considered to be *latest* by default
- *repository_name* can be a registry DNS or the name of a registry in the Docker Hub

We'll soon see more about [Docker Hub](#) and [private registries](#). All of the images

we've been using until now were downloaded from Docker Hub as they are

not DNS names. When you have time, you should browse the [Docker Hub](#) and get familiar with the images it provides.

For instance, the [Jenkins image](#) may be found on the Docker Hub.

Although the *docker run* command downloads images automatically when missing, you may want to trigger the download manually. To do this, you can use the *docker pull* command. A pull command forces an image to download, whether it is already present or not.

Here are some scenarios where using a *docker pull* command is relevant:

- You expect that the machine which runs the containers does not have access to the registries (e.g., no internet connection) at the time of running the containers.
- You want to ensure you have the latest version of an image tagged as “latest,” which [wouldn't be downloaded](#) by the *docker run* command.

Let's wrap up this chapter with a quiz to test what you have learned so far.