This image has a tag for every release of blue ocean, to run the latest, ensure you run docker pull jenkinsci/blueocean from time to time.

A docker image to give BlueOcean a try

* run as docker run -p 8080:8080 jenkinsci/blueocean
* note the admin password dumped on log
* open a browser on [http://localhost:8080](http://localhost:8080/)
* run the initial setup wizard. Choose "recommended plugins"
* browse to <http://localhost:8080/blue>

The Jenkins project provides a Linux container image, not a Windows container image. Be sure that your Docker for Windows installation is configured to run Linux Containers rather than Windows Containers. See the Docker documentation for instructions to [switch to Linux containers](https://docs.docker.com/docker-for-windows/#switch-between-windows-and-linux-containers). Once configured to run Linux Containers, the steps are:

1. Open up a command prompt window.
2. Create a [bridge network](https://docs.docker.com/network/bridge/) in Docker using the following [docker network create](https://docs.docker.com/engine/reference/commandline/network_create/) command:

docker network create jenkins

1. Create the following [volumes](https://docs.docker.com/storage/volumes/) to share the Docker client TLS certificates needed to connect to the Docker daemon and persist the Jenkins data using the following [docker volume create](https://docs.docker.com/engine/reference/commandline/volume_create/) commands:

docker volume create jenkins-docker-certs

docker volume create jenkins-data

1. In order to execute Docker commands inside Jenkins nodes, download and run the docker:dind Docker image using the following [docker container run](https://docs.docker.com/engine/reference/commandline/container_run/) command:

docker container run --name jenkins-docker --rm --detach ^

--privileged --network jenkins --network-alias docker ^

--env DOCKER\_TLS\_CERTDIR=/certs ^

--volume jenkins-docker-certs:/certs/client ^

--volume jenkins-data:/var/jenkins\_home ^

--volume "%HOMEDRIVE%%HOMEPATH%":/home ^

docker:dind

1. Run the jenkinsci/blueocean image as a container in Docker using the following [docker container run](https://docs.docker.com/engine/reference/commandline/container_run/) command (bearing in mind that this command automatically downloads the image if this hasn’t been done):

docker container run --name jenkins-blueocean --rm --detach ^

--network jenkins --env DOCKER\_HOST=tcp://docker:2376 ^

--env DOCKER\_CERT\_PATH=/certs/client --env DOCKER\_TLS\_VERIFY=1 ^

--volume jenkins-data:/var/jenkins\_home ^

--volume jenkins-docker-certs:/certs/client:ro ^

--volume "%HOMEDRIVE%%HOMEPATH%":/home ^

--publish 8080:8080 --publish 50000:50000 jenkinsci/blueocean

|  |
| --- |
| Maps the /var/jenkins\_home directory in the container to the Docker [volume](https://docs.docker.com/engine/admin/volumes/volumes/) with the name jenkins-data. If this volume does not exist, then this docker container run command will automatically create the volume for you. |
|  | Maps the $HOME directory on the host (i.e. your local) machine (usually the /Users/<your-username> directory) to the /home directory in the container. |

**Note:** If copying and pasting the command snippet above doesn’t work, try copying and pasting this annotation-free version here:

docker container run --name jenkins-tutorial --rm --detach \

--network jenkins --env DOCKER\_HOST=tcp://docker:2376 \

--env DOCKER\_CERT\_PATH=/certs/client --env DOCKER\_TLS\_VERIFY=1 \

--volume jenkins-data:/var/jenkins\_home \

--volume jenkins-docker-certs:/certs/client:ro \

--volume "$HOME":/home --publish 8080:8080 jenkinsci/blueocean

1. Proceed to the [Setup wizard](https://jenkins.io/doc/tutorials/build-a-node-js-and-react-app-with-npm/#setup-wizard).

Clone the project for node and react from <https://github.com/jenkins-docs/simple-node-js-react-npm-app>

References

<https://hub.docker.com/r/jenkinsci/blueocean/>

<https://jenkins.io/doc/tutorials/build-a-node-js-and-react-app-with-npm/>