

# Checkpoint

- ☐ Container
- ☐ Image
- ☐ Dockerfile
- ☐ Registry
- ☐ Auto build with Github (Bonus!)

# Checkpoint

- ☐ Basic docker volume
- ☐ Docker network
- ☐ Docker multistage build
- ☐ Real-world example project
- ☐ Deploy container to Heroku (Bonus!)

# Container

## Lab 1. Run first container

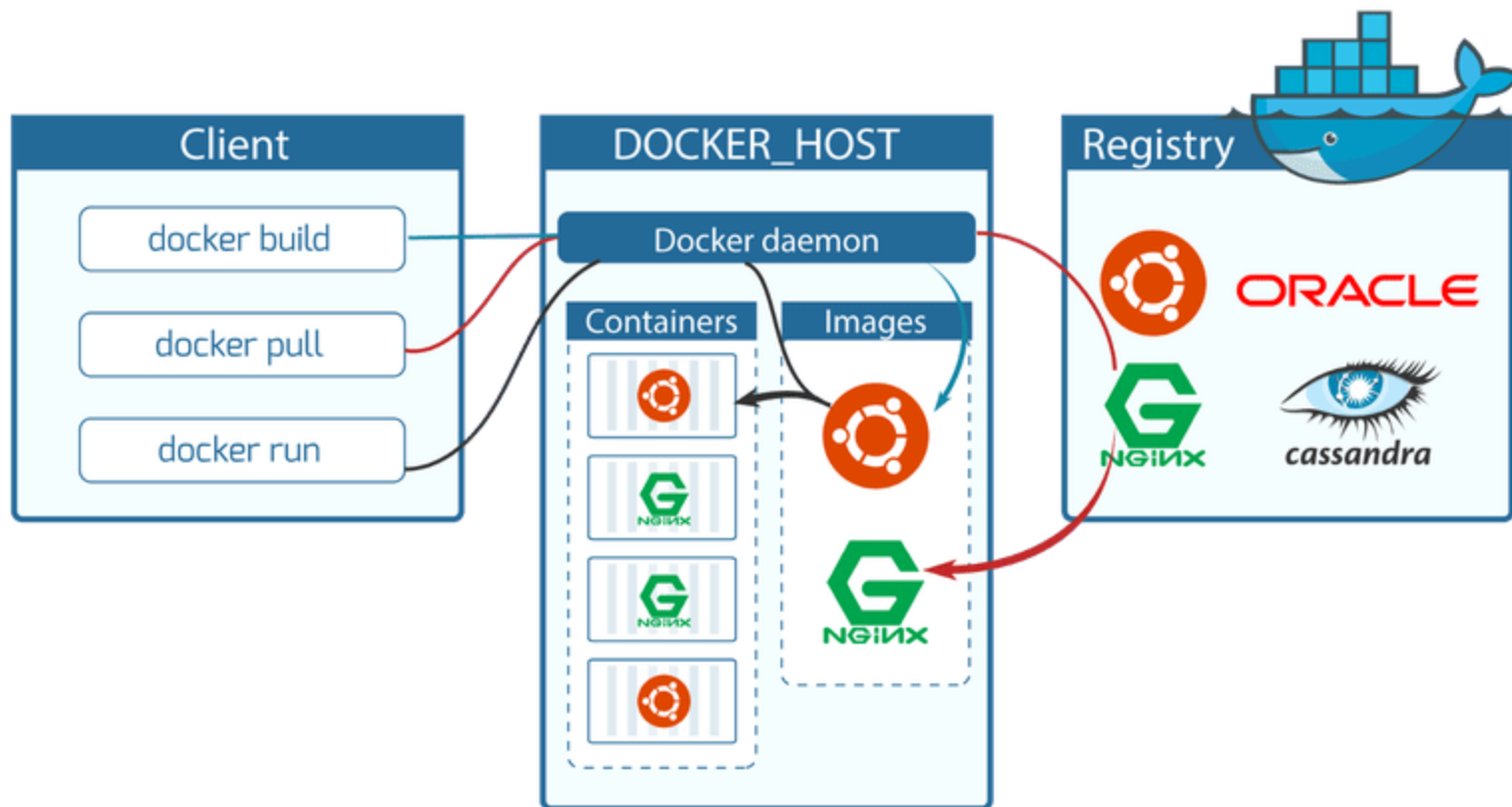
```
docker container run hello-world
```

```
docker run hello-world
```

# Hello World: What Happened?



# DOCKER COMPONENTS



# Image

## Lab 2. Image management

List of images

```
docker image ls
```

Remove image

```
docker image rm {{imageName}} -f
```

```
docker rmi {{imageName}} -f
```

Pull image

```
docker image pull alpine
```

```
docker pull alpine
```

## Lab 3. Execute command in container

```
docker container run alpine ls -l
```

```
docker run alpine ls -l
```

```
docker container run alpine echo "hello from alpine"
```

```
docker run alpine echo "hello from alpine"
```

```
docker run alpine sh -c 'echo "hello from alpine"'
```

## Lab 4. Playing in container

```
docker run alpine sleep 3600
```

```
docker ps -a
```

```
docker exec {{containerId}} ls
```



```
docker exec {{containerId}} echo "Hello to file" > hello.txt
```

```
docker exec {{containerId}} cat hello.txt
```

## Kill all container

```
docker kill $(docker ps -q)
```

## *docker run* Details

- 1 Run the **alpine** container and send **ls -l**

```
$ docker run alpine ls -l
```

Docker Engine

Linux



- 2 **alpine** launches and runs **ls -l**



Docker Engine

Linux



- 3 **alpine** shuts down and output of **ls -l** sent back to host OS

```
total 8
drwxr-xr-x  ...  bin
drwxr-xr-x  ...  dev
drwxr-xr-x  ...  etc
drwxr-xr-x  ...  home
.
```



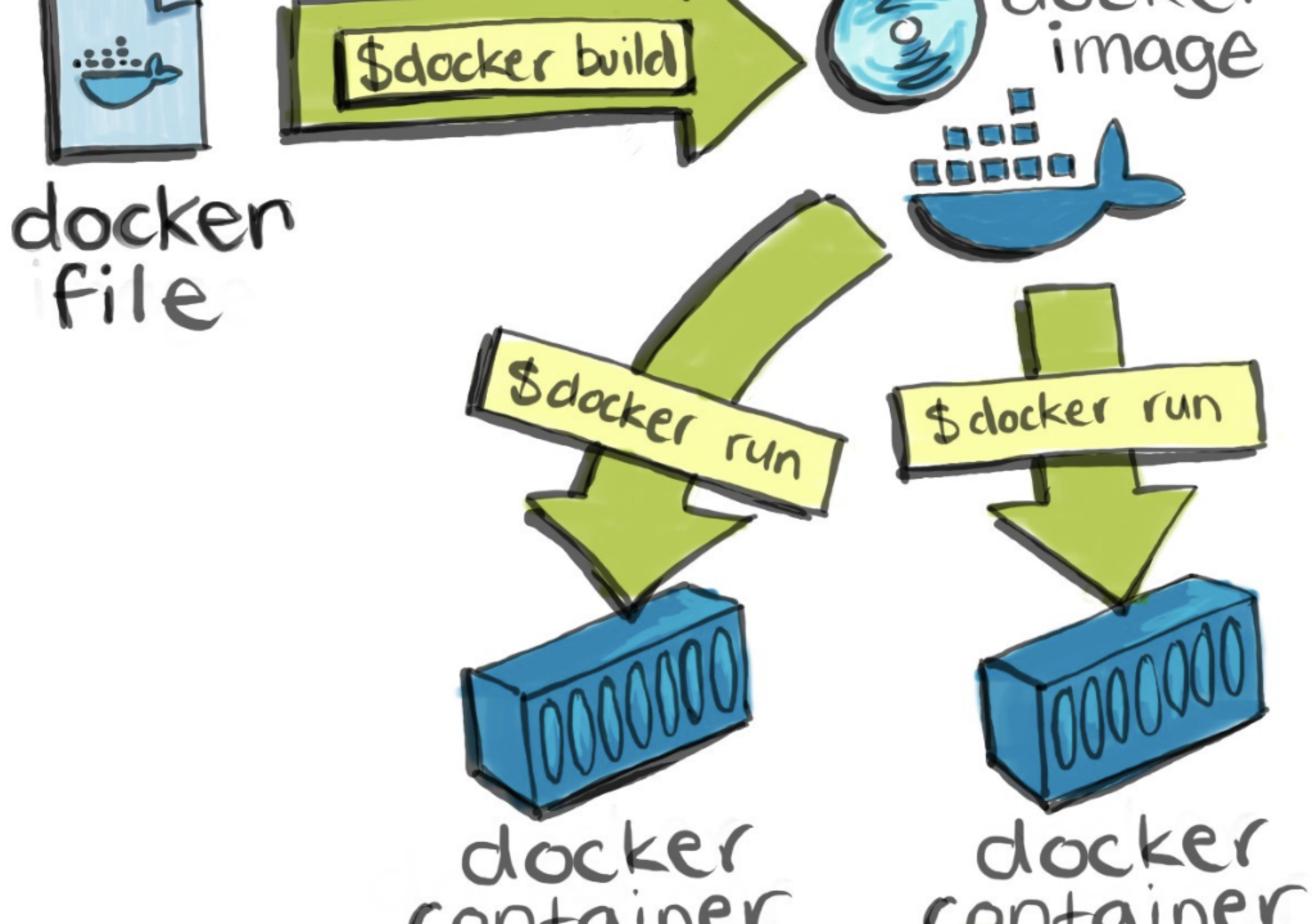
Docker Engine

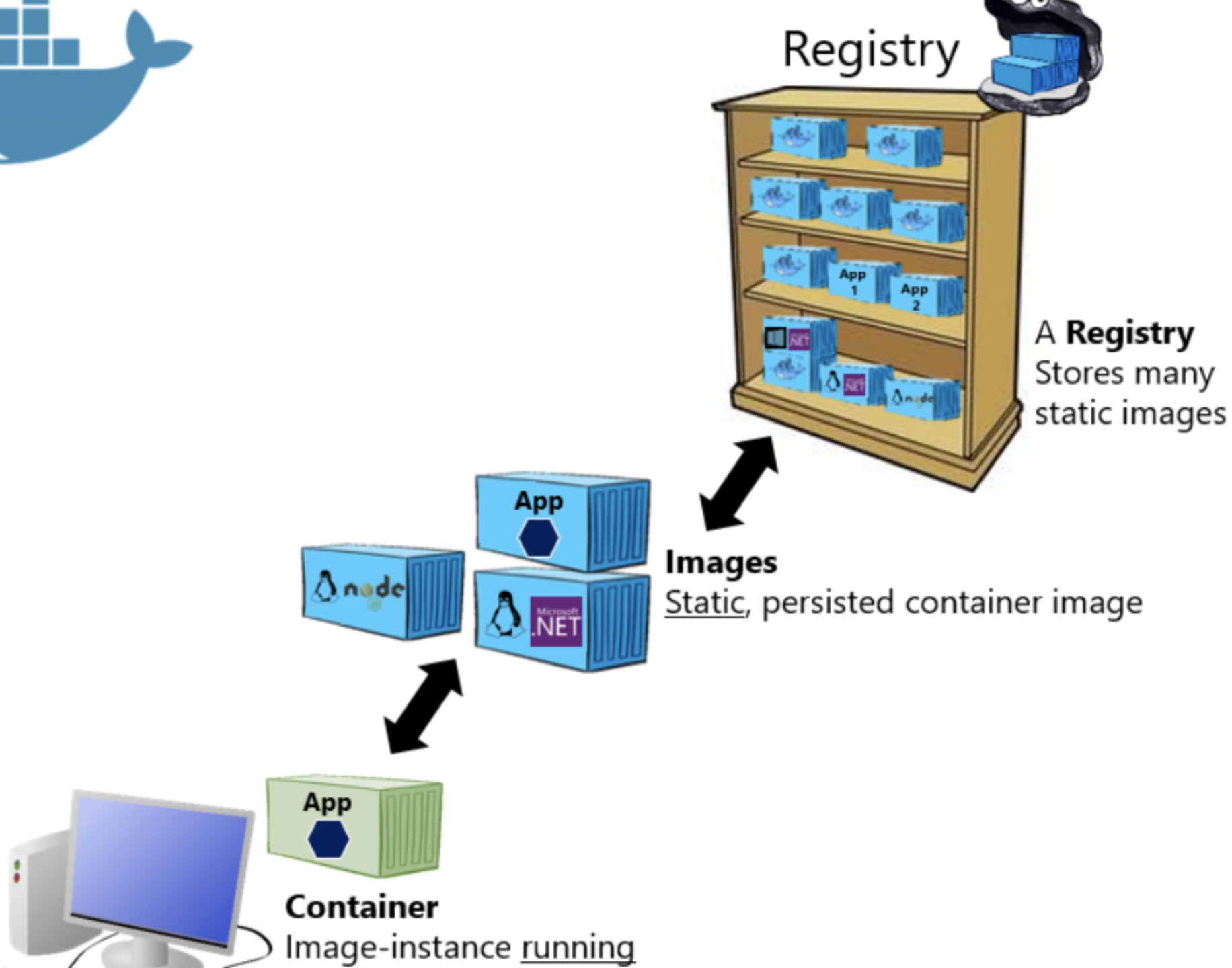
Linux



# Docker way

- 1 application = 1 container.
- Run process in the foreground.
- Keep data out of container -- use volumes.
- Do not use SSH (if you need to step into container you can use `docker exec` command).
- Avoid manual configurations (or actions) inside container.





# Dockerfile

```
FROM -- set base image
RUN -- execute command in container
ENV -- set environment variable
COPY -- copy file to container
WORKDIR -- set working directory
VOLUME -- create mount-point for a volume
CMD -- set executable for container that maybe used
ENTRYPOINT -- set executable for container that always used
```

## CMD VS RUN

- RUN - command triggers while we build the docker image.
- CMD - command triggers while we launch the created docker image.



## CMD VS ENTRYPOINT

- Only one command `CMD` or `ENTRYPOINT` in Dockerfile
- `CMD` - can replace with command when run container
- `ENTRYPOINT` - can't replace with command when run container

## Lab 6. Create Hello world image

```
docker run alpine echo "Hello world"
```

```
docker build -t alpine-hello:0.1 .
```

```
docker run alpine-hello:0.1
```

```
docker history {{imageName}}
```

## Lab 7. Pass variable

```
docker build -t alpine-hello:0.2 .
```

```
docker run alpine-hello:0.2
```

```
docker run -e WELCOME="Hi!!" alpine-hello:0.2
```

## Lab 8.1 Run Node.js application

- [https://hub.docker.com/\\_/node](https://hub.docker.com/_/node) (node:14-alpine3.12)

```
docker pull node:14-alpine3.12
```

```
docker run node:14-alpine3.12 node -v
```

## Lab 8.2 Run Node.js application by yourself

1. Add server.js files to own Dockerfile
2. Build own Dockerfile ~> Image
3. Run own image

## Build own image

```
docker build -t app-js:0.1 .
```

## Run image in foreground process

```
docker run -p 81:80 app-js:0.1
```

## Run image in background process

```
docker run -d -p 81:80 app-js:0.1
```



# Checkpoint

- ☒ Container
- ☒ Image
- ☒ Dockerfile
- ☐ Registry
- ☐ Auto build with Github (Bonus!)

## Push own image to public registry

- Go to ~> <http://dockerhub.com>
- Register new account
- Create new public registry

```
docker login
```

```
docker build -t {{Registry name}}:{{Tag}} .
```

```
docker push {{Registry name}}:{{Tag}}
```

```
docker pull {{Registry name}}:{{Tag}}
```

```
docker run {{Registry name}}:{{Tag}}
```



# Create new repository on Github

- Go to ~> <http://github.com>
- Register new account
- Create new public registry
- [:Ref](#)



Search or jump to...



Pull requests

Issues

Marketplace

Explore



Shehanka / ITP-GitHub-intro

<> Code



Issues 1



Pull requests



Actions



Projects



Wiki



Security



Insights



Settings



master ▾



1 branch



0 tags

Go to file

Add file ▾



Code ▾



Shehanka Delete node.js.yml



ad97946 11 minutes ago



9 commits

📁 Shehanka / ITP-GitHub-intro

<> Code

⚠ Issues 1

🔗 Pull requests

▶ Actions

📁 Projects

📖 Wiki

🛡 Security

📈 Insights

⚙ Settings

Workflows

New workflow

All workflows

🔗 Docker Image CI

All workflows

🔍 Filter workflows



# Choose a workflow template

Build, test, and deploy your code. Make code reviews, branch management, and issue triaging work the way you want. Select a workflow template to get started.

Skip this and [set up a workflow yourself](#) →

## Workflows made for your Dockerfile repository Suggested

### Publish Docker Container

By GitHub Actions

Build, test and push Docker image to GitHub Packages.

[Set up this workflow](#)

```
if [ -f docker-compose.test.yml ]; then
docker-compose --file docker-compose.test.yml build
docker-compose --file docker-compose.test.yml run sut
```

 actions/starter-workflows

Dockerfile ●

### Docker image

By GitHub Actions

Build a Docker image to deploy, run, or push to a registry.

[Set up this workflow](#)

```
docker build . --file Dockerfile --tag my-image-name:$(date +%s)
```

 actions/starter-workflows

Dockerfile ●

name: Docker Image CI

on:

push:

branches: [ master ]

pull\_request:

branches: [ master ]

jobs:

build:

runs-on: ubuntu-latest

steps:

- uses: actions/checkout@v2

- name: docker login

env:

DOCKER\_USER: \${secrets.DOCKER\_USER}}

DOCKER\_PASSWORD: \${secrets.DOCKER\_PASSWORD}}

run: |

docker login -u \$DOCKER\_USER -p \$DOCKER\_PASSWORD


- name: Build the Docker image

run: docker build . --file Dockerfile --tag chamodshehanka/node-

test:\$(date +%s)


name: Docker Push

 Shehanka / ITP-GitHub-intro

 Unw

<> Code

! Issues 1


 Pull requests

 Actions

 Projects

 Wiki

 Security

 Insights

 Settings

 master ▾

 1 branch

 0 tags

Go to file

Add file ▾

 Code ▾

About



Shehanka Delete node.js.yml

✓ ad97946 21 minutes ago ⌚ 9 commits

No des  
provid



.github/workflows

Delete node.js.yml

21 minutes ago

Manage access

Security & analysis

Branches

Webhooks

Notifications

Integrations

Deploy keys

Autolink references

Secrets

Actions

Repos

ITP-

☐ Ter

Ten

Social

Upload

Image

[Down](#)

---

## Secrets

New secret

Secrets are environment variables that are **encrypted** and only exposed to selected actions. Anyone with **collaborator** access to this repository can use these secrets in a workflow.

Secrets are not passed to workflows that are triggered by a pull request from a fork. [Learn more](#).

**Name**

DOCKER\_USER

**Value**

YOUR\_USER\_NAME|



Add secret

# Secrets

New secret

Secrets are environment variables that are **encrypted** and only exposed to selected actions. Anyone with **collaborator** access to this repository can use these secrets in a workflow.

Secrets are not passed to workflows that are triggered by a pull request from a fork. [Learn more](#).

 DOCKER_PASSWORD	Updated 30 minutes ago	<a href="#">Update</a>	<a href="#">Remove</a>
 DOCKER_USER	Updated 31 minutes ago	<a href="#">Update</a>	<a href="#">Remove</a>

✓ Docker Image CI

on: push

✓ build

Docker Image CI / build

succeeded 27 minutes ago in 47s

- ▶ ✓ Set up job
- ▶ ✓ Run actions/checkout@v2
- ▶ ✓ docker login
- ▶ ✓ Build the Docker image
- ▶ ✓ Docker Push

▼ ✓ Post Run actions/checkout@v2

```
1 Post job cleanup.  
2 /usr/bin/git version  
3 git version 2.27.0  
4 /usr/bin/git config --local --name-only --get-regexp core\.sshCommand  
5 /usr/bin/git submodule foreach --recursive git config --local --name-only --get-regexp  
all 'core.sshCommand' || :  
6 /usr/bin/git config --local --name-only --get-regexp http\.https\:\/\/github\.com\/  
7 http.https://github.com/.extraheader  
8 /usr/bin/git config --local --unset-all http.https://github.com/.extraheader  
9 /usr/bin/git submodule foreach --recursive git config --local --name-only --get-regexp  
git config --local --unset-all 'http.https://github.com/.extraheader' || :
```

- ▶ ✓ Complete job



## chamodshehanka / node-test

*This repository does not have a description* 

 Last pushed: 29 minutes ago

### Docker commands

[Public View](#)

To push a new tag to this repository,

```
docker push chamodshehanka/node-test:tagname
```

### Tags

This repository contains 2 tag(s).

1596277810



 29 minutes ago

1596277716



 31 minutes ago

[See all](#)

### Recent builds

*Link a source provider and run a build to see build results here.*

# Checkpoint

- [x] Container
- [x] Image
- [x] Dockerfile
- [x] Registry
- [x] Auto build with Github (Bonus!)

