DOCKER

DEVELOPMENT TO PRODUCTION

@kellyjandrews - @nexmodev

SOFTWARE ARCHITECT

INCONSISTENT
TOOLCHAIN

2 WEEK
ON-BOARDING

NO RELIABLE TESTING ENV

MANUAL RELEASES

HEAVY GROWTH

=

BIG PROBLEMS

SOLUTION?

WHAT *IS* DOCKER?

SO A VIRTUAL MACHINE?

TOOLCHAIN
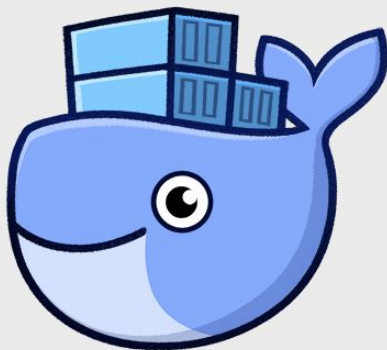
Docker Desktop
Community

Version 2.0.0.0-mac81 (29211)

Channel: stable

4271b9e6a8

Engine: 18.09.0

Compose: 1.23.2

Machine: 0.16.0

Notary: 0.6.1

Credential Helper: 0.6.0

Kubernetes: v1.10.3

Release Notes   Acknowledgements   License Agreement

```
FROM node:11.8.0-alpine

WORKDIR /usr/app

RUN apk update && apk add postgresql
COPY package.json .
RUN npm install --quiet

COPY . .

CMD ["npm", "start"]
```
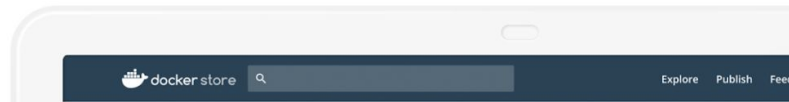
docker store

The Docker Store

Find Trusted and Enterprise Ready Containers, Plugins, and Docker Editions

🔍 Search for great content (e.g. mysql)

🐳 Docker EE   🐳 Docker CE   ▣ Containers   🧩 Plugins

GET DOCKER

Get started with Docker

Q node

node

By **Docker**

Node.js is a JavaScript-based platform for server-side and networking applications.

⬇ **10M+**

Container   Linux   x86-64   IBM Z   Application Infrastructure

Official Image

$0.00

Terms of Service

Linux - x86-64   ▼

Copy and paste to pull this image

`docker pull node`   ⧉

View Available Tags

DESCRIPTION   REVIEWS   RESOURCES

Supported tags and respective `Dockerfile` links

Average Rating: ★★★★★   11 Ratings

- `8.6.0`, `8.6`, `8`, `latest` (*8.6/Dockerfile*)
- `8.6.0-alpine`, `8.6-alpine`, `8-alpine`, `alpine` (*8.6/alpine/Dockerfile*)
- `8.6.0-onbuild`, `8.6-onbuild`, `8-onbuild`, `onbuild` (*8.6/onbuild/Dockerfile*)
- `8.6.0-slim`, `8.6-slim`, `8-slim`, `slim` (*8.6/slim/Dockerfile*)
- `8.6.0-stretch`, `8.6-stretch`, `8-stretch`, `stretch` (*8.6/stretch/Dockerfile*)
- `8.6.0-wheezy`, `8.6-wheezy`, `8-wheezy`, `wheezy` (*8.6/wheezy/Dockerfile*)
- `6.11.4`, `6.11`, `6`, `boron` (*6.11/Dockerfile*)
- `6.11.4-alpine`, `6.11-alpine`, `6-alpine`, `boron-alpine` (*6.11/alpine/Dockerfile*)
- `6.11.4-onbuild`, `6.11-onbuild`, `6-onbuild`, `boron-onbuild` (*6.11/onbuild/Dockerfile*)

```yaml
version: '3.7'
services:
  web:
    build: .
    command: npm run dev
    volumes:
    - .:/usr/app/
    - /usr/app/node_modules
    ports:
      - "3000:3000"
    depends_on:
      - postgres
    environment:
      DATABASE_URL: postgres://postgres@postgres
  postgres:
    image: postgres:9.6.2-alpine
```

CREATE
DOCKERFILE

```
FROM node:11.8.0-alpine as dev-stage

MAINTAINER Kelly J Andrews <kelly@kellyjandrews.com>

WORKDIR /usr/src

COPY package.json .

RUN npm install --quiet

ENV PATH /usr/src/node_modules/.bin:$PATH

COPY . .

EXPOSE 3000

CMD ["npm", "start"]
```

DEFINE DEV ENVIRONMENT

```yaml
version: "3.7"
services:
  app:
    build:
      context: .
      target: dev-stage
    command: ["npm", "run", "dev"]
    volumes:
    - .:/usr/src/
    - /usr/src/node_modules
    ports:
      - "3000:3000"
```

RUN THE ENVIRONMENT

IMMEDIATE
ON-BOARDING

WHAT ABOUT TESTS?

DOCKER IS
EPHEMERAL

RELIABLE TESTING ENVIRONMENT

REMOTE TESTING IS HARD

DEPLOYMENTS?

MANUAL OR AUTOMATED

ENVIRONMENT IS DONE

SWARM/K8S FOR ORCHESTRATION

TEXT "JOKE"
613-482-0609

THANKS!
@kellyjandrews