总结

# 安装Docker

<https://docs.docker.com/engine/installation/linux/ubuntu/#install-from-a-package>

# 安装Compose

<https://docs.docker.com/compose/install/>

Docker 与 Compose 版本对照

Docker version 17.03.1-ce, build c6d412e

查看发布版本

<https://github.com/docker/compose/releases>

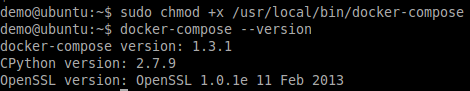
安装/升级方法(需要vpn)

sudo curl -L "https://github.com/docker/compose/releases/download/1.11.2/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

sudo chmod +x /usr/local/bin/docker-compose

验证安装

docker-compose --version



# 使用

docker images

docker run meanjs/mean

删除所有容器

docker container rm -f $(docker container ls -qa)

# MEANJS

## Docker启动

|  |
| --- |
| 镜像地址 <https://hub.docker.com/r/meanjs/mean/>  获取镜像 docker pull meanjs/mean  删除所有容器和数据  docker container rm -f $(docker container ls -qa)  sudo rm -fr /home/demo/nodejs\_projects/docker\_meanjs/mongodb\_data/\*  #分别启动数据库和meanjs  sudo docker run \  -p 27017:27017 \  --name db \  -v /home/demo/nodejs\_projects/docker\_meanjs/mongodb\_data:/data/db \  -d mongo  sudo docker run \  -p 3000:3000 \  -p 35729:35729 \  --name meanjs4docker \  -v /home/demo/nodejs\_projects/docker\_meanjs/public:/home/mean/public \  -v /home/demo/nodejs\_projects/docker\_meanjs/app:/home/mean/app \  --link db:db\_1 meanjs/mean  docker container rm -f db meanjs4docker  sudo rm -fr /home/demo/nodejs\_projects/docker\_meanjs/mongodb\_data/\*  docker run -p 27017:27017 -d --name db mongo  docker run -p 3000:3000 --link db:db\_1 mean  **Express**  docker run \  --name mongo-express \  --link db:mongo \  -p 8081:8081 \  -e ME\_CONFIG\_OPTIONS\_EDITORTHEME="ambiance" \  -e ME\_CONFIG\_BASICAUTH\_USERNAME="user" \  -e ME\_CONFIG\_BASICAUTH\_PASSWORD="user" \  -d mongo-express |

# MEANIO

## Docker启动

镜像地址： <https://hub.docker.com/r/gbevan/meanio/>

获取镜像： docker pull gbevan/meanio

运行容器： docker run -d -p 80:3000 --name yourapp gbevan/meanio

安装meanio 应用必须使用 mean 用户的目录~/appserver中 (密码也是 'mean')

其他详细如下：

Mean.io scaffolding unit-tests are run as part of the docker image build, as a

filter to ensure released images pass all upstream unit-tests.

Entrypoint for the image is /start-image.sh (see source on GitHub for details),

this ensures mongod is started, the css and javascript are minified/uglified for

performance, and starts up the meanio app using forever to always keep it running.

The start-image.sh script also provides a hook called /start-image-hook.sh which

can be embedded in downstream (aka docker images using meanio as a base) and

called during the startup of the container.

安装Docker Compose

取得镜像： docker pull meanjs/mean:removeGrunt

* Local development and testing with compose:
* $ docker-compose up
* Local development and testing with just Docker:
* $ docker build -t mean .
* $ docker run -p 27017:27017 -d --name db mongo
* $ docker run -p 3000:3000 --link db:db\_1 mean
* $
* To enable live reload, forward port 35729 and mount /app and /public as volumes:
* $ docker run -p 3000:3000 -p 35729:35729 -v /Users/mdl/workspace/mean-stack/mean/public:/home/mean/public -v /Users/mdl/workspace/mean-stack/mean/app:/home/mean/app --link db:db\_1 mean

### Production deploy with Docker

* Production deployment with compose:
* $ docker-compose -f docker-compose-production.yml up -d
* Production deployment with just Docker:
* $ docker build -t mean -f Dockerfile-production .
* $ docker run -p 27017:27017 -d --name db mongo
* $ docker run -p 3000:3000 --link db:db\_1 mean

## Getting Started With MEAN.JS

You have your application running, but there is a lot of stuff to understand. We recommend you go over the [Official Documentation](http://meanjs.org/docs.html).  
In the docs we'll try to explain both general concepts of MEAN components and give you some guidelines to help you improve your development process. We tried covering as many aspects as possible, and will keep it updated by your request. You can also help us develop and improve the documentation by checking out the gh-pages branch of this repository.