Docker-compose部署显示helloworld

Docker-comopose安装

下载

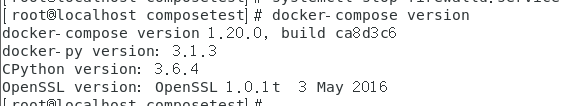
sudocurl-Lhttps://github.com/docker/compose/releases/download/1.20.0/docker-compose-`uname -s`-`uname -m` -o /usr/local/bin/docker-compose

安装

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

查看版本

Docker-compose version



出现上面信息就说明安装完成

创建项目路径

Mkdir composetest

Cd composetest

在当前目录下创建app.py文件

import time

import redis

from flask import Flask

app = Flask(\_\_name\_\_)

cache = redis.Redis(host='redis', port=6379)

def get\_hit\_count():

retries = 5

while True:

try:

return cache.incr('hits')

except redis.exceptions.ConnectionError as exc:

if retries == 0:

raise exc

retries -= 1

time.sleep(0.5)

@app.route('/')

def hello():

count = get\_hit\_count()

return 'Hello World! I have been seen {} times.\n'.format(count)

if \_\_name\_\_ == "\_\_main\_\_":

app.run(host="0.0.0.0", debug=True)

在同目录下创建requirements.txt文件，添加项目依赖python包

Flask

Redis

在同样目录下，创建一个Dockerfile文件

FROM python:3.4-alpine

ADD . /code

WORKDIR /code

RUN pip install -r requirements.txt

CMD ["python", "app.py"]

在当前目录下创建一个docker-compose.yml

version: '2'

services:

web:

build: .

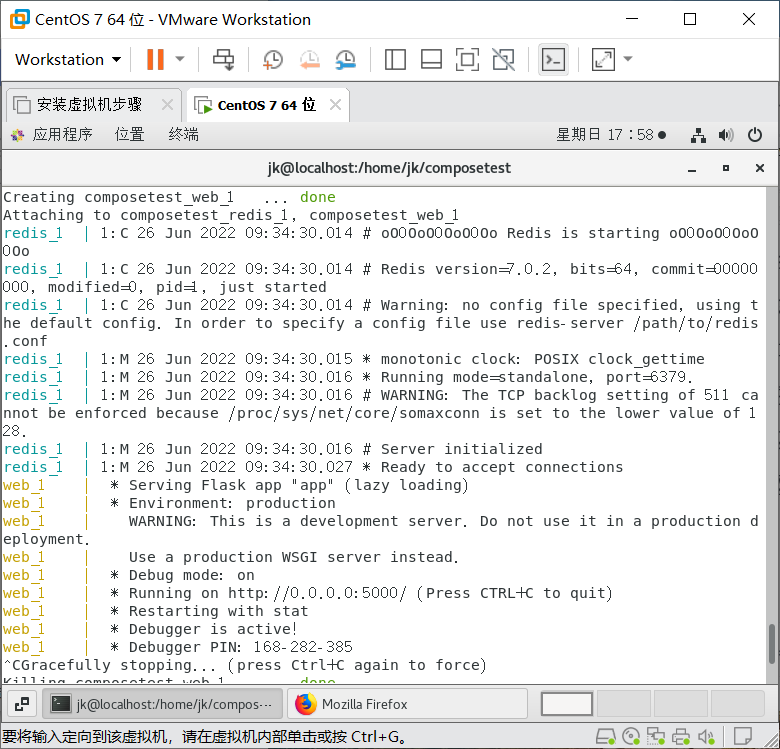
ports:

- "5000:5000"

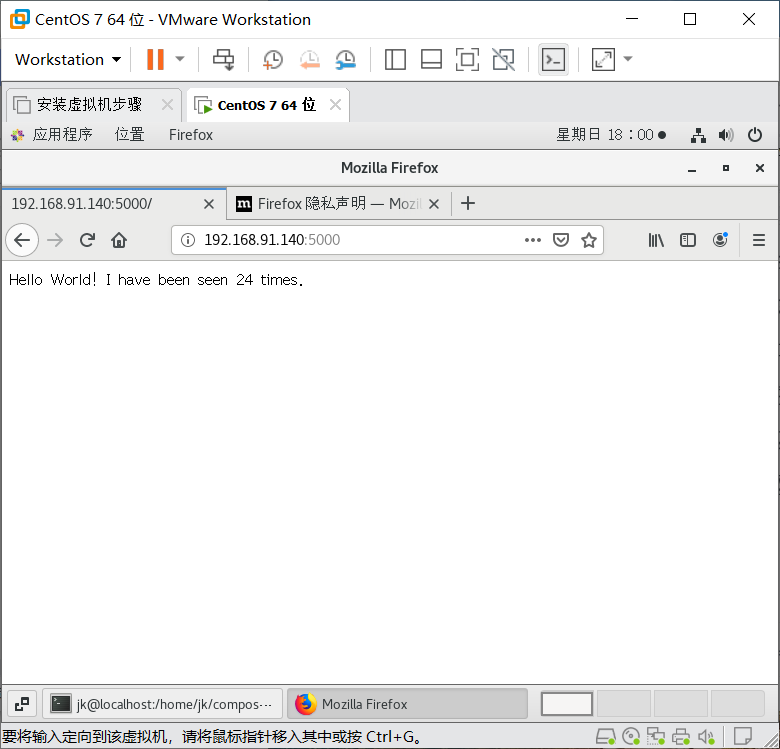
redis:

image: "redis:alpine

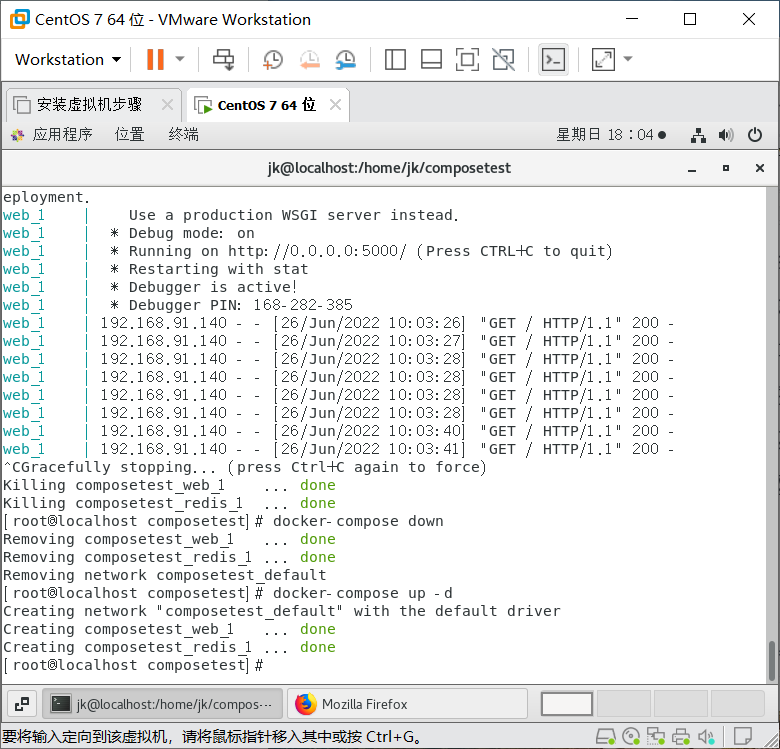
使用命令docker-comopose up启动



启动成功后在浏览器访问 http：//192.168.91.140:5000/



Docker-compose up -d 在后台启动服务



Docker-compose ps 查看服务状态

