



22、docker-compose.yml文件，编写配置规则

1、官网此文件的内容规则

- <https://docs.docker.com/compose/compose-file/compose-file-v3/>
- build和image需要一个即可



version #版本

services #服务列表

redis: #服务名

build:#构建镜像时，去context指定位置去找dockerfile指定的自定义命名的文件

context: .

dockerfile: 自定义的dockerfile文件名（如果文件命名为Dockerfile，此参数不用写，只需用context来指定名字为Dockerfile的文件的路径即可。其实就是docker-compose.yml文件和Dockerfile文件同级）

cache_from: alpine:latest

network: host

shm_size: '2gb'

target: prod

args: (dockerfile文件中需要配置ARG命令)

buildno: 1

gitcommithash: cdc3b19

configs:

my_config:

file: ./my_config.txt

my_other_config:

external: true

secrets:

my_first_secret:

file: ./secret_data

my_second_secret:

external: true

```
    external: true
    name: redis_secret
deploy:
  mode: replicated
  replicas: 1
  placement:
    max_replicas_per_node: 1
  update_config:
    parallelism: 2
    delay: 10s
  restart_policy:
    condition: on-failure
  resources:
    limits:
      cpus: '0.50'
      memory: 50M
    reservations:
      cpus: '0.25'
      memory: 20M
```

image: redis:latest #拉取的镜像

volumes:

- "/var/run/postgres/postgres.sock:/var/run/postgres/postgres.sock"
- "dbdata:/var/lib/postgresql/data"

volumes:

- type: volume
 - source: mydata
 - target: /data
 - volume:
 - nocopy: true
- type: bind
 - source: ./static
 - target: /opt/app/static



```
hostname: nct-wlhy-01
container_name: nct-wlhy-01
environment:
  - TZ=Asia/Shanghai
ports:
  - 8281:8281
privileged: true
entrypoint:
  - "java"
  - "-Djava.security.egd=file:/dev/./urandom"
  - "-Dblade.log.elk.destination=10.0.0.37:9000"
  - "-jar"
  - "app.jar"
entrypoint: /code/entrypoint.sh
entrypoint: ["php", "-d", "memory_limit=-1", "vendor/bin/phpunit"]
restart: always
command:
  - --spring.profiles.active=${PROFILE}
  - --Dblade.log.elk.destination=127.0.0.1:9000
  - --spring.cloud.nacos.config.server-addr=${NACOS_SERVER_ADDR}
  - --spring.cloud.nacos.discovery.server-addr=${NACOS_SERVER_ADDR}
  - --spring.cloud.sentinel.transport.dashboard=${SENTINEL_DASHBOARD_ADDR}
networks:
  app_net: #自定义网络名，使用的driver模式是默认的bridge
    ipv4_address: 172.16.238.99
    ipv6_address: 2001:3984:3989::10
    ipam:
      driver: default
      config:
        - subnet: "172.16.238.0/24"
        - subnet: "2001:3984:3989::/64"
```



network_mode: "bridge"

depends_on: #依赖关系，保证启动顺序必须先是被依赖的db数据库容器，然后才是redis容器

- db

links:

- db #依赖的服务名

db: #服务名

image: postgres

#全局的环境配置

volumes:

dbdata:

