

服务端开发环境配置

1. docker基本指令

创建新容器	<code>docker run -d --name 容器名称 镜像名称</code>
列出所有的镜像	<code>docker image</code>
列出所有的容器	<code>docker ps -a</code>
查看容器信息	<code>docker inspect 容器ID</code>
进入运行中的容器	<code>docker exec -t -i 容器ID bash</code>
停止容器	<code>docker stop 容器ID</code>
启动容器	<code>docker start 容器ID</code>
删除容器	<code>docker rm 容器ID</code>
删除镜像	<code>docker mi 镜像ID</code>
删除全部停止容器	<code>docker rm \$(docker ps -a -q)</code>
删除全部镜像	<code>docker rmi \$(docker images -q)</code>
提交修改过的容器	<code>docker commit 容器ID 镜像名称</code>
提交修改过的镜像	<code>docker push 镜像名</code>
镜像重命名	<code>docker tag 镜像ID 镜像名</code>
获取镜像	<code>docker pull 镜像名</code>

2.启动服务

运行一下命令依次安装mongo、redis。

```
docker run -d -p 27017:27017 --name mongoddb mongo --auth
```

```
docker run -d -p 6379:6379 --name redis redis
```

```
docker run -d -p 27017:27017 -v /data/mongo:/data/db --name mongoddb
```

```
192.168.221.45:5000/mongo --auth
```

```
docker run -d -p 6379:6379 --name redis 192.168.221.45:5000/redis
```

```
docker run -ti --name node -p 3010:3010 -p 3100:3100 -p 3101:3101 -p 4001:4001  
-p 31017:31017 -p 31027:31027 -p 31037:31037 -p 31047:31047 -v /work:/work --  
link mongoddb:mongoddb --link redis:redis 192.168.221.45:5000/node
```

3.数据库配置

进入mongo虚拟机

```
docker exec -t -i mongoddb mongo
```

进入mongo命令模式

运行

```
use admin
```

创建管理员

```
db.createUser( { user: "admin", pwd: "6Wq9Zl8tq7lgV85e", roles: [ { role:  
"userAdminAnyDatabase", db: "admin" } ] } )
```

使用管理员授权

```
db.auth('admin', '6Wq9Zl8tq7lgV85e')
```

创建Root用户

```
db.createUser( {user: "root",pwd: "K0HuO5gvbYTgDJaM",roles: [ { role: "root", db:
```

```
"admin" } ]});  
使用Root授权  
db.auth('root', 'K0HuO5gvbYTgDJaM')
```

```
创建库  
use Game  
创建库用户  
db.createUser({user:'Game',pwd:'ALbbhxeGO0nwkyLX',roles:  
[{'role':'readWrite',db:'Game'}]})
```

```
创建库  
use logs  
创建库用户  
db.createUser({user:'logs',pwd:'JEpxQs7pfGyXawUB',roles:  
[{'role':'readWrite',db:'logs'}]})
```

4.内存数据库配置

```
启动chess服务器  
docker run -ti --name chess -p 3010:3010 -p 3100:3100 -p 3101:3101 -p  
4001:4001 -v e:/work:/work --link mongodb:mongodb --link redis:redis  
node  
进入chess环境  
docker exec -ti chess bash  
安装内存数据库  
cd /work  
npm install -g git+http://gitlab.369.com/zhipeng/memdb.git  
启动内存数据库  
cd /work/memdb  
./bin/memdbcluster start
```

5.日志服务

```
在chess容器内  
cd /work  
git clone http://gitlab.369.com/zhipeng/pomelo-udp-logger.git  
启动服务  
cd /work/pomelo-udp-logger  
NODE_ENV=development node app
```

6.管理服务

```
在chess容器内  
cd /work  
git clone http://gitlab.369.com/zhipeng/pomelo-admin-web.git  
启动服务  
cd /work/pomelo-admin-web  
NODE_ENV=development node app
```

7.游戏逻辑服务

启动

```
docker run -ti --name chess -p 3010:3010 -p 3100:3100 -p 3101:3101 -p  
4001:4001 -p 3001:3001 -p 3002:3002 -p 5000:5000 -p 5870:5870 -p  
5871:5871 -p 5100:5100 -p 5101:5101 -p 5860:5860 -p 5880:5880 -p  
5881:5881 -p 5890:5890 -p 5891:5891 -p 5900:5900 -p 5901:5901 -p  
5910:5910 -p 5911:5911 -p 13100:13100 -p 13101:13101 -p 22:22 -p  
6010:6010 -p 6011:6011 -v e:/work:/work 192.168.221.45:5000/node
```

在chess容器内

```
cd /work
```

```
git clone http://gitlab.369.com/zhipepeng/chess-server.git
```

启动服务

```
cd /work/chess-server
```

```
NODE_ENV=development node app
```