1. Docker部署Mysql
2. 需求分析

在Docker容器中部署Mysql，并通过外部的Mysql客户端操作Mysql Server。

1. 端口映射

由于外部机器与容器不能直接通信，而外部机器与宿主机器是可以通信的，因此可以通过将容器中提供服务的端口映射到宿主机的端口上，当外部机器访问宿主机的该端口时，就能间接地访问容器的服务。这种操作称为端口映射。

1. 部署步骤
2. 搜索mysql镜像

docker search mysql

1. 拉取mysql镜像

docker pull mysql

1. 创建容器，设置端口映射，目录映射

首先创建宿主机的mysql目录：/root/mysql，进入mysql目录。

mkdir /root/mysql

cd /root/mysql

然后创建容器，-p参数表示设置端口映射，前面是宿主机的端口，后面是容器的端口， -v是映射目录，前面是宿主机数据卷，后面是容器目录；-e是设置容器环境，MYSQL\_ROOT\_PASSWORD初始化root用户的密码。

docker run -id -p 3307:3306 --name=c\_mysql -v $PWD/conf:/etc/mysql/conf.d -v $PWD/logs:/logs -v $PWD/data:/var/lib/mysql -e MYSQL\_ROOT\_PASSWORD=123456 mysql

1. 进入容器，操作Mysql

docker exec -it c\_mysql /bin/bash

1. 登录mysql

mysql -uroot -p123456

1. 在本地机器中连接宿主机3307端口

连接宿主机的3307端口，就映射到了容器的3306端口，那么就是直接操作容器的mysql了，这样，就实现了外部机器与容器之间的通信。

1. Docker部署Tomcat
2. 搜索Tomcat镜像

docker search tomcat

1. 拉取镜像

docker pull tomcat

1. 创建容器，设置端口映射，目录映射

mkdir /root/tomcat

cd /root/tomcat

docker run -id --name=c\_tomcat -p 8080:8080 -v $PWD:/usr/local/tomcat/webapps tomcat

1. Docker 部署Redis
2. 搜索Redis镜像

docker search redis

1. 拉取Redis镜像

docker pull redis

1. 创建容器，设置端口映射

docker run -id --name=c\_redis -p 6379:6379 redis

1. 使用外部机器连接Redis

在本地机器中进入Redis目录中，使用如下命令连接宿主机的端口。

redis-cli.ext -h 192.168.000.000 -p 6379