实验一

[root@localhost ~]# vim Dockerfile

FROM centos:9

RUN dnf install iproute -y

[root@localhost ~]# docker build -t gyx ./

[root@localhost ~]# vim Dockerfile

FROM gyx

ADD nginx-1.24.0.tar.gz /

WORKDIR nginx-1.24.0

RUN dnf install gcc zlib-devel make pcre-devel openssl openssl-devel -y

RUN ./configure --prefix=/usr/local/ --user=nginx --group=nginx

RUN make && make install

RUN useradd nginx

RUN echo "gaoyuxin" > /usr/local/html/index.html

CMD ["nginx","-g","daemon off;"] [root@localhost ~]# docker build -t nginx0621:v1.0 . --no-cache

[root@localhost ~]# docker run -d --name nginx nginx0710:v1.0

[root@localhost ~]# nginx

docker run -d --name web1 nginx0710:v1.0

docker run -d --name web2 nginx0710:v1.0

docker run -d --name web3 nginx0710:v1.0

实验二

[root@localhost ~]# vim Dockerfile

FROM rockylinux:9

COPY mysql-8.0.38-linux-glibc2.28-x86\_64.tar.xz /root

RUN dnf -y install xz perl-JSON perl-Data-Dumper libaio numactl

RUN tar -xf /root/mysql-8.0.38-linux-glibc2.28-x86\_64.tar.xz -C /usr/local/

RUN mv /usr/local/mysql-8.0.38-linux-glibc2.28-x86\_64 /usr/local/mysql

RUN useradd mysql

RUN mkdir /var/lib/mysql \

&& chown -R mysql:mysql /var/lib/mysql

RUN mkdir /var/run/mysqld \

&& chown -R mysql:mysql /var/run/mysqld

RUN mkdir /var/log/mysql \

&& chown -R mysql:mysql /var/log/mysql

RUN ln -s /usr/local/mysql/bin/\* /usr/local/bin/

COPY start-mysql.sh /usr/local/bin/start-mysql.sh

RUN chmod +x /usr/local/bin/start-mysql.sh

EXPOSE 3306

ENTRYPOINT ["/usr/local/bin/start-mysql.sh"]

[root@localhost ~]# vim start-mysql.sh

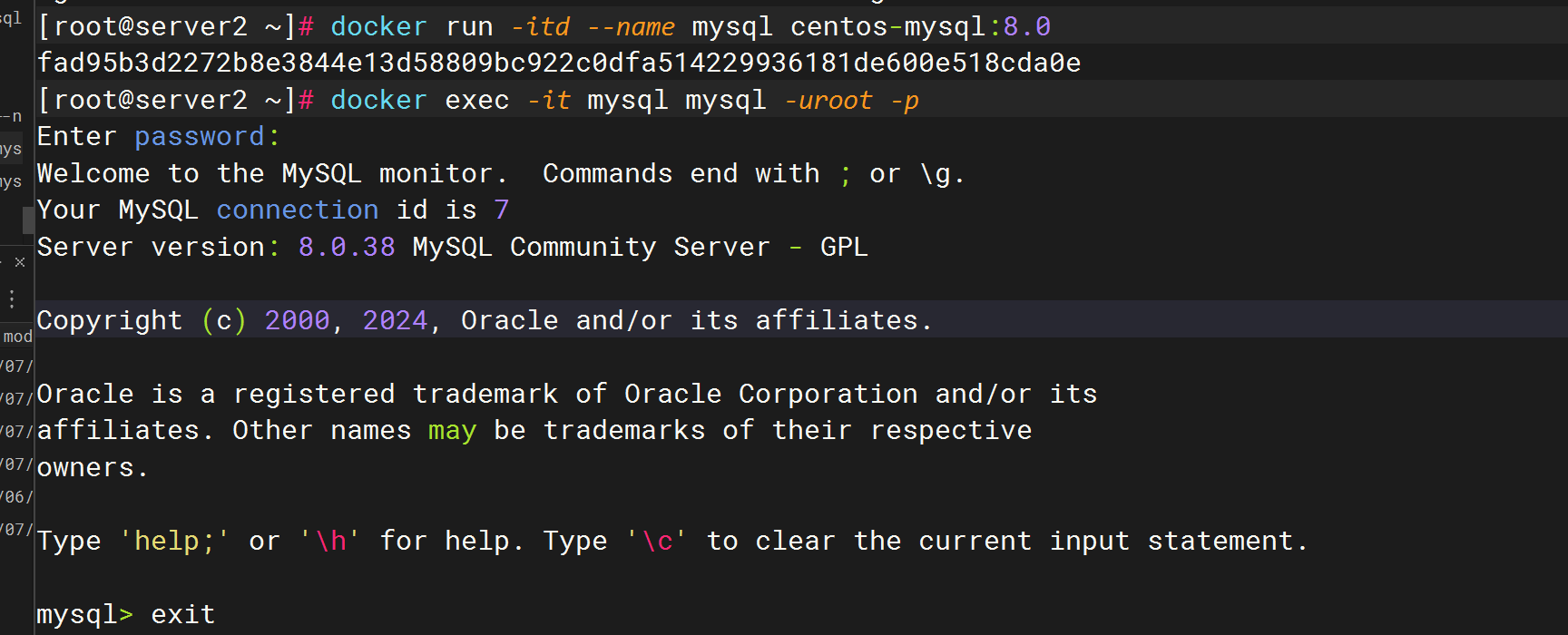
#!/bin/bash

if [ ! -d "/var/lib/mysql/mysql" ]; then

/usr/local/mysql/bin/mysqld --initialize-insecure --user=mysql

fi

exec /usr/local/mysql/bin/mysqld --user=mysql --skip-grant-tables



图片包含 图形用户界面

描述已自动生成

文本

中度可信度描述已自动生成