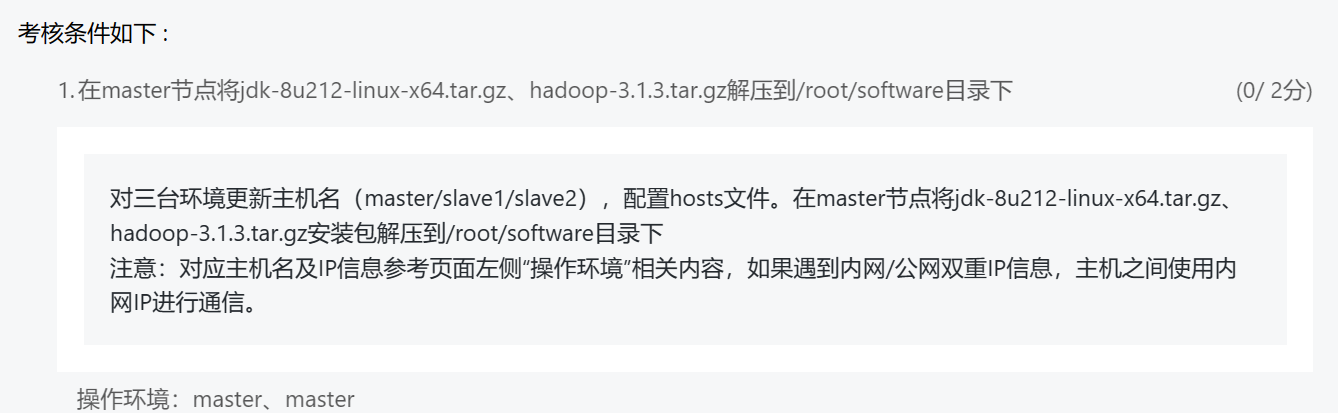




**# 一、模块1：平台搭建与运维**



本搭建模块对应虚拟机为：master、slave1、slave2

相关压缩包路径为：`/root/software/package`,对应安装路径为`/root/software/`

**## （一）任务1：大数据平台搭建**

**### 1.子任务1-1：Hadoop完全分布式安装配置**

（1） 在master节点将jdk-8u212-linux-x64.tar.gz、hadoop-3.1.3.tar.gz解压到/root/software目录下；

（2） 在master上生成SSH密钥对，实现三台机器间的免密登录。并同步jdk，配置环境变量并生效；

（3） 将hadoop分发至slave1、slave2中，其中三个节点节点均作为datanode，配置好相关环境，初始化Hadoop环境namenode；

（4）开启集群，查看各节点进程。

```shell

# 修改主机名 三台主机

hostnamectl set-hostname master && bash

hostnamectl set-hostname slave1 && bash

hostnamectl set-hostname slave2 && bash

# 修改 hosts 文件，添加主机IP映射（注意修改IP为实际内网IP！！！）

echo "

172.18.10.13 master

172.18.39.65 slave1

172.18.21.20 slave2" >> /etc/hosts

# 安装JDK、Hadoop

# （安装包所在路径：/root/software/）解压安装到指定路径（解压目标目录：/root/software）

tar -xzvf /root/software/package/jdk-8u212-linux-x64.tar.gz -C /root/software/

tar -zxvf /root/software/package/hadoop-3.1.3.tar.gz -C /root/software

# master 生成 rsa 密钥

ssh-keygen

# 实现 master 自身免密

ssh-copy-id -i /root/.ssh/id\_rsa.pub master

# 实现 master 到 slave1 免密

ssh-copy-id -i /root/.ssh/id\_rsa.pub slave1

# 实现 master 到 slave2 免密

ssh-copy-id -i /root/.ssh/id\_rsa.pub slave2

# 验证是否免密登录成功

ssh slave1

ssh slave2

#分发jdk

scp -r /root/software/jdk1.8.0\_212 root@slave1:/root/software/

scp -r /root/software/jdk1.8.0\_212 root@slave2:/root/software/

# 配置 JAVA\_HOME 环境变量

vim /etc/profile

# 添加以下内容：

export JAVA\_HOME=/root/software/jdk1.8.0\_212

export PATH=$PATH:$JAVA\_HOME/bin

# 分发环境变量配置文件（/etc/profile）

scp -r /etc/profile root@slave1:/etc/profile

scp -r /etc/profile root@slave2:/etc/profile

# 在 master、slave1、slave2 节点分别使环境变量生效

source /etc/profile

# 查看版本信息

java -version

# 查看 javac 命令

javac

# 2. 配置hadoop env

echo "export JAVA\_HOME=/root/software/jdk1.8.0\_212" >> /root/software/hadoop-3.1.3/etc/hadoop/hadoop-env.sh

```

（2）配置相关文件

1.修改 core-site.xml 配置文件

```xml

# vim /root/software/hadoop-3.1.3/etc/hadoop/core-site.xml

<configuration>

<property>

 <name>fs.defaultFS</name>

 <value>hdfs://master:9000</value>

</property>

<property>

 <name>hadoop.tmp.dir</name>

 <value>/root/software/hadoop-3.1.3/data</value>

</property>

<property>

 <name>hadoop.security.authorization</name>

 <value>true</value>

</property>

</configuration>

```

2.修改 hdfs-site.xml 配置文件

```xml

# vim /root/software/hadoop-3.1.3/etc/hadoop/hdfs-site.xml

<configuration>

<property>

 <name>dfs.namenode.http-address</name>

 <value>master:9870</value>

</property>

<property>

 <name>dfs.replication</name>

 <value>3</value>

</property>

<property>

 <name>dfs.permissions.enabled</name>

 <value>true</value>

</property>

<property>

 <name>dfs.permissions.superusergroup</name>

 <value>root</value>

</property>

</configuration>

```

3.修改 mapred-site.xml 配置文件

```xml

# vim /root/software/hadoop-3.1.3/etc/hadoop/mapred-site.xml

<configuration>

<property>

 <name>mapreduce.framework.name</name>

 <value>yarn</value>

</property>

<property>

 <name>yarn.app.mapreduce.am.env</name>

 <value>HADOOP\_MAPRED\_HOME=/root/software/hadoop-3.1.3</value>

</property>

<property>

 <name>mapreduce.map.env</name>

 <value>HADOOP\_MAPRED\_HOME=/root/software/hadoop-3.1.3</value>

</property>

<property>

 <name>mapreduce.reduce.env</name>

 <value>HADOOP\_MAPRED\_HOME=/root/software/hadoop-3.1.3</value>

</property>

</configuration>

```

4.修改 yarn-site.xml 配置文件

提示：通过配置 **\*\*yarn-site.xml\*\*** 禁用 Yarn 提供的 Web 服务接口和绑定本机地址，实现只允许内网可访问。同时禁用资源管理器的 RPC 调用地址防止 RPC 的未授权访问漏洞。

```xml

# vim /root/software/hadoop-3.1.3/etc/hadoop/yarn-site.xml

<configuration>

<property>

 <name>yarn.nodemanager.aux-services</name>

 <value>mapreduce\_shuffle</value>

</property>

<property>

 <name>yarn.resourcemanager.hostname</name>

 <value>master</value>

</property>

<property>

 <name>yarn.nodemanager.env-whitelist</name>

 <value>JAVA\_HOME,HADOOP\_COMMON\_HOME,HADOOP\_HDFS\_HOME,HADOOP\_CONF\_DIR,CLASSPATH\_PREPEND\_DISTCACHE,HADOOP\_YARN\_HOME,HADOOP\_MAPRED\_HOME</value>

</property>

</configuration>

```

5.修改 workers 配置文件(直接替换)

```sh

# vim /root/software/hadoop-3.1.3/etc/hadoop/workers

echo "master

slave1

slave2" > /root/software/hadoop-3.1.3/etc/hadoop/workers

```

6.修改 hadoop-policy.xml 服务认证配置文件（授予只有root用户才可操作的权限，直接替换整个文件！！！）

```xml

# vim /root/software/hadoop-3.1.3/etc/hadoop/hadoop-policy.xml

<?xml version="1.0"?>

<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<!--

 Licensed to the Apache Software Foundation (ASF) under one

 or more contributor license agreements.  See the NOTICE file

 distributed with this work for additional information

 regarding copyright ownership.  The ASF licenses this file

 to you under the Apache License, Version 2.0 (the

 "License"); you may not use this file except in compliance

 with the License.  You may obtain a copy of the License at

     http://www.apache.org/licenses/LICENSE-2.0

 Unless required by applicable law or agreed to in writing, software

 distributed under the License is distributed on an "AS IS" BASIS,

 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

 See the License for the specific language governing permissions and

 limitations under the License.

-->

<!-- Put site-specific property overrides in this file. -->

<configuration>

  <property>

    <name>security.client.protocol.acl</name>

    <value>root</value>

    <description>ACL for ClientProtocol, which is used by user code

    via the DistributedFileSystem.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.client.datanode.protocol.acl</name>

    <value>root</value>

    <description>ACL for ClientDatanodeProtocol, the client-to-datanode protocol

    for block recovery.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.datanode.protocol.acl</name>

    <value>root</value>

    <description>ACL for DatanodeProtocol, which is used by datanodes to

    communicate with the namenode.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.inter.datanode.protocol.acl</name>

    <value>root</value>

    <description>ACL for InterDatanodeProtocol, the inter-datanode protocol

    for updating generation timestamp.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.namenode.protocol.acl</name>

    <value>\*</value>

    <description>ACL for NamenodeProtocol, the protocol used by the secondary

    namenode to communicate with the namenode.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

 <property>

    <name>security.admin.operations.protocol.acl</name>

    <value>\*</value>

    <description>ACL for AdminOperationsProtocol. Used for admin commands.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.refresh.user.mappings.protocol.acl</name>

    <value>\*</value>

    <description>ACL for RefreshUserMappingsProtocol. Used to refresh

    users mappings. The ACL is a comma-separated list of user and

    group names. The user and group list is separated by a blank. For

    e.g. "alice,bob users,wheel".  A special value of "\*" means all

    users are allowed.</description>

  </property>

  <property>

    <name>security.refresh.policy.protocol.acl</name>

    <value>\*</value>

    <description>ACL for RefreshAuthorizationPolicyProtocol, used by the

    dfsadmin and mradmin commands to refresh the security policy in-effect.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.ha.service.protocol.acl</name>

    <value>\*</value>

    <description>ACL for HAService protocol used by HAAdmin to manage the

      active and stand-by states of namenode.</description>

  </property>

  <property>

    <name>security.zkfc.protocol.acl</name>

    <value>\*</value>

    <description>ACL for access to the ZK Failover Controller

    </description>

  </property>

  <property>

    <name>security.qjournal.service.protocol.acl</name>

    <value>\*</value>

    <description>ACL for QJournalProtocol, used by the NN to communicate with

    JNs when using the QuorumJournalManager for edit logs.</description>

  </property>

  <property>

    <name>security.interqjournal.service.protocol.acl</name>

    <value>\*</value>

    <description>ACL for InterQJournalProtocol, used by the JN to

    communicate with other JN

    </description>

  </property>

  <property>

    <name>security.mrhs.client.protocol.acl</name>

    <value>\*</value>

    <description>ACL for HSClientProtocol, used by job clients to

    communciate with the MR History Server job status etc.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <!-- YARN Protocols -->

  <property>

    <name>security.resourcetracker.protocol.acl</name>

    <value>\*</value>

    <description>ACL for ResourceTrackerProtocol, used by the

    ResourceManager and NodeManager to communicate with each other.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.resourcemanager-administration.protocol.acl</name>

    <value>\*</value>

    <description>ACL for ResourceManagerAdministrationProtocol, for admin commands.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.applicationclient.protocol.acl</name>

    <value>root</value>

    <description>ACL for ApplicationClientProtocol, used by the ResourceManager

    and applications submission clients to communicate with each other.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.applicationmaster.protocol.acl</name>

    <value>\*</value>

    <description>ACL for ApplicationMasterProtocol, used by the ResourceManager

    and ApplicationMasters to communicate with each other.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.containermanagement.protocol.acl</name>

    <value>\*</value>

    <description>ACL for ContainerManagementProtocol protocol, used by the NodeManager

    and ApplicationMasters to communicate with each other.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.resourcelocalizer.protocol.acl</name>

    <value>\*</value>

    <description>ACL for ResourceLocalizer protocol, used by the NodeManager

    and ResourceLocalizer to communicate with each other.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.job.task.protocol.acl</name>

    <value>\*</value>

    <description>ACL for TaskUmbilicalProtocol, used by the map and reduce

    tasks to communicate with the parent tasktracker.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.job.client.protocol.acl</name>

    <value>\*</value>

    <description>ACL for MRClientProtocol, used by job clients to

    communciate with the MR ApplicationMaster to query job status etc.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.applicationhistory.protocol.acl</name>

    <value>\*</value>

    <description>ACL for ApplicationHistoryProtocol, used by the timeline

    server and the generic history service client to communicate with each other.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.collector-nodemanager.protocol.acl</name>

    <value>\*</value>

    <description>ACL for CollectorNodemanagerProtocol, used by nodemanager

    if timeline service v2 is enabled, for the timeline collector and nodemanager

    to communicate with each other.

    The ACL is a comma-separated list of user and group names. The user and

    group list is separated by a blank. For e.g. "alice,bob users,wheel".

    A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.applicationmaster-nodemanager.applicationmaster.protocol.acl</name>

    <value>\*</value>

    <description>ACL for ApplicationMasterProtocol, used by the Nodemanager

        and ApplicationMasters to communicate.

        The ACL is a comma-separated list of user and group names. The user and

        group list is separated by a blank. For e.g. "alice,bob users,wheel".

        A special value of "\*" means all users are allowed.</description>

  </property>

  <property>

    <name>security.distributedscheduling.protocol.acl</name>

    <value>\*</value>

    <description>ACL for DistributedSchedulingAMProtocol, used by the Nodemanager

        and Resourcemanager to communicate.

        The ACL is a comma-separated list of user and group names. The user and

        group list is separated by a blank. For e.g. "alice,bob users,wheel".

        A special value of "\*" means all users are allowed.</description>

    </property>

</configuration>

```

（3）分发hadoop

```sh

scp -r /root/software/hadoop-3.1.3 root@slave1:/root/software/

scp -r /root/software/hadoop-3.1.3 root@slave2:/root/software/

```

（4）配置 hadoop 环境变量

```sh

# 编辑

vim /etc/profile

# 文件底部追加以下内容：

export HADOOP\_HOME=/root/software/hadoop-3.1.3

export PATH=$PATH:$HADOOP\_HOME/bin:$HADOOP\_HOME/sbin

export HADOOP\_CONF\_DIR=$HADOOP\_HOME/etc/hadoop

export HADOOP\_CLASSPATH=$(hadoop classpath)

export HDFS\_NAMENODE\_USER=root

export HDFS\_DATANODE\_USER=root

export HDFS\_SECONDARYNAMENODE\_USER=root

export YARN\_RESOURCEMANAGER\_USER=root

export YARN\_NODEMANAGER\_USER=root

export HDFS\_JOURNALNODE\_USER=root

export HDFS\_ZKFC\_USER=root

```

（5）分发环境变量配置文件

```sh

scp /etc/profile root@slave1:/etc/profile

scp /etc/profile root@slave2:/etc/profile

```

（6）使环境变量生效（**\*\*三台节点均需执行\*\***）

```sh

source /etc/profile

```

（7）格式化文件系统（在master节点执行）

```sh

hdfs namenode -format

```

（8）启动集群（在master节点执行）

```sh

# 开启集群

start-all.sh

# 启动 hdfs

# start-dfs.sh

# 启动 yarn

# start-yarn.sh

# 启动历史服务器

# mr-jobhistory-daemon.sh start historyserver

```



**### 2.子任务1-2：MySQL安装配置**

**\*\*注意：要安装 mysql 之前需要先卸载掉 mariadb-libs（命令：yum remove -y mariadb-libs）\*\***

（1）解压MySQL 5.7.25到/root/software目录下，并安装MySQL组件。使用mysql用户初始化和启动数据库；

（2）无密码登录MySQL，并修改root用户的密码为123456，验证登录；

（3）更改“mysql”数据库里的user表里的host项，实现用户远程登录；设置完成刷新配置信息，让其生效。

```sh

# 解压数据库

tar -xvf /root/software/package/mysql-5.7.25-1.el7.x86\_64.rpm-bundle.tar -C /root/software/

# 删除依赖库

yum remove -y mariadb-libs

# 安装mysql各个组件

cd /root/software/

rpm -ivh mysql-community-common-5.7.25-1.el7.x86\_64.rpm

rpm -ivh mysql-community-libs-5.7.25-1.el7.x86\_64.rpm

rpm -ivh mysql-community-libs-compat-5.7.25-1.el7.x86\_64.rpm

rpm -ivh mysql-community-client-5.7.25-1.el7.x86\_64.rpm

rpm -ivh mysql-community-server-5.7.25-1.el7.x86\_64.rpm

# 初始化

/usr/sbin/mysqld --initialize-insecure --console --user=mysql

# 启动服务

systemctl start mysqld.service

# 进入数据库（自行修改密码及权限，注意刷新权限）

mysql -uroot

# 修改用户密码

mysql> alter user 'root'@'localhost' identified by '123456';

Query OK, 0 rows affected (0.00 sec)

# 修改远程登录权限

mysql> update mysql.user set host='%' where host='localhost';

Query OK, 3 rows affected (0.00 sec)

Rows matched: 3  Changed: 3  Warnings: 0

# 刷新权限

mysql> flush privileges;

Query OK, 0 rows affected (0.00 sec)

# 退出

mysql> exit;

```

**### 3.子任务1-3：Hive安装配置**

（1） 将Hive3.1.2安装包解压到/root/software目录下；并配置其环境变量，让其立即生效，查看Hive版本；

（2） 修改相关配置，添加mysql-connector-java-5.1.47.jar依赖包，将MySQL数据库作为Hive元数据库，初始化Hive元数据；

```shell

# 解压安装包

tar -xzvf /root/software/package/apache-hive-3.1.2-bin.tar.gz -C /root/software/

# 设置环境变量

# 编辑

vim /etc/profile

# 文件底部追加以下内容：

export HIVE\_HOME=/root/software/apache-hive-3.1.2-bin

export PATH=$PATH:$HIVE\_HOME/bin

# 生效环境变量

source /etc/profile

# 查看hive版本

hive --version

# 配置相关文件，添加依赖包

# hive-env.sh 配置文件

cd /root/software/apache-hive-3.1.2-bin/conf

mv hive-env.sh.template hive-env.sh

# 相关配置直接追加到 hive-env.sh 文件

echo "

HADOOP\_HOME=/root/software/hadoop-3.1.3

export HIVE\_CONF\_DIR=/root/software/apache-hive-3.1.2-bin/conf

export HIVE\_AUX\_JARS\_PATH=/root/software/apache-hive-3.1.2-bin/lib" >> /root/software/apache-hive-3.1.2-bin/conf/hive-env.sh

# 配置 hive-site.xml 文件

vim /root/software/apache-hive-3.1.2-bin/conf/hive-site.xml

<?xml version="1.0"?>

<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<configuration>

<property>

 <name>javax.jdo.option.ConnectionURL</name>

 <value>jdbc:mysql://master:3306/hivedb?createDatabaseIfNotExist=true&amp;useSSL=false&amp;useUnicode=true&amp;characterEncoding=UTF-8</value>

</property>

<property>

 <name>javax.jdo.option.ConnectionDriverName</name>

 <value>com.mysql.jdbc.Driver</value>

</property>

<property>

 <name>javax.jdo.option.ConnectionUserName</name>

 <value>root</value>

</property>

<property>

 <name>javax.jdo.option.ConnectionPassword</name>

 <value>123456</value>

</property>

</configuration>

# 添加 mysql-connector 驱动

cp /root/software/package/mysql-connector-java-5.1.47-bin.jar  /root/software/apache-hive-3.1.2-bin/lib/

# 初始化元数据库

schematool -dbType mysql -initSchema

```



## 任务二：数据库配置维护

### 子任务一：创建数据库及相关数据表

**【具体任务】**

1. 在 MySQL 数据库中创建“test”数据库，并在“test”数据库中分别创建“shopping”、“fooditems”共 2 个数据表。各个数据表的表字段格式如下；

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 字段中文名 | 类型 | 备注 |
| id | 行号 | INT | 自增，主键 |
| city | 城市 | VARCHAR(255) |  |
| food\_name | 美食名称 | VARCHAR(255) |  |
| likelihood\_of\_liking | 喜爱度 | INT |  |
| restaurant\_list | 餐馆列表 | TEXT |  |
| food\_detail\_link | 美食详情链接 | TEXT |  |
| food\_image\_link | 美食图片链接 | TEXT |  |
| food\_description | 美食介绍 | TEXT |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 字段中文名 | 类型 | 备注 |
| id | 行号 | INT | 自增，主键 |
| city | 城市 | VARCHAR(255) |  |
| shop\_name | 购物地名称 | VARCHAR(500) |  |
| address | 地址 | VARCHAR(50) |  |
| contact\_phone | 联系电话 | VARCHAR(100) |  |
| business\_hours | 营业时间 | VARCHAR(100) |  |
| ranking | 排名 | VARCHAR(100) |  |
| overall\_rating | 综合评分 | VARCHAR(50) |  |
| reviews\_count | 点评数 | VARCHAR(50) |  |
| review\_category | 评价类别 | VARCHAR(100) |  |
| visitor\_rating | 游客评分 | VARCHAR(100) |  |
| visitor\_review | 游客评价 | TEXT |  |

systemctl start mysqld.service

# 进入数据库（自行修改密码及权限，注意刷新权限）

mysql -uroot -p123456