# 连数据库

**1.安装MySQL，pymysql**

在 Ubuntu 18.04 中，默认情况下，只有最新版本的 MySQL 包含在 APT 软件包存储库中,要安装它，只需更新服务器上的包索引并安装默认包 apt-get。

|  |
| --- |
| #命令1  sudo apt-get update  #命令2  sudo apt-get install mysql-server  #命令3  pip install pymysql |

**2.配置MySQL**

**2.1 安装配置**

|  |
| --- |
| sudo mysql\_secure\_installation |

配置项较多，如下所示：

|  |
| --- |
| #1  VALIDATE PASSWORD PLUGIN can be used to test passwords...  Press y|Y for Yes, any other key for No: N (我的选项)  #2  Please set the password for root here...  New password: (输入密码)  Re-enter new password: (重复输入)  #3  By default, a MySQL installation has an anonymous user,  allowing anyone to log into MySQL without having to have  a user account created for them...  Remove anonymous users? (Press y|Y for Yes, any other key for No) : N (我的选项)  #4  Normally, root should only be allowed to connect from  'localhost'. This ensures that someone cannot guess at  the root password from the network...  Disallow root login remotely? (Press y|Y for Yes, any other key for No) : Y (我的选项)  #5  By default, MySQL comes with a database named 'test' that  anyone can access...  Remove test database and access to it? (Press y|Y for Yes, any other key for No) : N (我的选项)  #6  Reloading the privilege tables will ensure that all changes  made so far will take effect immediately.  Reload privilege tables now? (Press y|Y for Yes, any other key for No) : Y (我的选项) |

**2.2 检查mysql服务状态：**

|  |
| --- |
| systemctl status mysql.service |

显示如下结果说明mysql服务是正常的：

|  |
| --- |
| ● mysql.service - MySQL Community Server  Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: en  Active: active (running) since Sat 2018-06-23 11:09:03 CST; 6min ago  Main PID: 14078 (mysqld)  Tasks: 29 (limit: 4915)  CGroup: /system.slice/mysql.service  └─14078 /usr/sbin/mysqld --daemonize --pid-file=/run/mysqld/mysqld.pi  6月 23 11:09:02 weision systemd[1]: Starting MySQL Community Server...  6月 23 11:09:03 weision systemd[1]: Started MySQL Community Server. |

进入mysql的方法：命令行输入

mysql -u root -p

*注：mysql安装以后输入mysql -u root -p无法登陆的的解决方法：*

* 1.sudo mysql -u root -p*

* 2.select user, plugin from mysql.user;*

*3.update mysql.user set authentication\_string=PASSWORD('123456'), plugin='mysql\_native\_password' where user='root';*

* 4.flush privileges;*

* 5.exit*

* 6.重启进入*

**3.创建数据库：**

|  |
| --- |
| create database songDB;  use songDB;  create table song (name char(100), singer char(100), style char(100)); |

# 读取csv文件存到数据库中

1. 打开数据库连接

db = pymysql.connect("localhost","root","123456","songDB",charset='utf8')

1. 使用 cursor() 方法创建一个游标对象 cursor

cursor = db.cursor()

1. 打开csv文件

with open("playlist\_style1.csv",'r',encoding="utf-8") as file:

1. 读取每一行，每一行再通过“,”分隔符将name,singer,style分别提取出来

for line in file:

line1=line.split('\n')

str = line1[0].split(',')

1. sql插入语句

sql = "insert into song(name,singer,style) values (\'{}\',\'{}\',\'{}\')".format(str[0],str[1],str[2])

1. 执行sql语句

try:

# 执行sql语句

cursor.execute(sql)

# 提交到数据库执行

db.commit()

except:

# 如果发生错误则回滚

db.rollback()

1. 关闭数据库连接

db.close()

# bot.py查库

获取到相应的槽值name,singer,style后，调用ActionSearchConsume类, 根据槽值的填充情况进行数据库查询，基本查询方法如下：

1. 打开数据库连接

db = pymysql.connect("localhost","root","123456","songDB",charset='utf8')

1. 使用 cursor() 方法创建一个游标对象 cursor

cursor = db.cursor()

1. 根据不同情况写sql查询语句，例如：

sql = "select \* from song where name = \'{}\' ".format(item)

1. 执行sql语句

try:

# 执行sql语句

cursor.execute(sql)

# 提交到数据库执行

db.commit()

except:

# 如果发生错误则回滚

db.rollback()

1. 关闭数据库连接

db.close()

1. 利用dispatcher.utter\_message()回复用户