

# 目录结构和部署文档

---

开发环境

语言: `python 3.7.4`

python包管理工具: `pip 20.1`

后端: `django 3.0.6`

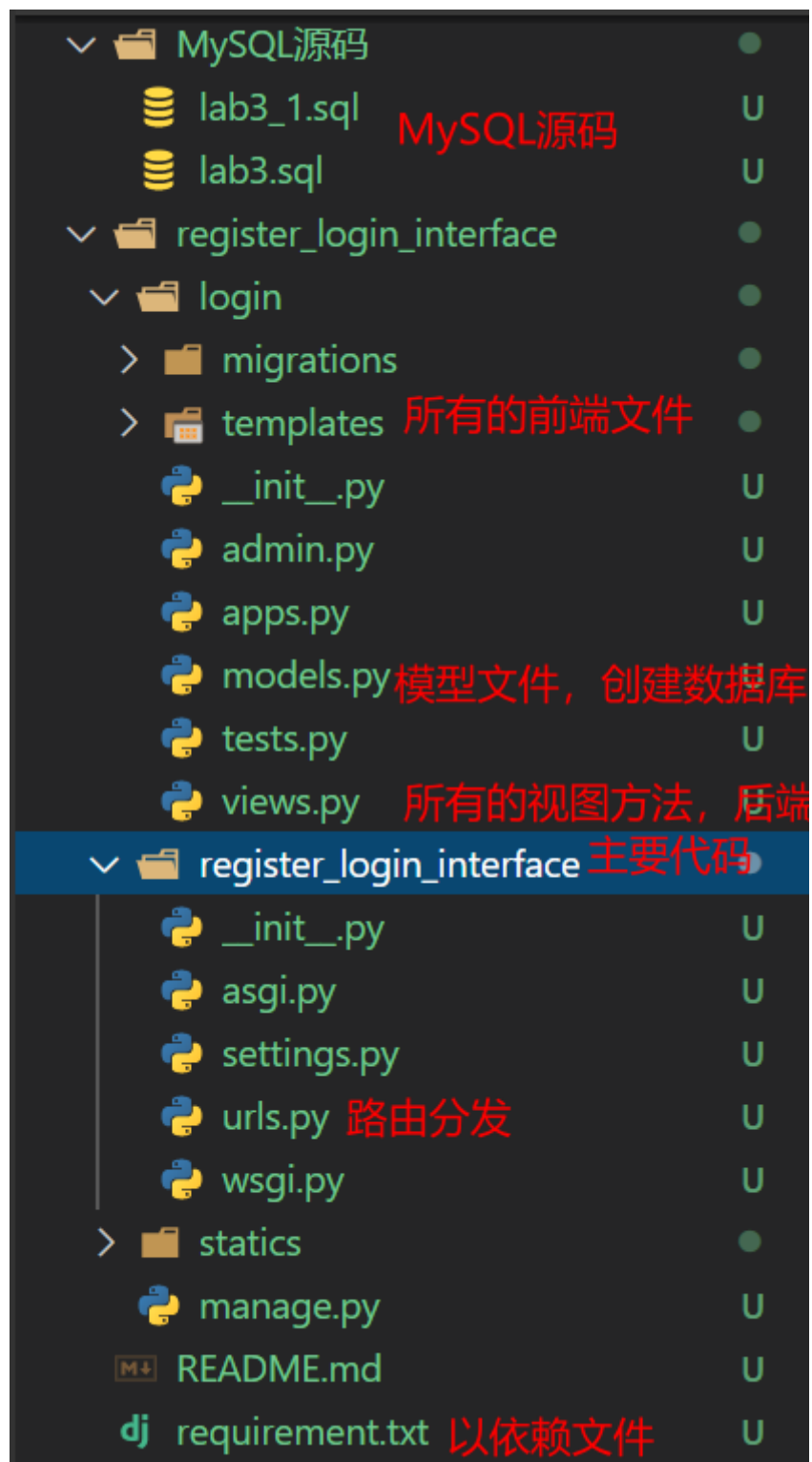
前端: `layui echarts.js`

数据库: `MySQL`

操作系统: `windows 10`

## 目录结构

---



## 1.创建虚拟环境

1. 若环境中没有virtualenv,首先安装virtualenv,若有则跳过此步

```
pip install virtualenv
```

2. 创建虚拟环境, 安装依赖

windows下

```
virtualenv venv  
venv\Scripts\activate  
pip install -r requirements.txt
```

linux下

```
python3.7 -m venv venv
source venv/bin/activate
pip install -r requirements.txt
```

## 2.开启数据库

- 主机安装了MySQL的情况下  
进入root用户，也可以选择其它用户

```
mysql -uroot -p
```

创建数据库

```
create database bankmanagementsystem;
```

- 也可以使用docker创建MySQL数据库

```
docker run --name mariadb -e MYSQL_ROOT_PASSWORD=1234 -d mariadb
docker inspect -f "{{ .NetworkSettings.IPAddress }}" mariadb //查看数据库地址，需要在settings.py中设置相应HOST
docker exec -it mariadb /bin/bash //进入数据库终端
mysql -uroot -p //进入root用户数据库，确定设定的密码可以进入数据库，如果设置的密码不正确，可能为空，之后修改密码
create database bankmanagementsystem default character set utf8 collate utf8_unicode_ci;
```

使用了不同的用户和数据库，需要在settings.py中对数据库和用户进行修改

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'bankmanagementsystem', 数据库名
        'USER': 'root', 用户名
        'PASSWORD': '1234', 用户密码
        'HOST': 'localhost',
        'PORT': '3306',
    }
}
```

## 3.运行项目

创建记录和数据表，注意在源码\bankmanagementsystem目录下运行

```
python manage.py migrate
python manage.py makemigrations login
python manage.py migrate login
python manage.py runserver 0.0.0.0:8000
```

如果需要创建管理员

```
python manage.py createsuperuser
```

## 4.打开网页

---

1. 主页: <http://localhost:8000/index/>

## 5.创建数据库触发器

---

lab3.sql文件包含了本次数据库的表结构，但已经用Django自带的数据库扫描方法将数据库模型存在了models.py文件中了，因此数据库表文件会自动创建，不需要允许lab3.sql脚本。但仍创建了几个额外的触发器，因此在项目运行后请运行MySQL源码中的lab3\_1.sql脚本