寒假项目经验整理总结

1. Django的创建及使用

首先，在pip3安装Django后，我的django版本为2.2.8. 然后在终端直接运行可执行文件django-admin。windows需要在对应目录下，如果没有添加到环境变量的话。

在想要创建的目录下执行命令：django-admin startproject mysite 就会自动在目录下创建一个mystie的文件夹。里面会有自动创建的一些文件。而mysite就是项目名称。

Pycharm新建项目，直接里面就有django的项目，创建后就会有相同的一些文件，但是Pycharm创建多了一个templates文件夹和db.sqlit3文件

下面详细说明这些文件的一些个作用

-manage.py 管理的一个文件，项目各种管理可以通过这个文件执行，比如./manage.py runserver

-settings.py 一个配置文件，可以在里面配置数据库信息等

-urls.py 路由系统 url->函数的对应关系

-wsgi.py 定义Django用什么socket来实现 默认wsgiref，uwsgi

-templates文件夹 放入前端文件

-db.sqlite3 Django自带的数据库

1. 采用Django使用MVC设计模式

1、创建app

之前没有使用过MVC模式，然而Django自带了类似MVC的MVT模式，也就是模型-视图-模版这样的模式，这种模式必须要在创建了一个app的情况下才能使用具体的这个项目也就创建了一个app，在终端使用manage.py直接python manage.py startapp XXX就可以创建一个app了，然后再把app添加到settings.py文件里面，就可以使用app了。在这里我创建了一个app，名字叫equipment。



可以看到这个app里面同样是有很多文件的，具体的模版和一些views文件都放在这里面。

2、使用views控制交互

在前端的数据展示和数据交互都需要通过views里面的函数进行处理，可以看到我前面的工程目录里面除了本身app创建时自带的views.py以外，我还自己创建了login\_views.py, excel\_views.py等，因为全部放在一个文件里面我觉得太多了，所以分开了。

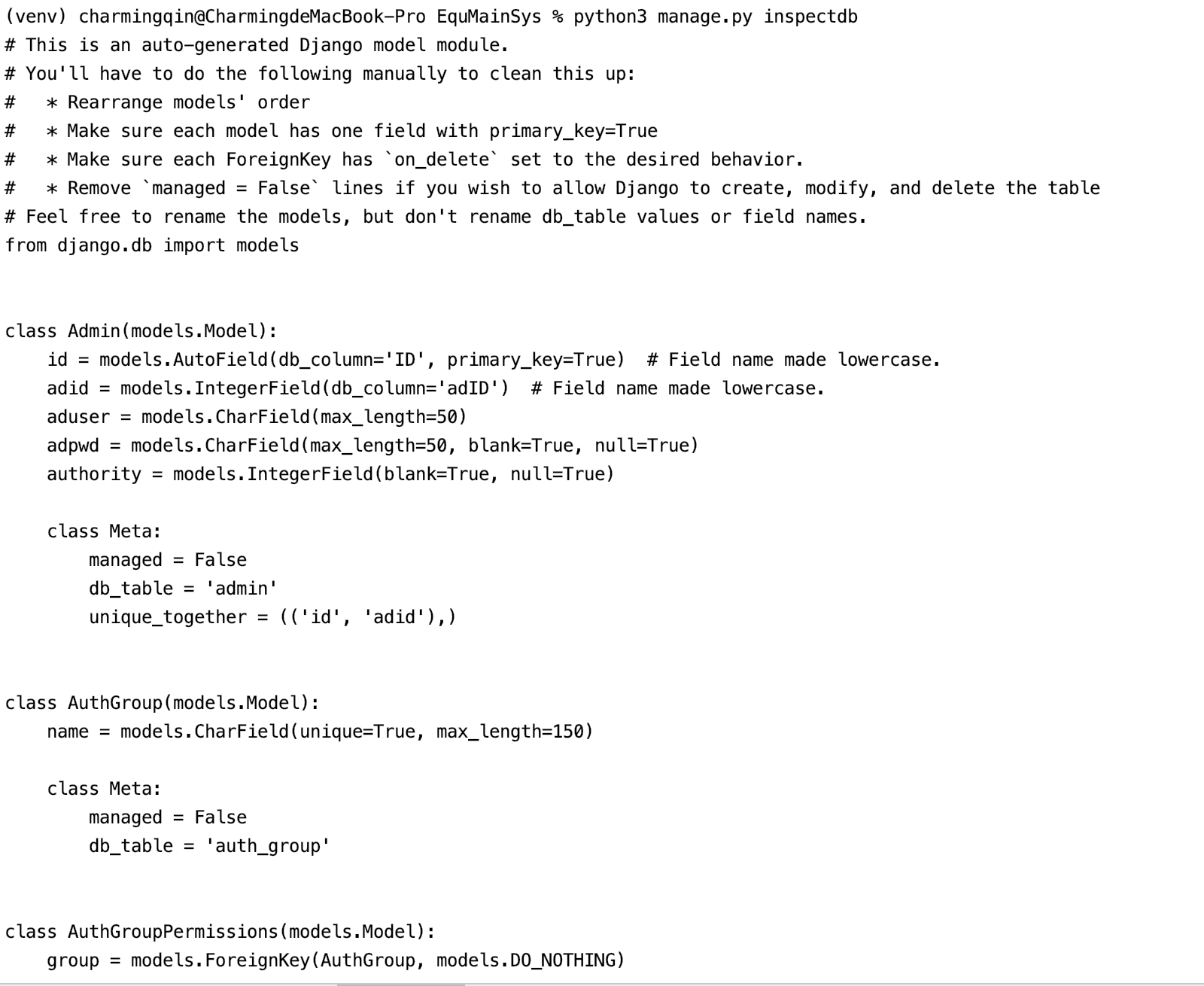
这个是其中的一个方法，对应控制管理员登陆的验证，具体的urls文件对应如下：

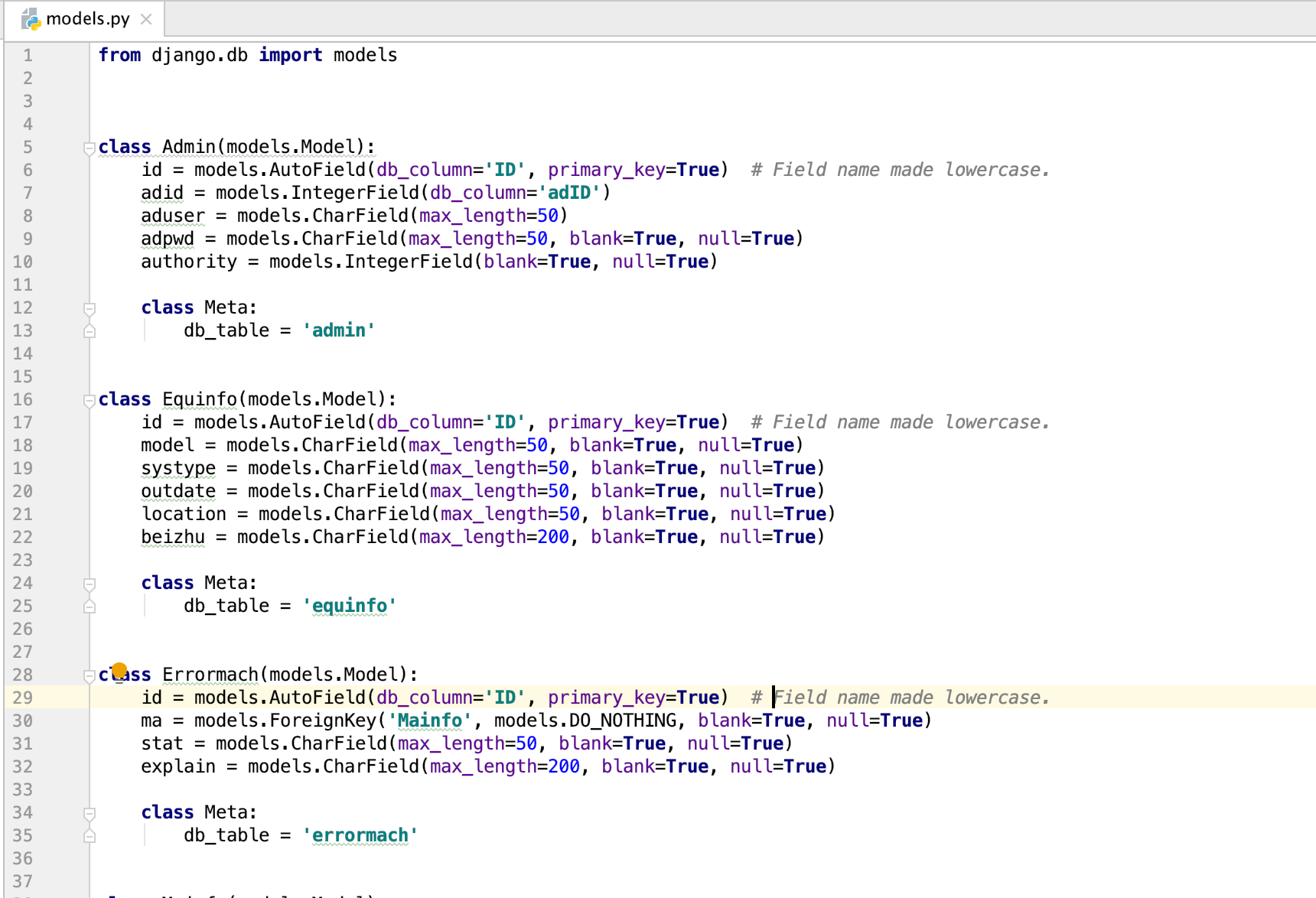


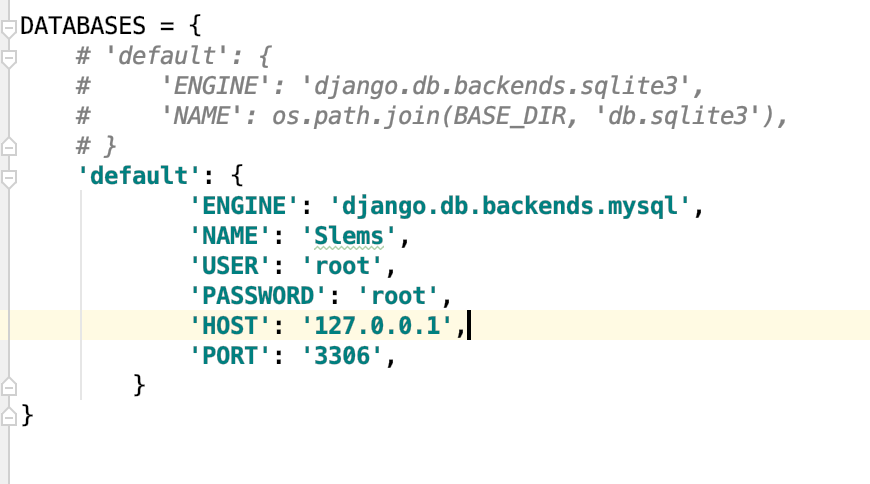
可以看到是对于login\_views.py里面的user\_login函数，这个函数注释的内容是以前还不是很懂一些方法以及操作的时候写的，很繁琐，后面使用模型之后就很短的几行就可以实现了。

3、使用模型控制数据库

模型控制数据库是我第一次接触的东西，以往进行C#开发的时候都是自己写sql语句，这个模型直接封装成方法使用不用自己写sql语句让我很是惊讶，所以导致为什么我电脑上使用的是mysql数据库而第一次部署服务器因为服务器端没有装mysql直接使用自带的sqlite一样可以运行，所以可以跨数据库使用了，让我再一次感受到了python的方便。具体的模型使用：

首先因为已经在本地数据库中创建表了，字段什么都添加好了，但是直接可以使用manage.py一个命令自动生成模型代码，

也就是如这样所见，直接生成model代码，

这个代码放在models.py里面，如果我想在admin表里面新加一个字段，可以直接使用Admin.objects.create(adid= ,aduser= ,adpwd= ,)就直接这样一句就可以代替sql语句中的INSERT INTO了，我使用的mysql数据库也需要在settings.py里面声明

非常方便

1. 前后端的数据交互
2. 使用form标签交互

我想这个方式是最普通的一种交互方式了，如下图，一点登陆submit就直接讲input里面的数据提交到后台，通过定义的name获取数据，方式是post，在后台直接使用request.POST.get()就可以获取数据。



但是这样会有一个问题，每一次点按钮都会再来一次http请求，刷新一次页面，从而导致一些我不想要的效果

1. Ajax交互

使用ajax进行数据交互；比如在管理员修改界面，我想使用ajax进行数据交互，如果数据出错不至于刷新页面



有了这些代码，一些空判断甚至都放在了前端，但是后面发现其实放在后端也是可以的，然后为了定位到id也就是主键还是用了parent().prevAll函数进行获取父标签获取id，期间也遇到了很多困难，但是ajax的交互还是成功了。

1. 前端知识补充

实际自己的前端知识是很差的，有时候调一个样式很久都调不好，为了一个弹窗效果有时候就很费时间，但是在这个项目中大量使用了模态框，也就是bootstrap框架的模态框，很能解决一些问题，通过模态框的一些数据展示以及数据修改都能很方便。但是这个项目的前端知识远远没有那么的多，一些js也只是很浅的东西。

1. 服务器部署

项目的demo完成后，第一次部署在阿里云上的时候就遇到很多麻烦，拟采用nginx服务器部署，但是由于自己的不熟练导致配环境的时候这个包又装不上，那个地方又崩了这样的情况，还有就是虽然有linux基础但是还远远没有那么熟练可以达到一个运维工程师的水平，后面换了windows server进行部署，换在了腾讯云上，因为以前有过部署java和ASP.NET项目成功在腾讯云上的经历，所以换在了腾讯云上的windows server2016上，直接cmd运行

python manage.py runserver 0.0.0.0:8000然后ip加8000端口访问就成功了，外网可以访问了，然后我就想为什么阿里云不行，我想一定是安全组没有配好，果然现在放行了8000端口直接终端运行python3 manage.py runserver一样可以外网访问，因为阿里云的服务器要好一点访问还比较流畅，但是这也只是一个暂时的展示效果，如果数据量大可能还是不行，目前项目部署在阿里云上，阿里云服务器使用的是centos6，数据库换成mysql5.7了，然后使用的是默认的80端口。

维护员端地址<http://39.107.254.79/stulogin/>

管理员端地址<http://39.107.254.79/adlogin/> （默认账号密码都是2017001，可以在管理员端查看更多用户）

1. 关于参加创青春比赛的算法想法

目前认为要求的功能基本实现，可能具体一些细节还达不到要求，但是问题都不是很大。但是身为一个在图书馆勤工助学了的人，经历过以前拿一个本子去各个楼层查看机器的情况，也遇到过经常这个电脑蓝屏啊或者是这个电脑它显示屏不亮等情况出现，所以在开发这个登记系统等时候有一个想法冒出：

可以设计一个算法实现一个预警系统吗？或者说是一个设备崩溃预警报告这么个功能；

因为每次巡查的都是机器的固定的一些项目，比如鼠标键盘灵不灵，或者是蓝屏没有这么个情况。这个功能可以通过大量的维护员的记录情况，根据这些记录情况来推算这个机器会不会在近期出现问题，并给出具体出现什么问题的可能性，并反映在后台管理系统上，或者更加花哨一点，发给管理员邮件或者是短信之类的。

关于算法的实现，目前来说想的是使用LOF算法，一个离群点检测算法，目前刚刚了解到这个算法还在接触过程中，所以还不是很熟悉，感觉有点类似于k近邻算法，一种离群点检测的算法。

目前组了个队，王越豪负责人，加我还有周小萱和一个地科院的同学。