## 只不过是从头~再来

1. 创建项目

首先需要配置env2的虚拟环境

打开黑窗口，输入workon env2进入env2的虚拟环境中

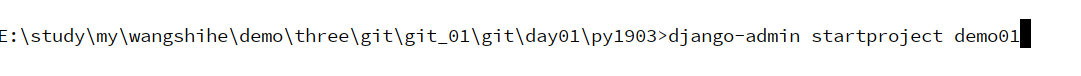
然后输入pip install mongodb配置env2的mongodb环境

在pycharm中下方选择terminal选项



首先输入workon env2进入env2虚拟环境中

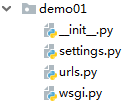
然后在里边输入:django-admin startproject 文件名然后回车



此时在我们左侧project中会出现我们创建的项目



我们点开文件夹





# 我们会发现我们有内外两个demo01文件

~ 外部demo01文件夹：项目文件夹

~ 内部demo01文件夹：创建项目自动生成的文件夹，储存的内部文件

~ manage.py文件：命令行工具

~ \_\_init\_\_.py：表示我们创建的是一个模块

~ setting.py：项目配置模块

~ url.py：项目路由模块

~ wsgi.py：wsgi协议实现

1. 创建应用：

# 创建应用需要在与manage.py同级的文件夹创建，所以我们需要cd到文件夹中

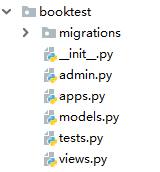
cd 文件夹名称



进入同级文件夹后输入：python manage.py startapp 文件名

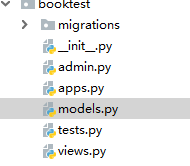


我们会发现了这样一个文件夹



1. 创建类

首先我们创建类需要在给定的models.py模块中创建，双击进入



根据需求，我们依照上方的思想创建类



# models模块中已经为我们引入了models模块

# 我们创建的类需要继承父类Model，这样我们就可以使用封装好的方法进行操作

~ models.CharField(max\_length=20)：创建的列需要的数据是CharField类型的，虽大长度为20

~ models.DateTimeField()：创建的列需要的数据是DateTimeField类型的。

~ default=True：默认值是True

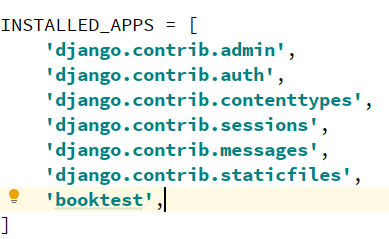
# 我们在创建表时所需的数据类型在这里都有封装，可以直接选择使用

# 我们类属性的名称会作为表的列名

外键的创建：外键名 = Models.Foreignkey(外键关联的类名, on\_delete=models.CASCADE)

1. 类与表的转化

进入setting文件夹中，配置数据库。



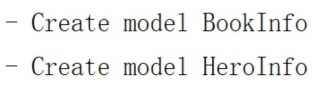
# 在最后一行按照格式将我们创建的应用文件夹名称填写进去

生成迁移文件：根据模型类生成sql语句

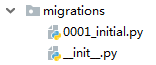
命令：python manage.py makemigrations 回车



显示创建即为成功



我们在migrations文件夹中会出现这样一个文件

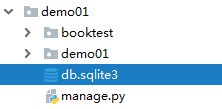


执行迁移：执行sql语句生成表

命令：python manage.py migrate



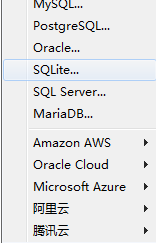
执行成功后我们会在外层目录中找到数据库文件



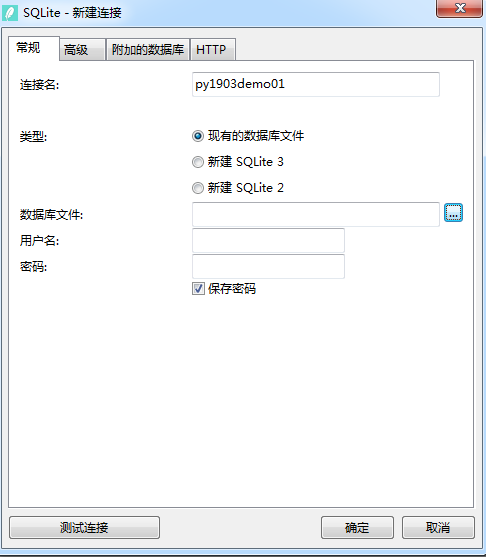
之后我们进入数据库点击连接



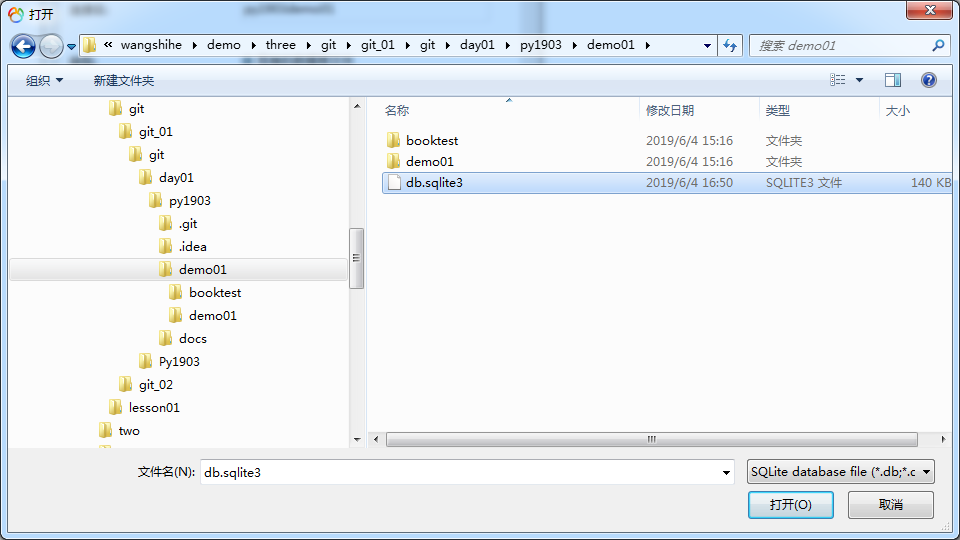
选择SQLite，因为我们python封装的转换是这个数据库的



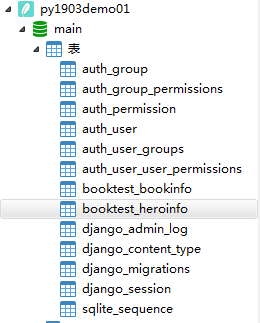
然后我们填写连接名，点击数据库文件的三个小点…



我们找到我们生成的db文件点击打开，然后完成



我们的表就生成完成了



其中booktest开头的两个表就是我们需要的表

1.打开后台:

在虚拟环境下输入python manage.py runseruer

2.创建一个超级管理员用户:python manage.py createsuperuser

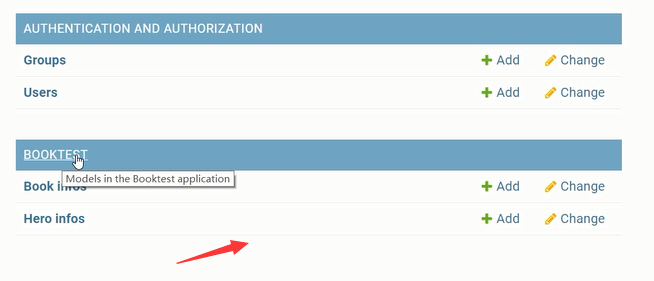
3.在admin中引入 from.models import BookInfo,HeroInfo

输入注册类:注册一个BookInfo和一个HeroInfo

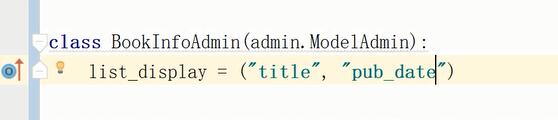
admin.site.register(BookInfo)

admin.site.register(HeroInfo)

得到如下效果

4.重写模型的后台管理类：如图:



重写方法: 



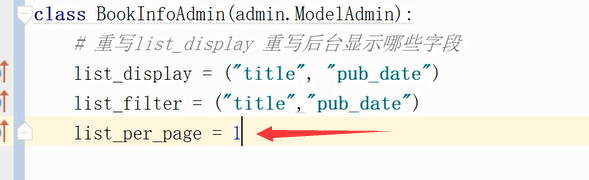
重写英雄: 

过滤器:

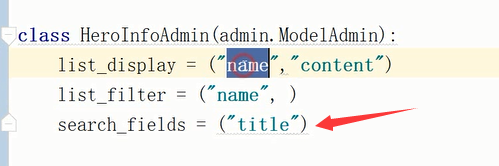


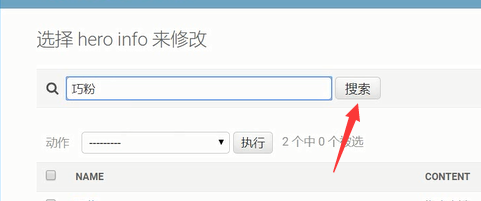


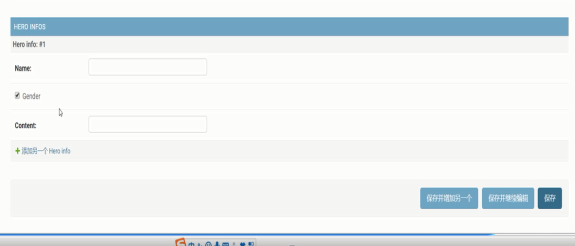
分页:

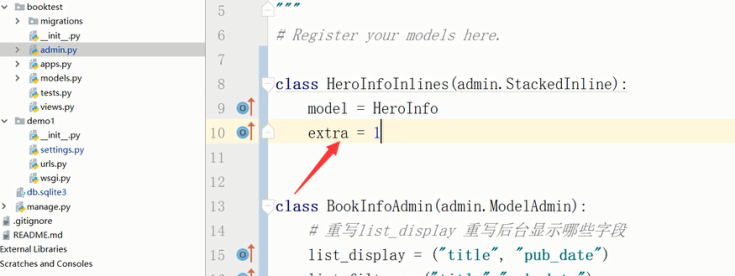




搜索: 



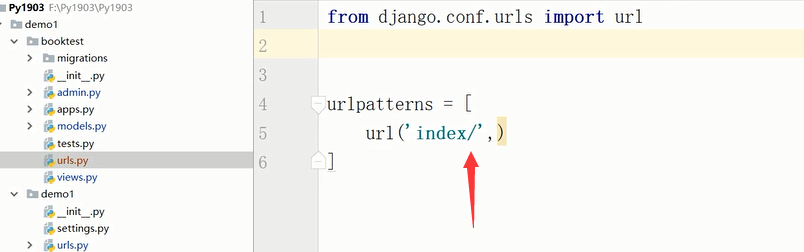
在添加一本书 默认添加一个英雄如图：

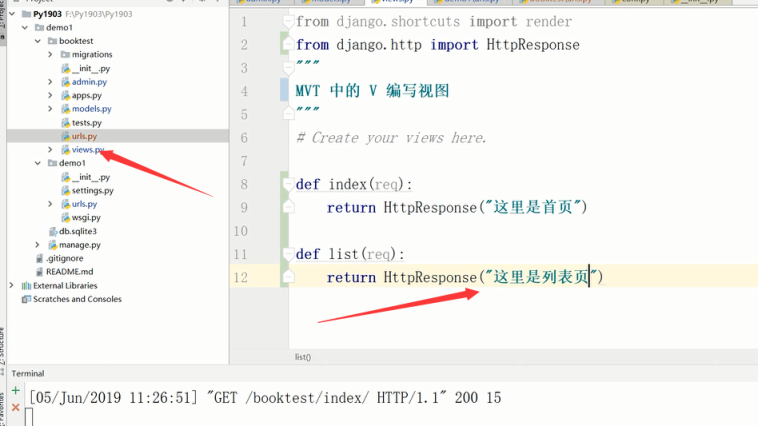


创建视图：1.

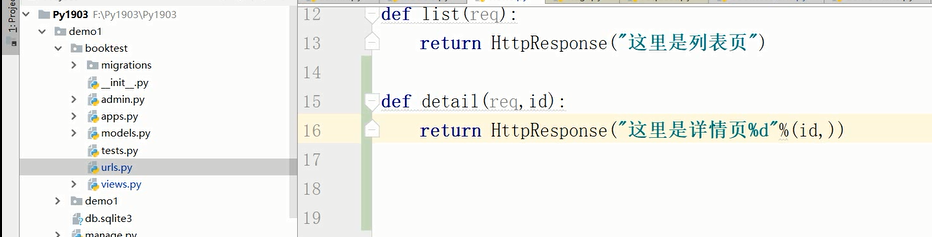
引入一个命名空间

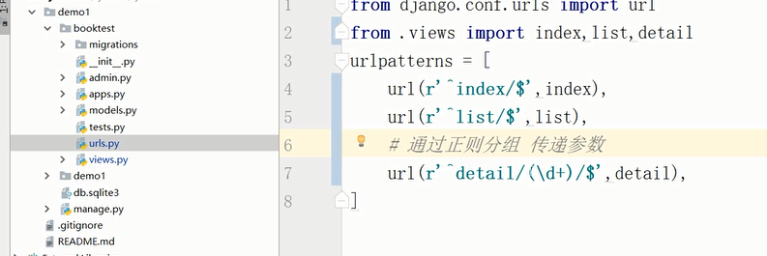


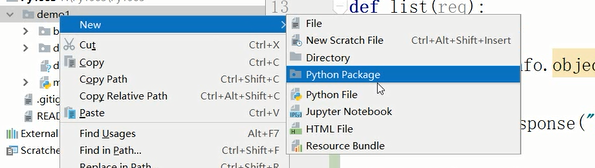
进入新建的urls: 

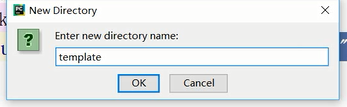


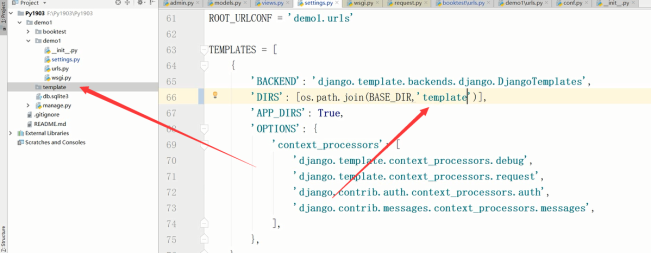


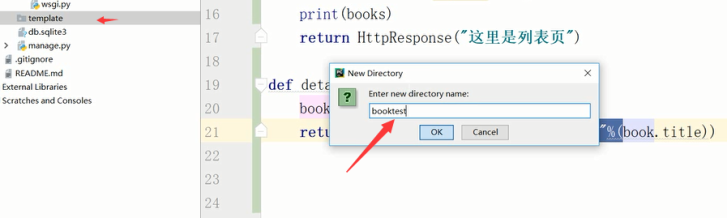
使用正则：

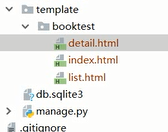


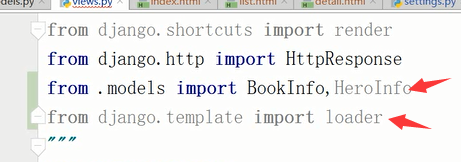
新建文件夹



配置静态文件夹存放

在新建文件夹下方 在创建一个用于存放HTML文件

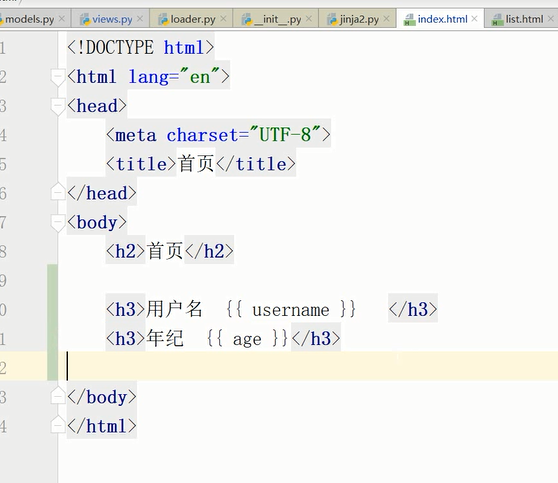
创建HTML文件：

引入模块：

获取模板：

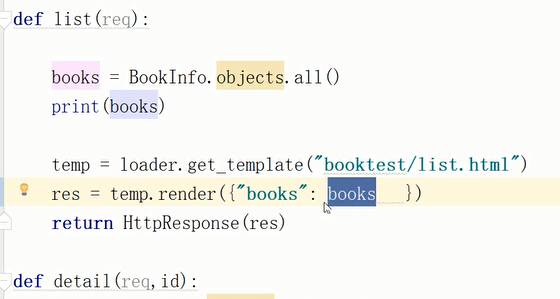


使用模板并传入参数：



使用动态数据渲染静态页面

列表页:





详情页：