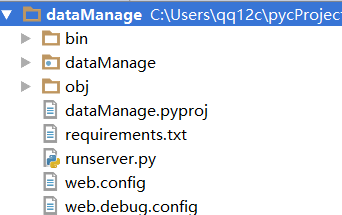
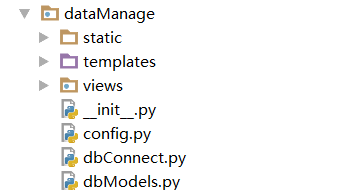
首先是根文件夹：

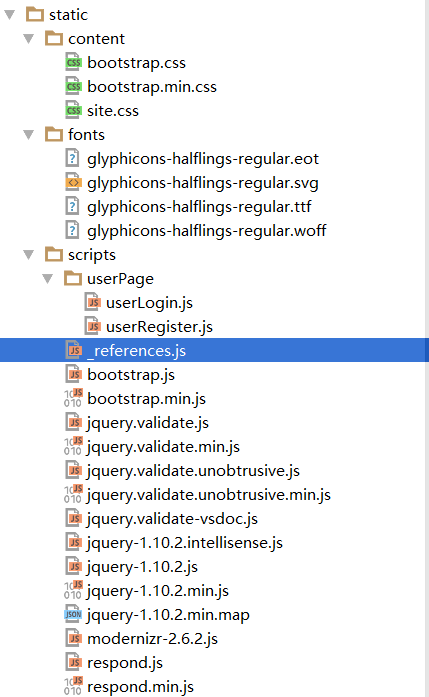


，结构如图。Runserver.py是与服务器交接的程序，即main函数在其中声明。app 是我们的python web程序。剩下的bin文件夹，obj文件夹，dataManage.pyproj文件，web.config文件，web.debug.config文件不需要理会，大部分为visual Studio生成，无需操作。

进入dataManage文件夹，会有如下结构;



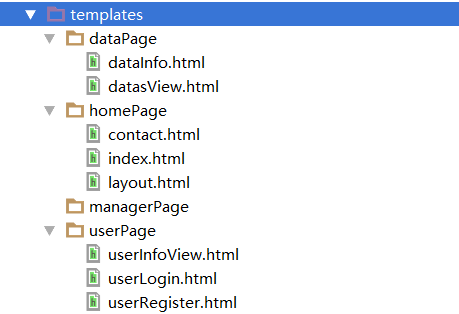
Static文件夹，用于存放静态文件，如图片文件、javaScript文件、css样式文件等。Static文件夹已经在flask框架下指明路径，因此以后有涉及到静态文件（大部分为非.py文件以及非html文件）请在static文件下存放。

如图中的结构。

Content文件夹中是bootstrap的css样式表

Fonts文件夹中存放字体，这里不需关注

Script除了放置了bootstrap、jquery等项目必须js之外，还有我们自定义的userPage文件夹（后面可能还会有homePage、dataPage、managerPage等文件夹，他们都是我们这个项目的分区，只是所需js文件存放在这里。）

Templates文件夹 ：

根据程序的分区，里面分成了dataPage,homepage,managerPage,

UserPage等四个分区，各个分区通过

BluePrint进行结构分区（以后会说明）

打开每个分区的文件夹会看到不同的html文件，这些就是我们要访问的基本页面。从字面应该能知道每个页面是说明意思。

Views文件夹：

里面存放.py文件，这个文件夹里面每个.py文件存放的是视图。就是通过浏览器中url的不同来访问不同的页面。

如，

@userPage.route('/userLogin')  
def userLogin():  
 return render\_template('userPage/userLogin.html',  
 title = '登录')

这里，@userPage.route(‘/userLogin’),意思是，你浏览器访问服务器网址跟后缀/userLogin ,如在本地就是（localhost:5000/userLogin）,则通过下面的 render\_template，返回

UserPage分区下的userLogin.html文件，对这个html文件进行渲染，从而访问这个页面。

Title是后台返回给title端的参数，后面会说明。

其余文件：



这三个，第一个是flask的配置，第二个是数据库连接文件，第三个是数据库模型。这三个文件不需要管。