总体任务：

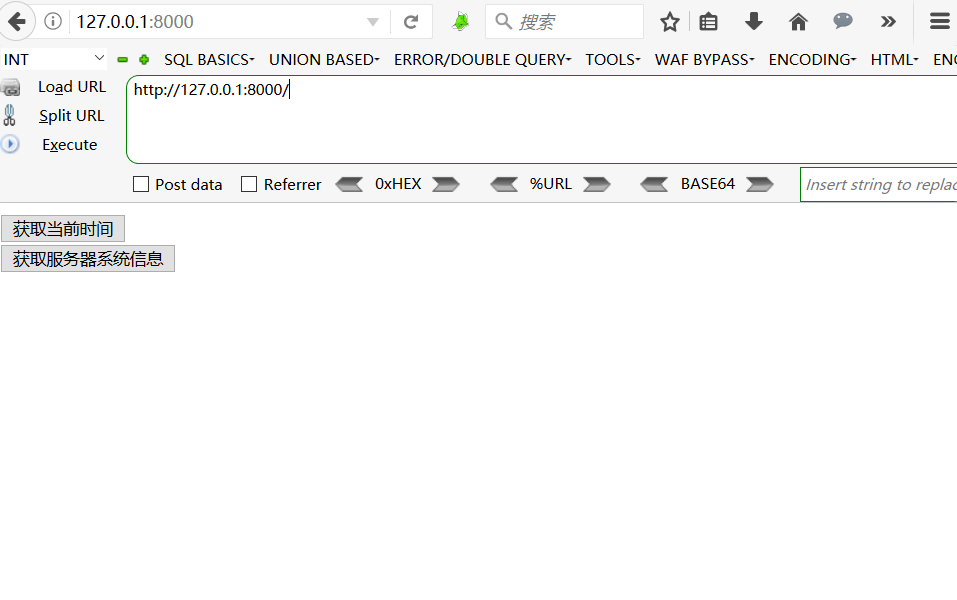
1. 搭建一个本地web服务器, 访问指定页面能够显示当前服务器的资源信息(CPU/内存/磁盘占用率/网口的实时速率)2. 搭建一个本地留言系统, 访问页面能够填写发布人的名称-标题-内容, 也能够看到之前发布的历史留言3. 在2的基础上增加注册系统, 需要注册-登陆-验证后才能发布留言4. 在3的基础上增加管理员系统, 管理员能够管理注册账号和留言(一些常见的python web框架都会有后台管理功能, 要注意理解其原理和流程)5. 使用爬虫框架, 在留言系统的首页显示最新爬取的信息(选择自己感兴趣的网站爬吧, 最好不要都选同一个)

我完成了第一个，主要时间还是花在学习python Django框架上

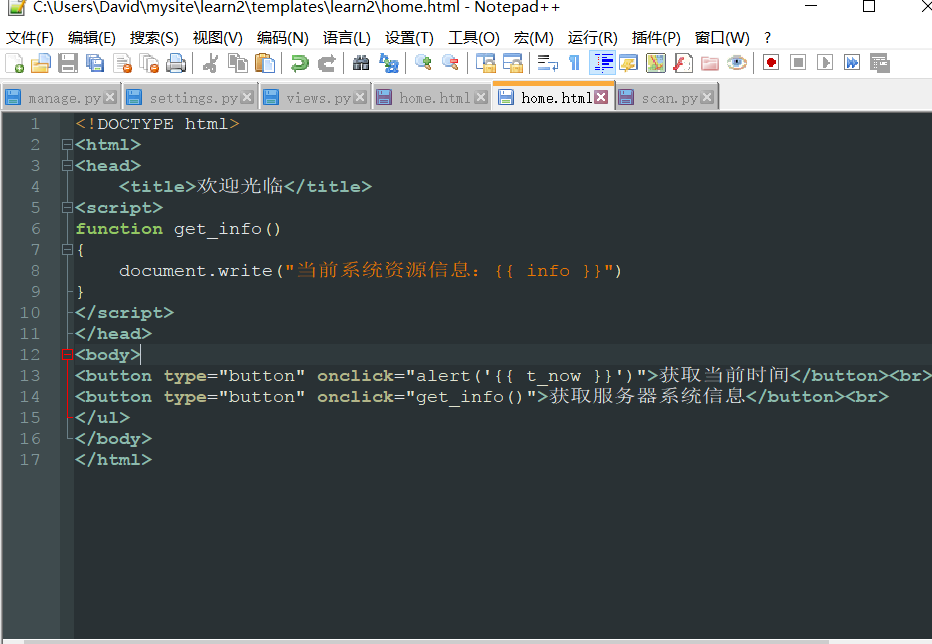
一：第一个任务是在django学习初级阶段完成的：

Python manage.py runserver之后，开发服务器运行在本机8000端口

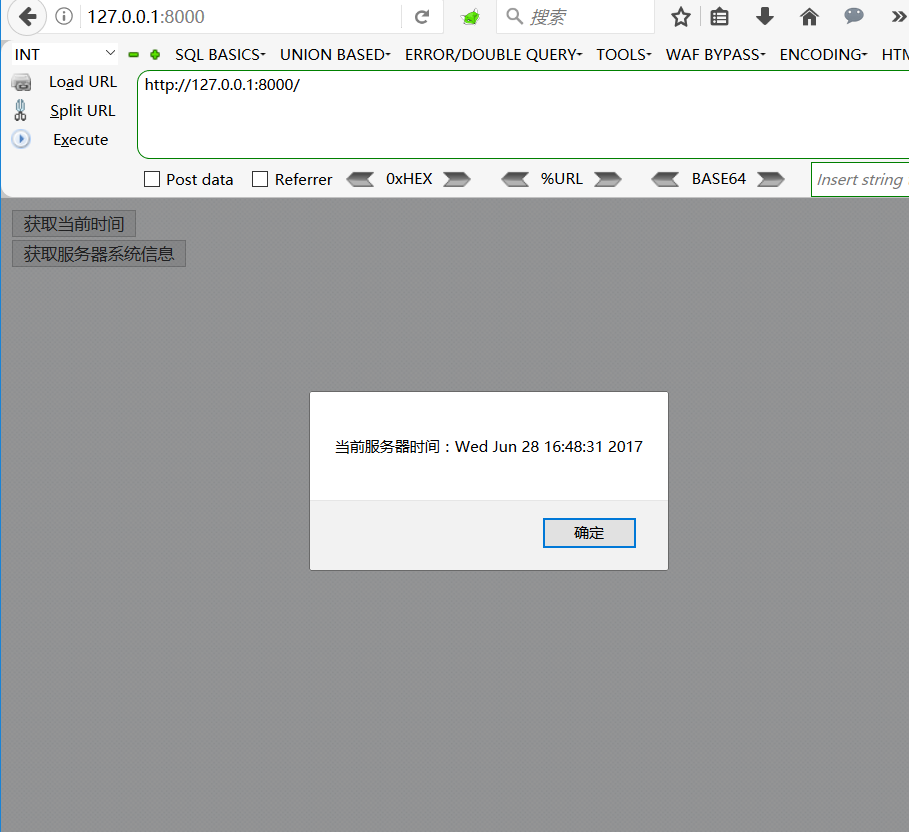
浏览器访问：



这是主页对应的模板html文件，比较简单，只是初步完成任务。

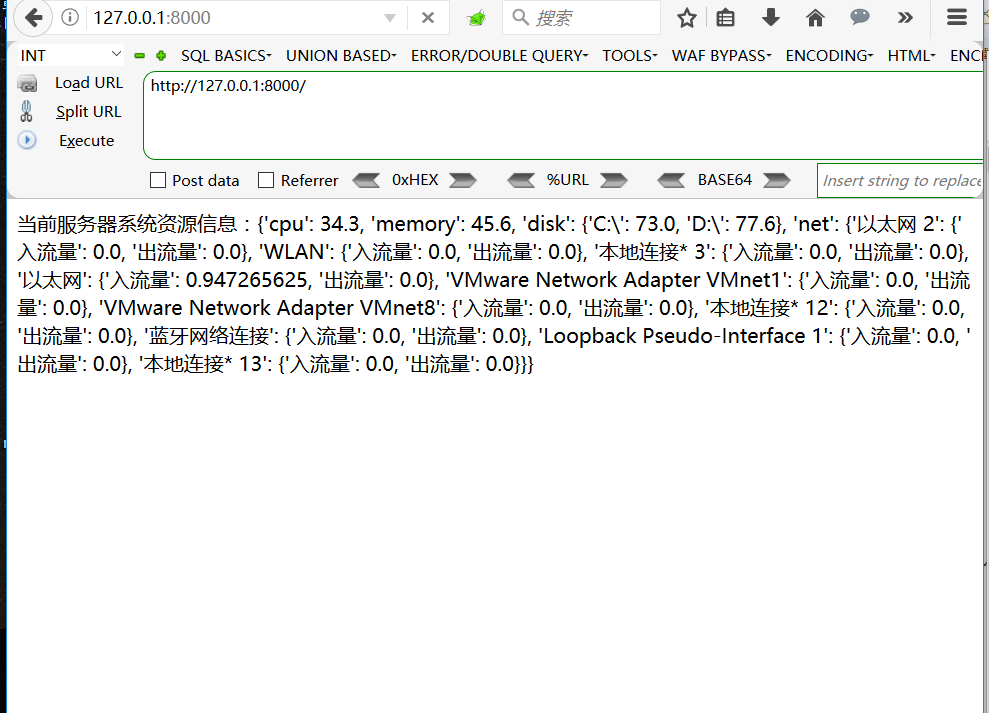


点击“获取当前时间”：弹出服务器当前时间



点击“获取服务器系统资源信息”

页面刷新输出，获取服务器相关信息用的是之前的getSysInfo模块完成的：



二：django学习过程的收获：

1. 初步了解了django框架的各个组成部分，以及环境配置，到初步写一个web app，运行开发服务器，浏览器访问得到相应的界面。
2. 学习了解sqlite数据库的安装配置和使用，以及若干常用命令，包括：创建数据库，数据库表，打开已有的数据库，对数据库进行增删改查操作。Django支持sqlite3，因此学习sqlite3对django的学习也有帮助
3. 进一步学习巩固html/xhtml/html5,css,javascript的相关知识。因为django模板开发需要运用到相关知识，因此我花时间学习并写了部分模板，为之后的任务做准备
4. 使用django写了计算app，将django教程各部分结合在一起练习对django的使用。

三：学习过程的问题：

问题主要集中在django使用中对模板的动态更新以及与数据库的实时交互，查阅相关资料，在进一步学习中

四：下周工作计划：

完成全部5个任务，尽可能完善。