

基于维基的人物信息检索系统

一、功能：

1.爬取维基中的人物信息：

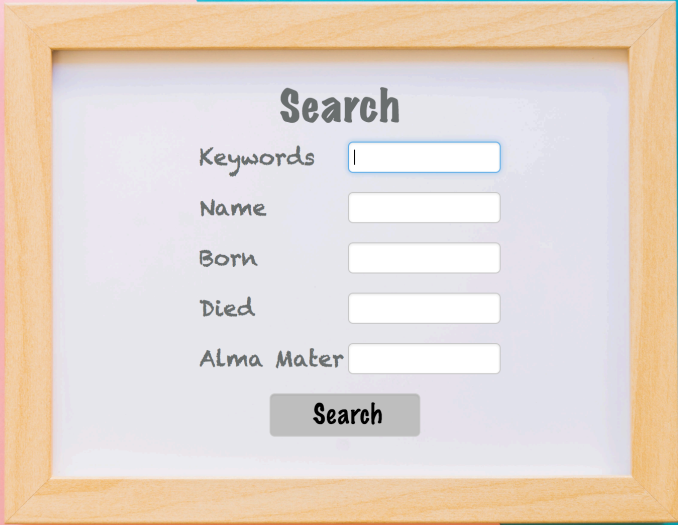
选择的分类是美国的男性和女性电影演员。共爬取了13012个人物的信息，提取了每个人物在维基中的页面的infobox中的信息，并且保存在.json格式的文件中。

2.处理爬取页面信息：

根据爬取到得先后顺序为每个人物赋予了独有的整数ID。分别根据名字、出生时间、死亡时间、母校等方面的信息各自建立了一个倒排列表，将相应的词语与ID对应。最后根据infobox中的所有信息建立了一个倒排列表，用于关键词查询。

3.实现网页：

建立了首页、查询页、信息页三种页面，首页和查询页支持用户进行查询、提交表单。查询的方法有两种：模糊查询和精确查询。模糊查询根据一个或多个关键词来查询，用户输入多个关键词，服务器返回包含全部关键词的文档的链接，同时也返回包含一部分关键词的文档的链接。



The image shows a search interface titled "Search" displayed within a wooden frame. The interface includes five input fields for "Keywords", "Name", "Born", "Died", and "Alma Mater". Below these fields is a "Search" button. The background is a gradient of pink and blue.

Keywords

Name Born Died Alma Mater

Search

Time consumed:0.623636s

Jon Bernthal

- Education: [Harvard](#) University Moscow Art Theatre Skidmore College

Amy Aquino

- Alma mater: [Harvard](#) University (B.S., Biology) Yale University (M.F.A.)

Ted Hartley

- Alma mater: U.S. Naval Academy [Harvard](#) Business School

Al Franken

- Education: [Harvard](#) University (BA)

Tommy Lee Jones

- Alma mater: [Harvard](#) University (BA)

1 2 3 4 5 6 7

信息页可以显示人物的姓名、照片以及一个包含人物其他信息

Tom Hiddleston



Residence	Belsize Park, North West London
Alma mater	Pembroke College, Cambridge Royal Academy of Dramatic Art
Home town	Wimbledon, London, England Oxford, England
Born	Thomas William Hiddleston 9 February 1981 (age 36) Westminster, London, England, UK
Years active	2001–present
Education	Dragon School Eton College
Occupation	Actor

的表格。信息页的地址包含任务的ID，由此来建立相应的人物页面。

4.使用CSS美化了页面：

经过网上查询和自己尝试之后，使用CSS的多种语句和多种、HTML部件的属性，利用CSS静态文件美化了自己的页面。

5.分页功能：

若符合条件的搜索结果超过了5个，将采取分页显示。每页显示5个搜索结果。每个搜索结果页的地址包含了页码数，服务器根据页码数进行动态渲染，返回网页。

6.标红关键词：

对于每一个符合条件的搜索结果，显示符合关键词的相关内容，并且重点标出其中的关键词。

7.支持多字段查询：

即前文提到的精确查询，用户可以输入想要查询的人物的名字、出生时间、死亡时间、母校信息，服务器只返回包含这些关键词并且关键词在人物信息中的类别正确的文档。

8.显示搜索时间：

在用户点击搜索之后，服务器在返回符合条件的文档链接的同时，也会返回搜索所用时间，经过测试，所需最长时间为0.7s到0.8s，不会超过1s。

二、写大作业的感想：

首先是爬取页面时的感想，我大概花了九个小时来爬这一万三千多个页面，中间有过一些中断，但不会出现从头再来的这种情况。我觉得这是一个很漫长的过程，可能也和我在图书馆的网络里爬取有关系，网络的阻塞可能会有所增加，但是我在宿舍爬取的时候，其实也没有快很多。爬取速度慢主要会因为网络的阻塞，本地的IO速度几乎对速度没有任何影响。总的来说，爬取这一部分比较心累，还是最好在代码里写好一些异常抛出和保存当前爬取状态的功能，然后让电脑在自己睡觉的时候爬取。

接下来就是django的使用了，其实我之前根本想不到django的使用这一块是最轻松的。这一部分其实就主要是度django的文档，学会这一个框架的使用方法。我觉得这个框架还蛮简单的，反正我在写

代码的时候经常出现直接编译成功、没有任何bug的情况，让我心情很好。这部分没有什么多说的，就遵守django的规定就好了。

最后就是每一次大作业最心累的部分了，也就是设计UI。虽然我每次都想要就此打住，不再修改了，但是每次看到自己的UI都想再多优化一点，就不断地投入了更多的时间。CSS相对Qt的UI设计框架来说，限制比较多，也比较难懂，所以这方面我花了差不多一天的时间来优化、设计界面。在这个过程中，我学会了很多关于HTML的知识，之前自己完全看不懂的网页源代码现在看来就好像一篇文章一样了。

这个小学期我真的是痛并快乐着，虽然任务和考试比较多，自己也熬了多次夜，但是同时也学到了很多新的知识，同时也做出了很多虽然不是很完善但是自己很喜欢的产品成果，我的代码量可能也提高了一倍吧。自己真的有了一种软件工程师的感觉，从产品的构思、功能实现到UI设计、文档编译，都需要靠一个人来完成，这样当你的产品诞生的时候，心里真的是有很多的成就感。

最后一个体会就是，一定要培养自己的搭建系统的直觉。我在写国际跳棋的时候，本来自己脑子里有个想法，但是和另外一个同学聊了之后，发现自己这个系统的想法真的是既不好调试、代码量又大。一个敏锐的直觉和好的想法真的能为自己省不少代码量和调试的时间。