1. 爬虫的基本原理以及使用的基本方法（如果使用框架，简述参数和各项module的功能）

由于本学院官网上的教师资料信息较为简单，本次作业的信息爬取使用Python的Requests库完成。

使用的基本方法：

首先需要设置一个headers，虽然学校官网没有反爬虫措施，但是还是要做好基本参数的设置。

headers = { 'user-agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4430.212 Safari/537.36 Edg/90.0.818.66',}

然后就是使用requests库进行信息的提取了

data = requests.get(url1,headers=headers)

data.encoding = 'utf-8'

baseInfo = data.text

1. 实现的简单流程：包括爬虫工具的使用、数据库设计、xpath/css路径匹配等等；

以一个老师的信息框架为例

<div class="teacher-left">

  <img class="teacher-img" src="/\_\_local/A/F6/48/BF5B51E0276CAE514B0C31EDDD5\_2952AE48\_AFC4.png">

  <p class="teacher-name">徐增林</p>

</div>

<div class="teacher-box">

  <dl>

    <dt>任职：</dt>

    <dd>国家青年特聘专家、计算机学院教授/博导</dd>

  </dl>

  <dl>

    <dt>电话：</dt>

    <dd> </dd>

  </dl>

  <dl>

    <dt>传真：</dt>

    <dd> </dd>

  </dl>

  <dl>

    <dt>Email：</dt>

    <dd><a href="mailto:xuzenglin@hit.edu.cn">xuzenglin@hit.edu.cn</a></dd>

  </dl>

  <dl>

    <dt>研究方向：</dt>

    <dd>

      致力于解决涉及现代大数据分析的关键建模和计算的挑战，实现复杂系统在自然语言处理，计算机视觉，社会计算，网络空间安全，生物信息学和生物医学应用等领域的研究。为此研究由各种应用驱动的稀疏、关系、动态、深度学习模型，并为这些模型开发精确、高效和可扩展的算法。

    </dd>

  </dl>

  <a href="../info/1021/2300.htm" class="teacher-link">查看简历</a>

</div>

可以看到所有老师的信息开头都有<div class="teacher-left">，因此可以以此为依据判断这一段代码是不是老师的信息。

判断出老师的信息后，还需要从中提取出老师信息中的各个字段，这里使用re库正则匹配就可以了。

    teacherInfo1 = re.findall(r'<div class="teacher-left">',baseInfo1)

    teacherName1 = re.findall(r'<p class="teacher-name">(.\*?)</p>', baseInfo1)

    teacherTitle1 = re.findall(r'<dt>任职：</dt><dd>(.\*?)</dd>', baseInfo1)

    teacherTelephone1 = re.findall(r'<dt>电话：</dt><dd> (.\*?)</dd>', baseInfo1)

    teacherFax1 = re.findall(r'<dt>传真：</dt><dd> (.\*?)</dd>', baseInfo1)

    teacherEmail1 = re.findall(r'<dt>Email：</dt><dd><a href="mailto:(.\*?)">', baseInfo1)

    teacherField1 = re.findall(r'<dt>研究方向：</dt><dd>(.\*?)</dd>', baseInfo1)

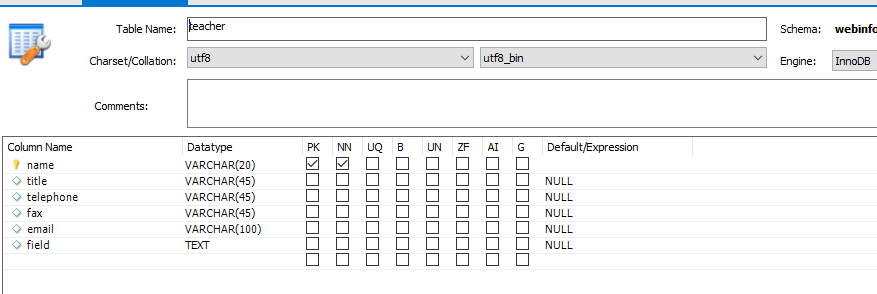
之后将数据存到MySQL中就完成该实验了。

for j in range(0,len(teacherInfo)):

    sql = 'insert into `teacher`(`name`,`title`,`telephone`,`fax`,`email`,`field`) values ("{}","{}","{}","{}","{}","{}")'.format(teacherName[j],teacherTitle[j],teacherTelephone[j],teacherFax[j],teacherEmail[j],teacherField[j])

    cursor.execute(sql)

db.commit()

数据库设计如下：

1. 简单的心得体会

本次作业相较来说比较简单，主要考察爬虫的使用以及数据库的设计。小小吐槽一下学院官网的设计，URL没有按顺序来，爬起来有一点麻烦，需要先对首页进行一个单独处理。