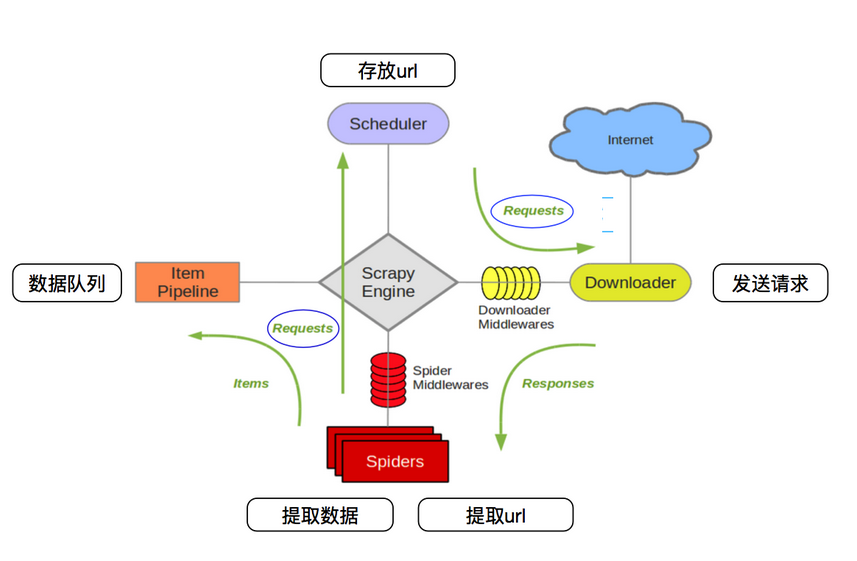
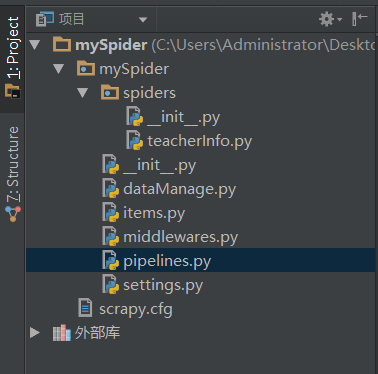
作业1：简单爬虫的实现

1. scrapy框架的基本原理

scrapy框架架构：



　一个简易的scapy项目目录结构:



·Scrapy Engine（引擎）：负责Spider，itemPipeline，Downloader, Scheduler中间的通讯，信号、数据传递等。

·Scheduler(调度器): 它负责接受引擎发送过来的Request请求，并按照一定的方式进行整理排列，入队，当引擎需要时，交还给引擎。

· Downloader（下载器）：负责下载Scrapy Engine(引擎)发送的所有Requests请求，并将其获取到的Responses交还给Scrapy Engine(引擎)，由引擎交给Spider来处理，

·Spider（爬虫）：它负责处理所有Responses,从中分析提取数据，获取Item字段需要的数据，并将需要跟进的URL提交给引擎，再次进入Scheduler(调度器).

　－－－spiders文件夹

·Item Pipeline(管道)：它负责处理Spider中获取到的Item，并进行进行后期处理（详细分析、过滤、存储等）的地方。－－－pipelines.py

·Downloader Middlewares（下载中间件）：你可以当作是一个可以自定义扩展下载功能的组件。

·Spider Middlewares（Spider中间件）：你可以理解为是一个可以自定扩展和操作引擎和Spider中间通信的功能组件（比如进入Spider的Responses;和从Spider出去的Requests）－－－－middlewares.py

最后setting.py是项目的设置文件；scrapy.cfg: 项目的配置文件。

1. 实现流程

（1）数据库的简单设计

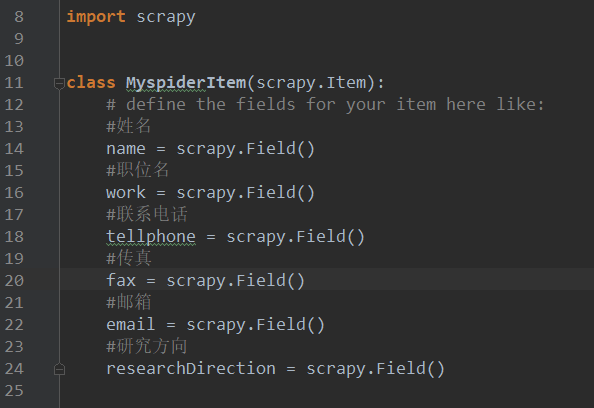
创建数据库：teachers

创建数据库表：teacher(name char(5) primary key,work char(20),tellphone char(20),fax char(20),email char(30),researchDirection char(50));

（2）爬取页面并利用xpath匹配相应数据

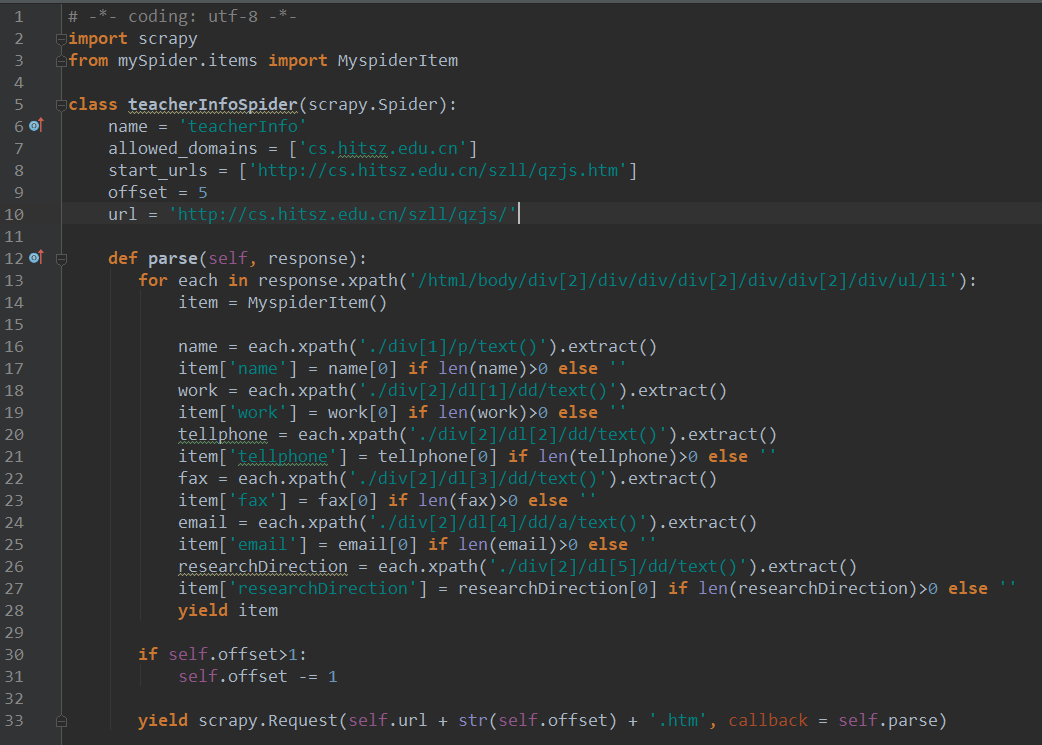
·创建scrapy项目：scrapy startproject mySpider

·在mySpider目录下的items.py中定义结构化数据字段（根据数据库表的结构）



·创建一个名为teacherInfo的爬虫：scrapy genspider teacherInfo

·在得到的teacherInfo.py文件中编写爬虫逻辑：



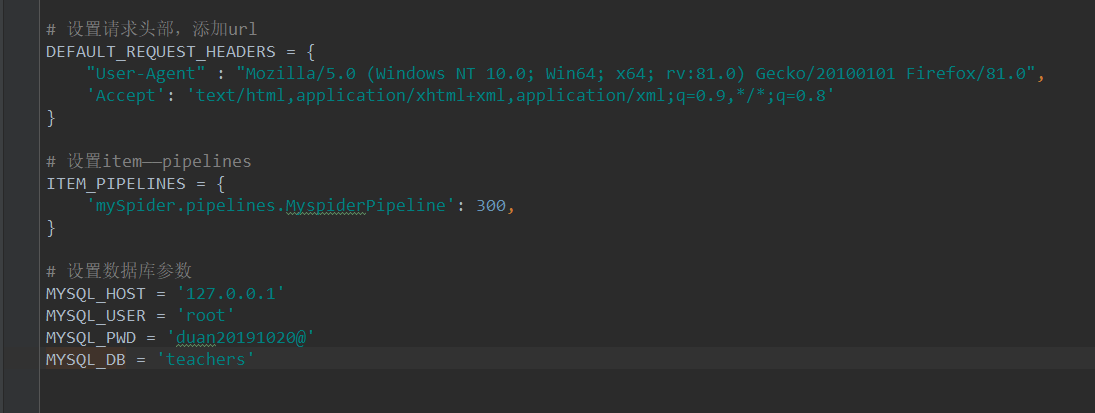
1首先设置域名为’cs.hitsx.edu.cn’,初始url为’https://cs.hitsz.edu/szll/qzjs.htm

2观察这个网页，能发现其中共有5页，首页的url即为初始url,其后的4页url为为’https://cs.hitsz.edu/szll/qzjs/’+offset+’.htm’,其中offset从4到1。

3在parse函数中进行XPath匹配：通过查看页面源代码，找到老师信息所在的元素，右键单击选择复制XPath,并分析页面结构即可知道该页所有老师的信息的xpath为’/html/body/div[2]/div/div/div[2]/div/div[2]/div/ul/li’；遍历通过该XPath匹配得到的结果，对结果进行解析，分别取出name,work,tellphone,fax,email,researchDirection,此处需注意判断的到的结果是否为空！

4.遍历剩下的页，使用scrapy.Request函数，回调函数设置为self.parse。

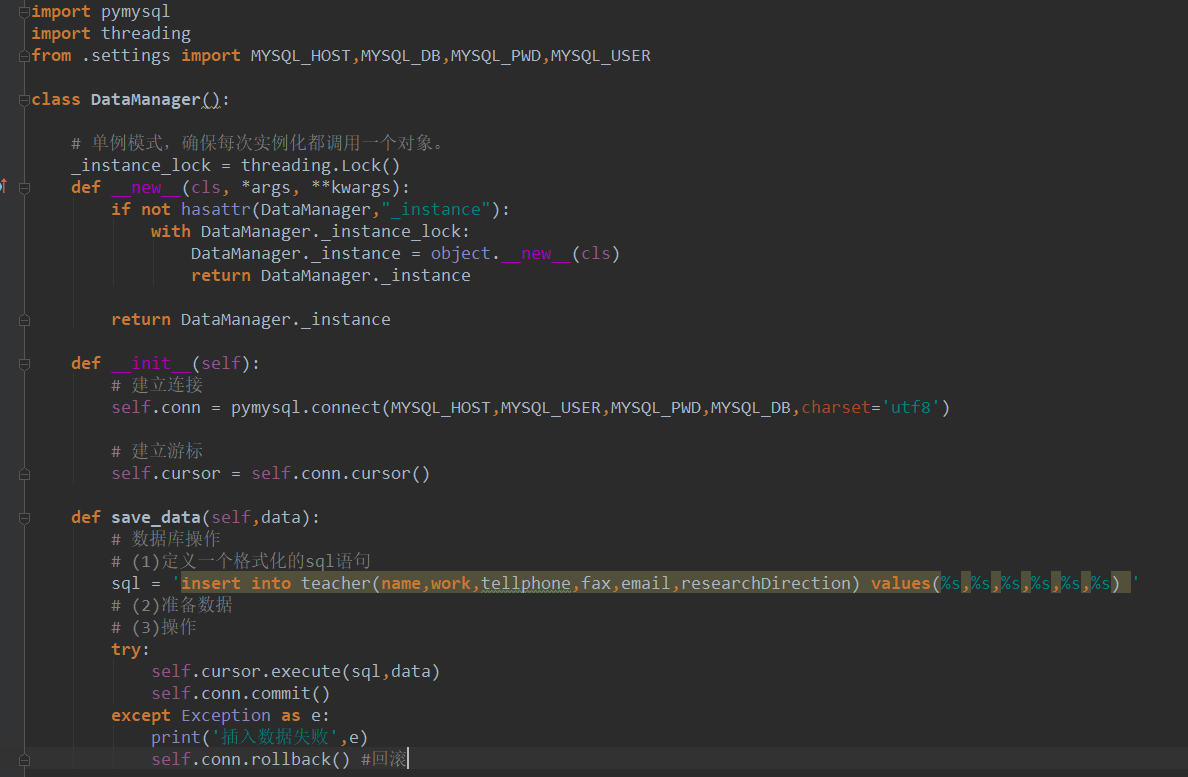
·在setting.py文件中设置参数



设置请求头部，item-pipelines以及数据库的一些参数

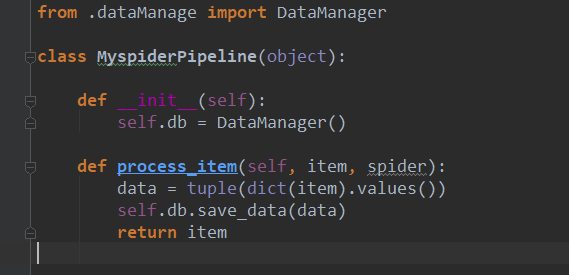
1. 将爬取到的结果存入数据库表中

·首先编写数据库操作相关的类

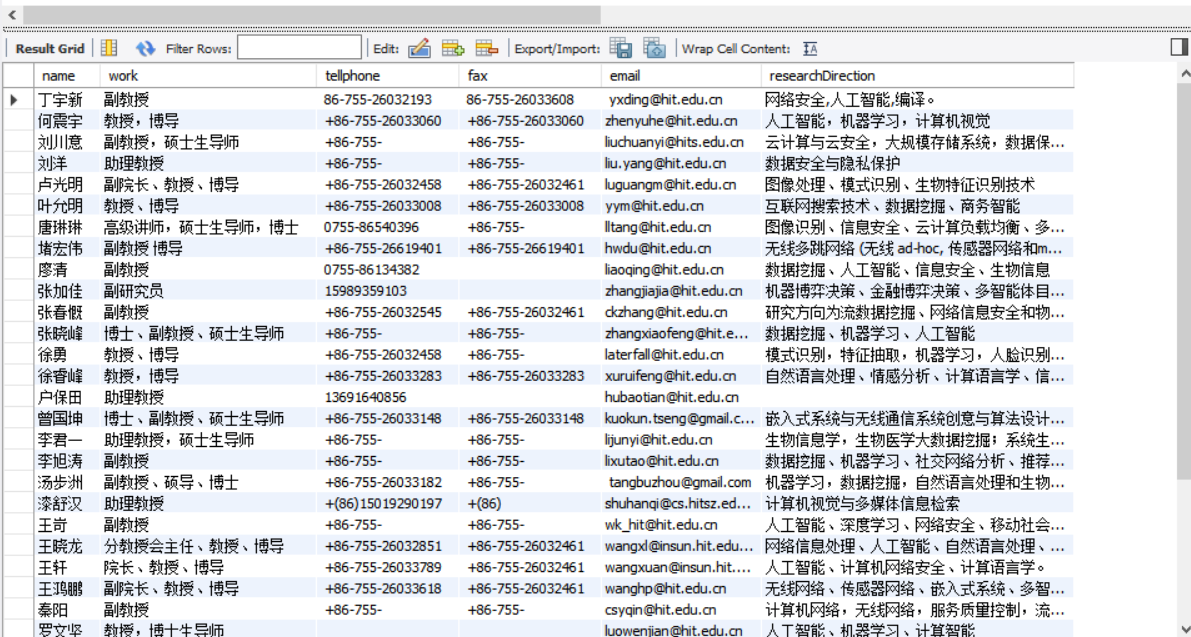


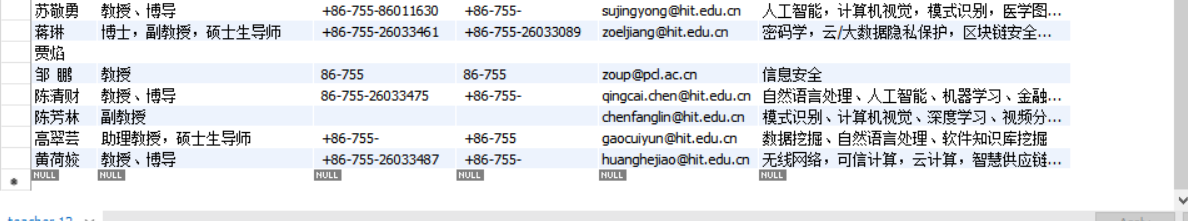
1. 建立与数据库的连接（主机名，［，端口（默认３３０６）］用户名，密码，数据库名，编码方式）以及建立游标（＿＿init＿＿函数）
2. 使用insert 语句将数据插入数据库,执行游标，然后提交，注意失败需要回滚保证数据库的完整性。

·然后，在pipelines.py文件中编写处理数据（将数据存入数据库）的逻辑：



1. 最终结果





1. 心得体会

　首先就是，使用爬虫框架会使得爬虫的逻辑编写起来更加清晰简单，基本上值用考虑数据的匹配和数据的存储；

　第二点就是对于数据的处理一定要注意空数据的情况的特殊处理，否则就会出现错误，比如说这次爬虫中，存在有老师的相关信息为空的情况：



　这种情况下，each.xpath('./div[1]/p/text()').extract()[0]会报错，

　因为此时使用each.xpath('./div[1]/p/text()').extract()得到的数组是一个空数组。