1. List

有序集合，占用空间小，查找和插入采用顺序查找，效率低。初始化之后可以修改 len计算长度 append追加 insert插入



1. tuple

元祖，也是一种有序列表，和list类似，但是初始化之后就不能修改

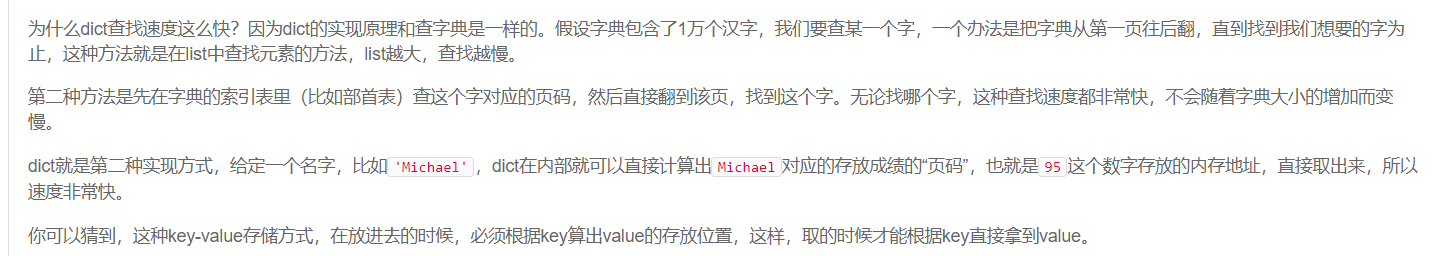


1

1. dict

字典 （键值对） key-value存储，查询速度极快

声明：d={‘key’:value}



删除：d.pop(key)

获取：d.get(key)

查找和插入的速度极快，不会随着key的增加而变慢；

需要占用大量的内存，内存浪费多。

1. set

不存储value，只存储key的集合，要求里面的元素不能重复，如果重复会自动过滤。

初始化：s = set([1,2,3]);

添加 ：s.add(4);

删除： s.remove(4);

1. 函数

定义：Def

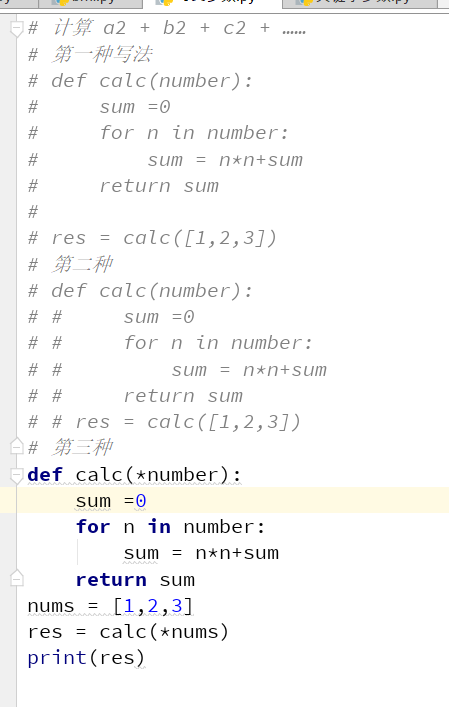
空函数：需要函数但是没想好这个函数怎么写的时候需要使用pass来占位

**Python解释器会检查出函数参数个数的错误，但是不会检查参数类型的错误**

Python函数可以返回return多个值 return a,b;(**表面上，实际上返回的是一个tuple**)

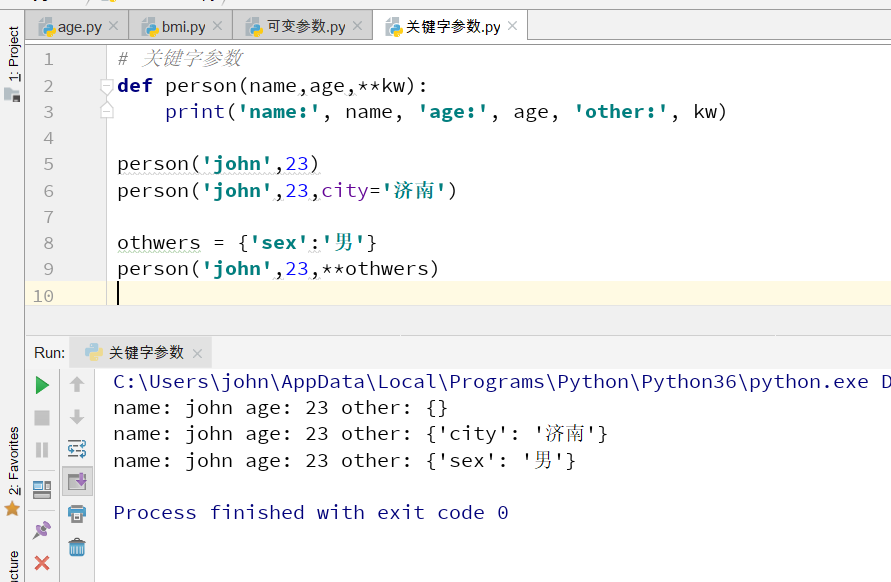
默认参数：默认参数必须指向不变对象 以下例子参数可以设置L=none，none是不变对象



可变参数：

关键字参数：除了必要的参数之外，还可以传入其他的参数

关键字参数有什么用？它可以扩展函数的功能。比如，在person函数里，我们保证能接收到name和age这两个参数，但是，如果调用者愿意提供更多的参数，我们也能收到。试想你正在做一个用户注册的功能，除了用户名和年龄是必填项外，其他都是可选项，利用关键字参数来定义这个函数就能满足注册的需求。



1. 切片操作

取list或者tuple中的部分元素（一个范围内的元素）

L = ['Michael', 'Sarah', 'Tracy', 'Bob', 'Jack']

取前三的元素：L[0:3] 🡺 ['Michael', 'Sarah', 'Tracy']

倒切片：

取最后两个元素：L[-2:]

取倒数第二个元素L[-2:-1]

**记住倒数第一个元素的索引是-1**

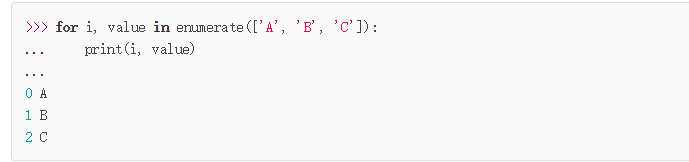
1. 迭代

使用for in遍历list或tuple

Python中的for可以用在其他迭代对象上，list,tuple,dict,字符串等，可以使用以下方法判断对象是否可迭代



示例：





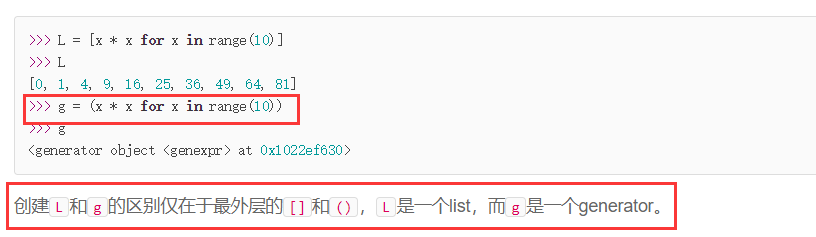
1. 列表生成式

用来创建list



1. 生成器

保存列表的规则（算法），需要时根据算法取生成结果

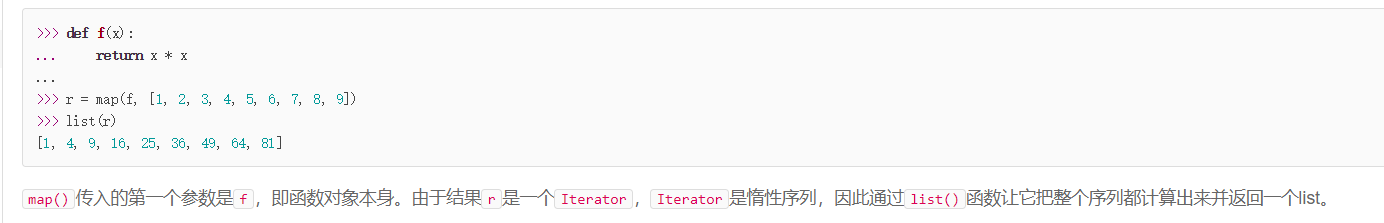


如果在函数中的某个地方添加了yield关键字，那么这个函数就不再是一个普通函数，而是一个generator

**最难理解的就是generator和函数的执行流程不一样。函数是顺序执行，遇到return语句或者最后一行函数语句就返回。而变成generator的函数，在每次调用next()的时候执行，遇到yield语句返回，再次执行时从上次返回的yield语句处继续执行。**

1. 高阶函数map

map()函数接收两个参数，一个是函数，一个是Iterable，map将传入的函数依次作用到序列的每个元素，并把结果作为新的Iterator返回。



**Reduce,filter和map类似不在说明，需要去查手册**

1. 返回函数

一个函数A,作为另一个函数B的返回值被返回而不执行函数A,只有调用B的返回值的时候，才会调用A



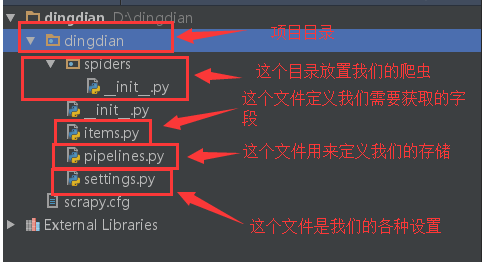
**Scrapy爬虫教程：**

1. **安装scrapy (在Anaconda 环境下)**

code install Scrapy

1. 创建一个爬虫项目

scrapy startproject XXXXX  XXXXX代表你项目的名字



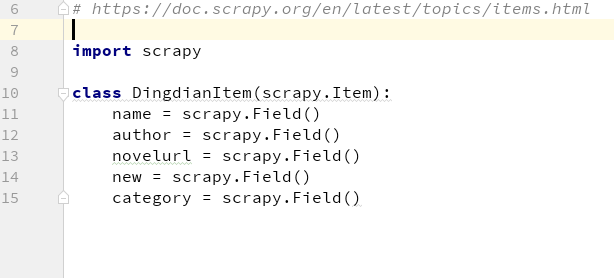
1. 爬虫默认只能从命令行中启动，如果想在IDE中运行，需要创建一个entrypoint.py文件，用来模拟从命令行中启动

from scrapy.cmdline import execute

execute([ 'scrapy' , 'crawl' , XXX ]) XXX是你自己的爬虫的名字（不是项目的名字）

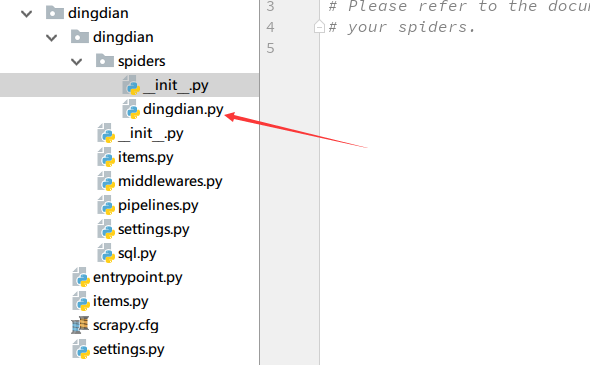
爬虫的名字要定义在自己写的Spider类的成员name里面

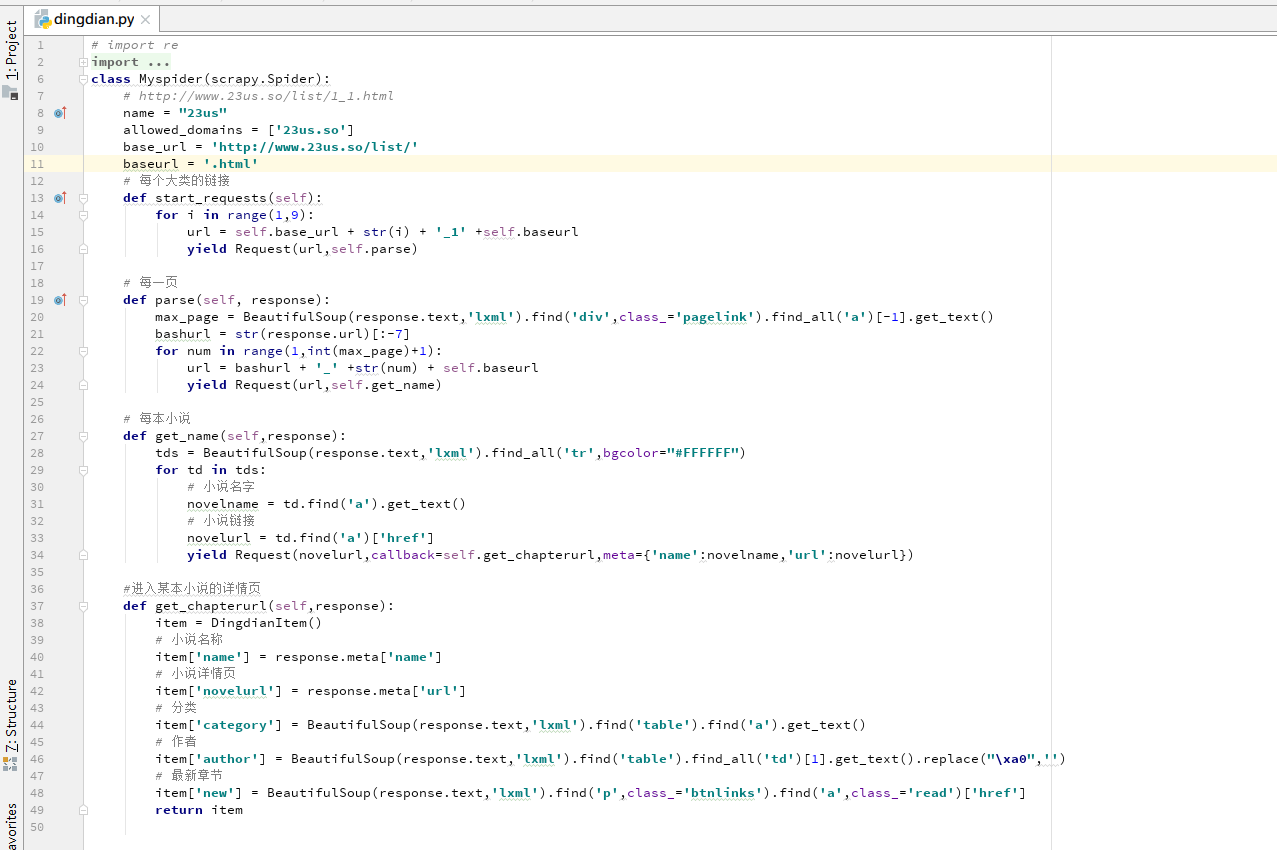
1. 在items.py定义要爬取的字段



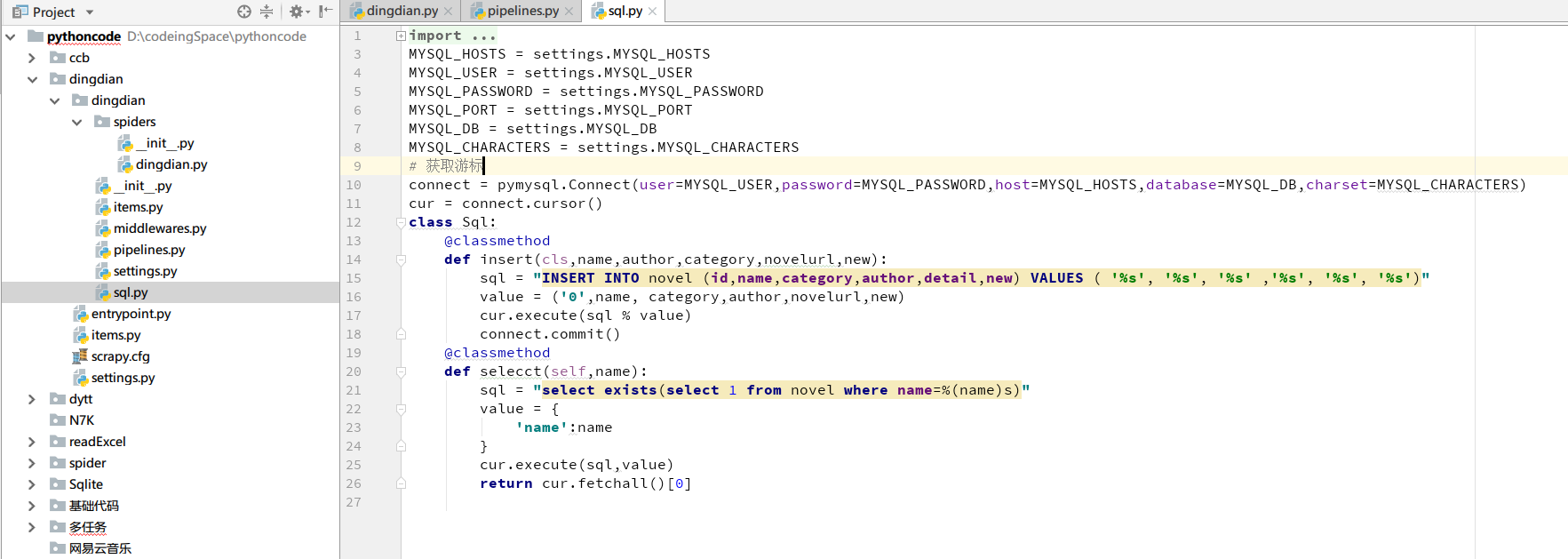
1. spider文件中编写自己的爬虫

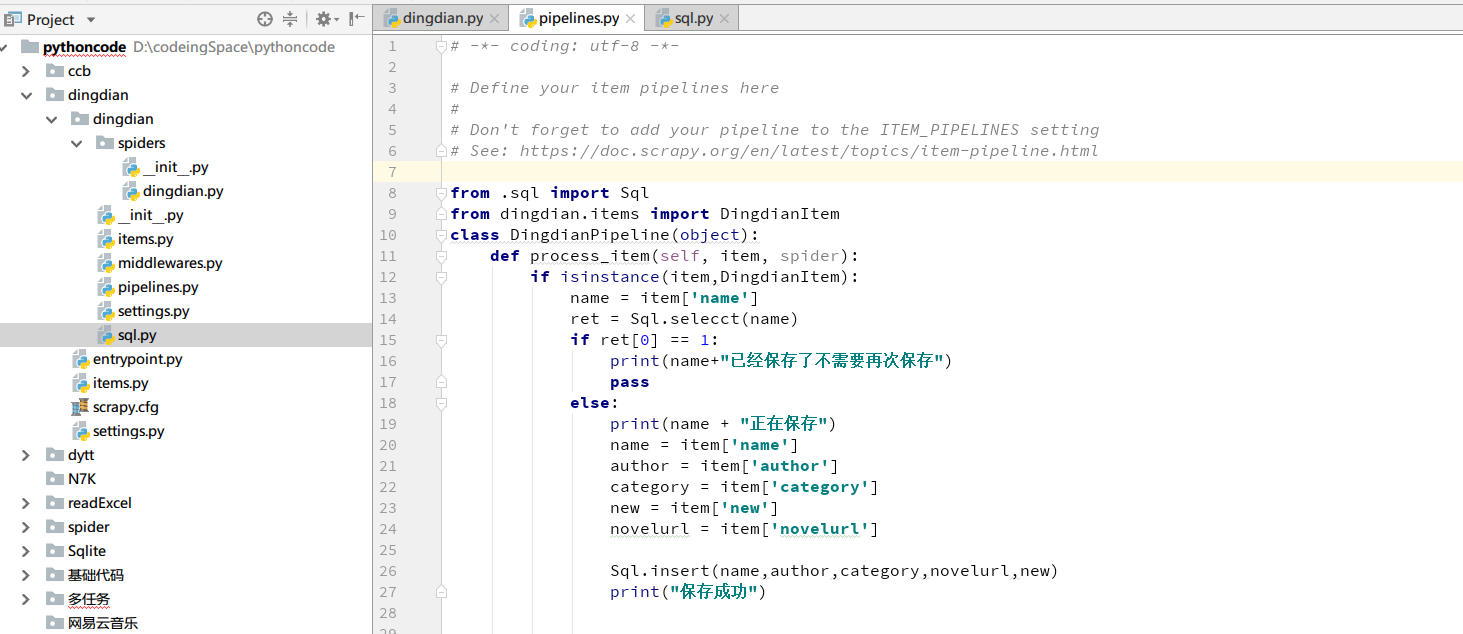
这里我使用的BeautifulSoup分析库，但是这个类库的缺点是满，可以使用Lxml，它是基于ElementTree



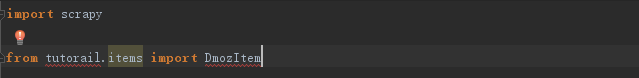


1. pipelines中定义保存数据，在sql中定义插入数据库的方法





在实验的过程中可能会出现如下问题



就是导入包不成功，这时候要去清理缓存，并重新指定项目的根目录