**python编码规范：**

**基本要求**

主要考虑到照顾python2与三之间的迁徙以及一些pep8的编码规范：

1. print函数需要添加括号，照顾python3中的编码需要；
2. 不要将自己写的类、模块、方法、变量的名字写成和python自带的模块、类一样；
3. 所有代码都需要添加上#\_\*\_coding:utf-8\_\*\_，因为在python2如果出现中文会需要；
4. 注意整数的除法，2与3之间有细微的差别；
5. 编码缩进时尽量使用tab，或者4个空格键，二选一。一旦确定，就尽量使用一种，而不要混合使用两种；
6. 每行最大长度79（不要太长就行），换行可以使用反斜杠，最好使用圆括号。换行点要在操作符的后边敲回车；
7. 类和top-level函数定义之间空两行；类中的方法定义之间空一行；函数内逻辑无关段落之间空一行；其他地方尽量不要再空行。
8. 模块内容的顺序：模块说明和docstring(文档描述，比如这个爬虫的特殊之处（反扒，验证）)—import—globals&constants—其他定义。其中import部分，又按标准、三方和自己编写顺序依次排放，之间空一行。
9. 不要在一句import中多个库，比如import os, sys不推荐。

**个人小建议：**

1. 单独添加了一个setdefault模块，会自动设置每个item的默认值，也就不用再在spider中初始化为none；
2. 在spider中提取数据时，目前大家都是用xpath或者css之后extract（），之后再做进一步处理，如果这样写的话建议添加上try…except部分，python字符处理是个容易出的地方；
3. 推荐使用itemloader，在后续的数据处理中使用自定义函数，函数尽量定义在这个方法的内部开始位置，命名规则尽量时deal\_content,或者deal\_title,deal\_publish\_time这样的名字，自己写了一个itemloader,在itemloader\_ll.py中，添加了add\_to\_riginal()函数，因为add\_value跟官方文档解释的不一样，文档解释说是apeend新的值，代码却是覆盖原来的值，添加的add\_to\_original相当于补全官方文档中对add\_value描述的功能，名字不要吐槽，大家可以自己换一个名，也可以添加一些自己的功能；
4. 在访问下一页的时候，如果下一页不是需要**重新设置**访问信息的话，比如重新设定cookie，user-agent，或者其它字段，建议使用yield response.follow()方法而不是yield scrapy.Request()因为这样会自动补全一些request的headers中的字段，比如refer。
5. 关于spider类的内部方法：建议保留def parse方法，尽量不要在这个方法中做数据处理，可以写成一个打印异常的函数，在yield request中，尽量写好callback，因为parse是spider的默认的调用方法，如果意外的request，会自动的到这个parse中。以免出现因response找不到callback而导致项目停下来；
6. Spider中的函数部分，建议统一函数名，比如deal\_index,deal\_content,deal\_others(如果有的话)。建议在start\_requests()函数中发起请求，因为不太能网站要求的请求头不一样，建议将各个网站的headers写在对应的spider函数中，而不统一在middleware中再添加；
7. 在代码中变量名含义应尽量明确，避免使用不同变量之间区别只有只有数字的这些命名方式，如i1,i2,j1,j2等，考虑到实际情况可能比较复杂，不建议规范死了，所以大家注意一下，因为爬虫多了，碰到个别网站改版的情况会大些，规范化有助于将来维护；
8. 最终部署的时候，因尽可能将spider中的print语句注释掉。貌似打印太多，cmd缓存空间也是有一定大小的，过了量会造成程序不稳定。

**一些不太重要的建议：**

1. 关于url的处理尽量使用标准库w3lib来处理，避免使用字符串操作，因为python2中字符串操作比较容易出错。其它字符串操作的地方应考虑是否添加try…except…。
2. 在scrapy.Filed()中有input\_processor和output\_processor这两个参数，分别处理输入和输出，因为itemloader得到的是list，如果想取一个值的还，可以使用Takefirest（），如publish\_time等，但publisher建议保留list结构，因为有些文章作者不止一个。Field()参数里边可以添加scrapy.loader.processor中的函数，也可以参照里边函数的写法写一些自定义的函数。