**简介**

scrapyd是scrapy官方出品的scrapy项目调度工具，实现通过简单的web api调度scrapy项目

**安装**

$ pip install scrapyd

**启动**

安装scrapyd，安装完成后，在控制台输入scrapyd，服务便启动成功，随后，访问localhost:6800便可以访问scrapy的主界面

**发布**

$ pip install scrapyd-client

安装scrapyd-client 用于发布你的scrapy项目至scrapyd调度，即scrapyd在进行爬虫调度的时候需要知道调度哪一个scrapy项目

**发布scrapy项目至scrapyd**

$ cd到你要发布的scrapy项目目录

$ python D:\Anaconda3\Scripts\scrapyd-deploy [scrapyd-deploy在anaconda中的默认的安装地址] -p mockdata [项目名称] --version r01 [版本号]

这里可能会报错 unknown target

**修改**scrapy项目中的scrappy.cfg文件，去掉在[deploy]下url以及project的注释，url即为命令行中的 target, project即为命令行中的 -p projectname

[deploy]

url = <http://localhost:6800/> # 去掉注释

project = mockdata

修改过后可以cd到项目目录通过以下语句发布项目

$ python D:\Anaconda3\Scripts\scrapyd-deploy [scrapyd-deploy在anaconda中的默认的安装地址] --version r01 [版本号]

可以看出，每次都需要输入scrapyd-deploy在anaconda中的默认的安装地址非常麻烦，在D:\Anaconda3\Scripts\下

**新增**scrapyd-deploy.bat文件，文件内容如下

@echo off

"D:\Anaconda3\python.exe" "D:\Anaconda3\Scripts\scrapyd-deploy" %\*

最后我们发布项目只需

$ cd到你要发布的scrapy项目目录

$ scrapyd-deploy --version r01

注意！！！

有些时候我们可能会手残直接scrapyd-deploy没有加版本号，如果是这样scrapyd-deploy会默认会以当前的时间戳作为版本号，但是这样做时，在启动项目的时候 scrapyd会使用 < 符号去比较版本大小，因此当你某个版本加了字符串版本号，比如r01之类的

就会报错 '<' not supported between instances of 'str' and 'int'

这就是因为scrapyd直接使用了 < 去比较版本号，字符串类型不能跟int类型直接比较

**因此我们在发布版本的时候**

**要加版本号就一直加（最好加数字版本号不要加字符串的）**

**不加就一直不加、就默认按照时间戳更新版本**