玩转Raspberry Pi 4之抓取美女图

Raspberry Pi 4终于到手了。折腾开始！

今天我们折腾宅男最爱的美女图爬虫！

1. 系统配置

1.1 安装scrapy

Raspberry Pi 4自带Python 2.7.16和Python 3.7.3，我们在Python3环境中用scrapy做抓取美女图的爬虫。

1.1.1 安装scrapy

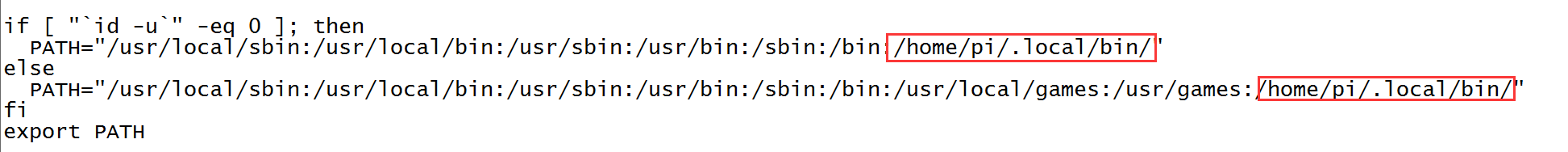
pip3 install scrapy

1.1.2 设置PATH

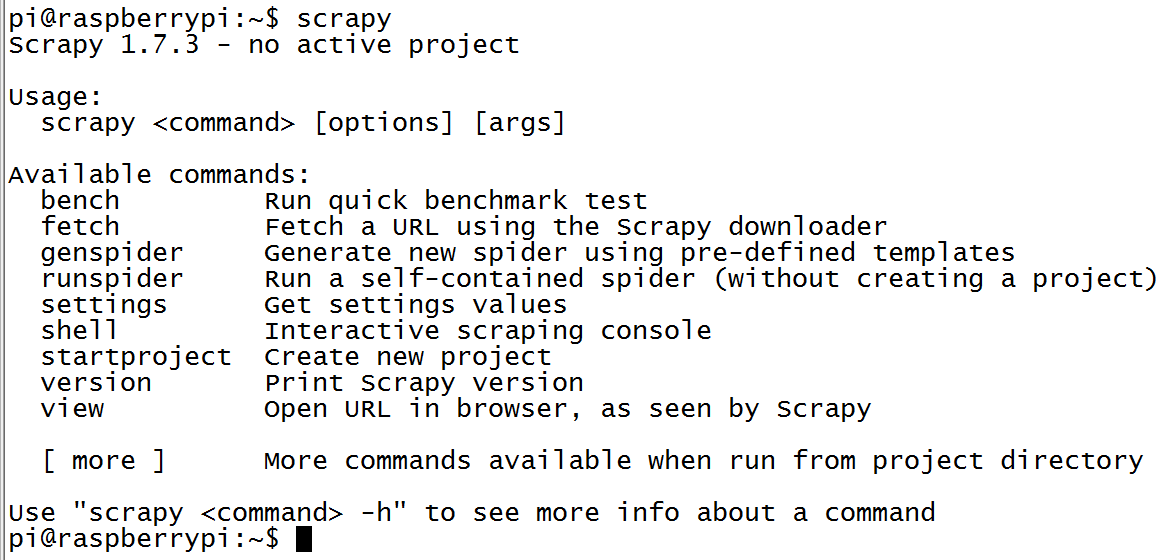
在命令行中输入scrapy命令，如果提示没有该命令，则手动把scrapy的安装目录加入到PATH中。

sudo vim /etc/profile

把scrapy的安装目录/home/pi/.local/bin/加入到PATH中。



重新启动系统后，再尝试在命令行中输入scrapy命令，OK!



1.2 设置U盘

Raspberry Pi 4支持U盘，我们把抓取的美女图存储到U盘上。

1.2.1 插入U盘，查看U盘的设备名

ls /dev/hd\*

会看到/dev目录下有两个设备，/dev/sda和/dev/sda1，/dev/sda1就是U盘

1.2.2 挂载U盘

在/home/pi目录下，创建U盘的挂载目录，并用mount命令挂载。

mkdir usbdisk

sudo mount -t vfat -o uid=pi,gid=pi -o utf8 /dev/sda1 /home/pi/usbdisk

注：

1) 我的U盘是fat32格式，参数为-t vfat，如果是ntfs格式的U盘，参数改为-t ntfs-3g

2) 下载图片时创建的文件夹名字中有中文，需要加上-o utf8参数

1.2.3 卸载U盘

U盘访问结束后，不能直接拔出，需要先umount

sudo umount /home/pi/usbdisk

2. 编写scrapy爬虫

系统配置完成后，我们开始编写爬虫抓取www.mzitu.com网站上的美女图片。

2.1 新建爬虫

2.1.1创建工程

scrapy startproject scrapy\_beauty

2.1.2创建爬虫

scrapy genspider mzitu [www.mzitu.com](http://www.mzitu.com)

2.1.3 去除setting.py中把ITEM\_PIPELINES开关注释

2.1.4 图片类网站抓取图片太快的话，会出现服务器访问错误，因此我们设置一秒钟发送一个请求，在setting.py中修改DOWNLOAD\_DELAY = 1

2.2 分析网站

打开主页，网站上的美女套图有233页。用chrome浏览器的“开发者工具”查看网页代码，发现这233页的网页URL非常有规律，如下图所示：



因此，我们可以构建scrapy爬虫的start\_urls，不需要从网页中回去URL:

begin = 1 #起始页

end = 233 #结束页

start\_urls = ['https://www.mzitu.com/page/{cnt}'.format(cnt=cnt) for cnt in range(begin, end)]

页面中的所有美女套图的列表如下图所示：



我们获取该页面上所有的li标签中的美女套图页面URL，用list存储，并循环访问美女套图的页面，封装parse\_piclist函数处理美女套图页面。

urls = []

postlist = response.xpath('//div[@class="postlist"]')

for li in postlist.xpath('./ul/li'):

urls.append(li.xpath('./a/@href').extract\_first())

pic\_urls = set(urls) #避免存在重复的URL

for url in pic\_urls:

yield scrapy.Request(url, callback=self.parse\_piclist, headers=self.headers)

进一步分析美女套图页面的HTML源码，我们发现每套美女图的URL也是有规律的。



美女套图中不同图片的URL都是https://www.mzitu.com/196214后面附带图片的编号，因此我们可以获取套图中最后一个图片的数字，来知道这个套图有多少张图片，用list来生成每个图片URL，并循环去获取整套图片资源，封装parse\_pic函数来处理套图里面的每一张图片，代码如下：

def parse\_piclist(self, response):

pagenavi = response.xpath('//div[@class="pagenavi"]')

atags = pagenavi.xpath('./a')

picnum = atags[-2].xpath('./span/text()').extract\_first()

last = atags[-2].xpath('./@href').extract\_first().rindex('/')

baseurl = atags[-2].xpath('./@href').extract\_first()[:last]

urls = [ baseurl + '/' + str(i) for i in range(0, (int(picnum) + 1)) ]

for url in urls:

yield scrapy.Request(url, callback=self.parse\_pic, headers=self.headers)

我们继续分析美女套图页面的HTML源码，可以看到<div class = "main-image">中有每张图片的具体信息，包括获取套图的名字和图片的真实URL，代码如下：

picinfo = response.xpath('//div[@class="main-image"]/p/a/img')

foldname = picinfo.xpath('./@alt').extract\_first()

picurl = picinfo.xpath('./@src').extract\_first()

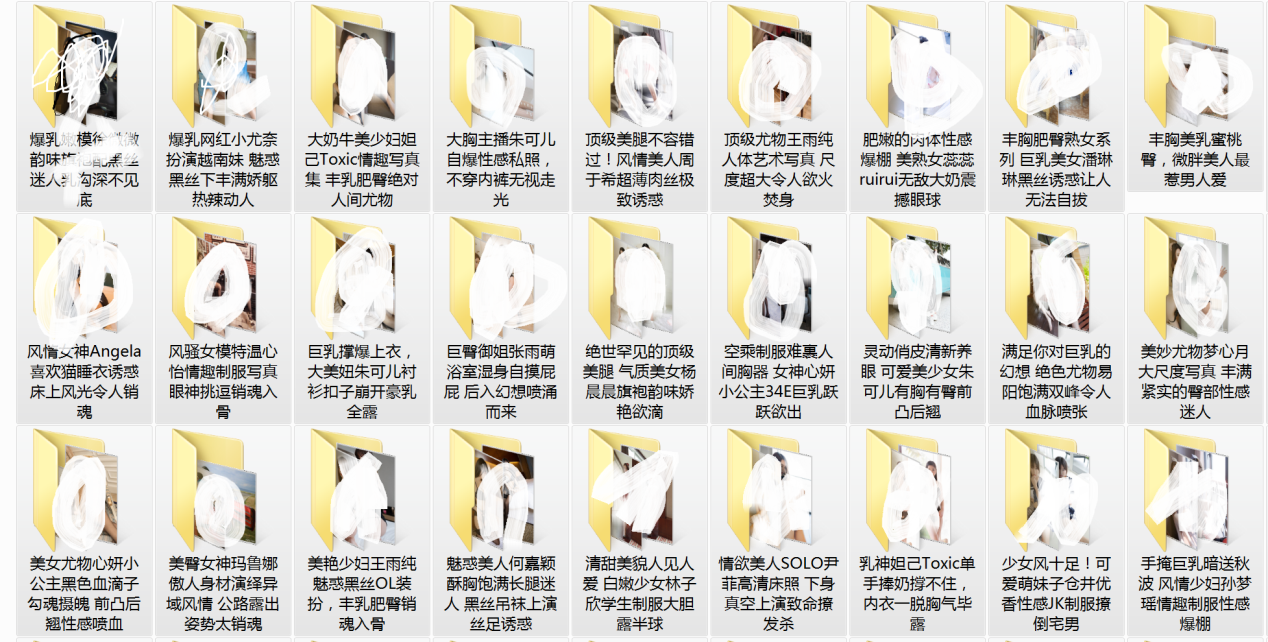
有了真实的URL，我们就可以用封装的save\_pic函数直接去下载啦！下载时自动存储在挂载的U盘的pic目录中。

3. 运行爬虫

在命令行中，运行scrapy crawl mzitu, 爬虫就开始工作啦！

记住，爬虫结束后,拔出U盘时，一定要先卸载U盘。

爬虫的战果：



4. 小结

爬虫代码的编写，没有固定的套路，需要仔细分析所爬取的网站的HTML源码，找到最便捷的抓取方法！我们的目的是抓数据，不是写爬虫代码！

美女图爬虫代码获取请访问：<https://github.com/dangelzjj/scrapy_beauty>