pycurl的使用方法

pycurl是curl的一个python版本。

pycurl的使用说明：

pycurl的使用主要是一些参数的设定。

1，c.setopt(pycurl.URL,myurl)

设定链接的地址

2,c.setopt(pycurl.HTTPHEADER,['Content-Type:application/json','Content-Length: '+str(len(remove\_str))])

设置http的包头信息。注意，长度的字符传是用于put或者post等方法传参数的。

3，c.setopt(pycurl.CUSTOMREQUEST,"DELETE")

设置封装方法，有put，post，get，delete等多种方法

4， c.setopt(pycurl.POSTFIELDS,remove\_str)

设置psot过去的数据，注意是一个字典样式的字符串

5，c.setopt(pycurl.WRITEFUNCTION,b.write)

c.setopt(pycurl.FOLLOWLOCATION, 1)

设置写的回调，所有输出都定向到b.write中。

6，c.setopt(pycurl.MAXDEDIRS,5)

设置重定向次数

7，c.setopt(pycurl.CONNECTTIMEOUT,60)

c.setopt(pycurl.TIMEOUT,600)

设置链接超时，设置下载超时

8，c.setopt(pycurl.USERAGENT,"xxxx")

设置代理浏览器

9，c.setopt(pycurl.HEADER,1)

开启包头输出

c.setopt(pycurl.HEADERFUNCTION,header\_str.write)

将包头输出到header\_str.write流中

10，c.perform()

执行curl命令

11，print b.getvalue()打印消息

12，print c.getinfo(c.HTTP\_CODE) //答应返回值

Print c.getinfo(c.CONTENT\_TYPE) //打印文本类型

Print c.getinfo(c.EFFECTIVE\_URL) //打印重定向URL

具体代码：

1. Import  pycurl
2. Import StringIO
4. checkurl="www.test.com/abc?afgf=afd
5. "
6. b=StringIO.StringIO()
7. c=pycurl.Curl()
8. c.setopt(pycurl.URL, checkurl)
9. c.setopt(pycurl.HTTPHEADER, ["Accept:"])
10. c.setopt(pycurl.WRITEFUNCTION, b.write)
11. c.setopt(pycurl.FOLLOWLOCATION, 1)
12. c.setopt(pycurl.MAXREDIRS, 5)
13. c.perform()
14. Print b.getvalue()
15. Print c.getinfo(c.HTTP\_CODE)
16. b.close()
17. c.close()

传参：

1. b = StringIO.StringIO()
2. c = pycurl.Curl()
3. mkdir\_str = '[{"op":"MKDIRS","permission"=permission}]'
4. mkdir\_url="http://192.168.0.112/abdf?op=MKDIRS&permission=%s" % (self.url\_path,path,permission)
5. c.setopt(pycurl.URL, mkdir\_url)
6. c.setopt(pycurl.HTTPHEADER,['Content-Type:application/json','Content-Length: '+str(len(mkdir\_str))])
7. c.setopt(pycurl.CUSTOMREQUEST,"PUT")
8. c.setopt(pycurl.POSTFIELDS,mkdir\_str)
10. c.setopt(pycurl.WRITEFUNCTION, b.write)
11. c.setopt(pycurl.FOLLOWLOCATION, 1)
12. c.setopt(pycurl.MAXREDIRS, 5)
13. c.perform()
14. status = c.getinfo(c.HTTP\_CODE)
15. bbody = b.getvalue()
16. b.close()