# Python3中使用builtwith的方法

1. 首先通过pip install builtwith安装builtwith

|  |
| --- |
| C:\Users\Administrator>pip install builtwith  Collecting builtwith  Downloading builtwith-1.3.2.tar.gz  Installing collected packages: builtwith  Running setup.py install for builtwith ... done  Successfully installed builtwith-1.3.2 |

1. 新建test.py并输入一下测试代码

|  |
| --- |
| import builtwith  tech\_used = builtwith.parse('http://www.baidu.com')  print(tech\_used) |

运行得到下面的错误：

|  |
| --- |
| Traceback (most recent call last):  File "test.py", line 4, in <module>  import builtwith  File "C:\Users\door\AppData\Local\Programs\Python\Python36-32\lib\site-packages\builtwith\\_\_init\_\_.py", line 43  except Exception, e:  ^  SyntaxError: invalid syntax |

原因是builtwith是基于2.x版本的，需要修改几个地方，在pycharm出错信息中双击出错文件，进行修改，主要修改下面三种：  
1. Python2中的 “Exception ,e”的写法已经不支持，需要修改为“Exception as e”。  
2. Python2中print后的表达式在Python3中都需要用括号括起来。  
3. builtwith中使用的是Python2中的urllib2工具包，这个工具包在Python3中是不存在的，需要修改urllib2相关的代码。  
1和2容易修改，下面主要针对第3点进行修改：  
首先将import urllib2替换为下面的代码：

|  |
| --- |
| import urllib.request  import urllib.error |

然后将urllib2的相关方法替换如下：

|  |
| --- |
| request = urllib.request.Request(url, None, {'User-Agent': user\_agent})  response = urllib.request.urlopen(request) |

再次运行项目，遇到下面错误：

|  |
| --- |
| Traceback (most recent call last):  File "test.py", line 6, in <module>  tech\_used = builtwith.parse('http://www.baidu.com')  File "C:\Users\door\AppData\Local\Programs\Python\Python36-32\lib\site-packages\builtwith\\_\_init\_\_.py", line 61, in builtwith  if contains(html, snippet):  File "C:\Users\door\AppData\Local\Programs\Python\Python36-32\lib\site-packages\builtwith\\_\_init\_\_.py", line 104, in contains  return re.compile(regex.split('\\;')[0], flags=re.IGNORECASE).search(v)  TypeError: cannot use a string pattern on a bytes-like object |

这是因为urllib返回的数据格式已经发生了改变，需要进行转码，将下面的代码：

|  |
| --- |
| if html is None:  html = response.read() |

修改为：

|  |
| --- |
| if html is None:  html = response.read()  html = html.decode('utf-8') |

再次运行得到最终结果如下：

|  |
| --- |
| {'javascript-frameworks': ['jQuery']} |

但是如果把网站换成 'www.163.com'，运行再次报错如下：

|  |
| --- |
| Error: 'utf-8' codec can't decode byte 0xcd in position 565: invalid continuation byte  Traceback (most recent call last):  File "test.py", line 6, in <module>  tech\_used = builtwith.parse('http://www.163.com')  File "C:\Users\door\AppData\Local\Programs\Python\Python36-32\lib\site-packages\builtwith\\_\_init\_\_.py", line 62, in builtwith  if contains(html, snippet):  File "C:\Users\door\AppData\Local\Programs\Python\Python36-32\lib\site-packages\builtwith\\_\_init\_\_.py", line 105, in contains  return re.compile(regex.split('\\;')[0], flags=re.IGNORECASE).search(v)  TypeError: cannot use a string pattern on a bytes-like object |

所以不同的网站需要用不同的解码方式，所以我们需要安装一个叫chardet的工具包，如下：

|  |
| --- |
| C:\Users\Administrator>pip install chardet  Collecting chardet    Downloading chardet-2.3.0-py2.py3-none-any.whl (180kB)     100% |████████████████████████████████| 184kB 616kB/s  Installing collected packages: chardet  Successfully installed chardet-2.3.0 |

先将html = html.decode('utf-8')注释掉，再将builtwith对应的代码做下面修改：

|  |
| --- |
| encode\_type = chardet.detect(html)  if encode\_type['encoding'] == 'utf-8':  html = html.decode('utf-8')  else:  html = html.decode('gbk') |

最后记得import chardet！

注：上面的步骤做完，运行test.py，现在两个网站都能识别了。但是这种做法究竟是否有误，还有待求证。另外，为了方便查看我将修改的\_\_init\_\_.py拷贝到目录下，并改名为[modification](C:/Users/door/AppData/Local/youdao/DictBeta/Application/7.1.0.0421/resultui/dict/?keyword=modification).py。