

项目目标

request库和BeautifulSoup模块获取豆瓣民谣音乐数据，并用Tkinter建立图形化显示界面。

豆瓣民谣音乐



署前街少年

赵雷 Lei Zhao / 2022-08-29 / 专辑 / 数字



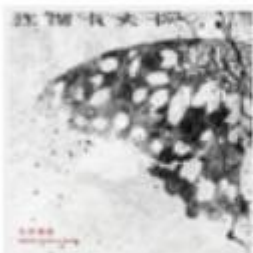
後會有期

白水 / 2022-11-05 / CD / 民谣



瓦尔登湖

周云蓬 / 2022-07-11 / 专辑 / CD / 民谣



江湖卡夫卡

生祥乐队 Sheng Xiang & Band / 2022-10-20



白银饭店

张玮玮, 郭龙 / 2012-05-26 / 专辑 / CD / 民谣

获取民谣网页

进入豆瓣音乐，选择分类浏览民谣：

公众号：黑猫编程

网址：<https://noi.hioier.co>



豆瓣音乐标签: 民谣

综合排序 / 评分排序 / 日期排序 / 标注次数排序



署前街少年 Teen on Shuqian Street

赵雷 Lei Zhao / 2022-08-29 / 专辑 / 数字(Digital) / 民谣

★★★★★ 8.7 (7083人评价)



後會有期 The Long Goodbye

白水 / 2022-11-05 / CD / 民谣

★★★★★ 8.1 (505人评价)



瓦尔登湖

周云蓬 / 2022-07-11 / 专辑 / CD / 民谣

★★★★★ 7.7 (692人评价)

```
1 import requests
2
3 headers = {'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; WOW64)
4           AppleWebKit/537.36 (KHTML, like Gecko) Chrome/56.0.2924.87 Safari/537.36'}
5
6 #设置url地址
7 url = 'https://music.douban.com/tag/民谣'
8 #发送网络请求
9 response = requests.get(url, headers=headers)
10 print(response.text)
```

BeautifulSoup解析数据

- BeautifulSoup是一个可以从HTML或XML文件中提取数据的Python库。其功能简单而强大、容错能力高、文档相对完善，清晰易懂。
- 非Python标准模块，需要安装才能使用。

安装方式: `pip install bs4`



BeautifulSoup

BeautifulSoup支持Python标准库中的HTML解析器，还支持一些第三方的解析器，如果不安装第三方解析器，则Python会使用默认解析器。

序号	解析器	使用方法	优点	缺点
1	标准库	BeautifulSoup(html,'html.parser')	内置标准库，速度适中，文档容错能力强	Python3.2版本前的文档容错能力差
2	lxml HTML	BeautifulSoup(html,'lxml')	速度快，文档容错能力强	安装C语言库
3	lxml XML	BeautifulSoup(html,'xml')	速度快，唯一支持XML	安装C语言库
4	html5lib	BeautifulSoup(html,'html5lib')	容错能力最强，可生成HTML5	运行慢，扩展差

BeautifulSoup提取数据的常用方法：

返回值类型	方法	功能	语法	举例
Tag	find()	提取满足要求的首个数据	bs.find(标签,属性)	bs.find('div',class_='books')
Tag	find_all()	提取满足要求的所有数据	bs.find_all(标签,属性)	bs.find_all('div',class_='books')

CSS选择器：

功能	举例
通过ID查找	bs.select('#abc')
通过classa查找	bs.select('.abc')

tag对象：

功能	举例
获取标签	bs.title
获取所有属性	bs.title.attrs
获取单个属性的值	bs.div.get('class') bs.div['class'] bs.a['href']

html文件:

```

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <title>黑猫编程</title>
6  </head>
7  <body>
8      <div class="box">
9          <ul name="list">
10             <li>1</li>
11             <li>2</li>
12             <li>3</li>
13          </ul>
14          <a href="https://noi.hioier.com/" target="_blank">黑猫编程</a>
15          <a href="https://www.hioier.com/">在线测评题库</a>
16      </div>
17
18      <div class="box">
19          <span id="d1">aa</span>
20          <span>bb</span>
21          <span>cc</span>
22      </div>
23 </body>
24 </html>

```

python文件:

```

1  from bs4 import BeautifulSoup
2
3  with open('test02.html', 'r', encoding="utf-8") as f:
4      html = f.read()
5
6  # print(html)
7  # print(type(html))
8  bs = BeautifulSoup(html, 'lxml')
9
10 ret = bs.find("div", class_="box")
11 print(ret)
12
13 ret = bs.findAll("div", class_="box")
14 print(ret)
15 print(len(ret))

```

```

16
17 ret = bs.find("ul", attrs={"name":"list"})
18 print(ret)
19
20 ret = bs.select("#d1")
21 print(ret)
22
23 ret = bs.select(".box")
24 print(ret)
25
26 ret = bs.title
27 print(ret)
28
29 ret = bs.div.attrs
30 print(ret)

```

获取音乐详细信息

```

1 bs = BeautifulSoup(response.text, "html.parser")
2
3 listResult = bs.select("tr.item", limit=5)
4
5 resultList = []
6
7 for result in listResult:
8
9     resultDict = {}
10
11     resultDict['musicImage'] = result.select('img')[0]['src']
12     resultDict['musicName'] = result.select('a')[0]['title'].split('-')
13     resultDict['musicInfo'] = result.select("p.pl")[0].get_text()
14
15     resultList.append(resultDict)
16
17 print(resultList)

```

完整项目

```

1 w = tk.Tk()
2 w.geometry('400x510')
3
4 w.resizable(0, 0)
5
6 w.title("黑猫音乐爬虫")
7
8 for i in range(5):
9
10     # 创建Frame组件
11     frame = tk.Frame(w, width=400, height=100, bg='white')
12     frame.place(relx=0, rely=i*0.2)
13
14     # 显示音乐封面
15     imgResponse = requests.get(resultList[i]['musicImage'])
16     image = Image.open(BytesIO(imgResponse.content))

```

公众号：黑猫编程

网址：<https://noi.hioqier.co>

```
17
18     photoImage = ImageTk.PhotoImage(image.resize((81, 81)))
19
20     labelImg = tk.Label(frame, image=photoImage, width=81, height=81)
21     labelImg.place(relx=0.04, rely=0.05)
22     labelImg.image = photoImage
23
24     # 显示音乐
25     labelName = tk.Label(frame, text=resultList[i]['musicName'], bg='white',
26 fg='#3377AA', font=("./font/simhei.ttf", 11))
27     labelName.place(relx=0.3, rely=0.1)
28
29     # 显示音乐信息
30     labelInfo = tk.Label(frame, text=resultList[i]['musicInfo'], bg='white',
31 fg='#666666', font=("./font/simhei.ttf", 9))
32     labelInfo.place(relx=0.3, rely=0.6)
33
34 w.mainloop()
```