**Python3 爬虫之网易云音乐下载**

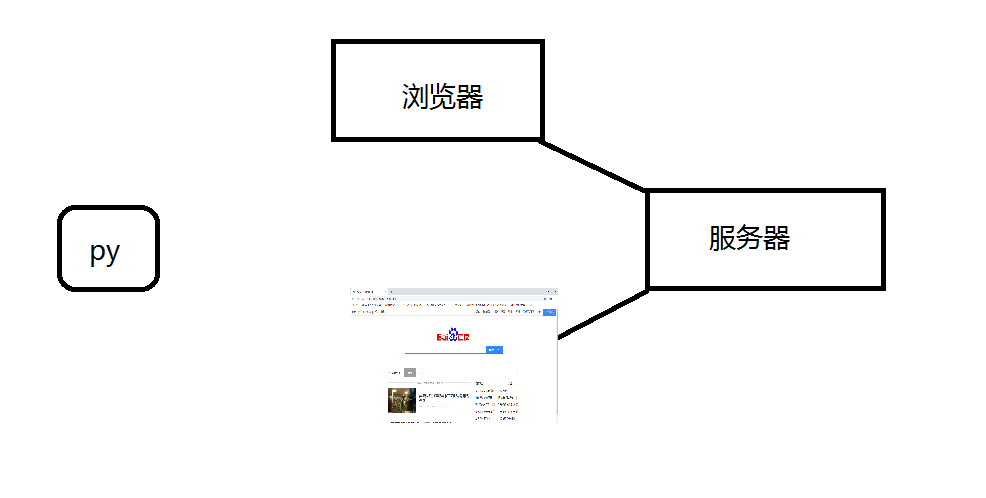
**本节所讲内容：**

1. **爬虫的简单概述**
2. **必要模块&简单网站分析**

**实战1：根据ID爬取单个歌曲下载，  
实战2：根据利用进程爬取网银云音乐热门歌单排行榜，下载到本地**

1. **爬虫的简单概述**

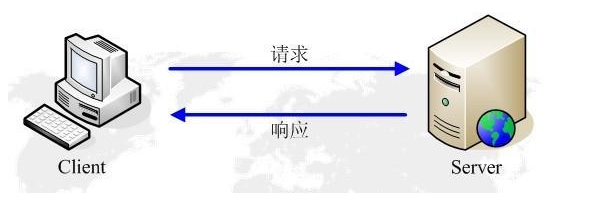
**爬虫（spider：网络蜘蛛）:是一个用脚本代替浏览器请求服务器获取服务器资源的程序。**



1. **必要模块&网站分析**
   1. **模块安装**

**requests是python实现的简单易用的HTTP库，使用起来比urllib简洁很多**

**什么是http请求 知道的1 不知道2**



**是指从客户端到**[**服务器**](https://baike.baidu.com/item/%E6%9C%8D%E5%8A%A1%E5%99%A8)**端的请求消息。包括：消息首行中，对资源的请求方法、资源的标识符及使用的协议。**

**因为是第三方库，所以使用前需要安装:pip install requests**

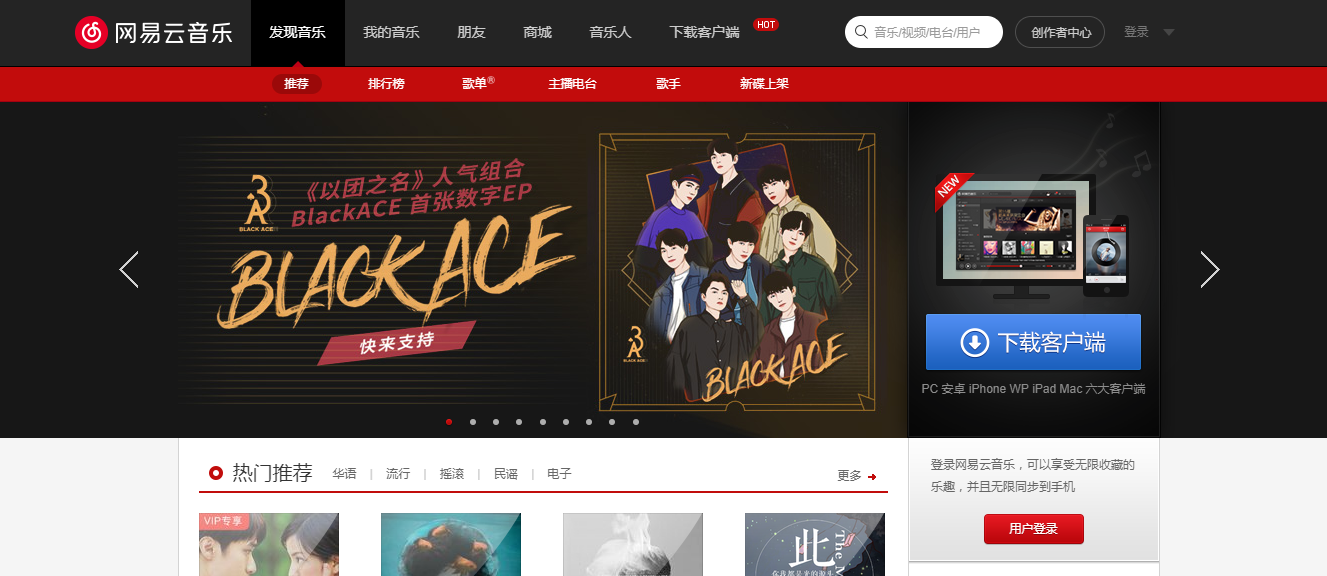
**requests，multiprocessing，re 需要用pip install reqeuests**

**网络请求模块，requests是进行网络访问的最重要的模块**

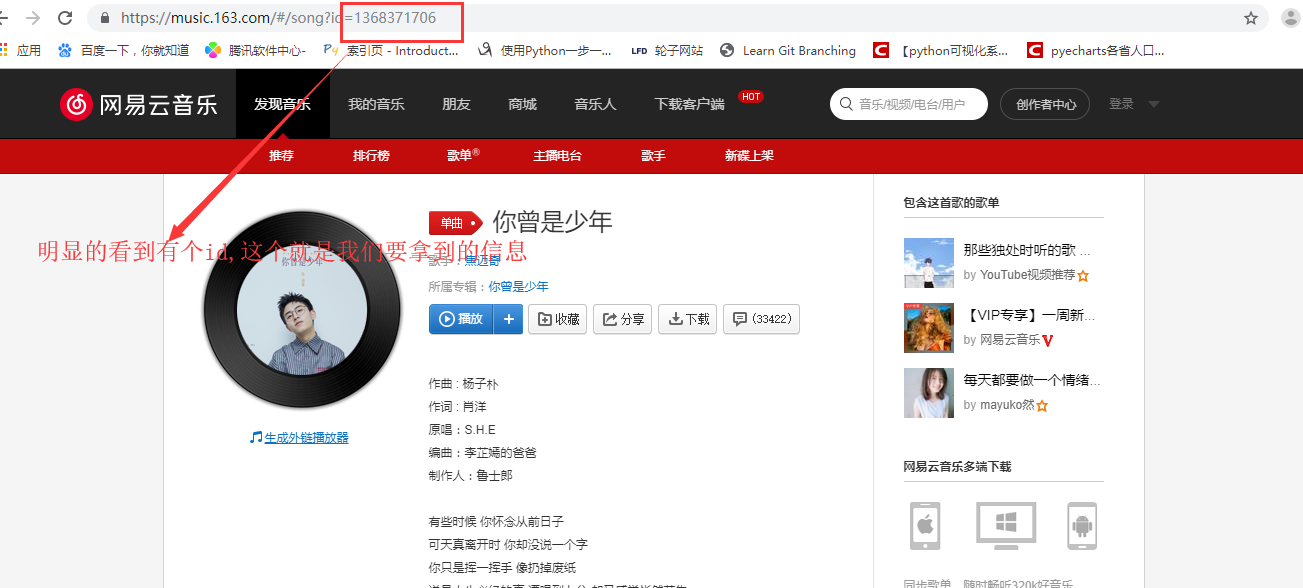
**Multiprocessing 进程,re正则**

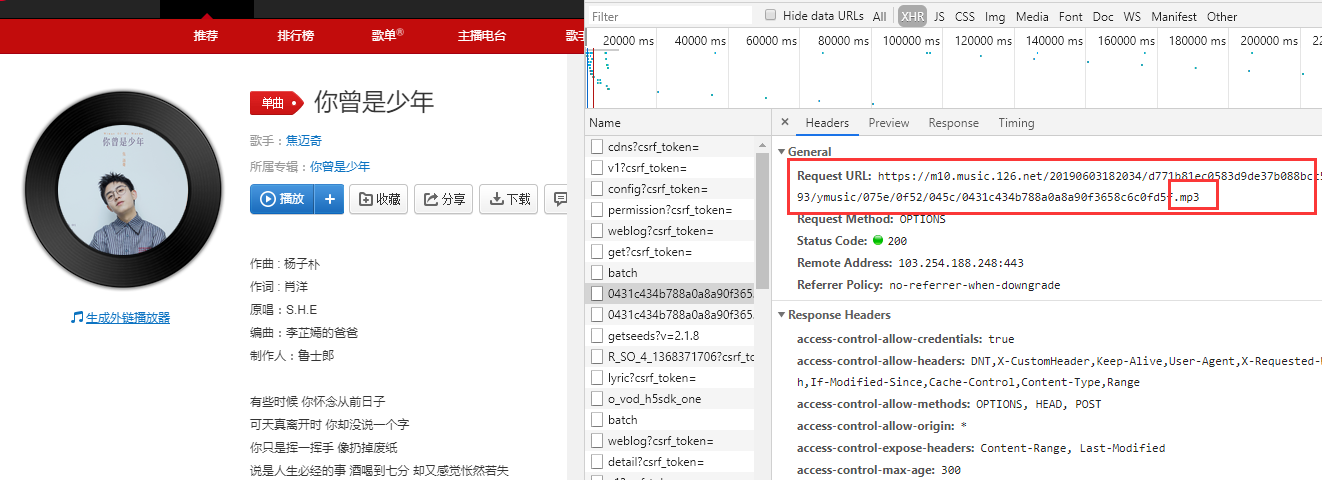
**2.2网站分析**

**页面分析：首先打开网易云音乐(**[**https://music.163.com/#**](https://music.163.com/)**)**

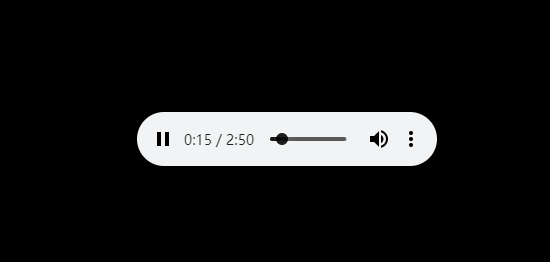


**紧着着我们随便点击一个歌曲如下图所示：**



**我们再次进行分析可明显看到这个request中有MP3** 

**点击该网站获取如下图：**



**进而进行下载。**

**然而，这个网址中的一些数据是加密的，不可能我们每次操作用这种流程，所以我通过Fidder抓包获取到如下网址：http://music.163.com/song/media/outer/url?id=%s" %(歌曲id)**

**该网址就为咱们MP3的网站。**

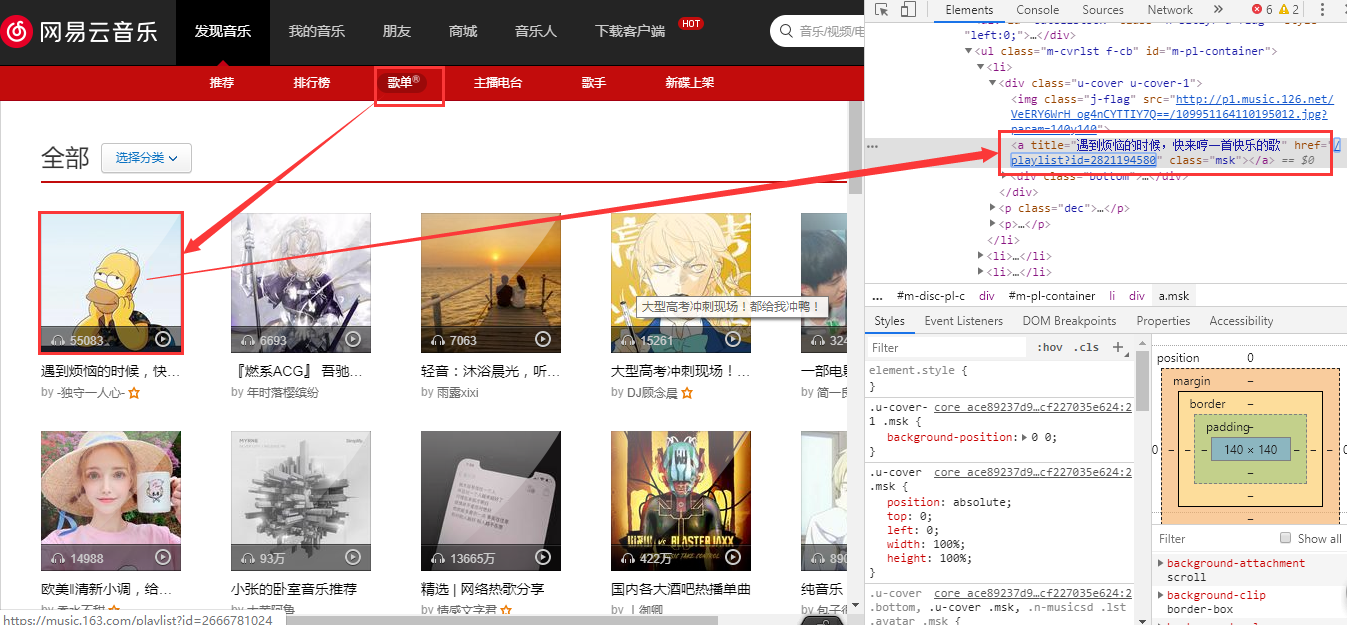
**实战1：网易云音乐单个音乐下载  
import requests  
import urllib.request  
from fake\_useragent import UserAgent  
from bs4 import BeautifulSoup  
from requests.exceptions import RequestException  
*#设置下载提示*downloaded = '0'  
def download\_listener(a, b, c):  
 per = 100.0 \* a \* b / c  
 if per > 100:  
 per = 100  
 new\_downloaded = '%.1f' % per  
 global downloaded  
 if new\_downloaded != downloaded:  
 downloaded = new\_downloaded  
 print ('download %s%% %s/%s' % (downloaded, a \* b, c))  
*#网络请求，进行响应*def getResponse(url,headers):  
 try:  
 response = requests.get(url,headers = headers)  
 if response.status\_code == 200:  
 return response  
 return None  
 except RequestException:  
 return None  
*#获取歌曲名字*def getSongName(songid):  
 headers = {"user-agent":"Mozilla/5.0 (iPhone; CPU iPhone OS 11\_0 like Mac OS X) AppleWebKit/604.1.38 (KHTML, like Gecko) Version/11.0 Mobile/15A372 Safari/604.1",}  
 url = "https://music.163.com/m/song?id={}".format(songid)  
 Text = getResponse(url,headers).text  
 soup = BeautifulSoup(Text,"lxml")  
 title = soup.title.text  
 name,\*\_ = title.split('-')  
 return name.strip()  
*#进行配置*if \_\_name\_\_=='\_\_main\_\_':  
 songid = input("请输入网站中的id:")  
 url = 'http://music.163.com/song/media/outer/url?id={}.mp3'.format(songid)  
 headers = {"user-agent":UserAgent().random}  
 downurl = getResponse(url,headers).url  
 SongName = getSongName(songid)  
 urllib.request.urlretrieve(downurl, SongName + ".mp3",download\_listener)**

**实战2：根据歌单ID实现歌单歌曲批量下载**

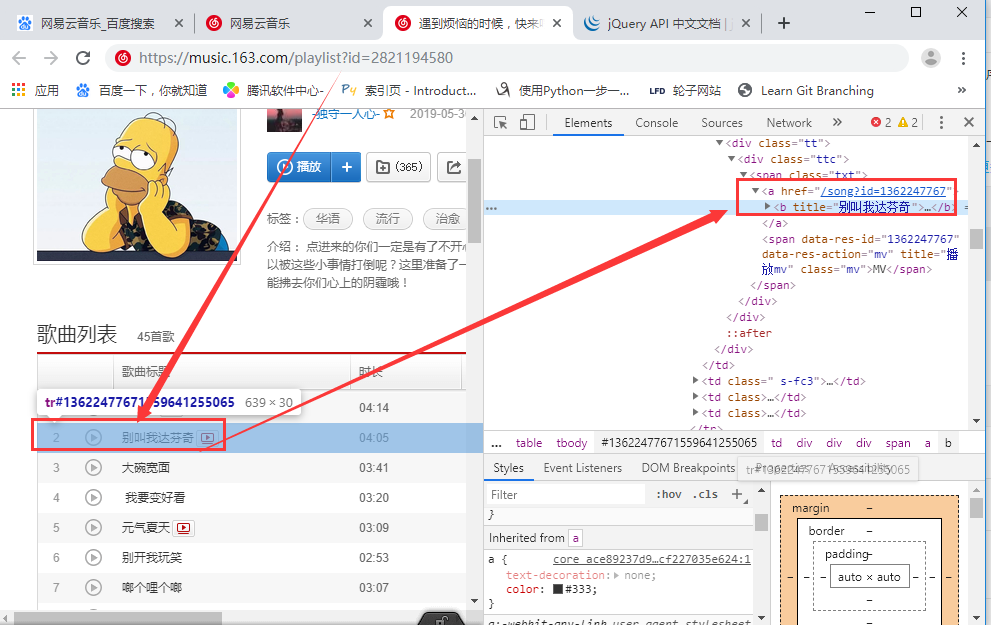
**接下来进行歌单下载，下载最近的hot的歌单列表。进而分析网站：**

**歌单排行版的URL地址如：https://music.163.com/discover/playlist/?order=hot**

**每个歌单的具体URL地址如下：**[**https://music.163.com/playlist?id=2821194580**](https://music.163.com/playlist?id=2821194580)



**每个歌单中的歌曲都有固定的ID如下图所示：**



**现在我们只要在原有下载单个歌曲的基础上开始下载，并访问就好了  
import requests  
import re  
from multiprocessing import Pool  
  
headers = {  
 'Referer': 'https://music.163.com/',  
 "User-Agent": "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/53.0.2785.89 "  
 "Safari/537.36"  
}  
  
def get\_page(url):  
 res = requests.get(url, headers=headers)  
 data = re.findall('<a title="(.\*?)" href="/playlist\?id=(\d+)" class="msk"></a>', res.text)  
 print(data)  
 pool = Pool(processes=4)  
 pool.map(get\_songs, data)  
 print("下载完毕！")  
  
  
def get\_songs(data):  
 playlist\_url = "https://music.163.com/playlist?id=%s" % data[1]  
 print(playlist\_url)  
 res = requests.get(playlist\_url, headers=headers)  
 for i in re.findall(r'<a href="/song\?id=(\d+)">(.\*?)</a>', res.text):  
 download\_url = "http://music.163.com/song/media/outer/url?id=%s" % i[0]  
 print(download\_url)  
 try:  
 with open('./music/' + i[1]+'.mp3', 'wb') as f:  
 f.write(requests.get(download\_url, headers=headers).content)  
 except FileNotFoundError:  
 pass  
 except OSError:  
 pass  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 hot\_url = "https://music.163.com/discover/playlist/?order=hot"  
 get\_page(hot\_url)**

**总结：**

1. **爬虫的简单概述**
2. **必要模块&简单网站分析**

**实战1：根据ID爬取单个歌曲下载，  
实战2：根据利用进程爬取网银云音乐热门歌单排行榜，下载到本地**