import requests # 用于获取网页内容的模块

from bs4 import BeautifulSoup # 用于解析网页源代码的模块

header={ # 伪造浏览器头部，不然获取不到网易云音乐的页面源代码。

'User-Agent':'Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/46.0.2490.80 Safari/537.36',

}

link = "<https://music.163.com/#/discover/toplist?id=3778678>" # 这是网易云音乐热歌榜的链接（其实是嵌套在网页里面含有歌曲数据的页面框架的真实链接）

r = requests.get(link, headers=header) # 通过 requests 模块的 get 方法获取网页数据

html = r.content # 获取网页内容

soup = BeautifulSoup(html, "html.parser") # 通过 BeautifulSoup 模块解析网页，具体请参考官方文档。

songs = soup.find("ul", class\_="f-hide").select("a", limit=10) #通过分析网页源代码发现排行榜中的歌曲信息全部放在类名称为 f-hide 的 ul 中，于是根据特殊的类名称查找相应 ul，然后找到里面的全部 a 标签，限制数量为10，即排行榜的前 10 首歌。

i = 1 # 设置一个自增参数，表示歌曲的数目

for s in songs: # 遍历输出数组 songs 中的内容

song\_id = s['href'][9:] # 只截取歌曲链接中的 ID 部分，因为网页中链接的形式为“/song?id=496870798”，从 = 号之后的就是歌曲的 ID 号。

song\_name = s.text # 获取 a 标签的文本内容，即歌曲的名称。

song\_down\_link = "<http://music.163.com/song/media/outer/url?id=>" + song\_id + ".mp3" # 根据歌曲的 ID 号拼接出下载的链接。歌曲直链获取的方法参考文前的注释部分。

print("第 " + str(i) + " 首歌曲：" + song\_down\_link)

print("正在下载...")

response = requests.get(song\_down\_link, headers=header).content # 亲测必须要加 headers 信息，不然获取不了。

f = open(song\_name + ".mp3", 'wb') # 以二进制的形式写入文件中

f.write(response)

f.close()

print("下载完成.\n\r")

i = i + 1