我们以酷狗top500为例（ <http://www.kugou.com/yy/rank/home/1-8888.html> ），展示音乐信息的整理。



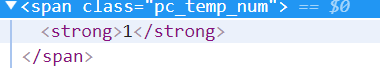
首先，这个网页没有设计（或者说隐藏了）翻页的地方，但我们不难从网址发现，’-8888’前的数字指代页码。

然后，我们明确我们需要抓取的条目有：排名、歌手名、歌名、时长、网址。



找对应元素，然后构造语句这些都不难。需要注意的一点是，在抓取排名时，我们很容易把匹配规则写成<span class="pc\_temp\_num">.\*?</span>就完事了，但是实际抓取下来，前三名的排名会莫名多出strong标签。



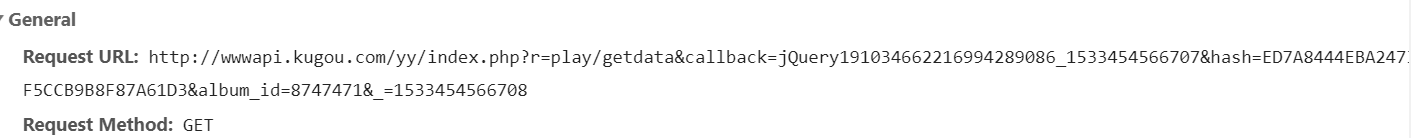




这是因为，前三名明显在样式上有强调的意思，所以会被放在strong标签下。且.\*?是匹配任意字符，这里我们只匹配数字，完全没必要这么写。所以正确写法是：

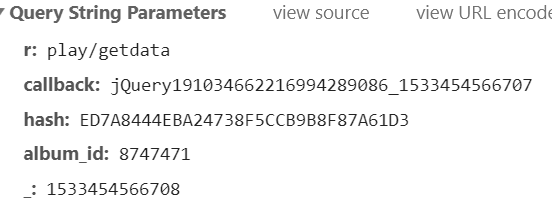
<span class="pc\_temp\_num">.\*?(\d{1,3}).\*?</span>

对于下载地址的获取，我们先随便点开一首歌，在其中，找到这样一个json文件，他会包含下载地址的信息。





然而访问这个json文件需要很多参数。



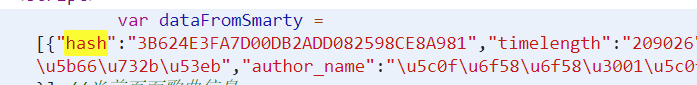
对于这个json文件的网址

<http://wwwapi.kugou.com/yy/index.php?r=play/getdata&callback=jQuery191034662216994289086_1533454566707&hash=ED7A8444EBA24738F5CCB9B8F87A61D3&album_id=8747471&_=1533454566708>

我们试着一步步简化，会发现只保留hash参数时他依然可以正常打开这个json文件

<http://wwwapi.kugou.com/yy/index.php?r=play/getdata&hash=ED7A8444EBA24738F5CCB9B8F87A61D3>

所以，我们需要获取hash参数。我们发现，正好在网页上有这个参数。



于是我们整理获取下载地址的思路：获取播放地址--->获取源码--->获取hash参数--->带参数访问json文件([http://wwwapi.kugou.com/yy/index.php?r=play/getdata&hash={}](http://wwwapi.kugou.com/yy/index.php?r=play/getdata&hash=ED7A8444EBA24738F5CCB9B8F87A61D3) .format(hash))--->从json文件获取下载地址。