

跟我学SQL：(三)使用SQL子选择来合并查询 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/137/2021_2022__E8_B7_9F_E6_88_91_E5_AD_A6S_c98_137964.htm 你是否曾经为了得到所需要的信息而反复查询？子选择，也被称为子查询，也许正是你在寻找的。SQL的这项功能使你可以在一组结果中查询，创造性地给结果组加以限定，或是在向数据库的单一调用中将结果与一个无关系的查询做相关。这篇文章中我将给出几个子选择的例子并就何时使用他们进行讨论。在一个结果组中搜索子选择的理念很简单：一个选择查询安置在另一个查询内部，创建一个在单一声明搜索中不可用的资源。子选择允许查询的合并，结果组比较的责任落到了数据库中而不是应用软件代码中。使用这个功能的一个途径是对两个表格中的可比数据专栏中的值进行定位。例如，我的一个数据库有两个表格，Album和Lyric。我可以很容易地通过下面的子查询声明来找到每一个Metallica的歌曲中包含“justice”的歌名：

```
SELECT song_name FROM Album WHERE band_name = 'Metallica' AND song_name IN (SELECT song_name FROM Lyric WHERE song_lyric LIKE '%justice%')
```

这个例子是很简单的，我从Album表格中选择了所有Metallica的歌曲，接着，我在lyric表格中选择所有包含“justice”的歌曲，最后，我使用IN关键字来从Lyric表格结果组中显示的Album表格中返回歌曲名称。我使用Lyric表格结果组来给Album表格中的结果做限定。WHERE子句中的子选择部分是完全自包含的，因此我不需要使用例如Album.song_name和Lyric.song_name等完整的专栏名称。我没有从最终结果组的Lyric表格中返回任何值

，如果我需要歌曲的Lyric，我会使用一个JOIN声明。使用NOT IN排除结果 你可以使用NOT IN关键字来获得明确地不被包含在另一个结果组中的结果。例如，我想要通过下面的代码来返回Metallica在“ And Justice for All ”专辑中不包含单词“ justice ”的歌曲：
`SELECT song_name FROM Album WHERE album_name = ' And Justice for All ' AND band_name = ' Metallica ' AND song_name NOT IN (SELECT song_name FROM Lyric WHERE song_lyric LIKE '%justice% ')`. 在前面的SQL代码中，我选择了Metallica的“ And Justice for All, ”专辑中的所有歌曲，接着是带有歌词中带有“ justice ”所有歌曲，最后从在Lyric结果组中没有出现的Album结果组返回了所有歌曲。较之于返回两个查询并使用代码来比较数组，你通过一个单独的声明就可以得到确切的结果。 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com