

深入浅出SQL系列教程之SELECT语句中的左连接、右连接和全连接 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/261/2021_2022__E6_B7_B1_E5_85_A5_E6_B5_85_E5_c67_261369.htm 上节我们介绍了表连接，更确切的说是inner joins连接。连接仅选出两张表中互相匹配的记录。因此，这会导致有时我们需要的记录没有包含进来。为更好的理解这个概念，我们介绍两个表作演示。苏格兰议会中的政党表(party)和议员表(msp)。

party(Code,Name,Leader)Code: 政党代码Name: 政党名称

Leader: 政党领袖msp(Name,Party,Constituency)Name: 议员名

Party: 议员所在政党代码Constituency: 选区在介绍左连接、

右连接和全连接前，有一个数据库中重要的概念要介绍一下，即空值(NULL)。有时表中，更确切的说是某些字段值，可能会出现空值，这是因为这个数据不知道是什么值或根本就不存在。空值不等同于字符串中的空格，也不是数字类型的0。

因此，判断某个字段值是否为空值时不能使用=,这些判断符。必需有专用的短语：IS NULL 来选出有空值字段的记录，同理，可用 IS NOT NULL 选出不包含空值的记录。例如：下面的语句选出了没有领导者的政党。（不要奇怪，苏格兰议会中确实存在这样的政党）

```
SELECT code, name FROM party
```

```
WHERE leader IS NULL
```

```
又如：一个议员被开除出党，看看他是谁。（即该议员的政党为空值）
```

```
SELECT name FROM msp
```

```
WHERE party IS NULL
```

好了，让我们言归正传，看看什么叫左连接、右连接和全连接。A left join（左连接）包含所有的左边表中的记录甚至是右边表中没有和它匹配的记录。同理，

也存在着相同道理的 right join（右连接），即包含所有的右

边表中的记录甚至是左边表中没有和它匹配的记录。而full join(全连接)顾名思义，左右表中所有记录都会选出来。讲到这里，有人可能要问，到底什么叫：包含所有的左边表中的记录甚至是右边表中没有和它匹配的记录。Ok,我们来看一个实例：SELECT msp.name, party.name FROM msp JOIN party ON party=code 这个是我们上一节所学的Join(注意：也叫inner join)，这个语句的本意是列出所有议员的名字和他所属政党。你可以在亲自执行一下该语句，看看结果是什么。很遗憾，我们发现该查询的结果少了个议员：Canavan MSP, Dennis。为什么，因为这个议员不属于任和政党，即他们的政党字段(Party)为空值。那么为什么不属于任何政党就查不出来了？这是因为空值在作怪。因为议员表中政党字段(Party)的空值在政党表中找不到对应的记录作匹配，即FROM msp JOIN party ON party=code 没有把该记录连接起来，而是过滤出去了。在该短语中，msp在Join的左边，所有称为左表。party在Join的右边，所有称为右表。Ok,现在再看看这句话，“包含所有的左边表中的记录甚至是右边表中没有和它匹配的记录”，意思应该很明白了吧。执行下面这个语句，那两个没有政党的议员就漏不了了。SELECT msp.name, party.name FROM msp LEFT JOIN party ON party=code关于右连接,看看这个查询就明白了：SELECT msp.name, party.name FROM msp RIGHT JOIN party ON msp.party=party.code这个查询的结果列出所有的议员和政党，包含没有议员的政党，但不包含没有政党的议员。那么既要包含没有议员的政党，又要包含没有政党的议员该怎么办呢，对了，全连接(full join)。SELECT msp.name, party.name FROM msp FULL JOIN party ON

msp.party=party.code 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