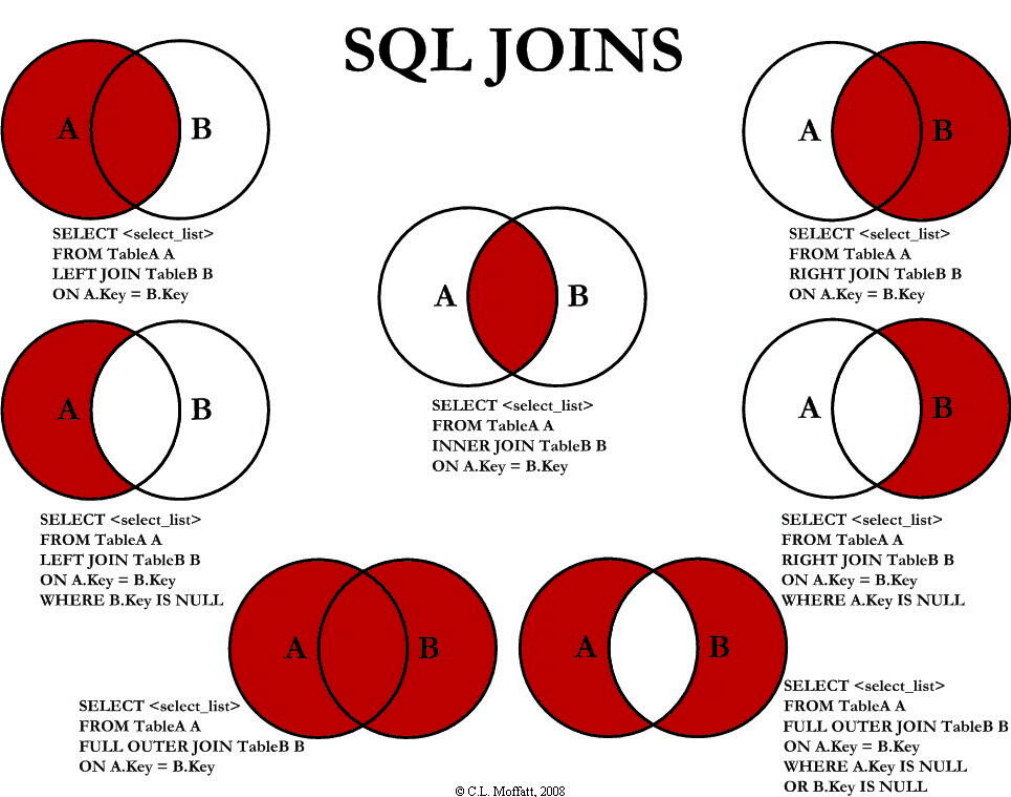


SQL 连接(JOIN)

SQL join 用于把来自两个或多个表的行结合起来。
下图展示了 LEFT JOIN、RIGHT JOIN、INNER JOIN、OUTER JOIN 相关的 7 种用法。



SQL JOIN

SQL JOIN 子句用于把来自两个或多个表的行结合起来，基于这些表之间的共同字段。
最常见的 JOIN 类型：**SQL INNER JOIN (简单的 JOIN)**。SQL INNER JOIN 从多个表中返回满足 JOIN 条件的所有行。

演示数据库

在本教程中，我们将使用 RUNOOB 样本数据库。
下面是选自 "Websites" 表的数据：

id	name	url	alexa	country
1	Google	https://www.google.cm/	1	USA
2	淘宝	https://www.taobao.com/	13	CN
3	菜鸟教程	http://www.runoob.com/	4689	CN
4	微博	http://weibo.com/	20	CN
5	Facebook	https://www.facebook.com/	3	USA

```
| 7 | stackoverflow | http://stackoverflow.com/ | 0 | IND |
+---+-----+-----+-----+-----+-----+
```

下面是 "access_log" 网站访问记录表的数据：

```
mysql> SELECT * FROM access_log;
+---+-----+-----+-----+
| aid | site_id | count | date       |
+---+-----+-----+-----+
| 1   | 1       | 45    | 2016-05-10 |
| 2   | 3       | 100   | 2016-05-13 |
| 3   | 1       | 230   | 2016-05-14 |
| 4   | 2       | 10    | 2016-05-14 |
| 5   | 5       | 205   | 2016-05-14 |
| 6   | 4       | 13    | 2016-05-15 |
| 7   | 3       | 220   | 2016-05-15 |
| 8   | 5       | 545   | 2016-05-16 |
| 9   | 3       | 201   | 2016-05-17 |
+---+-----+-----+-----+
9 rows in set (0.00 sec)
```

请注意，"Websites" 表中的 "id" 列指向 "access_log" 表中的字段 "site_id"。上面这两个表是通过 "site_id" 列联系起来的。

然后，如果我们运行下面的 SQL 语句（包含 INNER JOIN）：

实例

```
SELECT Websites.id, Websites.name, access_log.count, access_log.date
FROM Websites
INNER JOIN access_log
ON Websites.id=access_log.site_id;
```

执行以上 SQL 输出结果如下：

```
mysql> SELECT Websites.id, Websites.name, access_log.count, access_log.date
-> FROM Websites
-> INNER JOIN access_log
-> ON Websites.id=access_log.site_id;
+---+-----+-----+-----+
| id | name       | count | date       |
+---+-----+-----+-----+
| 1   | Google     | 45    | 2016-05-10 |
| 1   | Google     | 230   | 2016-05-14 |
| 2   | 淘宝       | 10    | 2016-05-14 |
| 3   | 菜鸟教程   | 100   | 2016-05-13 |
| 3   | 菜鸟教程   | 220   | 2016-05-15 |
| 3   | 菜鸟教程   | 201   | 2016-05-17 |
| 4   | 微博       | 13    | 2016-05-15 |
| 5   | Facebook   | 205   | 2016-05-14 |
| 5   | Facebook   | 545   | 2016-05-16 |
+---+-----+-----+-----+
9 rows in set (0.00 sec)
```

不同的 SQL JOIN

在我们继续讲解实例之前，我们先列出您可以使用的不同的 SQL JOIN 类型：

- **INNER JOIN** : 如果表中有至少一个匹配, 则返回行
- **LEFT JOIN** : 即使右表中没有匹配, 也从左表返回所有的行
- **RIGHT JOIN** : 即使左表中没有匹配, 也从右表返回所有的行
- **FULL JOIN** : 只要其中一个表中存在匹配, 则返回行

[← SQL 别名](#)[SQL INNER JOIN 关键字 →](#)

2 篇笔记

[写笔记](#)

首先, 连接的结果可以在逻辑上看作是由SELECT语句指定的列组成的新表。

左连接与右连接的左右指的是以两张表中的哪一张为基准, 它们都是外连接。

外连接就好像是为非基准表添加了一行全为空值的万能行, 用来与基准表中找不到匹配的行进行匹配。假设两个没有空值的表进行左连接, 左表是基准表, 左表的所有行都出现在结果中, 右表则可能因为无法与基准表匹配而出现是空值的字段。

这部分主要涉及的是表连接的逻辑问题, 教程主讲语法的话这部分确实需要自己再去搜资料

来源: 《数据库系统原理教程》, 王珊, 陈红编著, P86

Roam 2年前 (2017-09-12)



得到的结果数:

```
inner join <= min(left join, right join)
full join >= max(left join, right join)
当 inner join < min(left join, right join) 时, full join > max(left join, right join)
```

悠闲的某某某 11个月前 (05-03)