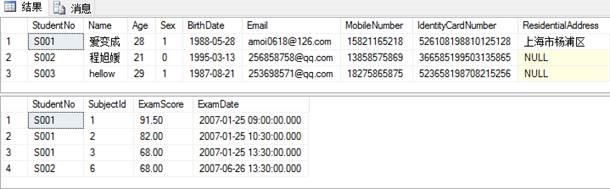
**[使用 CROSS APPLY 与 OUTER APPLY 连接查询](https://www.cnblogs.com/abeam/p/7137993.html)**

  前言

日常开发中遇到多表查询时，首先会想到 INNER JOIN 或 LEFT OUTER JOIN 等等，但是这两种查询有时候不能满足需求。比如，左表一条关联右表多条记录时，我需要控制右表的某**一条**或**多条**记录跟左表匹配。貌似，INNER JOIN 或 LEFT OUTER JOIN 不能很好完成。但是 CROSS APPLY 与 OUTER APPLY 可以，下面用示例说明。

1.   示例一

  有两张表：Student（学生表）和 Score（成绩表），数据如下：

[](http://images2015.cnblogs.com/blog/654920/201707/654920-20170708210653222-1099792135.jpg)

1)   查询每个学生最近两次的考试成绩

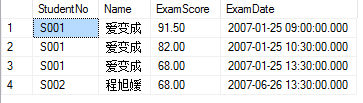
1.   先试下 INNER JOIN

1)   SQL 代码

SELECT T1.StudentNo, T1.Name, T2.ExamScore, T2.ExamDate FROM Student AS T1

INNER JOIN Score AS T2 ON T1.StudentNo = T2.StudentNo

2)   结果：

[](http://images2015.cnblogs.com/blog/654920/201707/654920-20170708210653894-723689019.png)

3)   咦，不对，这不是我想要的结果。

2.   再看看 CROSS APPLY

1)   SQL 代码

SELECT T1.StudentNo, T1.Name, T2.ExamScore, T2.ExamDate FROM Student AS T1

CROSS APPLY(

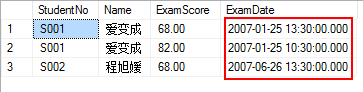
    SELECT TOP 2 \* FROM Score AS T

    WHERE T1.StudentNo = T.StudentNo

    ORDER BY T.ExamDate DESC

) AS T2

2)   结果：

[](http://images2015.cnblogs.com/blog/654920/201707/654920-20170708210654894-320193055.png)

3)   嗯，这次对了，并且还是按照“考试时间”倒序排序的。

2)   查询每个学生最近两次的考试成绩,没有参加考试的同学成绩补 null

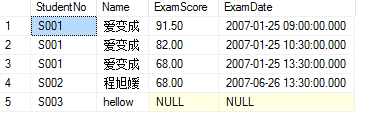
1.   先试下 LEFT OUTER JOIN

1)   SQL 代码

SELECT T1.StudentNo, T1.Name, T2.ExamScore, T2.ExamDate FROM Student AS T1

LEFT OUTERJOIN Score AS T2 ON T1.StudentNo = T2.StudentNo

2)   结果：

[](http://images2015.cnblogs.com/blog/654920/201707/654920-20170708210656112-2067612597.png)

3)   咦，不对，这又不是我想要的结果。

2.   再看看 OUTER APPLY

1)   SQL 代码

SELECT T1.StudentNo, T1.Name, T2.ExamScore, T2.ExamDate FROM Student AS T1

OUTER APPLY(

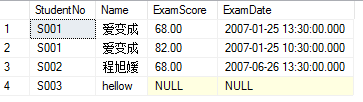
    SELECT TOP 2 \* FROM Score AS T

    WHERE T1.StudentNo = T.StudentNo

    ORDER BY T.ExamDate DESC

) AS T2

2)   结果：

[](http://images2015.cnblogs.com/blog/654920/201707/654920-20170708210657081-1674257416.png)

3)   嗯，这次对了，不但按照“考试时间”倒序排序的，而且没有考试的同学也被查出来了。

  总结

1.   理解 CROSS APPLY 与 OUTER APPLY（个人理解）

1)   CROSS APPLY 的意思是“交叉应用”，在查询时首先查询左表，然后右表的每一条记录跟左表的当前记录进行匹配。匹配成功则将左表与右表的记录合并为一条记录输出；匹配失败则抛弃左表与右表的记录。（与 INNER JOIN 类似）

2)   OUTER APPLY 的意思是“外部应用”，与 CROSS APPLY 的原理一致，只是在匹配失败时，左表与右表也将合并为一条记录输出，不过右表的输出字段为 null。（与 LEFT OUTER JOIN 类似）

2.   CROSS APPLY 与 INNER JOIN 的区别

1)   CROSS APPLY 可以根据当前左表的当前记录去查询右表，但是 INNER JOIN 不可以，INNER JOIN 是根据左表的当前记录匹配右表整个结果集。

2)   两者都是匹配成功才输出。

3.   OUTER APPLY 与 LEFT OUTER JOIN 的区别

1)   它们和（CROSS APPLY 与 INNER JOIN）类似。

2)   只是两者都是匹配失败也会输出。

4.   使用场景：

1)   一个商品有多张图片，但是只想取最近的一张图片跟商品匹配。

5.   总结一句话：右表可以是有条件的跟左表的记录匹配，而条件的值可以来至于左表。

cross apply sometimes enables you to do things that you cannot do with inner join.

Example (a syntax error):

select F.\* from sys.objects O

inner join dbo.myTableFun(O.name) F

on F.schema\_id= O.schema\_id

This is a **syntax error**, because, when used with inner join, table functions can only take *variables or constants* as parameters. (I.e., the table function parameter cannot depend on another table's column.)

However:

select F.\* from sys.objects O

cross apply ( select \* from dbo.myTableFun(O.name) ) F

where F.schema\_id= O.schema\_id

This is legal.

**Edit:** Or alternatively, shorter syntax: (by ErikE)

select F.\* from sys.objects O

cross apply dbo.myTableFun(O.name) F

where F.schema\_id= O.schema\_id