(T28)討論CrossApply和OuterApply  
CourseGUID: e48417fc-9db5-4e99-822c-706c5ccef6cc  
=======================================================================  
(T28)討論CrossApply和OuterApply  
=======================================================================  
0. Summary

1. Create Sample Data

2. InnerJoin V.S. LeftJoin

-----------

3. TableValueFunction need Cross Apply and Outter Apply instead of Join

3.1. Create Table Value Function, fn\_GetPersonAByDepartmentAID

3.2. Table Value Function can not use INNER JOIN

3.3. Table Value Function must use CROSS APPLY

3.4. fnTableValueFunction must be the right hand side of CROSS APPLY

3.5. Table Value Function must use OUTER APPLY

-----------

4. Clean up  
=======================================================================

0. Summary

1.

INNER JOIN   V.S.  CROSS APPLY

----Table Value Function must use CROSS APPLY

--SELECT  d.DepartmentName ,

--        p.[Name] ,

--        p.Gender ,

--        p.Salary

--FROM    DepartmentA d

--        CROSS APPLY fn\_GetPersonAByDepartmentAID(d.ID) p

--ORDER BY d.ID;

--GO -- Run the previous command and begins new batch

1.1.

--FROM    DepartmentA d

--        CROSS APPLY fn\_GetPersonAByDepartmentAID(d.ID) p

Pass each DepartmentID into fn\_GetPersonAByDepartmentAID()

This will return all the Persons who has Department.

Thus, fn\_GetPersonAByDepartmentAID() CROSS APPLY DepartmentA

will return all the Persons with their DepartmentName.

1.2.

--TableA INNER JOIN TableB

--ON TableA.ColumnAB = TableB.ColumnAB

INNER JOIN is for join 2 tables.

1.3.

--fnTableValueFunction CROSS APPLY TableA

This will cause ERROR,

fnTableValueFunction must be the right hand side of CROSS APPLY

1.4.

--TableA CROSS APPLY fnTableValueFunction

fnTableValueFunction must be the right hand side of CROSS APPLY

CROSS APPLY is similar to INNER JOIN

which retrieves only the matching rows.

However,

INNER JOIN is for join 2 tables.

CROSS APPLY is join 1 table(Left Hand Side)

and fnTableValueFunction(Right Hand Side).

fnTableValueFunction can not use INNER JOIN

-------------------------------

2.

LEFT JOIN   V.S.  OUTER APPLY

--SELECT  d.DepartmentName ,

--        p.[Name] ,

--        p.Gender ,

--        p.Salary

--FROM    DepartmentA d

--        OUTER APPLY fn\_GetPersonAByDepartmentAID(d.ID) p

--ORDER BY d.ID;

--GO -- Run the previous command and begins new batch

2.1.

--FROM    DepartmentA d

--        OUTER APPLY fn\_GetPersonAByDepartmentAID(d.ID) p

Pass each DepartmentID into fn\_GetPersonAByDepartmentAID()

This will return all the Persons who has Department.

DepartmentA d is in Left Hand Side of OUTER APPLY.

Thus, the query will return

all the Persons with their DepartmentName

plus all departments name which has no persons.

2.2.

--TableA LEFT JOIN TableB

--ON TableA.ColumnAB = TableB.ColumnAB

LEFT JOIN is for join 2 tables.

2.3.

--fnTableValueFunction OUTER APPLY TableA

This will cause ERROR,

fnTableValueFunction must be the right hand side of OUTER APPLY

2.4.

--TableA OUTER APPLY fnTableValueFunction

fnTableValueFunction must be the right hand side of OUTER APPLY

OUTER APPLY is similar to LEFT JOIN

which retrieves only the matching rows + Left Hand Side un-matching rows

However,

LEFT JOIN is for join 2 tables.

OUTER APPLY is join 1 table(Left Hand Side)

and fnTableValueFunction(Right Hand Side).

fnTableValueFunction can not use LEFT JOIN

==================================================

1. Create Sample Data

--=========================================================================

--T028\_01\_Create Sample Data

--=========================================================================

IF ( EXISTS ( SELECT    \*

              FROM      INFORMATION\_SCHEMA.TABLES

              WHERE     TABLE\_NAME = 'PersonA' ) )

    BEGIN

        TRUNCATE TABLE dbo.PersonA;

        DROP TABLE PersonA;

    END;

GO -- Run the previous command and begins new batch

IF ( EXISTS ( SELECT    \*

              FROM      INFORMATION\_SCHEMA.TABLES

              WHERE     TABLE\_NAME = 'DepartmentA' ) )

    BEGIN

        TRUNCATE TABLE dbo.DepartmentA;

        DROP TABLE DepartmentA;

    END;

GO -- Run the previous command and begins new batch

CREATE TABLE DepartmentA

    (

      ID INT PRIMARY KEY ,

      DepartmentName NVARCHAR(50)

    );

GO -- Run the previous command and begins new batch

INSERT  INTO DepartmentA

VALUES  ( 1, 'Department01' );

INSERT  INTO DepartmentA

VALUES  ( 2, 'Department02' );

INSERT  INTO DepartmentA

VALUES  ( 3, 'Department03' );

INSERT  INTO DepartmentA

VALUES  ( 4, 'Department04' );

INSERT  INTO DepartmentA

VALUES  ( 5, 'Department05' );

GO -- Run the previous command and begins new batch

CREATE TABLE PersonA

    (

      ID INT PRIMARY KEY ,

      [Name] NVARCHAR(50) ,

      Gender NVARCHAR(10) ,

      Salary MONEY ,

      DepartmentID INT FOREIGN KEY REFERENCES DepartmentA ( ID )

    );

GO -- Run the previous command and begins new batch

INSERT  INTO PersonA

VALUES  ( 1, 'Name01', 'Male', 41000, 1 );

INSERT  INTO PersonA

VALUES  ( 2, 'Name02', 'Female', 75000, 3 );

INSERT  INTO PersonA

VALUES  ( 3, 'Name03', 'Female', 65000, 2 );

INSERT  INTO PersonA

VALUES  ( 4, 'Name04', 'Female', 44000, 3 );

INSERT  INTO PersonA

VALUES  ( 5, 'Name05', 'Male', 38000, 1 );

GO -- Run the previous command and begins new batch

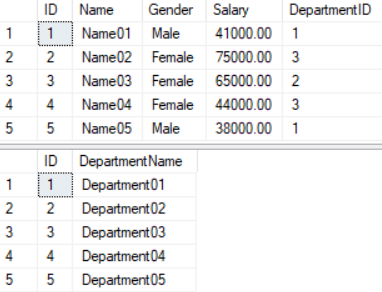
SELECT  \*

FROM    dbo.PersonA;

SELECT  \*

FROM    dbo.DepartmentA;

GO -- Run the previous command and begins new batch



==================================================

2. InnerJoin V.S. LeftJoin

--=========================================================================

--T028\_02\_InnerJoin V.S. LeftJoin

--=========================================================================

--=========================================================================

--T028\_02\_01

--Department INNER JOIN Person

SELECT  d.DepartmentName ,

        p.Name ,

        p.Gender ,

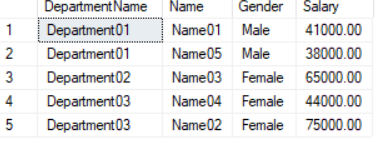
        p.Salary

FROM    DepartmentA d

        INNER JOIN dbo.PersonA p ON p.DepartmentID = d.ID

ORDER BY d.ID;

GO -- Run the previous command and begins new batch



--=========================================================================

--T028\_02\_02

--Department LEFT JOIN Person

SELECT  d.DepartmentName ,

        p.Name ,

        p.Gender ,

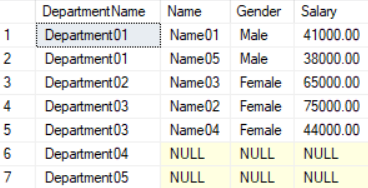
        p.Salary

FROM    DepartmentA d

        LEFT JOIN dbo.PersonA p ON p.DepartmentID = d.ID

ORDER BY d.ID;

GO -- Run the previous command and begins new batch



==================================================

3. TableValueFunction need Cross Apply and Outter Apply instead of Join

--=========================================================================

--T028\_03\_TableValueFunction need Cross Apply and Outter Apply instead of Join

--=========================================================================

3.1. Create Table Value Function, fn\_GetPersonAByDepartmentAID

--=========================================================================

--T028\_03\_01

--Create Table Value Function, fn\_GetPersonAByDepartmentAID

IF ( EXISTS ( SELECT    \*

              FROM      INFORMATION\_SCHEMA.ROUTINES

              WHERE     ROUTINE\_TYPE = 'FUNCTION'

                        AND LEFT(ROUTINE\_NAME, 2) NOT IN ( '@@' )

                        AND SPECIFIC\_NAME = 'fn\_GetPersonAByDepartmentAID' ) )

    BEGIN

        DROP FUNCTION fn\_GetPersonAByDepartmentAID;

    END;

GO -- Run the previous command and begins new batch

CREATE FUNCTION fn\_GetPersonAByDepartmentAID ( @DepartmentAID int )

RETURNS TABLE

AS

RETURN

    ( SELECT    p.ID ,

                p.[Name] ,

                p.Gender ,

                p.Salary ,

                p.DepartmentID

      FROM      dbo.PersonA p

      WHERE     DepartmentID = @DepartmentAID

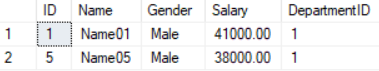
    );

GO -- Run the previous command and begins new batch

SELECT  \*

FROM    fn\_GetPersonAByDepartmentAID(1);

GO -- Run the previous command and begins new batch



3.2. Table Value Function can not use INNER JOIN

--=========================================================================

--T028\_03\_02

--Table Value Function can not use INNER JOIN

SELECT  d.DepartmentName ,

        p.[Name] ,

        p.Gender ,

        p.Salary

FROM    DepartmentA d

        INNER JOIN fn\_GetPersonAByDepartmentAID(d.ID) p ON d.ID = p.DepartmentID;

GO -- Run the previous command and begins new batch

/\*

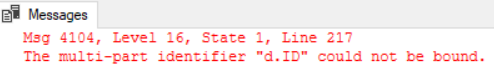
Error Message

--Msg 4104, Level 16, State 1, Line 133

--The multi-part identifier "d.ID" could not be bound.

Table Value Function can not use INNER JOIN

\*/



3.3. Table Value Function must use CROSS APPLY

--=========================================================================

--T028\_03\_03

--Table Value Function must use CROSS APPLY

SELECT  d.DepartmentName ,

        p.[Name] ,

        p.Gender ,

        p.Salary

FROM    DepartmentA d

        CROSS APPLY fn\_GetPersonAByDepartmentAID(d.ID) p

ORDER BY d.ID;

GO -- Run the previous command and begins new batch

/\*

1.

--FROM    DepartmentA d

--        CROSS APPLY fn\_GetPersonAByDepartmentAID(d.ID) p

Pass each DepartmentID into fn\_GetPersonAByDepartmentAID()

This will return all the Persons who has Department.

Thus, fn\_GetPersonAByDepartmentAID() CROSS APPLY DepartmentA

will return all the Persons with their DepartmentName.

1.1.

--TableA INNER JOIN TableB

--ON TableA.ColumnAB = TableB.ColumnAB

INNER JOIN is for join 2 tables.

1.2.

--fnTableValueFunction CROSS APPLY TableA

This will cause ERROR,

fnTableValueFunction must be the right hand side of CROSS APPLY

1.3.

--TableA CROSS APPLY fnTableValueFunction

fnTableValueFunction must be the right hand side of CROSS APPLY

CROSS APPLY is similar to INNER JOIN

which retrieves only the matching rows.

However,

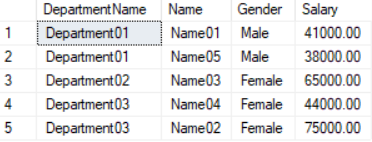
INNER JOIN is for join 2 tables.

CROSS APPLY is join 1 table(Left Hand Side)

and fnTableValueFunction(Right Hand Side).

fnTableValueFunction can not use INNER JOIN

\*/



3.4. fnTableValueFunction must be the right hand side of CROSS APPLY

--=========================================================================

--T028\_03\_04

--ERROR: fnTableValueFunction must be the right hand side of CROSS APPLY

SELECT  d.DepartmentName ,

        p.[Name] ,

        p.Gender ,

        p.Salary

FROM    fn\_GetPersonAByDepartmentAID(d.ID) p

        CROSS APPLY DepartmentA d

ORDER BY d.ID;

GO -- Run the previous command and begins new batch

/\*

1.

--fnTableValueFunction CROSS APPLY TableA

This will cause ERROR,

fnTableValueFunction must be the right hand side of CROSS APPLY

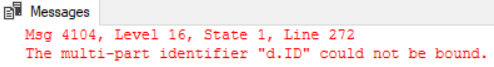
2.

Output

--Msg 4104, Level 16, State 1, Line 186

--The multi-part identifier "d.ID" could not be bound.

\*/



3.5. Table Value Function must use OUTER APPLY

--=========================================================================

--T028\_03\_05

--Table Value Function must use OUTER APPLY

SELECT  d.DepartmentName ,

        p.[Name] ,

        p.Gender ,

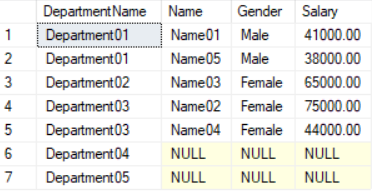
        p.Salary

FROM    DepartmentA d

        OUTER APPLY fn\_GetPersonAByDepartmentAID(d.ID) p

ORDER BY d.ID;

GO -- Run the previous command and begins new batch



/\*

1.

--FROM    DepartmentA d

--        OUTER APPLY fn\_GetPersonAByDepartmentAID(d.ID) p

Pass each DepartmentID into fn\_GetPersonAByDepartmentAID()

This will return all the Persons who has Department.

DepartmentA d is in Left Hand Side of OUTER APPLY.

Thus, the query will return

all the Persons with their DepartmentName

plus all departments name which has no persons.

1.1.

--TableA LEFT JOIN TableB

--ON TableA.ColumnAB = TableB.ColumnAB

LEFT JOIN is for join 2 tables.

1.2.

--fnTableValueFunction OUTER APPLY TableA

This will cause ERROR,

fnTableValueFunction must be the right hand side of OUTER APPLY

1.3.

--TableA OUTER APPLY fnTableValueFunction

fnTableValueFunction must be the right hand side of OUTER APPLY

OUTER APPLY is similar to LEFT JOIN

which retrieves only the matching rows + Left Hand Side un-matching rows

However,

LEFT JOIN is for join 2 tables.

OUTER APPLY is join 1 table(Left Hand Side)

and fnTableValueFunction(Right Hand Side).

fnTableValueFunction can not use LEFT JOIN

\*/

==================================================

4. Clean up

--=========================================================================

--T028\_04\_Clean up

--=========================================================================

--Drop Function if it exists.

IF ( EXISTS ( SELECT    \*

              FROM      INFORMATION\_SCHEMA.ROUTINES

              WHERE     ROUTINE\_TYPE = 'FUNCTION'

                        AND LEFT(ROUTINE\_NAME, 2) NOT IN ( '@@' )

                        AND SPECIFIC\_NAME = 'fn\_GetPersonAByDepartmentAID' ) )

    BEGIN

        DROP FUNCTION fn\_GetPersonAByDepartmentAID;

    END;

GO -- Run the previous command and begins new batch

---------------------------

--Drop Table if it exists.

IF ( EXISTS ( SELECT    \*

              FROM      INFORMATION\_SCHEMA.TABLES

              WHERE     TABLE\_NAME = 'PersonA' ) )

    BEGIN

        TRUNCATE TABLE dbo.PersonA;

        DROP TABLE PersonA;

    END;

GO -- Run the previous command and begins new batch

IF ( EXISTS ( SELECT    \*

              FROM      INFORMATION\_SCHEMA.TABLES

              WHERE     TABLE\_NAME = 'DepartmentA' ) )

    BEGIN

        TRUNCATE TABLE dbo.DepartmentA;

        DROP TABLE DepartmentA;

    END;

GO -- Run the previous command and begins new batch