LEFT JOIN vs. LEFT OUTER JOIN in SQL Server

Asked 10 years, 8 months ago Active 1 month ago Viewed 1.4m times

1474

What is the difference between `LEFT JOIN` and `LEFT OUTER JOIN` ?

sql-server tsql left-join outer-join

576

edited Aug 23 '16 at 15:39
Quiver
665 2 15 37

asked Jan 2 '09 at 8:30
KG Sosa
7,843 6 22 26

170 None! The `OUTER` keyword is optional. – jarlh Jun 2 '16 at 14:02

2 **Venn diagrams are not appropriate for answers to this question.** The answer to this question is **None**. See the first comment. – philipxy May 5 at 8:00

1 Venn diagrams illustrate *the difference in output rows* for *special cases* of *inner vs outer join*. *If no nulls or duplicate rows are input* (so we can take a table to be a set of row-valued values & use normal math equality) *then left & right circles hold output tables/sets of left & right join*. But if nulls or duplicate rows are input then it is so difficult to explain just what the circles are sets of & how those sets relate to input & output tables/bags that Venn diagrams are not helpful. See my comments here re other Venn diagram abuses. – philipxy May 5 at 8:14

12 Answers

2116

As per the documentation: FROM (Transact-SQL):

```
<join_type> ::=
    [ { INNER | { { LEFT | RIGHT | FULL } [ OUTER ] } } [ <join_hint> ] ]
    JOIN
```

The keyword `OUTER` is marked as optional (enclosed in square brackets). In this specific case, whether you specify `OUTER` or not makes no difference. Note that while the other elements of the join clause is also marked as optional, leaving *them* out *will* make a

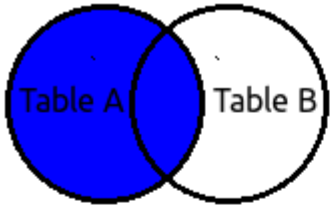
For instance, the entire type-part of the `JOIN` clause is optional, in which case the default is `INNER` if you just specify `JOIN`. In other words, this is legal:

```
SELECT *
FROM A JOIN B ON A.X = B.Y
```

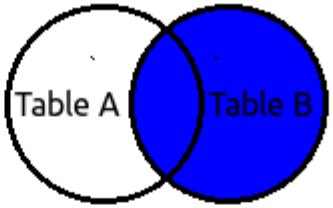
Here's a list of equivalent syntaxes:

```
A LEFT JOIN B            A LEFT OUTER JOIN B
A RIGHT JOIN B           A RIGHT OUTER JOIN B
A FULL JOIN B            A FULL OUTER JOIN B
A INNER JOIN B           A JOIN B
```

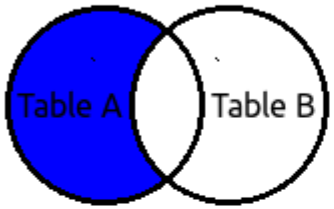
Also take a look at the answer I left on this other SO question: SQL left join vs multiple tables on FROM line?.



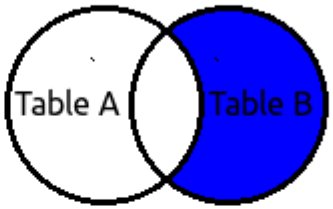

```
SELECT [list] FROM
    [Table A] A
LEFT JOIN
    [Table B] B
ON A.Value = B.Value
```



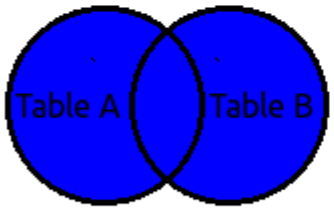

```
SELECT [list] FROM
    [Table A] A
RIGHT JOIN
    [Table B] B
ON A.Value = B.Value
```



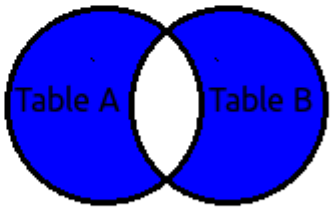

```
SELECT [list] FROM
    [Table A] A
LEFT JOIN
    [Table B] B
ON A.Value = B.Value
WHERE B.Value IS NULL
```



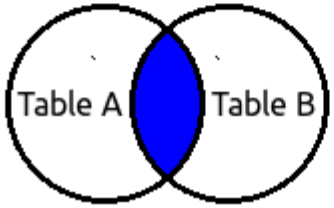

```
SELECT [list] FROM
    [Table A] A
RIGHT JOIN
    [Table B] B
ON A.Value = B.Value
WHERE A.Value IS NULL
```




```
SELECT [list] FROM
    [Table A] A
FULL OUTER JOIN
    [Table B] B
ON A.Value = B.Value
```




```
SELECT [list] FROM
    [Table A] A
FULL OUTER JOIN
    [Table B] B
ON A.Value = B.Value
WHERE A.Value IS NULL
OR B.Value IS NULL
```




```
SELECT [list] FROM
    [Table A] A
INNER JOIN
    [Table B] B
ON A.Value = B.Value
```

edited Jul 24 at 8:23

WBT
968 2 14 30

answered Jan 2 '09 at 8:51

Lasse Vågsæther Karlsen
304k 87 541 732

121 Absolutely correct. OUTER is allowed for ANSI-92 compatibility. – Sean Reilly Jan 2 '09 at 21:34

7 @LasseV.Karlsen I just meant that the left side has the concise form and the right side has the expanded form. I thought it would make it coherent if you followed the same for `JOIN` s as well. – nawfal May 2 '13 at 7:40

28 @Outlier That's your prerogative, but clearly 451 other people thought the answer was good. To be honest, if one answer says X and another answer says X and references the documentation, as well as copy the relevant pieces of the documentation into the answer, my vote goes to the second answer and that is why I write my answers the way I do. That someone *claims* X is not good. If you can *prove* X, that's better (not to slight sactiw's answer). But, its your prerogative to think whatever you want to. I question why you think it is pointless though, is the answer wrong in any way? – Lasse Vågsæther Karlsen Jul 20 '14 at 19:21

5 @Outlier And to be honest, if you don't feel like reading my "long" answers, then my answer was not for you. It is for everyone else coming here wondering about the same thing and want some background on why that is, and where to find more information, etc. Clearly you are the kind of guy that would want this answer: "None", but unfortunately this is not a legal answer here on Stack Overflow, nor should it be. – Lasse Vågsæther Karlsen Jul 20 '14 at 19:25

679

To answer your question **there is no difference between LEFT JOIN and LEFT OUTER JOIN, they are exactly same** that said...

At the top level there are mainly 3 types of joins:

1. INNER
2. OUTER
3. CROSS

1. **INNER JOIN** - fetches data if present in both the tables.
2. **OUTER JOIN** are of **3** types:
    1. `LEFT OUTER JOIN` - fetches data if present in the left table.
    2. `RIGHT OUTER JOIN` - fetches data if present in the right table.
    3. `FULL OUTER JOIN` - fetches data if present in either of the two tables.
3. **CROSS JOIN**, as the name suggests, does `[n X m]` that joins everything to everything.
   Similar to scenario where we simply lists the tables for joining (in the `FROM` clause of the `SELECT` statement), using commas to separate them.

- If you just mention `JOIN` then by default it is a `INNER JOIN`.
- An `OUTER` join has to be `LEFT` | `RIGHT` | `FULL` you can not simply say `OUTER JOIN`.
- You can drop `OUTER` keyword and just say `LEFT JOIN` or `RIGHT JOIN` or `FULL JOIN`.

For those who want to visualise these in a better way, please go to this link: A Visual Explanation of SQL Joins

edited May 14 '14 at 9:38

SriniV
8,958 13 48 72

answered Dec 9 '10 at 18:05

sactiw
18.4k 3 31 28

14 Very good answer. It will be clearer if you say "LEFT OUTER JOIN - fetches all data from the left table with matching data from right, if preset." for 2.1 (and similar change for 2.2) – ssh Dec 27 '12 at 19:27

Also you can do cross join by simply 'select * from TableA,TableB' – om471987 Feb 10 '13 at 18:22

1 Sorry if I'm necrobumping, but is `CROSS JOIN` the same as `FULL JOIN` ? – TechnicalTophat Jul 13 '16 at 13:11

7 @RhysO no, CROSS JOIN is a Cartesian product i.e. CROSS JOIN of a table, having n rows, with a table, having m rows, will always give (n*m) rows while FULL OUTER JOIN of a table, having n rows, with a table, having m rows, will give at max (n+m) rows – sactiw Jul 13 '16 at 15:34

336

> What is the difference between left join and left outer join?

Nothing. `LEFT JOIN` and `LEFT OUTER JOIN` are equivalent.

edited Aug 22 '14 at 0:03

answered Jan 2 '09 at 8:31

Mitch Wheat
263k 36 413 506

10 erm hello! who voted this down? LEFT JOIN is teh same as LEFT OUTER JOIN. – Mitch Wheat Jan 2 '09 at 8:35

6 This is the case in Microsoft SQL Server, and any other SQL-compliant RDBMS. – Bill Karwin Jan 2 '09 at 8:41

6 It would be nice if you added a reference or explanation about why the `OUTER` is optional. – Zero3 Apr 12 '16 at 12:50

original question straight to the point. – Bacon Brad Aug 9 '18 at 20:36

63

I'm a PostgreSQL DBA, as far as I could understand the difference between outer or not outer joins difference is a topic that has considerable discussion all around the internet. Until today I never saw a difference between those two; So I went further and I try to find the difference between those. At the end I read the whole documentation about it and I found the answer for this,

So if you look on documentation (at least in PostgreSQL) you can find this phrase:

"The words `INNER` and `OUTER` are optional in all forms. `INNER` is the default; `LEFT`, `RIGHT`, and `FULL` imply an outer join."

In another words,

`LEFT JOIN` **and** `LEFT OUTER JOIN` **ARE THE SAME**

`RIGHT JOIN` **and** `RIGHT OUTER JOIN` **ARE THE SAME**

I hope it can be a contribute for those who are still trying to find the answer.

edited May 23 '15 at 3:42

shA.t
13.5k 4 39 77

answered Aug 30 '10 at 2:33

andrefsp
639 5 2

57

`Left Join` and `Left Outer Join` are one and the **same**. The former is the shorthand for the latter. The same can be said about the `Right Join` and `Right Outer Join` relationship. The demonstration will illustrate the equality. Working examples of each query have been provided via *SQL Fiddle*. This tool will allow for hands on manipulation of the query.

Given

Table Left_Side	Table Right_Side
1	1
2	20
23	23
79	34
108	54

Left Join and Left Outer Join

```
SELECT
  LS.ID as "Left_Side",
  RS.ID as "Right_Side"
FROM Left_Side LS
LEFT JOIN Right_Side RS ON LS.id = RS.id
```

=

```
SELECT
  LS.ID as "Left_Side",
  RS.ID as "Right_Side"
FROM Left_Side LS
LEFT OUTER JOIN Right_Side RS ON LS.ID = RS.ID
```

Results

Left_Side	Right_Side
1	1
23	23
122	(null)
2	(null)
79	(null)
108	(null)
1526	(null)

=

Left_Side	Right_Side
1	1
23	23
122	(null)
2	(null)
79	(null)
108	(null)
1526	(null)

Right Join and Right Outer Join

```
SELECT
  LS.ID as "Left_Side",
  RS.ID as "Right_Side"
FROM Left_Side LS
RIGHT JOIN Right_Side RS ON LS.id = RS.id
```

=

```
SELECT
  LS.ID as "Left_Side",
  RS.ID as "Right_Side"
FROM Left_Side LS
RIGHT OUTER JOIN Right_Side RS ON LS.ID = RS.ID
```

Results

Left_Side	Right_Side
(null)	54
(null)	78
(null)	34
(null)	20
1	1
23	23
(null)	159

=

Left_Side	Right_Side
(null)	54
(null)	78
(null)	34
(null)	20
1	1
23	23
(null)	159


xdhmoore
2,567 4 28 51


WorkSmarter
3,099 2 21 29

40

I find it easier to think of Joins in the following order:

- CROSS JOIN - a Cartesian product of both tables. ALL joins begin here
- INNER JOIN - a CROSS JOIN with a filter added.
- OUTER JOIN - an INNER JOIN with missing elements (from either LEFT or RIGHT table) added afterward.

Until I figured out this (relatively) simple model, JOINS were always a bit more of a black art. Now they make perfect sense.

Hope this helps more than it confuses.

answered Dec 29 '11 at 21:01


frozenjim
520 4 9

This does not answer the question. – philipxy Jun 17 at 3:38

28

Why are LEFT/RIGHT and LEFT OUTER/RIGHT OUTER the same? Let's explain why this vocabulary. Understand that LEFT and RIGHT joins are specific cases of the OUTER join, and therefore couldn't be anything else than OUTER LEFT/OUTER RIGHT. The OUTER join is also called **FULL** OUTER as opposed to LEFT and RIGHT joins that are **PARTIAL** results of the OUTER join. Indeed:

Table A	Table B
1	5
2	1
3	6
4	2

Table A	Table B
1	1
2	2
3	null
4	null
null	5
null	6

OUTER JOIN (FULL)

Table A	Table B
1	1
2	2
3	null
4	null
-	-
-	-

LEFT OUTER (partial)

Table A	Table B
1	1
2	2
-	-
-	-
null	5
null	6

RIGHT OUTER (partial)

edited Apr 17 '14 at 21:15

answered Apr 17 '14 at 21:08

Yugo Amaryl

861 2 10 17

"it is clear only 3 cases exist": interesting but flawed. Consider that an inner join is a specialised cross join (i.e. move join predicates to the where clause). Further consider that outer join isn't a join at all, rather is a union where are used nulls in place of 'missing' columns. Therefore, it could be argued that cross is the only join required. Note the current thinking in relational theory is that natural join satisfies all join requirements. Aside: can you explain if/how the vocabulary " `JOIN` implies `INNER JOIN` " fits with your reasoning for outer join vocab? – onedaywhen Jul 6 '16 at 10:39

24

To answer your question

> In Sql Server joins syntax **OUTER** is optional

It is mentioned in msdn article : https://msdn.microsoft.com/en-us/library/ms177634(v=sql.130).aspx

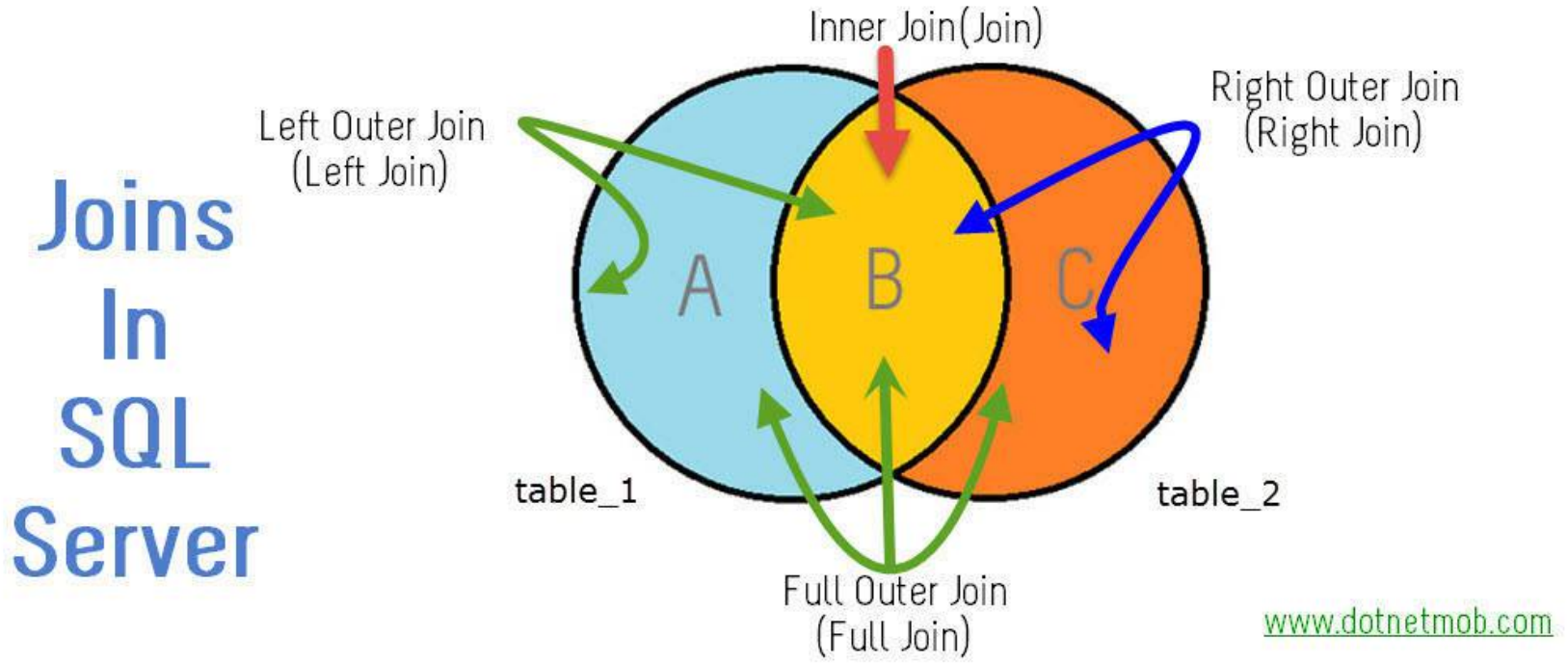
So following list shows join equivalent syntaxes with and without **OUTER**

```
LEFT OUTER JOIN => LEFT JOIN
RIGHT OUTER JOIN => RIGHT JOIN
FULL OUTER JOIN => FULL JOIN
```

Other equivalent syntaxes

```
INNER JOIN => JOIN
CROSS JOIN => ,
```

Strongly Recommend Dotnet Mob Artice : Joins in Sql Server



edited Jan 17 at 9:16
MD. Khairul Basar
3,020 10 25 44

answered Jun 25 '16 at 4:05
mass
364 2 6

Re abuse of Venn diagrams here see my comment on the question. – philipxy Apr 29 at 1:52

19

There are only 3 joins:

- A) Cross Join = Cartesian (E.g: Table A, Table B)
- B) Inner Join = JOIN (E.g: Table A Join/Inner Join Table B)
- C) Outer join:

```
There are three type of outer join
1)  Left Outer Join     = Left Join
2)  Right Outer Join    = Right Join
3)  Full Outer Join     = Full Join
```

Hope it'd help.

answered Mar 17 '17 at 9:56

Delickate
705 7 13

So 5 joins altogether. – Zeek2 Jul 20 '18 at 10:44

18

There are mainly three types of JOIN

1. Inner: fetches data, that are present in both tables
   - Only JOIN means INNER JOIN
2. Outer: are of three types
   - LEFT OUTER - - fetches data present only in left table & matching condition
   - RIGHT OUTER - - fetches data present only in right table & matching condition
   - FULL OUTER - - fetches data present any or both table
   - (LEFT or RIGHT or FULL) OUTER JOIN can be written w/o writing "OUTER"
3. Cross Join: joins everything to everything

edited Jan 18 '12 at 12:59

sjngm
8,488 13 64 100

answered Jan 18 '12 at 12:29

Harsh
189 1 3

Syntactic sugar, makes it more obvious to the casual reader that the join isn't an inner one.

3,678 2 34 48

8,095 7 46 76

1 So... what's a FULL OUTER JOIN then? – Amy B Jan 2 '09 at 20:29

4 tableA FULL OUTER JOIN tableB will give you three types of records: all records in tableA with no matching record in tableB, all records in tableB with no matching record in tableA, and all records in tableA with a matching record in tableB. – Dave DuPlantis Oct 5 '09 at 18:16

2

Just in the context of this question, I want to post the 2 'APPLY' operators as well:

JOINS:

1. **INNER JOIN** = JOIN
2. **OUTER JOIN**
   - LEFT OUTER JOIN = LEFT JOIN
   - RIGHT OUTER JOIN = RIGHT JOIN
   - FULL OUTER JOIN = FULL JOIN
3. **CROSS JOIN**

SELF-JOIN: This is not exactly a separate type of join. This is basically joining a table to itself using one of the above joins. But I felt it is worth mentioning in the context JOIN discussions as you will hear this term from many in the SQL Developer community.

APPLY:

1. **CROSS APPLY** -- Similar to INNER JOIN (But has added advantage of being able to compute something in the Right table for each row of the Left table and would return only the matching rows)
2. **OUTER APPLY** -- Similar to LEFT OUTER JOIN (But has added advantage of being able to compute something in the Right table for each row of the Left table and would return all the rows from the Left table irrespective of a match on the Right table)

https://www.mssqltips.com/sqlservertip/1958/sql-server-cross-apply-and-outer-apply/

https://sqlhints.com/2016/10/23/outer-apply-in-sql-server/

Real life example, when to use OUTER / CROSS APPLY in SQL

JOINS, one SQL developer should try to learn the APPLY operators next.

answered Jun 17 at 2:23

san
576 1 8

---

protected by Deduplicator Oct 8 '15 at 2:28

Thank you for your interest in this question. Because it has attracted low-quality or spam answers that had to be removed, posting an answer now requires 10 reputation on this site (the association bonus does not count).

Would you like to answer one of these unanswered questions instead?