



CODE
PROJECT®
For those who code

[articles](#)[Q&A](#)[forums](#)[lounge](#)

Visual Representation of SQL Joins



C.L. Moffatt, 3 Feb 2009

CPOL

Rate this:



4.96 (354 votes)

This article describes SQL Joins in a visual manner, and also the most efficient way to write the visualized Joins.

[Download Visual SQL JOINS examples - 1.09 KB](#)[Download Visual SQL JOINS cheat sheet - 143 KB](#)

Introduction

This is just a simple article visually explaining SQL `JOINS`.

Background

I'm a pretty visual person. Things seem to make more sense as a picture. I looked all over the Internet for a good graphical representation of SQL `JOINS`, but I couldn't find any to my liking. Some had good diagrams but lacked completeness (they didn't have all the possible JOINS), and some were just plain terrible. So, I decided to create my own and write an article about it.

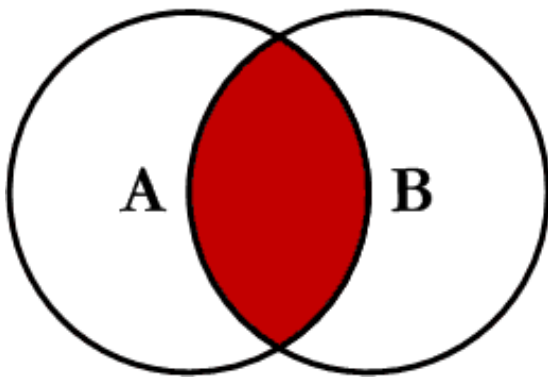
Using the code

I am going to discuss seven different ways you can return data from two relational tables. I will be excluding cross Joins and self referencing Joins. The seven Joins I will discuss are shown below:

1. `INNER JOIN`
2. `LEFT JOIN`
3. `RIGHT JOIN`
4. `OUTER JOIN`
5. `LEFT JOIN EXCLUDING INNER JOIN`
6. `RIGHT JOIN EXCLUDING INNER JOIN`
7. `OUTER JOIN EXCLUDING INNER JOIN`

For the sake of this article, I'll refer to 5, 6, and 7 as `LEFT EXCLUDING JOIN`, `RIGHT EXCLUDING JOIN`, and `OUTER EXCLUDING JOIN`, respectively. Some may argue that 5, 6, and 7 are not really joining the two tables, but for simplicity, I will still refer to these as Joins because you use a SQL Join in each of these queries (but exclude some records with a `WHERE` clause).

Inner JOIN

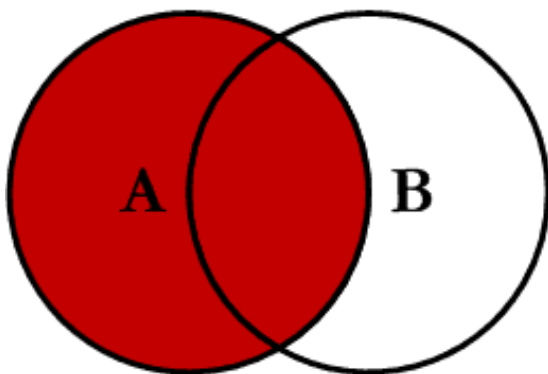


This is the simplest, most understood Join and is the most common. This query will return all of the records in the left table (table A) that have a matching record in the right table (table B). This Join is written as follows:

[Hide](#) [Copy Code](#)

```
SELECT <select_list>  
FROM Table_A A  
INNER JOIN Table_B B  
ON A.Key = B.Key
```

Left JOIN

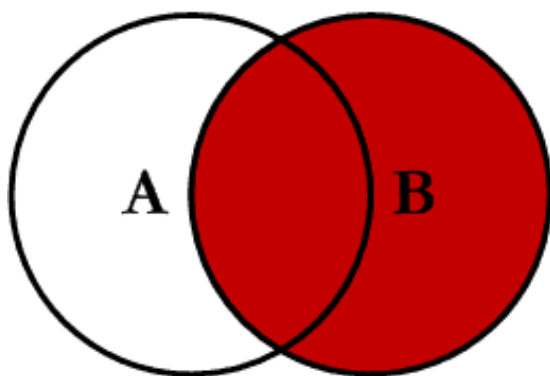


This query will return all of the records in the left table (table A) regardless if any of those records have a match in the right table (table B). It will also return any matching records from the right table. This Join is written as follows:

[Hide](#) [Copy Code](#)

```
SELECT <select_list>  
FROM Table_A A  
LEFT JOIN Table_B B  
ON A.Key = B.Key
```

Right JOIN

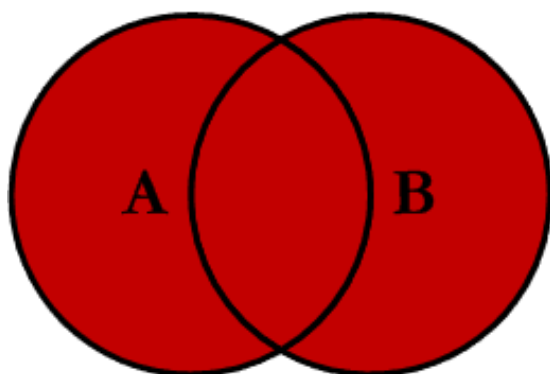


This query will return all of the records in the right table (table B) regardless if any of those records have a match in the left table (table A). It will also return any matching records from the left table. This Join is written as follows:

[Hide](#) [Copy Code](#)

```
SELECT <select_list>  
FROM Table_A A  
RIGHT JOIN Table_B B  
ON A.Key = B.Key
```

Outer JOIN

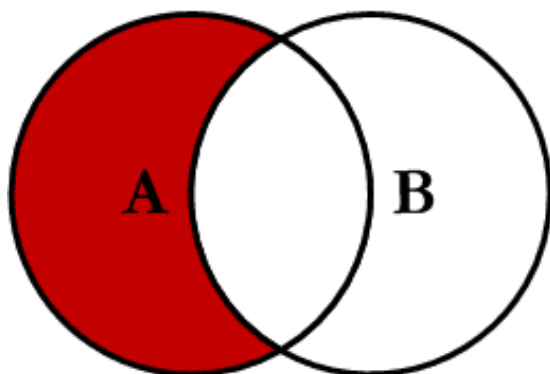


This Join can also be referred to as a **FULL OUTER JOIN** or a **FULL JOIN**. This query will return all of the records from both tables, joining records from the left table (table A) that match records from the right table (table B). This Join is written as follows:

[Hide](#) [Copy Code](#)

```
SELECT <select_list>  
FROM Table_A A  
FULL OUTER JOIN Table_B B  
ON A.Key = B.Key
```

Left Excluding JOIN

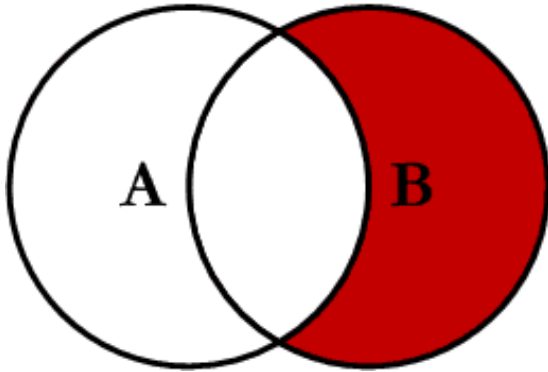


This query will return all of the records in the left table (table A) that do not match any records in the right table (table B). This Join is written as follows:

[Hide](#) [Copy Code](#)

```
SELECT <select_list>
FROM Table_A A
LEFT JOIN Table_B B
ON A.Key = B.Key
WHERE B.Key IS NULL
```

Right Excluding JOIN

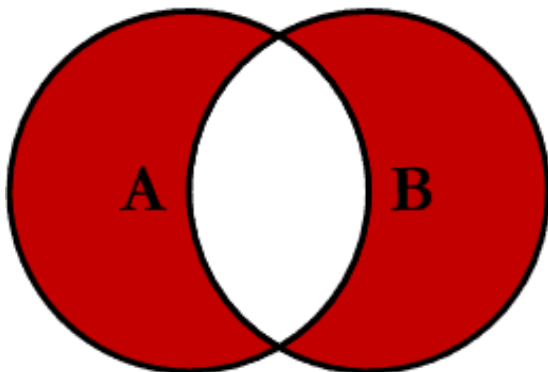


This query will return all of the records in the right table (table B) that do not match any records in the left table (table A). This Join is written as follows:

[Hide](#) [Copy Code](#)

```
SELECT <select_list>
FROM Table_A A
RIGHT JOIN Table_B B
ON A.Key = B.Key
WHERE A.Key IS NULL
```

Outer Excluding JOIN



This query will return all of the records in the left table (table A) and all of the records in the right table (table B) that do not match. I have yet to have a need for using this type of Join, but all of the others, I use quite frequently. This Join is written as follows:

[Hide](#) [Copy Code](#)

```
SELECT <select_list>
FROM Table_A A
FULL OUTER JOIN Table_B B
ON A.Key = B.Key
WHERE A.Key IS NULL OR B.Key IS NULL
```

Examples

Suppose we have two tables, *Table_A* and *Table_B*. The data in these tables are shown below:

[Hide](#) [Copy Code](#)
TABLE_A

PK	Value
1	FOX
2	COP
3	TAXI
6	WASHINGTON
7	DELL
5	ARIZONA
4	LINCOLN
10	LUCENT

TABLE_B

PK	Value
1	TROT
2	CAR
3	CAB
6	MONUMENT
7	PC
8	MICROSOFT
9	APPLE
11	SCOTCH

The results of the seven Joins are shown below:

[Hide](#) [Copy Code](#)

```
-- INNER JOIN
SELECT A.PK AS A_PK, A.Value AS A_Value,
       B.Value AS B_Value, B.PK AS B_PK
FROM Table_A A
INNER JOIN Table_B B
ON A.PK = B.PK
```

A_PK	A_Value	B_Value	B_PK
1	FOX	TROT	1
2	COP	CAR	2
3	TAXI	CAB	3
6	WASHINGTON	MONUMENT	6
7	DELL	PC	7

(5 row(s) affected)

[Hide](#) [Copy Code](#)

```
-- LEFT JOIN
SELECT A.PK AS A_PK, A.Value AS A_Value,
       B.Value AS B_Value, B.PK AS B_PK
FROM Table_A A
LEFT JOIN Table_B B
ON A.PK = B.PK
```

A_PK	A_Value	B_Value	B_PK
1	FOX	TROT	1
2	COP	CAR	2
3	TAXI	CAB	3
4	LINCOLN	NULL	NULL
5	ARIZONA	NULL	NULL
6	WASHINGTON	MONUMENT	6
7	DELL	PC	7
10	LUCENT	NULL	NULL

(8 row(s) affected)

-- RIGHT JOIN

```
SELECT A.PK AS A_PK, A.Value AS A_Value,
       B.Value AS B_Value, B.PK AS B_PK
FROM Table_A A
RIGHT JOIN Table_B B
ON A.PK = B.PK
```

A_PK	A_Value	B_Value	B_PK
1	FOX	TROT	1
2	COP	CAR	2
3	TAXI	CAB	3
6	WASHINGTON	MONUMENT	6
7	DELL	PC	7
NULL	NULL	MICROSOFT	8
NULL	NULL	APPLE	9
NULL	NULL	SCOTCH	11

(8 row(s) affected)

Hide Copy Code

-- OUTER JOIN

```
SELECT A.PK AS A_PK, A.Value AS A_Value,
       B.Value AS B_Value, B.PK AS B_PK
FROM Table_A A
FULL OUTER JOIN Table_B B
ON A.PK = B.PK
```

A_PK	A_Value	B_Value	B_PK
1	FOX	TROT	1
2	COP	CAR	2
3	TAXI	CAB	3
6	WASHINGTON	MONUMENT	6
7	DELL	PC	7
NULL	NULL	MICROSOFT	8
NULL	NULL	APPLE	9
NULL	NULL	SCOTCH	11
5	ARIZONA	NULL	NULL
4	LINCOLN	NULL	NULL
10	LUCENT	NULL	NULL

(11 row(s) affected)

Hide Copy Code

-- LEFT EXCLUDING JOIN

```
SELECT A.PK AS A_PK, A.Value AS A_Value,
       B.Value AS B_Value, B.PK AS B_PK
FROM Table_A A
LEFT JOIN Table_B B
ON A.PK = B.PK
WHERE B.PK IS NULL
```

A_PK	A_Value	B_Value	B_PK
4	LINCOLN	NULL	NULL
5	ARIZONA	NULL	NULL
10	LUCENT	NULL	NULL

(3 row(s) affected)

Hide Copy Code

-- RIGHT EXCLUDING JOIN

```
SELECT A.PK AS A_PK, A.Value AS A_Value,
       B.Value AS B_Value, B.PK AS B_PK
FROM Table_A A
RIGHT JOIN Table_B B
ON A.PK = B.PK
```

```
WHERE A.PK IS NULL
```

A_PK	A_Value	B_Value	B_PK
NULL	NULL	MICROSOFT	8
NULL	NULL	APPLE	9
NULL	NULL	SCOTCH	11

```
(3 row(s) affected)
```

[Hide](#) [Copy Code](#)

-- OUTER EXCLUDING JOIN

```
SELECT A.PK AS A_PK, A.Value AS A_Value,
B.Value AS B_Value, B.PK AS B_PK
FROM Table_A A
FULL OUTER JOIN Table_B B
ON A.PK = B.PK
WHERE A.PK IS NULL
OR B.PK IS NULL
```

A_PK	A_Value	B_Value	B_PK
NULL	NULL	MICROSOFT	8
NULL	NULL	APPLE	9
NULL	NULL	SCOTCH	11
5	ARIZONA	NULL	NULL
4	LINCOLN	NULL	NULL
10	LUCENT	NULL	NULL

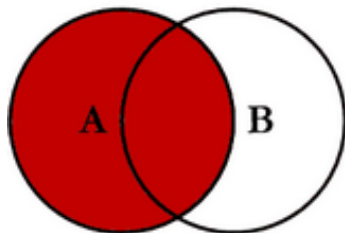
```
(6 row(s) affected)
```

Note on the **OUTER JOIN** that the inner joined records are returned first, followed by the right joined records, and then finally the left joined records (at least, that's how my Microsoft SQL Server did it; this, of course, is without using any **ORDER BY** statement).

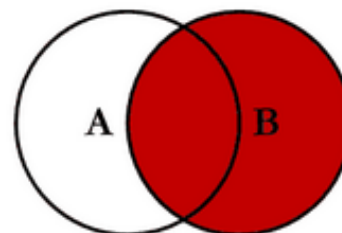
You can visit the Wikipedia article for more info [here](#) (however, the entry is not graphical).

I've also created a cheat sheet that you can print out if needed. If you right click on the image below and select "Save Target As...", you will download the full size image.

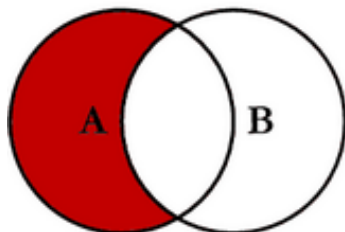
SQL JOINS



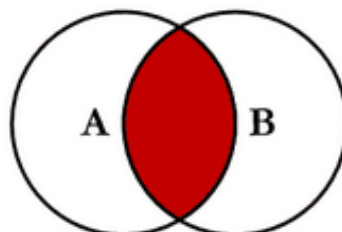
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
```



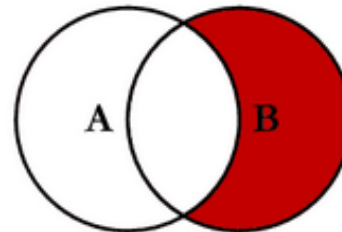
```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
```



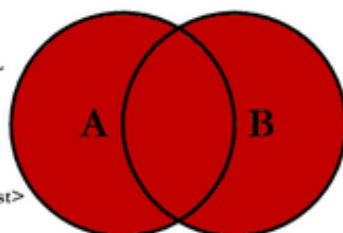
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
WHERE B.Key IS NULL
```



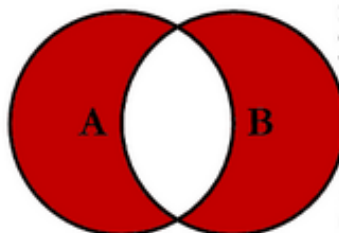
```
SELECT <select_list>
FROM TableA A
INNER JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
```



```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
OR B.Key IS NULL
```

© C.L. Moffitt, 2008

History

- Initial release -- 02/03/2009.
- Version 1.0 -- 02/04/2009 -- Fixed cheat sheet and minor typos.

License

This article, along with any associated source code and files, is licensed under [The Code Project Open License \(CPOL\)](#)

Share



EMAIL



About the Author



C.L. Moffatt

Green Sand Software
United States 🇺🇸

No Biography provided

Comments and Discussions

You must [Sign In](#) to use this message board.

Search Comments

Go

First Prev Next

How would we do two joins (on different variables)? 📌

Rumah Dijual 11-Jan-15 4:23

Re: How would we do two joins (on different variables)? 📌

C.L. Moffatt 23-Jan-15 14:23

Thanks 📌

Member 9859414 25-Dec-14 18:15

Credit notice on stackoverflow 📌

daantimmer 22-Dec-14 1:04

Re: Credit notice on stackoverflow 📌

C.L. Moffatt 23-Jan-15 14:24

Great article! 📌

dgirard 16-Oct-14 2:22

Explanation 📌

spgun 15-Oct-14 0:39

Re: Explanation 📌

C.L. Moffatt 23-Jan-15 14:24

My vote of 5 📌

Member 11136616 7-Oct-14 16:32

visual representation of sql joins 📌

pai1009 25-Sep-14 22:32

My vote of 5 📌

Sibeesh KV 19-Sep-14 19:44

My vote of 5 📌

Sibeesh Venu 1-Aug-14 6:17

My vote of 5 

Shmuel Zang 22-Jul-14 18:23

Thnak you so much...such a nice article 

Member 10678825 18-Jul-14 9:08

My vote of 1 

grenac 16-Jul-14 7:20

Thanks 

_shivu 4-Jul-14 19:32

My vote of 5 

Member 10911981 28-Jun-14 16:19

About SQL Server Joins tutorials 

vandykish 16-May-14 6:37

Joins Made EASY by C.L.Moffatt 

Bala-Seequel 26-Mar-14 19:21

joins 

Member 10670773 14-Mar-14 6:54

great post ! 

Member 10622226 24-Feb-14 11:09

Thank you... It was very Helpful 

Renuka Rane 28-Jan-14 17:56

Outer Excluding JOIN - I just needed this 

GuyThiebaut 26-Jan-14 23:10

My vote of 5 

mielniczuk 20-Jan-14 7:53

Nice Explanation :) Easily understood the concepts 

rajas2406 6-Jan-14 19:54

Maybe not a JOIN question... 


AlexEvans 26-Sep-13 0:01

Vote of 5 

TrollTier 23-Sep-13 4:08

How is join different from regular select ~ where ~ ? 

Member 10262760 9-Sep-13 9:32

Re: How is join different from regular select ~ where ~ ? 

C.L. Moffatt 10-Sep-13 8:15

My vote of 5 

colin harris 16-Aug-13 0:48

My vote of 5 

oustaz 22-Jul-13 16:55

My Vote of 5 

Anoop Kr Sharma 22-Jul-13 16:14

JOINS 

Newbies1002 11-Jun-13 10:16

My vote of 5 

Arlen Navasartian 30-May-13 5:10














My vote of 5 

Maciej Los 5-Apr-13 2:39

My vote of 5 

cycnus 31-Mar-13 14:53


thank u 

Amra Rama 12-Mar-13 5:22
My vote of 5  Chathura Roshan 22-Jan-13 22:15
My vote of 5  AndreHPoffo 23-Nov-12 5:59
JOIN  mitu sinha 13-Nov-12 21:59
Re: JOIN  C.L. Moffatt 14-Nov-12 4:04
Left Join  mah2244 6-Nov-12 2:59
Re: Left Join  C.L. Moffatt 14-Nov-12 3:59
My vote of 5  kamalsamant 25-Oct-12 21:37
My vote of 5  Farhan Ghumra 21-Sep-12 0:14
My vote of 5  wrecklass 9-Sep-12 13:21
My vote of 5  MANISH RASTOGI 11-Jun-12 2:30
My vote of 5  SeniorCrispy 14-May-12 5:57
Thanks a lot  ilneyo 3-Apr-12 11:54
My vote of 5  tgurinder 28-Mar-12 2:02
Refresh 1 2 3 Next »

 General
  News
  Suggestion
  Question
  Bug
  Answer
  Joke
  Rant
  Admin

Use Ctrl+Left/Right to switch messages, Ctrl+Up/Down to switch threads, Ctrl+Shift+Left/Right to switch pages.

[Permalink](#) |
 [Advertise](#) |
 [Privacy](#) |
 [Terms of Use](#) |
 [Mobile](#)
 Web04 | 2.8.150618.2 | Last Updated 4 Feb 2009

 Select Language | ▼
 Layout: fixed | fluid

Article Copyright 2009 by C.L. Moffatt
 Everything else Copyright © [CodeProject](#), 1999-2015