Merge Queries

Merge Queries are used when we need to add one or more columns to a Query from another Query. Merge is similar to JOINS in T-SQL or Other Databases.

Now we will see how to merge queries with an example. Take the EMP and DEPT Queries as shown below.

EMP

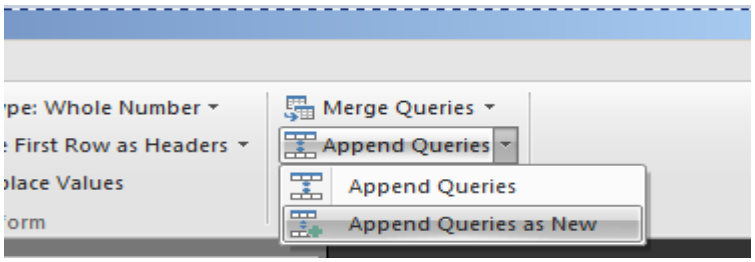
EMPNO	ENAME	JOB	MGR	SAL	DEPTNO
7369	SMITH	CLERK	1001	800	20
7499	ALLEN	SALESMAN	7566	1600	30
7521	WARD	SALESMAN	7566	1250	30
7566	JONES	MANAGER	7839	2975	20
7788	SCOTT	ANALYST	7839	3000	20
7839	KING	PRESIDENT		5000	10
7934	MILLER	CLERK	1001	1300	10
1001	SUNIL	TRAINER	7839	10000	50

DEPT

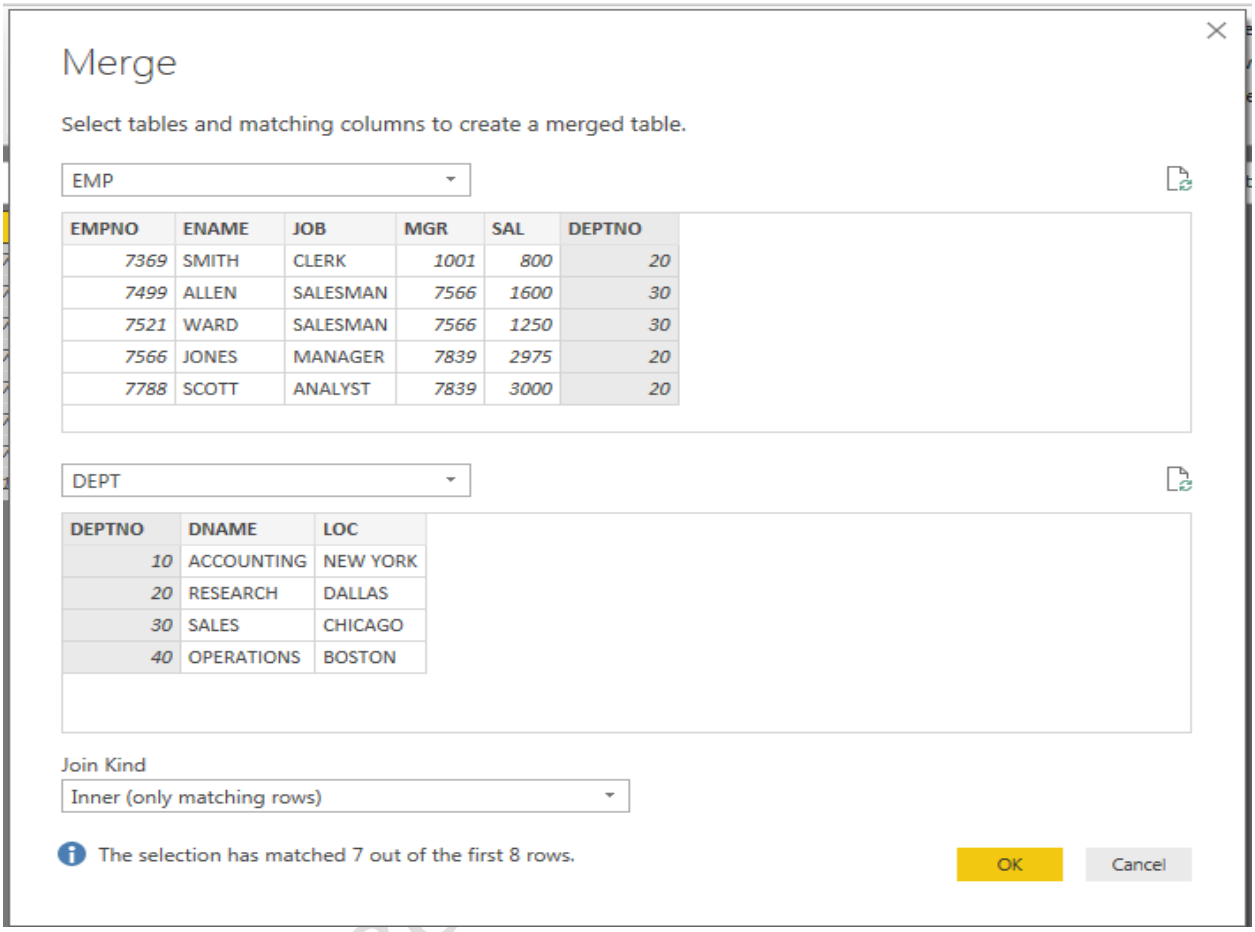
DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

If you want to see employee details along with the department details in the same query then you need to merge EMP, DEPT Queries. Now we will see how to merge both of them.

Get both the Queries (EMP, DEPT) into Power Query. Select EMP Query first, and then select Merge Queries (Merge Queries as New) from the Home tab on the ribbon as shown below.



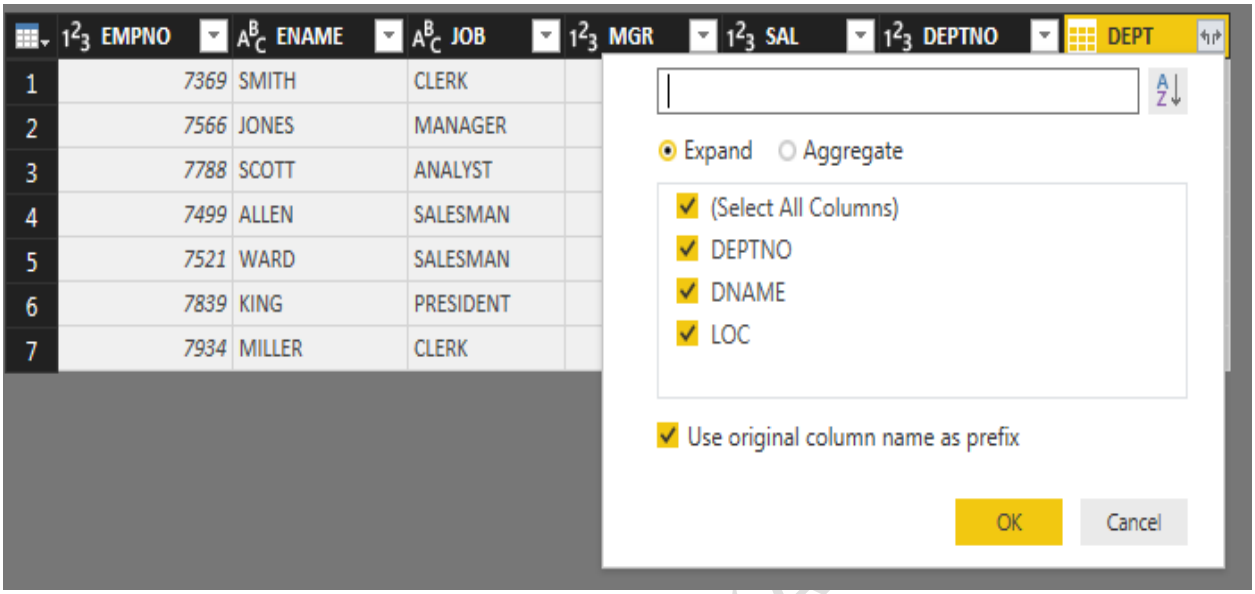
Next the Merge window appears prompting us to select the tables, matching columns and **Join Kind** to create a merged table as shown below. We will discuss more about Join Kind later.



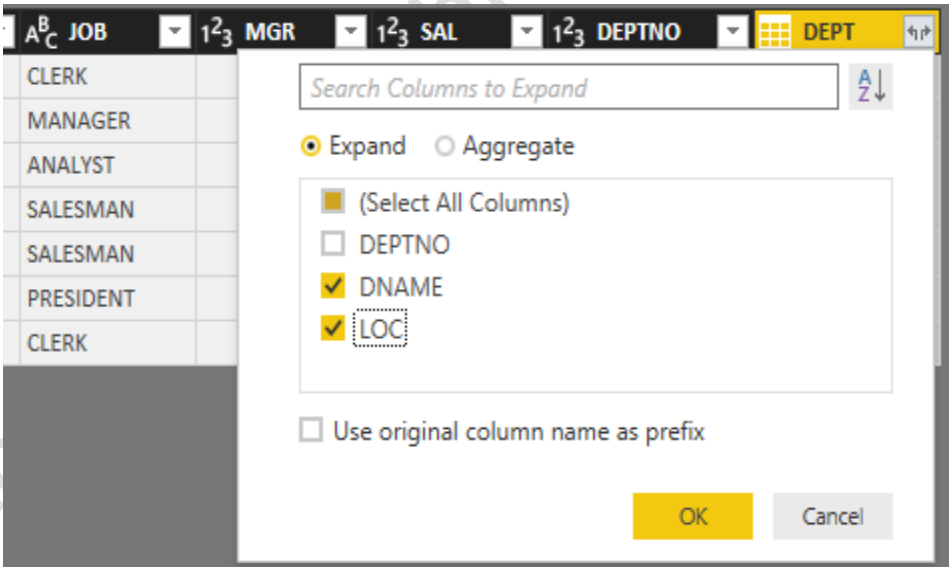
Once you select tables, matching columns and Join Kind Then Ok button enabled. Once you press **OK**, a NewColumn is created at the end of the query as shown below. The NewColumn is the contents of the table (query) that was merged with the existing query. All columns from the merged query are condensed into the NewColumn.

	EMPNO	ENAME	JOB	MGR	SAL	DEPTNO	DEPT
1	7369	SMITH	CLERK	1001	800	20	Table
2	7566	JONES	MANAGER	7839	2975	20	Table
3	7788	SCOTT	ANALYST	7839	3000	20	Table
4	7499	ALLEN	SALESMAN	7566	1600	30	Table
5	7521	WARD	SALESMAN	7566	1250	30	Table
6	7839	KING	PRESIDENT	null	5000	10	Table
7	7934	MILLER	CLERK	1001	1300	10	Table

To Expand the merged table, and select which columns to include, select the expand icon (⇲). The **Expand** window appears as shown below.



In this case, we only want the DNAME and LOC columns, so we select only that columns and then select OK. We clear the checkbox from Use original column name as prefix because we don't need or want that, if we leave that selected, the merged column would be named **DEPT.DNAME, DEPT.LOC** (the original column name, or NewColumn, then a dot, then the name of the column being brought into the query).



Select OK to see the merged query output as shown below.

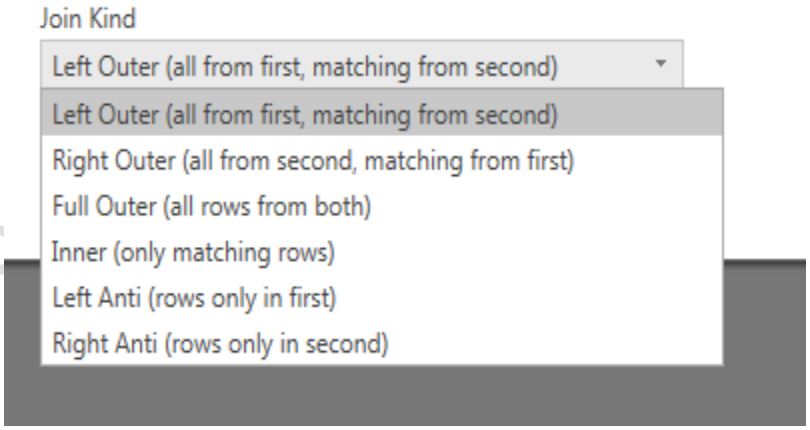
	EMPNO	ENAME	JOB	MGR	SAL	DEPTNO	DNAME	LOC
1	7369	SMITH	CLERK	1001	800	20	RESEARCH	DALLAS
2	7566	JONES	MANAGER	7839	2975	20	RESEARCH	DALLAS
3	7788	SCOTT	ANALYST	7839	3000	20	RESEARCH	DALLAS
4	7499	ALLEN	SALESMAN	7566	1600	30	SALES	CHICAGO
5	7521	WARD	SALESMAN	7566	1250	30	SALES	CHICAGO
6	7839	KING	PRESIDENT	null	5000	10	ACCOUNTING	NEW YORK
7	7934	MILLER	CLERK	1001	1300	10	ACCOUNTING	NEW YORK

You can easily use multiple columns in join condition for merging two data sets. Just select them in an order with holding Ctrl key of the keyboard.

Types of Joins / Join Kinds / Merge Type

There are 6 types of joins you can perform for merging the queries by default as shown below. Each of these joins gives you different results in merge query output. Default Join Kind is Left Outer. Let's see what their difference is.

1. Left Outer (all from first, matching from second)
2. Right Outer (all from second, matching from first)
3. Full Outer (all rows from both)
4. Inner (only matching rows)
5. Left Anti (rows only in first)
6. Right Anti (rows only in second)



We use below EMP and DEPT Tables to illustrate these Join Types

EMP

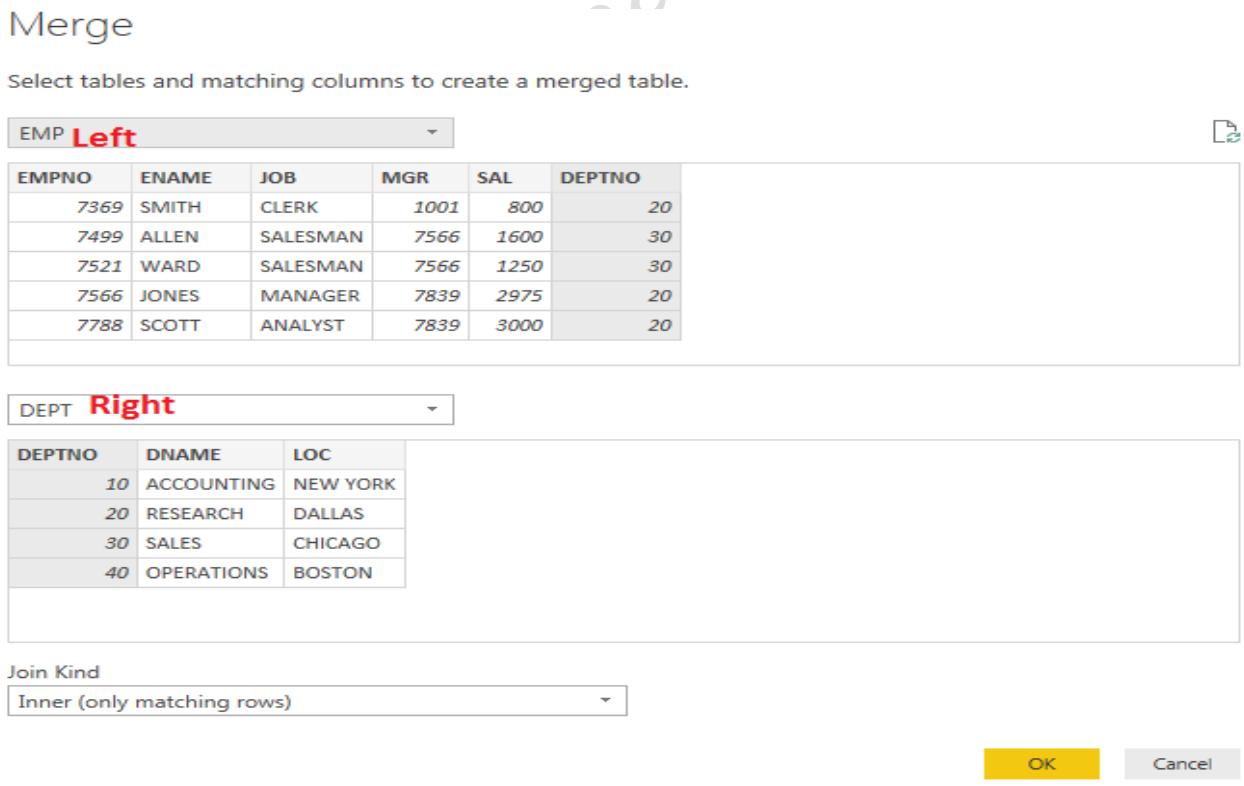
EMPNO	ENAME	JOB	MGR	SAL	DEPTNO
7369	SMITH	CLERK	1001	800	20
7499	ALLEN	SALESMAN	7566	1600	30
7521	WARD	SALESMAN	7566	1250	30
7566	JONES	MANAGER	7839	2975	20
7788	SCOTT	ANALYST	7839	3000	20
7839	KING	PRESIDENT		5000	10
7934	MILLER	CLERK	1001	1300	10
1001	SUNIL	TRAINER	7839	10000	50

DEPT

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

Left and Right

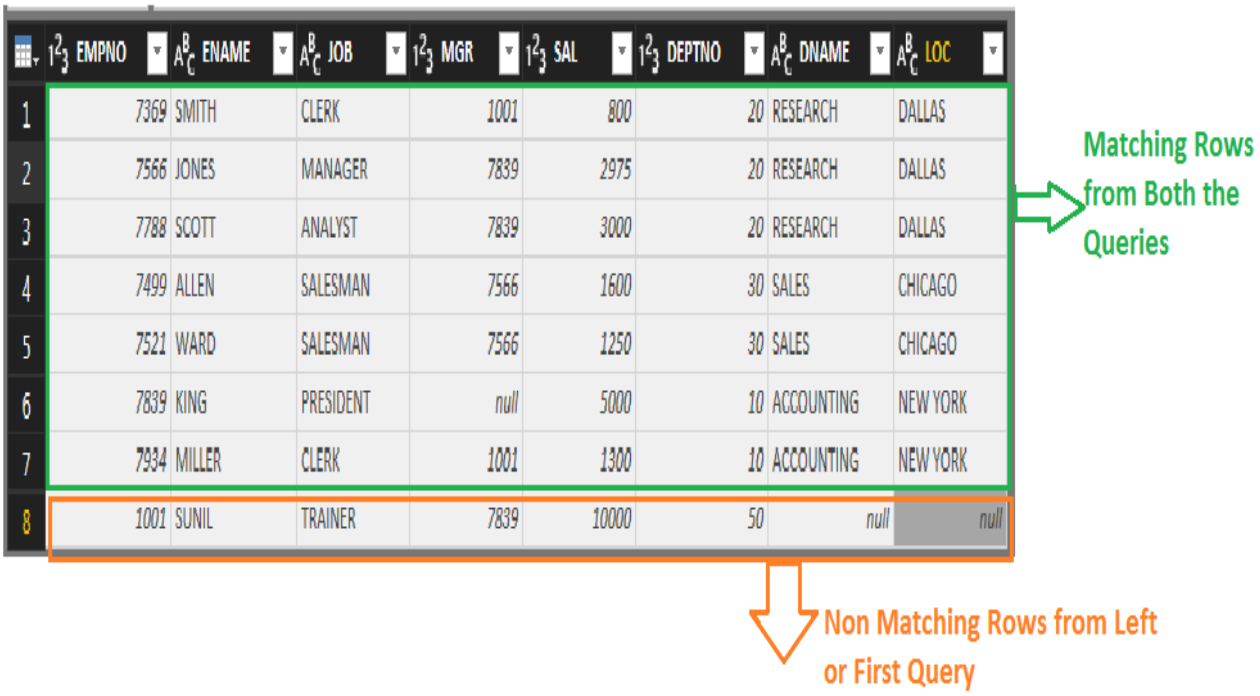
To start, you need to know the concept of Left and Right tables (or queries). When you merge two data sets with each other, the first query is considered as LEFT and the second as RIGHT.



In the above example EMP is Left and First and DEPT is Right and Second. Understanding this is important, because most of Join Kinds works with the concept of left or right or both.

Left Outer (All from first, matching from second)

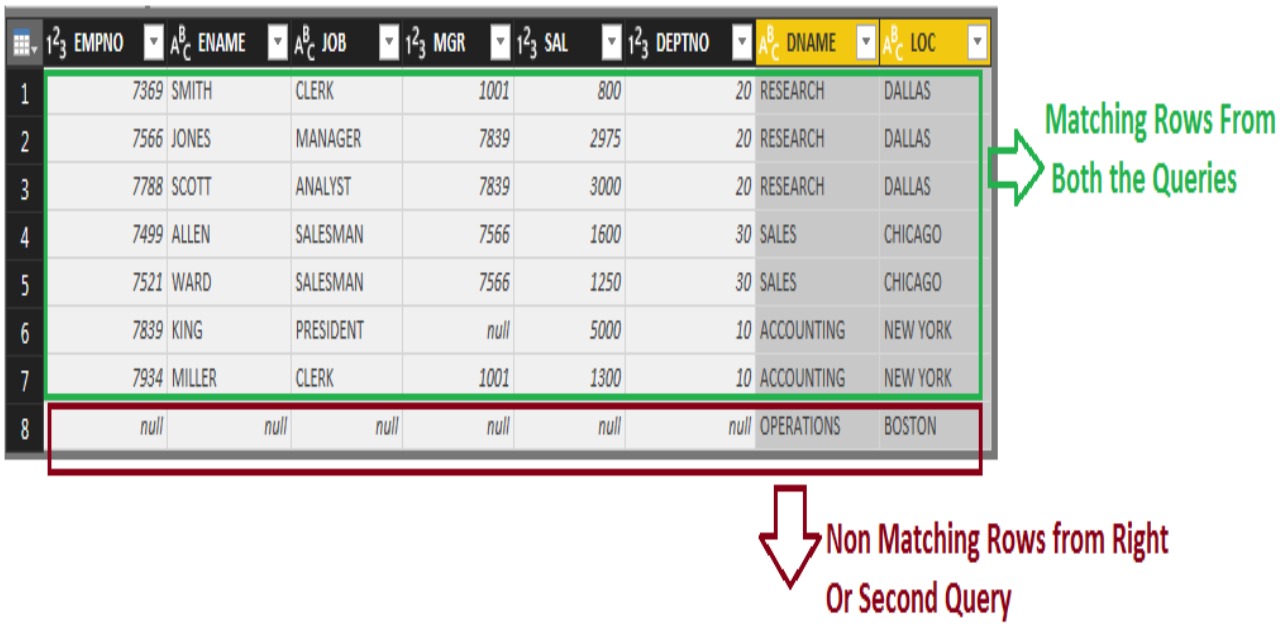
The first type of Join/Merge is Left Outer. All records from this query (LEFT or FIRST) will be showed in the result set plus their matching rows in the right (or second Query). This type of join is the default type. If you don't specify the Join Kind, it will be always Left Outer. The result set of EMP and DEPT with Join Kind Left Outer is shown below. We can see that all the Rows from EMP table are shown in the result. We don't have the DEPTNO=50 in DEPT table still it is showing in the merged query result as we select **Left Outer** Join Kind.



	EMPNO	ENAME	JOB	MGR	SAL	DEPTNO	DNAME	LOC
1	7369	SMITH	CLERK	1001	800	20	RESEARCH	DALLAS
2	7566	JONES	MANAGER	7839	2975	20	RESEARCH	DALLAS
3	7788	SCOTT	ANALYST	7839	3000	20	RESEARCH	DALLAS
4	7499	ALLEN	SALESMAN	7566	1600	30	SALES	CHICAGO
5	7521	WARD	SALESMAN	7566	1250	30	SALES	CHICAGO
6	7839	KING	PRESIDENT	null	5000	10	ACCOUNTING	NEW YORK
7	7934	MILLER	CLERK	1001	1300	10	ACCOUNTING	NEW YORK
8	1001	SUNIL	TRAINER	7839	10000	50	null	null

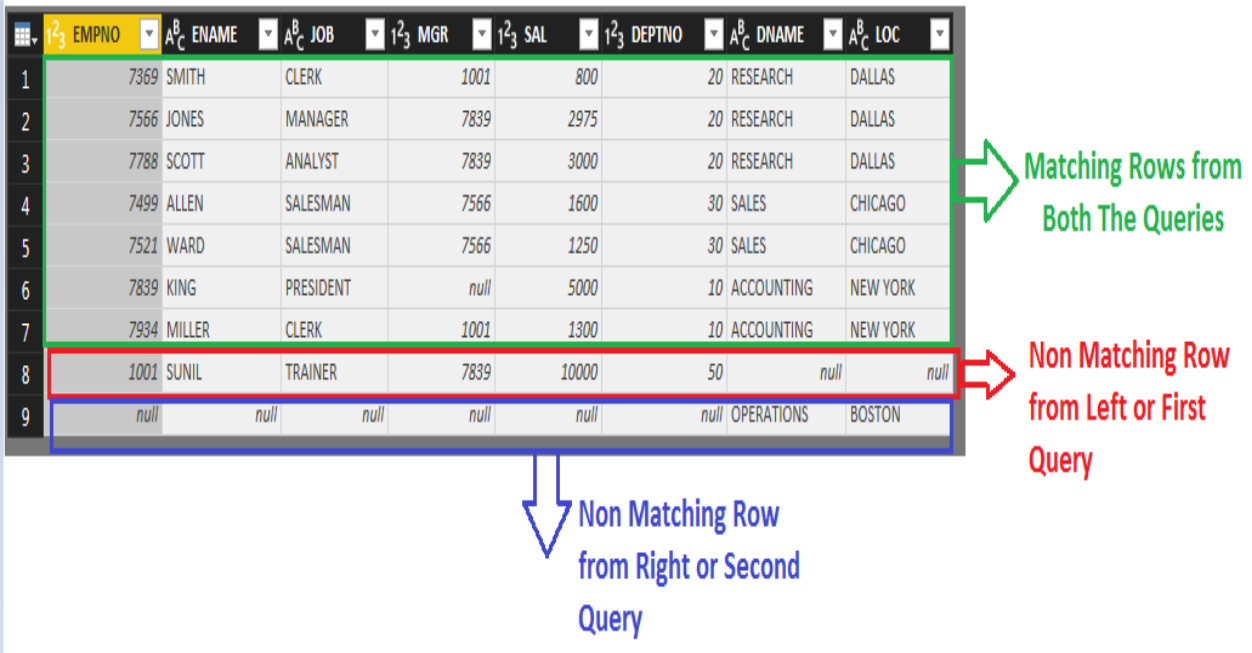
Right Outer (all rows from second, matching from first)

With this type of Join, you get all rows from the RIGHT (or second) Query, with their matching rows from left (or first Query). We can see that all the Rows from DEPT Query are shown in the result along with matching rows from Both the Queries.



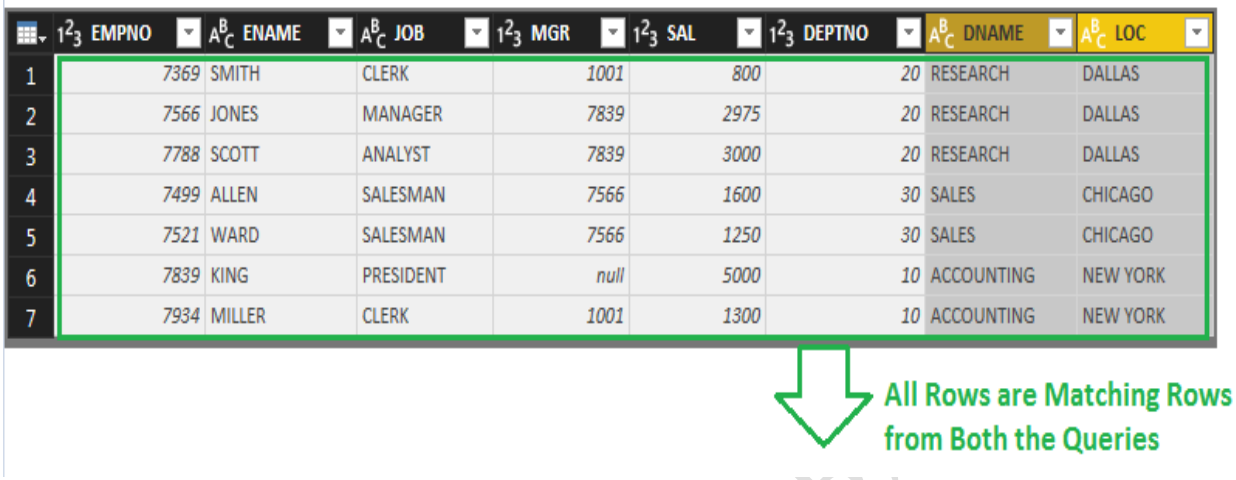
Full Outer (all rows from both)

This Join Kind will return all rows from both Queries (matching and non-matching). You will have all non-matching rows from first Query, and all non-matching rows from the second Query, and all matching rows from both the Queries. In the below image you can see Matching Rows from both the Queries in Green Box, Non matching rows from First or Left Query in Red Box and Non matching rows from Second or Right Query in Blue Box.



Inner (only matching rows)

This Join Kind will only return matching rows as shown below. You will not have any record with null values (because these records generate as a result of not matching).

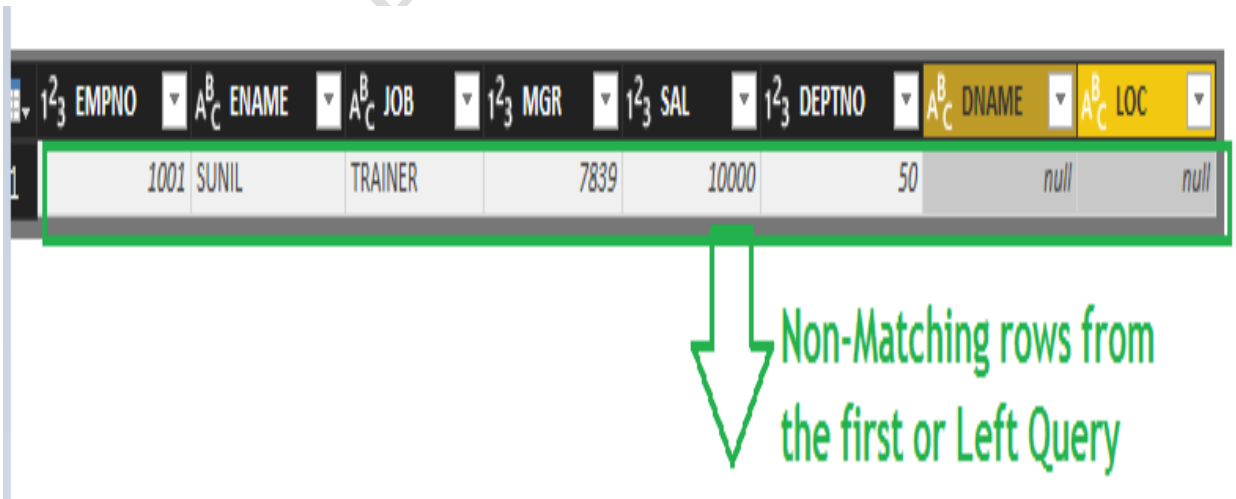


	EMPNO	ENAME	JOB	MGR	SAL	DEPTNO	DNAME	LOC
1	7369	SMITH	CLERK	1001	800	20	RESEARCH	DALLAS
2	7566	JONES	MANAGER	7839	2975	20	RESEARCH	DALLAS
3	7788	SCOTT	ANALYST	7839	3000	20	RESEARCH	DALLAS
4	7499	ALLEN	SALESMAN	7566	1600	30	SALES	CHICAGO
5	7521	WARD	SALESMAN	7566	1250	30	SALES	CHICAGO
6	7839	KING	PRESIDENT	null	5000	10	ACCOUNTING	NEW YORK
7	7934	MILLER	CLERK	1001	1300	10	ACCOUNTING	NEW YORK

Let Anti (rows only in first)

If you are only interested in rows from the LEFT (first) Query, then this is the option to select. This means rows that are in the First or Left Query and DO NOT match with the Second or Right Query. So this Join Kind returns only non-matching rows from the first or Left Query. With Anti options you always get null for the second data set, because these rows don't exist there. Anti-options are good for finding rows that exists in one Query but not in the other one.

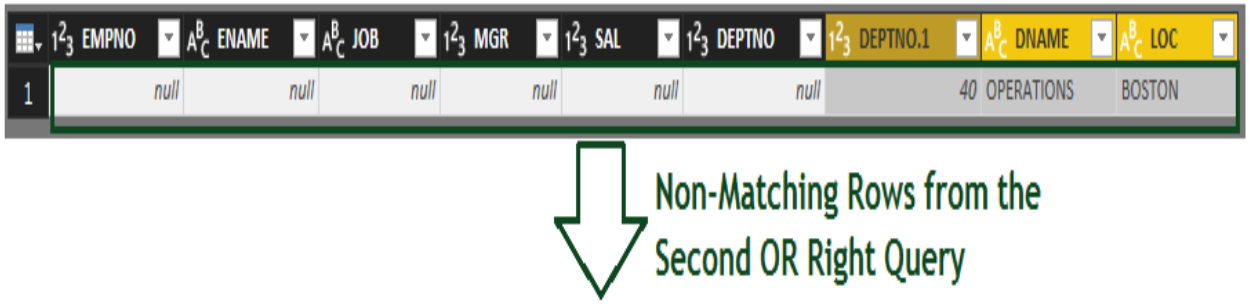
This Join Kind will find the only one row that exists in the EMP Query and does not match with any of rows in the DEPT Query.



	EMPNO	ENAME	JOB	MGR	SAL	DEPTNO	DNAME	LOC
1	1001	SUNIL	TRAINER	7839	10000	50	null	null

Right Anti (rows only in second)

This Join Kind will give you only non-matching rows, this time from the Second (Right) Query. You can find out what rows in the Right or Second Query are not matching with the Left or First Query.



Cartesian Join or Cross Join

By Default, in Power BI we have 6 types of joins and in that we don't have on more type of join that is Cross Join or Cartesian Join. To get the cross join output we have a work around that we need to add a custom column for both the queries and populate that column with a unique value.