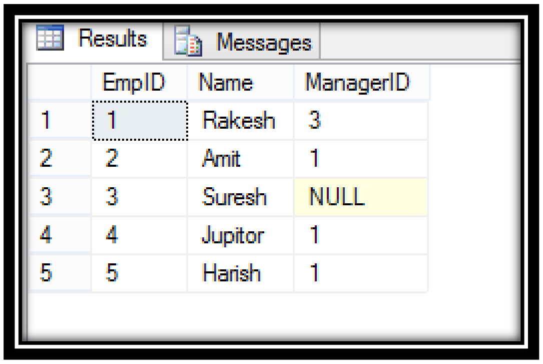
Ways of replace of NULL value

- ISNULL() function
- Case Statement
- COALESCE() function

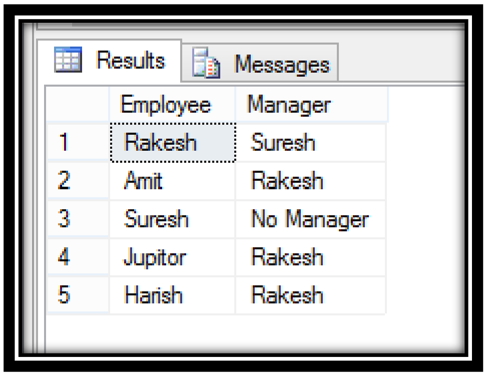


	EmpID	Name	ManagerID
1	1	Rakesh	3
2	2	Amit	1
3	3	Suresh	NULL
4	4	Jupitor	1
5	5	Harish	1

tblEmp Main Table

ISNull function

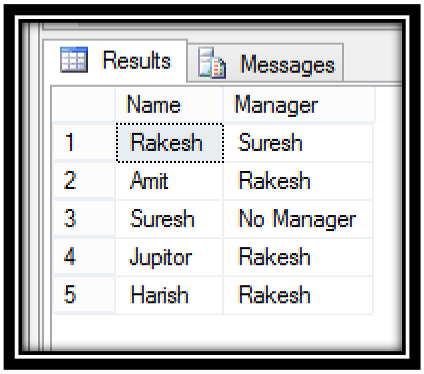
- Write a Query display NO MANAGER as per give below snap shot from the above table data.



	Employee	Manager
1	Rakesh	Suresh
2	Amit	Rakesh
3	Suresh	No Manager
4	Jupitor	Rakesh
5	Harish	Rakesh

COALESCE Function

- Write a Query display NO MANAGER using COALESCE function as per give below snap shot from the top given table data.



	Name	Manager
1	Rakesh	Suresh
2	Amit	Rakesh
3	Suresh	No Manager
4	Jupitor	Rakesh
5	Harish	Rakesh

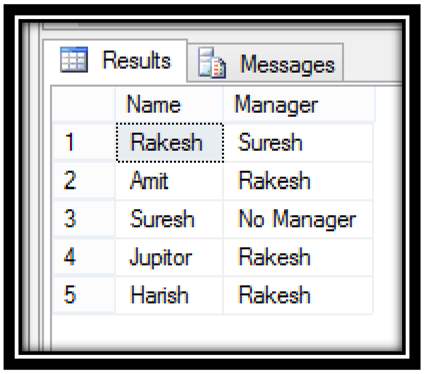
Case Statement

The CASE statement goes through conditions and returns a value when the first condition is met (like an IF-THEN-ELSE statement). So, once a condition is true, it will stop reading and return the result. If no conditions are true, it returns the value in the ELSE clause.

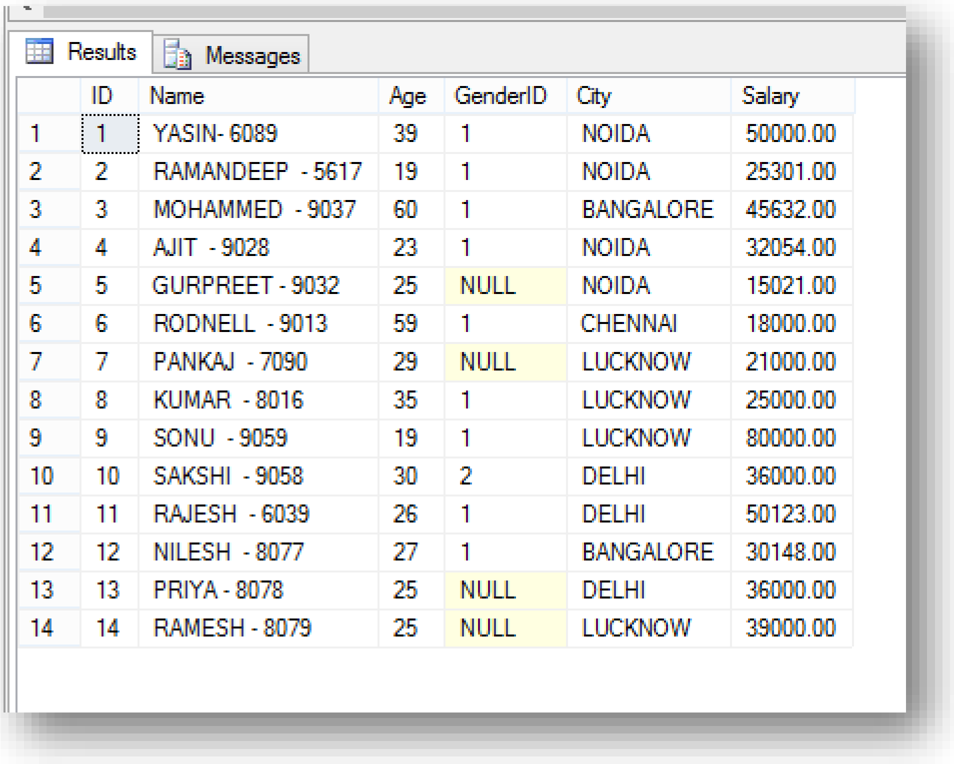
If there is no ELSE part and no conditions are true, it returns NULL

```
CASE
    WHEN condition1 THEN result1
    WHEN condition2 THEN result2
    WHEN conditionN THEN resultN
    ELSE result
END;
```

- Write a Query display NO MANAGER using Case function as per give below snap shot from the top given table data.

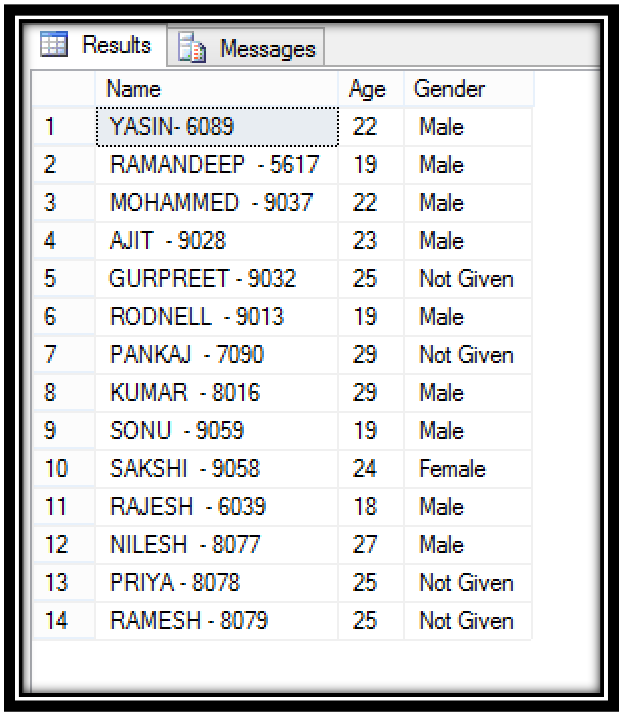


	Name	Manager
1	Rakesh	Suresh
2	Amit	Rakesh
3	Suresh	No Manager
4	Jupitor	Rakesh
5	Harish	Rakesh



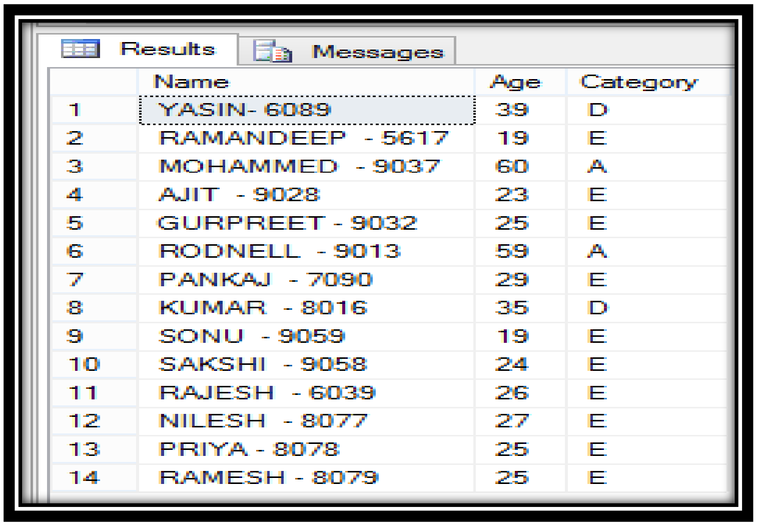
	ID	Name	Age	GenderID	City	Salary
1	1	YASIN- 6089	39	1	NOIDA	50000.00
2	2	RAMANDEEP - 5617	19	1	NOIDA	25301.00
3	3	MOHAMMED - 9037	60	1	BANGALORE	45632.00
4	4	AJIT - 9028	23	1	NOIDA	32054.00
5	5	GURPREET - 9032	25	NULL	NOIDA	15021.00
6	6	RODNELL - 9013	59	1	CHENNAI	18000.00
7	7	PANKAJ - 7090	29	NULL	LUCKNOW	21000.00
8	8	KUMAR - 8016	35	1	LUCKNOW	25000.00
9	9	SONU - 9059	19	1	LUCKNOW	80000.00
10	10	SAKSHI - 9058	30	2	DELHI	36000.00
11	11	RAJESH - 6039	26	1	DELHI	50123.00
12	12	NILESH - 8077	27	1	BANGALORE	30148.00
13	13	PRIYA - 8078	25	NULL	DELHI	36000.00
14	14	RAMESH - 8079	25	NULL	LUCKNOW	39000.00

- Write a Query display Male and Female in spite of 1 & 2 (1 for Male, 2 for Female). USING CASE STATEMENT



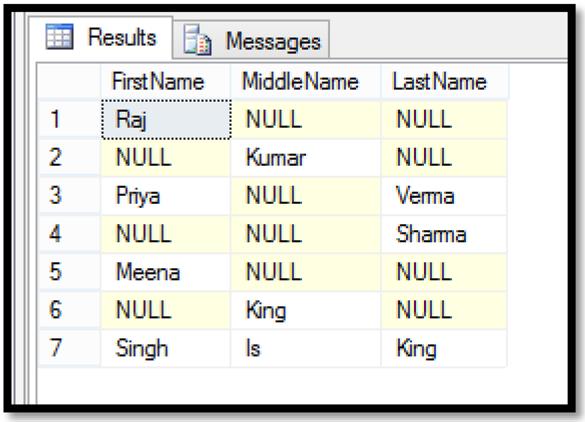
	Name	Age	Gender
1	YASIN- 6089	22	Male
2	RAMANDEEP - 5617	19	Male
3	MOHAMMED - 9037	22	Male
4	AJIT - 9028	23	Male
5	GURPREET - 9032	25	Not Given
6	RODNELL - 9013	19	Male
7	PANKAJ - 7090	29	Not Given
8	KUMAR - 8016	29	Male
9	SONU - 9059	19	Male
10	SAKSHI - 9058	24	Female
11	RAJESH - 6039	18	Male
12	NILESH - 8077	27	Male
13	PRIYA - 8078	25	Not Given
14	RAMESH - 8079	25	Not Given

- Write a Query display Category (A when age >=55, B when age (55->35), C when age(35 to 25) and E age <25)



	Name	Age	Category
1	YASIN- 6089	39	D
2	RAMANDEEP - 5617	19	E
3	MOHAMMED - 9037	60	A
4	AJIT - 9028	23	E
5	GURPREET - 9032	25	E
6	RODNELL - 9013	59	A
7	PANKAJ - 7090	29	E
8	KUMAR - 8016	35	D
9	SONU - 9059	19	E
10	SAKSHI - 9058	24	E
11	RAJESH - 6039	26	E
12	NILESH - 8077	27	E
13	PRIYA - 8078	25	E
14	RAMESH - 8079	25	E

COALESCE function – Return the first non-return value



	FirstName	MiddleName	LastName
1	Raj	NULL	NULL
2	NULL	Kumar	NULL
3	Priya	NULL	Verma
4	NULL	NULL	Sharma
5	Meena	NULL	NULL
6	NULL	King	NULL
7	Singh	Is	King

Union and Union All

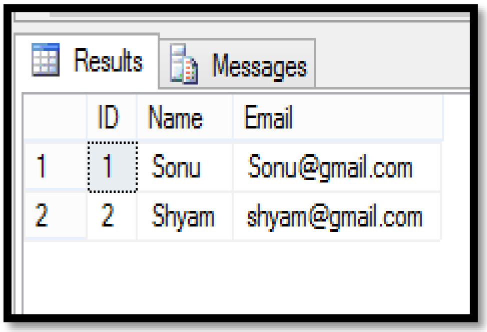
The UNION operator is used to combine the result-set of two or more SELECT statements.

- Each SELECT statement within UNION must have the same number of columns
- The columns must also have similar data types
- The columns in each SELECT statement must also be in the same order

Both UNION and UNION ALL concatenate the result of two different SQLs. They differ in the way they handle duplicates.

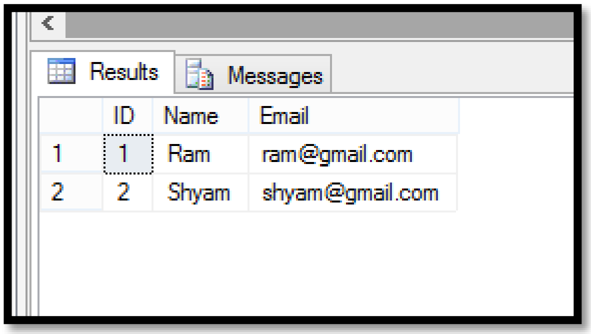
- UNION performs a DISTINCT on the result set, eliminating any duplicate rows.
- UNION ALL does not remove duplicates, and it therefore faster than UNION.

HR Table



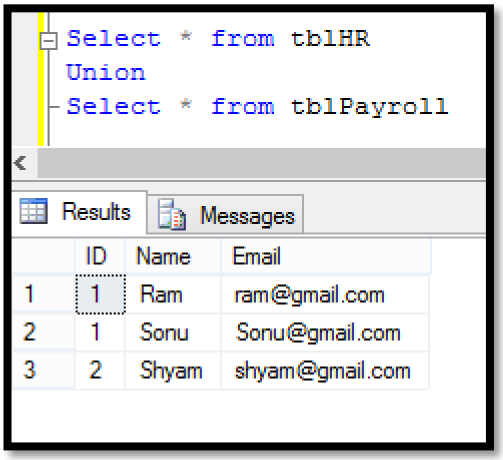
	ID	Name	Email
1	1	Sonu	Sonu@gmail.com
2	2	Shyam	shyam@gmail.com

Payroll Table



	ID	Name	Email
1	1	Ram	ram@gmail.com
2	2	Shyam	shyam@gmail.com

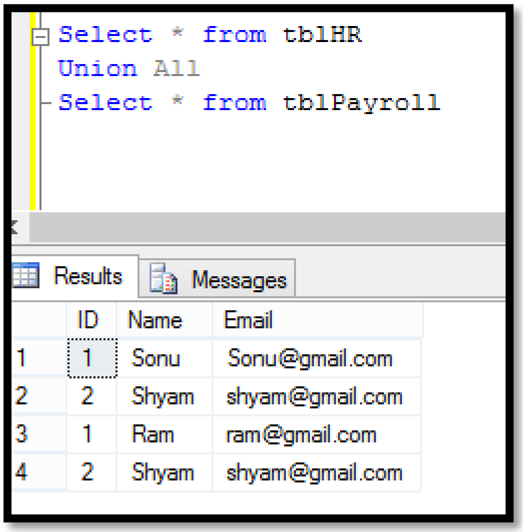
UNION Example



```
Select * from tblHR
Union
Select * from tblPayroll
```

	ID	Name	Email
1	1	Ram	ram@gmail.com
2	1	Sonu	Sonu@gmail.com
3	2	Shyam	shyam@gmail.com

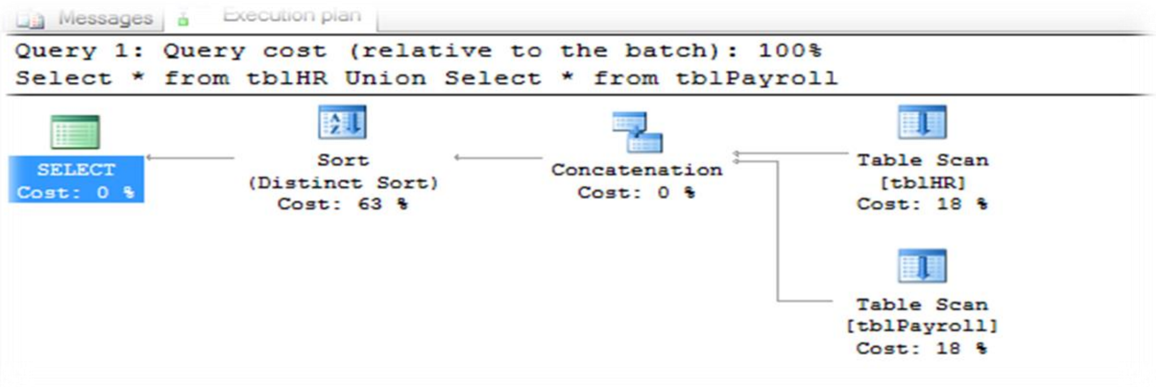
UNION ALL Example



```
Select * from tblHR
Union All
Select * from tblPayroll
```

	ID	Name	Email
1	1	Sonu	Sonu@gmail.com
2	2	Shyam	shyam@gmail.com
3	1	Ram	ram@gmail.com
4	2	Shyam	shyam@gmail.com

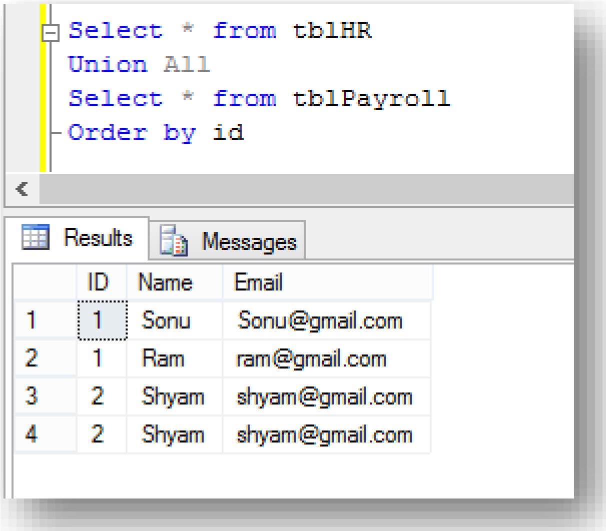
Estimate query execution plan – Ctrl + L



```
Query 1: Query cost (relative to the batch): 100%
Select * from tblHR Union Select * from tblPayroll
```

Sorting results of a UNION or UNION ALL

- Order by should be used only on the last SELECT statement in the UNION query
- If the Order by come before then it gives an error incorrect syntax.



```
Select * from tblHR
Union All
Select * from tblPayroll
Order by id
```

	ID	Name	Email
1	1	Sonu	Sonu@gmail.com
2	1	Ram	ram@gmail.com
3	2	Shyam	shyam@gmail.com
4	2	Shyam	shyam@gmail.com

Difference between UNION and JOIN

UNION combines the results of two or more queries into a single result set that includes all the rows that belong to all queries in the union.

By using JOINs, you can retrieve data from two or more tables based on logical relationships between the tables. Joins indicate how SQL should use data from one table to select the rows in another table