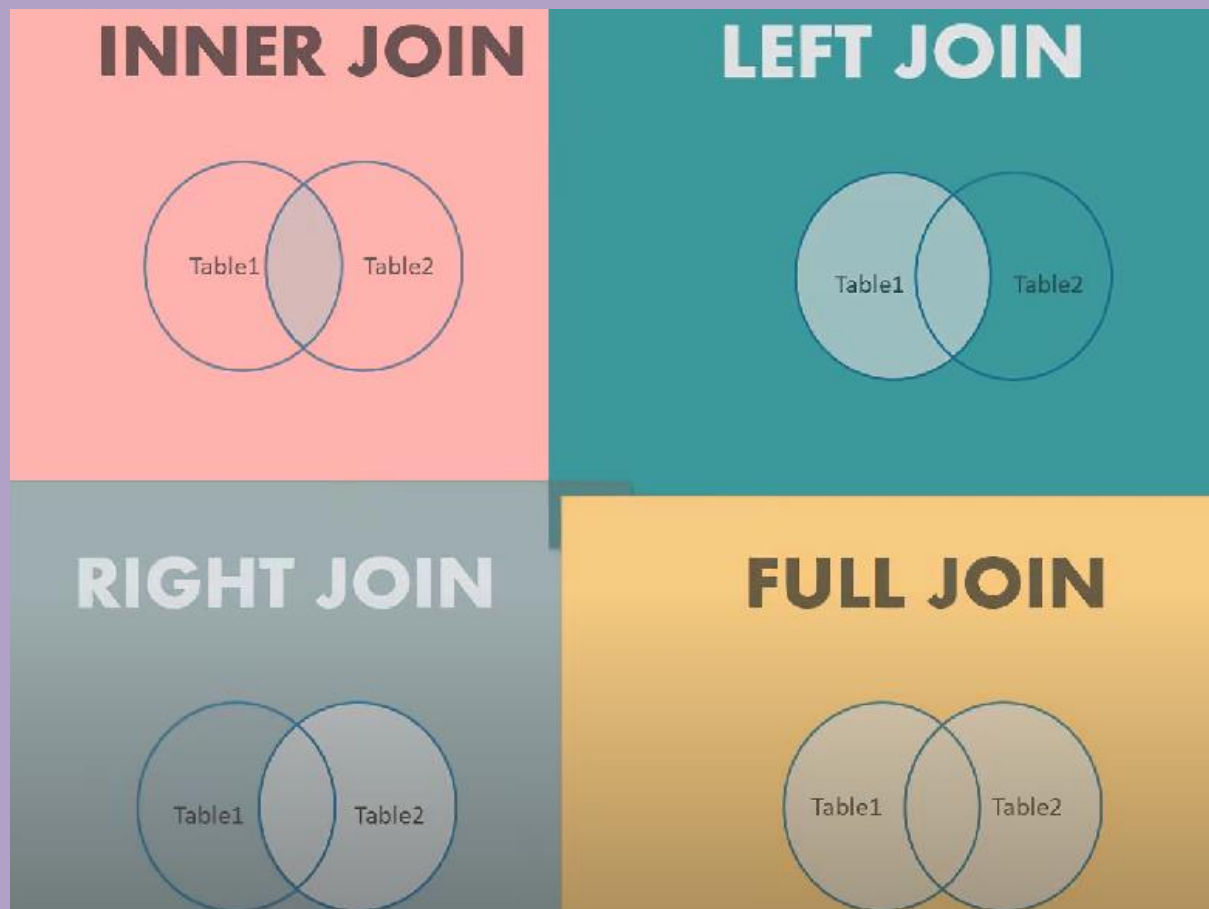


# COMPLETE JOINS TUTORIAL

## MySQL JOINS



### **What are JOINS?**

- \* Joins help retrieving data from two or more database tables.
- \* The tables are mutually related using primary and foreign keys.

**LET'S SEE ALL JOINS IN DETAIL**

# COMPLETE JOINS TUTORIAL

## INNER JOIN

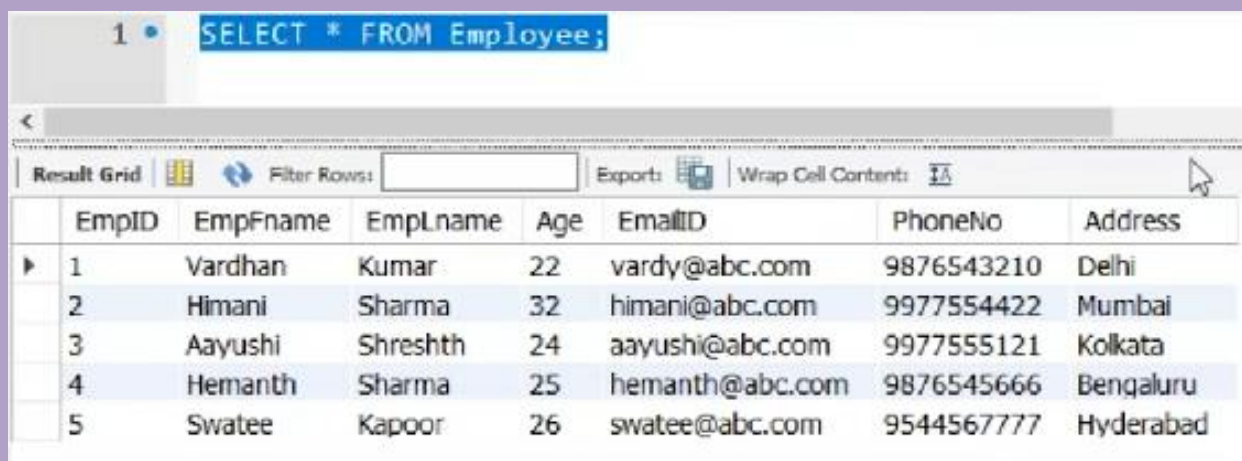
### INNER JOIN

This type of join returns those records which have matching values in both tables. So, if you perform an INNER JOIN operation between the Employee table and the Projects table, all the tuples which have matching values in both the tables will be given as output.

#### Syntax:

```
SELECT Table1.Column1,Table1.Column2,Table2.Column1,...  
FROM Table1  
INNER JOIN Table2  
ON Table1.MatchingColumnName = Table2.MatchingColumnName;
```

## Create Tables :



The screenshot shows a database query interface. At the top, a query editor displays the SQL statement: `SELECT * FROM Employee;`. Below the query editor, a toolbar contains options for 'Result Grid', 'Filter Rows', 'Exports', and 'Wrap Cell Contents'. The 'Result Grid' is active, displaying a table with 8 columns: EmpID, EmpFname, EmpLname, Age, EmailID, PhoneNo, and Address. The table contains 5 rows of data.

	EmpID	EmpFname	EmpLname	Age	EmailID	PhoneNo	Address
▶	1	Vardhan	Kumar	22	vardy@abc.com	9876543210	Delhi
	2	Himani	Sharma	32	himani@abc.com	9977554422	Mumbai
	3	Aayushi	Shreshth	24	aayushi@abc.com	9977555121	Kolkata
	4	Hemanth	Sharma	25	hemanth@abc.com	9876545666	Bengaluru
	5	Swatee	Kapoor	26	swatee@abc.com	9544567777	Hyderabad

# COMPLETE JOINS TUTORIAL

1 •

2 •

SELECT \* FROM Employee;

SELECT \* FROM Project;

<

Result Grid

Filter Rows:

Exports

Wrap Cell Contents

	ProjectID	EmpID	ClientID	ProjectName	ProjectStartDate
▶	111	1	3	Project1	2019-04-21
	222	2	1	Project2	2019-02-12
	333	3	5	Project3	2019-01-10
	444	3	2	Project4	2019-04-16
	555	5	4	Project5	2019-05-23
	666	9	1	Project6	2019-01-12
	777	7	2	Project7	2019-07-25
	888	8	3	Project8	2019-08-20

## Write Query for Inner Join

```
SELECT * FROM Employee;
SELECT * FROM Project;
SELECT Employee.EmpID, Employee.EmpFname, Employee.EmpLname, Project.ProjectID, Project.ProjectName
FROM Employee
INNER JOIN Project ON Employee.EmpID=Project.EmpID;
```

## OUTPUT :

	EmpID	EmpFname	EmpLname	ProjectID	ProjectName
▶	1	Vardhan	Kumar	111	Project1
	2	Himani	Sharma	222	Project2
	3	Aayushi	Shreshth	333	Project3
	3	Aayushi	Shreshth	444	Project4
	5	Swatee	Kapoor	555	Project5

# COMPLETE JOINS TUTORIAL

## LEFT JOIN

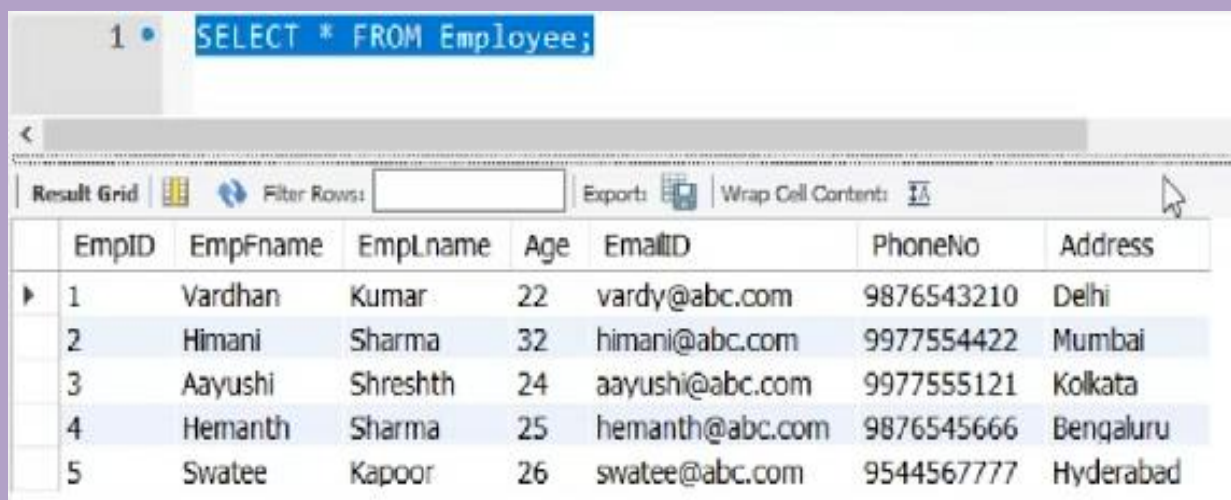
### LEFT JOIN

The LEFT JOIN or the LEFT OUTER JOIN returns all the records from the left table and also those records which satisfy a condition from the right table. Also, for the records having no matching values in the right table, the output or the result-set will contain the NULL values.

#### Syntax:

```
SELECT Table1.Column1,Table1.Column2,Table2.Column1,...  
FROM Table1  
LEFT JOIN Table2  
ON Table1.MatchingColumnName = Table2.MatchingColumnName;
```

## Create Tables :



The screenshot shows a database query editor with a query window at the top containing the SQL statement: `SELECT * FROM Employee;`. Below the query window is a toolbar with options like 'Result Grid', 'Filter Rows', 'Exports', and 'Wrap Cell Contents'. The 'Result Grid' is active, displaying a table with 8 columns: EmpID, EmpFname, EmpLname, Age, EmailID, PhoneNo, and Address. The table contains 5 rows of data.

	EmpID	EmpFname	EmpLname	Age	EmailID	PhoneNo	Address
▶	1	Vardhan	Kumar	22	vardy@abc.com	9876543210	Delhi
	2	Himani	Sharma	32	himani@abc.com	9977554422	Mumbai
	3	Aayushi	Shreshth	24	aayushi@abc.com	9977555121	Kolkata
	4	Hemanth	Sharma	25	hemanth@abc.com	9876545666	Bengaluru
	5	Swatee	Kapoor	26	swatee@abc.com	9544567777	Hyderabad



# COMPLETE JOINS TUTORIAL

```
1 • SELECT * FROM Employee;
2 • SELECT * FROM Project;
```

<

Result Grid | Filter Rows: | Exports: | Wrap Cell Contents: |

	ProjectID	EmpID	ClientID	ProjectName	ProjectStartDate
▶	111	1	3	Project1	2019-04-21
	222	2	1	Project2	2019-02-12
	333	3	5	Project3	2019-01-10
	444	3	2	Project4	2019-04-16
	555	5	4	Project5	2019-05-23
	666	9	1	Project6	2019-01-12
	777	7	2	Project7	2019-07-25
	888	8	3	Project8	2019-08-20

## Write Query for Left Join

```
SELECT * FROM Employee;
SELECT * FROM Project;
SELECT Employee.EmpID, Employee.EmpFname, Employee.EmpLname, Project.ProjectID, Project.ProjectName
FROM Employee
INNER JOIN Project ON Employee.EmpID=Project.EmpID;

SELECT Employee.EmpFname, Employee.EmpLname, Project.ProjectID, Project.ProjectName
FROM Employee
LEFT JOIN Project ON Employee.EmpID=Project.EmpID;
```

## OUTPUT :

	EmpFname	EmpLname	ProjectID	ProjectName
▶	Vardhan	Kumar	111	Project1
	Himani	Sharma	222	Project2
	Aayushi	Shreshth	333	Project3
	Aayushi	Shreshth	444	Project4
	Swatee	Kapoor	555	Project5
	Hemanth	Sharma	888	Project8

# COMPLETE JOINS TUTORIAL

## RIGHT JOIN

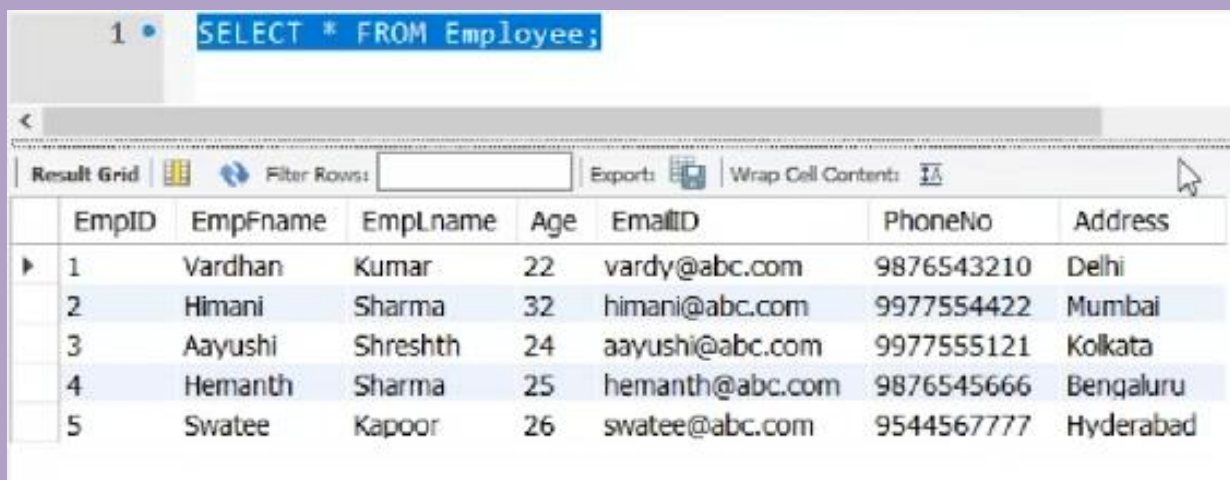
### RIGHT JOIN

The RIGHT JOIN or the RIGHT OUTER JOIN returns all the records from the right table and also those records which satisfy a condition from the left table. Also, for the records having no matching values in the left table, the output or the result-set will contain the NULL values.

#### Syntax:

```
SELECT Table1.Column1,Table1.Column2,Table2.Column1,....  
FROM Table1  
RIGHT JOIN Table2  
ON Table1.MatchingColumnName = Table2.MatchingColumnName;
```

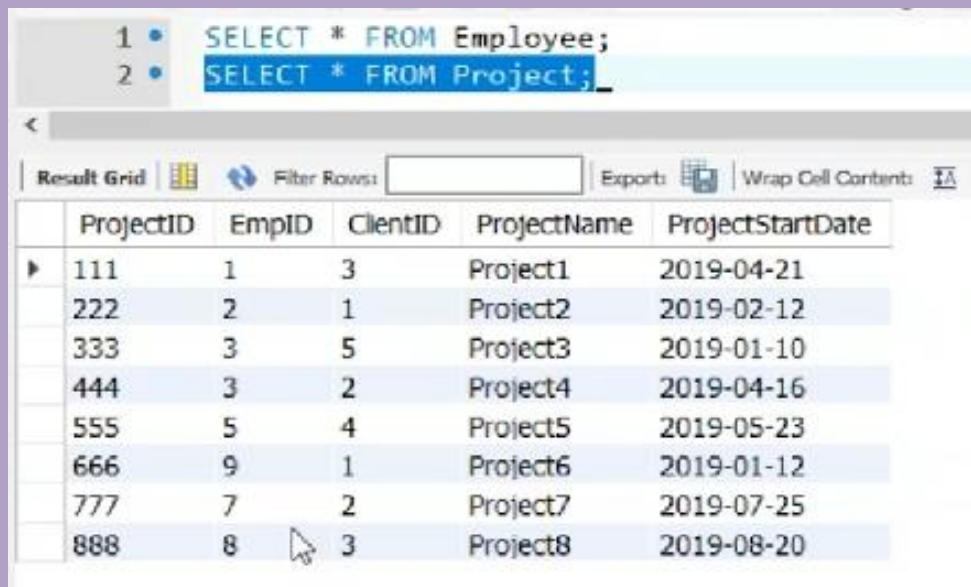
## Create Tables :



The screenshot shows a database interface with a query editor at the top containing the text "1 • SELECT \* FROM Employee;". Below the editor is a toolbar with options like "Result Grid", "Filter Rows", "Exports", and "Wrap Cell Contents". The main area displays a table with 8 columns: EmpID, EmpFname, EmpLname, Age, EmailID, PhoneNo, and Address. The table contains 5 rows of employee data.

	EmpID	EmpFname	EmpLname	Age	EmailID	PhoneNo	Address
▶	1	Vardhan	Kumar	22	vardy@abc.com	9876543210	Delhi
	2	Himani	Sharma	32	himani@abc.com	9977554422	Mumbai
	3	Aayushi	Shreshth	24	aayushi@abc.com	9977555121	Kolkata
	4	Hemanth	Sharma	25	hemanth@abc.com	9876545666	Bengaluru
	5	Swatee	Kapoor	26	swatee@abc.com	9544567777	Hyderabad

# COMPLETE JOINS TUTORIAL



The screenshot shows a SQL query editor with two queries listed:

- 1 • `SELECT * FROM Employee;`
- 2 • `SELECT * FROM Project;`

Below the queries is a toolbar with options like 'Result Grid', 'Filter Rows', 'Exports', and 'Wrap Cell Contents'. The 'Result Grid' is selected, displaying a table with the following data:

	ProjectID	EmpID	ClientID	ProjectName	ProjectStartDate
▶	111	1	3	Project1	2019-04-21
	222	2	1	Project2	2019-02-12
	333	3	5	Project3	2019-01-10
	444	3	2	Project4	2019-04-16
	555	5	4	Project5	2019-05-23
	666	9	1	Project6	2019-01-12
	777	7	2	Project7	2019-07-25
	888	8	3	Project8	2019-08-20

## Write Query for Right Join

```
SELECT Employee.EmpFname, Employee.EmpLname, Project.ProjectID, Project.ProjectName
FROM Employee
RIGHT JOIN Project ON Employee.EmpID=Project.EmpID;
```

## OUTPUT :

	EmpFname	EmpLname	ProjectID	ProjectName
▶	Vardhan	Kumar	111	Project1
	Himani	Sharma	222	Project2
	Aayushi	Shreshth	333	Project3
	Aayushi	Shreshth	444	Project4
	Swatee	Kapoor	555	Project5
	NULL	NULL	666	Project6
	NULL	NULL	777	Project7
	NULL	NULL	888	Project8

# COMPLETE JOINS TUTORIAL

## FULL JOIN

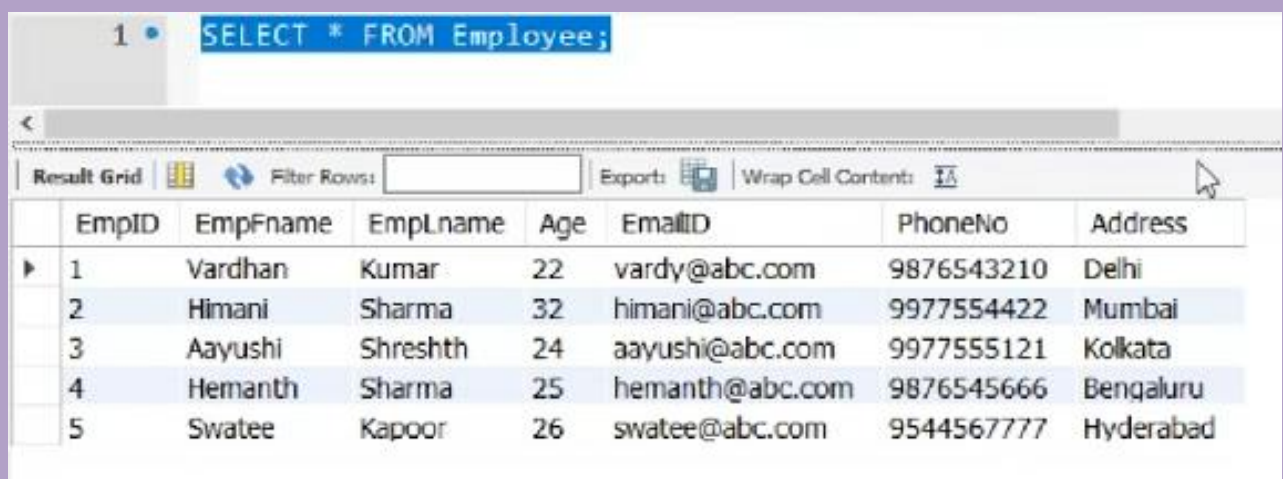
### FULL JOIN

Full Join or the Full Outer Join returns all those records which either have a match in the left(Table1) or the right(Table2) table.

#### Syntax:

```
SELECT Table1.Column1,Table1.Column2,Table2.Column1,....  
FROM Table1  
FULL JOIN Table2  
ON Table1.MatchingColumnName = Table2.MatchingColumnName;
```

## Create Tables :



The screenshot shows a database query interface. At the top, a SQL query is entered: `SELECT * FROM Employee;`. Below the query, there is a toolbar with options like 'Result Grid', 'Filter Rows', 'Exports', and 'Wrap Cell Contents'. The main area displays a table with 8 columns: EmpID, EmpFname, EmpLname, Age, EmailID, PhoneNo, and Address. The table contains 5 rows of data.

	EmpID	EmpFname	EmpLname	Age	EmailID	PhoneNo	Address
▶	1	Vardhan	Kumar	22	vardy@abc.com	9876543210	Delhi
	2	Himani	Sharma	32	himani@abc.com	9977554422	Mumbai
	3	Aayushi	Shreshth	24	aayushi@abc.com	9977555121	Kolkata
	4	Hemanth	Sharma	25	hemanth@abc.com	9876545666	Bengaluru
	5	Swatee	Kapoor	26	swatee@abc.com	9544567777	Hyderabad



# COMPLETE JOINS TUTORIAL

1 • `SELECT * FROM Employee;`  
2 • `SELECT * FROM Project;`

< Result Grid Filter Rows: Exports Wrap Cell Contents

	ProjectID	EmpID	ClientID	ProjectName	ProjectStartDate
▶	111	1	3	Project1	2019-04-21
	222	2	1	Project2	2019-02-12
	333	3	5	Project3	2019-01-10
	444	3	2	Project4	2019-04-16
	555	5	4	Project5	2019-05-23
	666	9	1	Project6	2019-01-12
	777	7	2	Project7	2019-07-25
	888	8	3	Project8	2019-08-20

## Write Query for Full Join

```
SELECT Employee.EmpFname, Employee.EmpLname, Project.ProjectID, Project.ProjectName
FROM Employee
LEFT JOIN Project ON Employee.EmpID=Project.EmpID;
UNION
SELECT Employee.EmpFname, Employee.EmpLname, Project.ProjectID, Project.ProjectName
FROM Employee
RIGHT JOIN Project ON Employee.EmpID=Project.EmpID;
```

## OUTPUT :

	EmpFname	EmpLname	ProjectID
▶	Vardhan	Kumar	111
	Himani	Sharma	222
	Aayushi	Shreshth	333
	Aayushi	Shreshth	444
	Swatee	Kapoor	555
	Hemanth	Sharma	NULL
	NULL	NULL	666
	NULL	NULL	777
	NULL	NULL	888