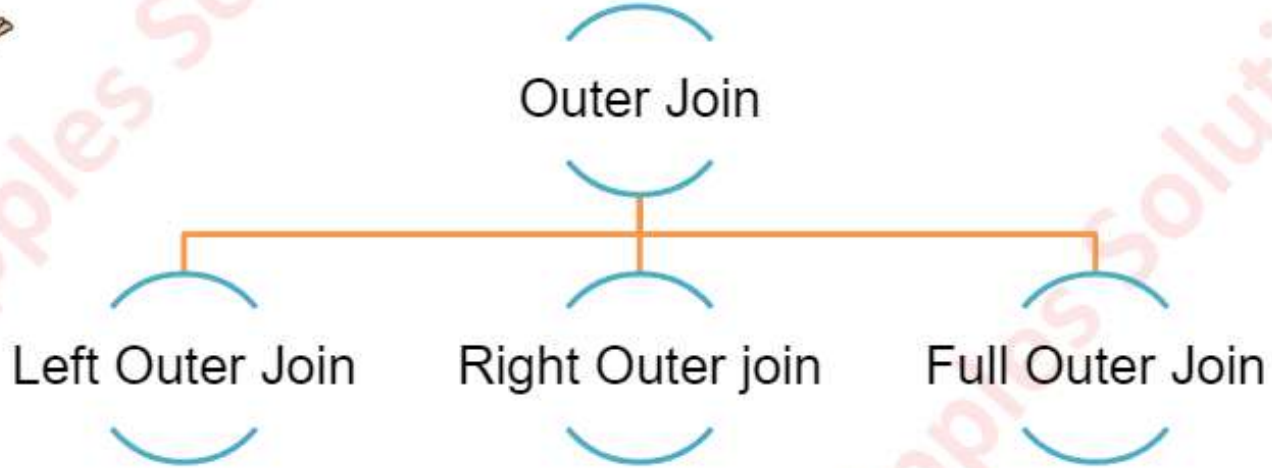
What is a outer join?

Click to Continue

Outer join returns all rows from primary table and only those rows from the secondary table which meets the condition.



What is Left Outer Join?

Click to Continue

In this join all the rows from the table specified in the left side of the join statement will be retrieved irrespective of whether the condition is satisfied or not.

When there are no matching data found in the right side table of the join, the fields fetched from the right side table will be set as null.

How to create a Left Outer Join?

```
SELECT column_name(s)
FROM table_1 LEFT JOIN table_2
ON  table_1.column_name=table_2.column_name
```

All the rows of table1 will be fetched even if a matching row is not found in table2

Illustration for Left Outer Join

Click to Continue

Let us assume a cinema theatre scenario to better understand joins, assuming two tables

- **Customer -** Has columns customer_id (Primary Key), FirstName,Last_Name, Address, Contact_no.
- **Tickets -** Has columns Ticket_Id , customer_id, RowName,Seat_no, Show_date,Show_Time.

Scenario #1: Retrieves all the customer even if ticket details are not available for a customer..

```
SELECT C.FirstName, C.LastName, T.Ticket_Id,T.RowName
FROM Customer C LEFT JOIN Ticket T
ON C.customer_Id=T.Customer_Id;
```

Result: The above query retrieves all the records where the customer id's in customer table matches with the tickets customer id's. In case the a customer does not have ticket details the record will be still fetched with the ticket id and row name displayed as null.

Illustration For Left Outer Join

Click to Continue

Customer

Customer_id	First Name	Last Name	Address	Contact_no
1	Adam	Job	Temple St,US	1234
2	Jack	Steve	Temple Road, UK	4567
3	Raj	Mohan	Anna Nagar, India	8901

Ticket_Details

Ticket_Id	Customer_Id	Seat_No
101	1	J1
201	1	J2
301	2	A3
401	2	A4
501	5	K8

Result :

First Name	Last Name	Ticket_ID
Adam	Job	101
Adam	Job	201
Jack	Steve	301
Jack	Steve	401
Raj	Mohan	Null

Though Raj does not have a ticket, his details are retrieved with the ticket id as null.

Right Outer Join

Click to Continue

This join fetches all rows from *table* specified on the right side of the join statement. When there are no matching data found in the left side table of the join, the fields fetched from the left side table will be set as null.

How to write a right outer join?

```
SELECT column_name(s)
FROM table1 RIGHT JOIN table2
ON table1.column_name=table2.column_name
```

All the rows of table2 will be fetched even if a matching row is not found in table1

Illustration for Right Outer Join

Click to Continue

Let us assume a cinema theatre scenario to better understand joins, assuming two tables

- **Customer -** Has columns customer_id (Primary Key), FirstName,Last_Name, Address, Contact_no.
- **Tickets -** Has columns Ticket_Id , customer_id, RowName,Seat_no, Show_date,Show_Time.

Scenario #1:

```
SELECT C.FirstName, C.LastName, T.Ticket_Id,T.RowName
FROM Customer C RIGHT JOIN Ticket T
ON C.customer_Id=T.Customer_Id;
```

Result: The above query retrieves all the records where the customer id's in customer table matches with the tickets customer id's. In case there is a ticket with no customers assigned the ticket will be fetched with the first name and last name displayed as null.

Illustration For Right Outer Join

Click to Continue

Customer

Customer_id	First Name	Last Name	Address	Contact_no
1	Adam	Job	Temple St,US	1234
2	Jack	Steve	Temple Road, UK	4567
3	Raj	Mohan	Anna Nagar, India	8901

Ticket_Details

Ticket_Id	Customer_Id	Seat_No
101	1	J1
201	1	J2
301	2	A3
401	2	A4
501		K8

Result :

First Name	Last Name	Ticket_ID
Adam	Job	101
Adam	Job	201
Jack	Steve	301
Jack	Steve	401
-	-	501

The ticket 501 data will be retrieved though there is no record in Customer assigned to it..

What is Full Outer Join ?

Click to Continue

The **Full Outer Join** retrieves all the records from the left side table in the join condition , and the right side table in the join, irrespective of whether the condition is met.

MySQL Does not support Full Outer Join.

You need to use work arounds to achieve full outer joins

What Is a Self Join?

Click to Continue

Self join is used to compared one row of a table to another row of the same table.

Illustration: Assume a theatre scenario where the user details are stored in a User table. Each of the user have a supervisor who is another user. To write a query to display the user name and supervisor name a self join to the same table should be performed.

user_ID	Name	Supervisor_Id
1	Tom	3
2	John	3
3	Joe	

```
select u.name user,s.name supervisor from user u inner join user s on
u.supervisor_id=s.user_id;
```

Result:

User	Supervisor
Tom	Joe
John	Joe