JOINS

→ Joins indicate how SQL Server should use data from one table to select the rows in another table.

→ A join condition defines the way two tables are related in a query by Specifying the column from each table to be used for the join

→ A JOIN clause is used to combine rows from two or more tables, based on a related column between them

→ In a real-world relational database, data is structured in a large number of tables and which is why, there is a constant need to join these multiple tables based on logical relationships between them.

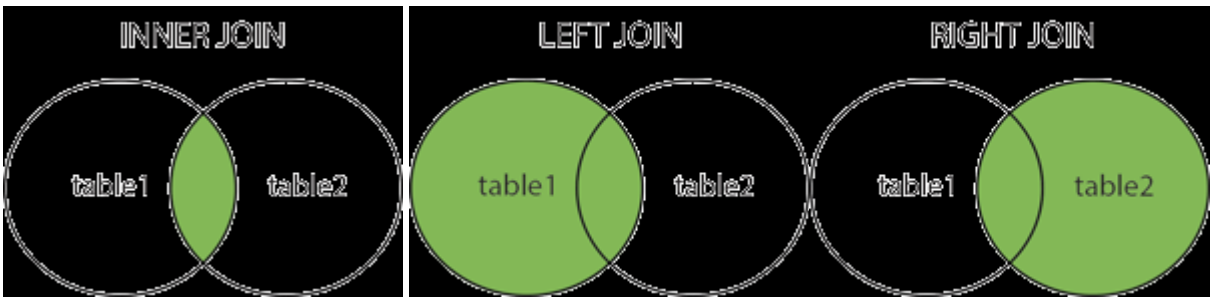
Types of Joins:

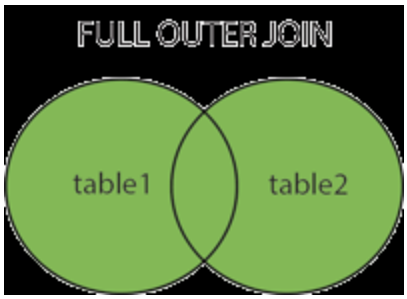
→ INNER JOIN: Returns records that have matching values in both tables

→ LEFT (OUTER) JOIN: Returns all records from the left table, and the matched records from the right table

→ RIGHT (OUTER) JOIN: Returns all records from the right table, and the matched records from the left table

→ FULL (OUTER) JOIN : Returns all records when there is a match in either left or right table





Example:

❖INNER JOIN:

This type of SQL server JOIN returns rows from all tables in which the join condition is true. It takes the following syntax:

We will use the following two tables to demonstrate this:

Students table-

	admission	firstName	lastName	age
1	3420	Nicholas	Samuel	14
2	3380	Joel	John	15
3	3410	Japheth	Becky	16
4	3398	George	Joshua	14
5	3386	John	Lucky	15
6	3403	Simon	Dan	13
7	3400	Calton	Becham	16

Fee Table-

	admission	course	amount_paid
1	3380	Electrical	20000
2	3420	ICT	15000
3	3398	Commerce	13000
4	3410	HR	12000

The following command demonstrates an INNER JOIN in SQL server with example:

```
SELECT Students.admission, Students.firstName,
Students.lastName, Fee.amount_paidFROM StudentsINNER
JOIN FeeON Students.admission = Fee.admission
```

OUTPUT:

	admission	firstName	lastName	amount_paid
1	3420	Nicholas	Samuel	15000
2	3380	Joel	John	20000
3	3410	Japheth	Becky	12000
4	3398	George	Joshua	13000

❖LEFT OUTER JOIN

This type of join will return all rows from the left-hand table plus records in the right-hand table with matching values. For example:

```
SELECT Students.admission, Students.firstName,
Students.lastName, Fee.amount_paidFROM StudentsLEFT
OUTER JOIN FeeON Students.admission = Fee.admission
```

OUTPUT:

	admission	firstName	lastName	amount_paid
1	3420	Nicholas	Samuel	15000
2	3380	Joel	John	20000
3	3410	Japheth	Becky	12000
4	3398	George	Joshua	13000
5	3386	John	Lucky	NULL
6	3403	Simon	Dan	NULL
7	3400	Calton	Becham	NULL

❖RIGHT OUTER JOIN

This type of join returns all rows from the right-hand table and only those with matching values in the left-hand table. For example:

```
SELECT Students.admission, Students.firstName,
Students.lastName, Fee.amount_paidFROM StudentsRIGHT
OUTER JOIN FeeON Students.admission = Fee.admission
```

OUTPUT:

	admission	firstName	lastName	amount_paid
1	3380	Joel	John	20000
2	3420	Nicholas	Samuel	15000
3	3398	George	Joshua	13000
4	3410	Japheth	Becky	12000

❖FULL OUTER JOIN

This type of join returns all rows from both tables with NULL values where the JOIN condition is not true. For example:

```
SELECT Students.admission, Students.firstName,
Students.lastName, Fee.amount_paidFROM StudentsFULL
OUTER JOIN FeeON Students.admission = Fee.admission
```

OUTPUT:

	admission	firstName	lastName	amount_paid
1	3420	Nicholas	Samuel	15000
2	3380	Joel	John	20000
3	3410	Japheth	Becky	12000
4	3398	George	Joshua	13000
5	3386	John	Lucky	NULL
6	3403	Simon	Dan	NULL
7	3400	Calton	Becham	NULL