**JOINS**

**INNER JOIN** - It joins two tables along the rows where the matching condition is satisfied.

* **FULL JOIN/OUTER JOIN** - It joins two tables along all the rows and returns a table containing the rows of both the tables. In case the join condition fails, the missing values would be returned as a NULL value in the table.
* **LEFT JOIN** - It preserves the rows of the left table while attaching the rows of the right table that have matching values in the corresponding column of the left table.
* **RIGHT JOIN** - It preserves the rows of the right table while attaching the rows of the left table that have matching values in the corresponding column of the right table.

The syntax for INNER JOIN is shown below.

(Note that JOIN and INNER JOIN are the same. The inner keyword is optional, as all joins are considered to be inner joins unless otherwise specified.)

**SELECT** **column\_name**(s)

**FROM** table1

**INNER** **JOIN** table2

**ON** table1.**column\_name** = table2.**column\_name**;

The syntax for FULL JOIN is as follows:

**SELECT** **column\_name**(s)

**FROM** table1

**FULL** **OUTER** **JOIN** table2

**ON** table1.**column\_name** = table2.**column\_name**

**WHERE** condition;

The syntax for LEFT JOIN is as follows:

**SELECT** **column\_name**(s)

**FROM** table1

**LEFT** **JOIN** table2

**ON** table1.**column\_name** = table2.**column\_name**;

The syntax for RIGHT JOIN is as follows:

**SELECT** **column\_name**(s)

**FROM** table1

**RIGHT** **JOIN** table2

**ON** table1.**column\_name** = table2.**column\_name**;

Through this segment, you must have realised the importance of joins and how we can use them effectively to derive insights.

For further reference, please refer to the queries listed in the file provided below.