Name: Saurabh Kadam

Roll No.: 47

Course: PGDACC (ESDS)

Subject: DBA

Assignment No. 3

**Title**: Understand the different kind of joins and their importance

**Aim**: Perform different kind of joins

* Create the normalised database with multiple tables
* Relate all tables with primary-foreign relationship
* Perform DML operations those extract data with following joins:
  + Inner Join
  + Left Join
  + Right Join
  + Full Outer Join

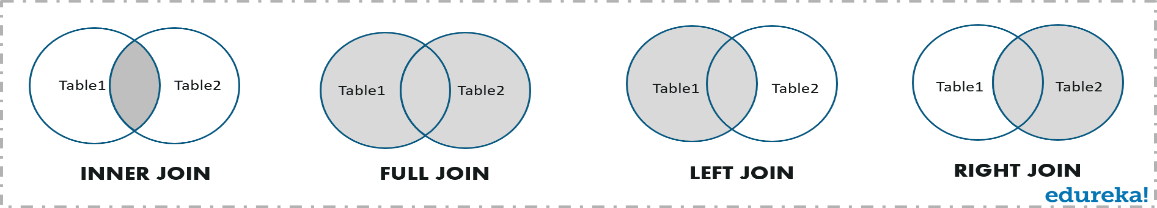
**Theory:**

1. **SQL Joins:**

JOINS in SQL are commands which are used to combine rows from two or more tables, based on a related column between those tables. There are predominantly used when a user is trying to extract data from tables which have one-to-many or many-to-many relationships between them.

Let’s say we have two sets of data in our relational database: table A and table B, with some sort of relation specified by primary and foreign keys. The result of joining these tables together can be visually represented by the following diagram: The extent of the overlap, if any, is determined by how many records in Table A match the records in Table B. Depending on what subset of data we would like to select from the two tables, the four join types can be visualized by highlighting the corresponding sections of the Venn diagram:

* (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



1. **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 two tables, 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;*

**Example:**

For all the join operations we are using following tables(data)

**Employee Table:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **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 |

**Project Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **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 |

**Client Table:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ClientID** | **ClientFname** | **ClientLname** | **Age** | **ClientEmailID** | **PhoneNo** | **Address** | **EmpID** |
| 1 | Susan | Smith | 30 | susan@adn.com | 9765411231 | Kolkata | 3 |
| 2 | Mois | Ali | 27 | mois@jsq.com | 9876543561 | Kolkata | 3 |
| 3 | Soma | Paul | 22 | soma@wja.com | 9966332211 | Delhi | 1 |
| 4 | Zainab | Daginawala | 40 | zainab@qkq.com | 9955884422 | Hyderabad | 5 |
| 5 | Bhaskar | Reddy | 32 | bhaskar@xyz.com | 9636963269 | Mumbai | 2 |

**SQL Query:**

*SELECT Employee.EmpID, Employee.EmpFname, Employee.EmpLname, Projects.ProjectID, Projects.ProjectName*

*FROM Employee*

*INNER JOIN Projects ON Employee.EmpID=Projects.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 |

1. **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;*

**Example:**

**SQL Query:**

*SELECT Employee.EmpFname, Employee.EmpLname, Projects.ProjectID, Projects.ProjectName*

*FROM Employee*

*LEFT JOIN*

*ON Employee.EmpID = Projects.EmpID ;*

|  |  |  |  |
| --- | --- | --- | --- |
| **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 | NULL | NULL |

**Output:**

1. **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;*

**Eaxmple:**

**SQL Query:**

*SELECT Employee.EmpFname, Employee.EmpLname, Projects.ProjectID, Projects.ProjectName*

*FROM Employee*

*RIGHT JOIN*

*ON Employee.EmpID = Projects.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 |

1. **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;*

**Example:**

**SQL Query:**

*SELECT Employee.EmpFname, Employee.EmpLname, Projects.ProjectID*

*FROM Employee*

*FULL JOIN Projects*

*ON Employee.EmpID = Projects.EmpID;*

**Output:**

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

**Conclusion:**

Hence, we understood different kind of joins and how to use them to combine rows from two or more tables, based on a related column between those tables.

OUTPUT:

