**Hive Joins for Datasets - Lab Manual**

**Aim:**

To understand and implement different types of joins in Apache Hive using datasets, and analyze how joins work in Hive for data processing and querying.

**Types of Joins in Hive:**

**1. Inner Join**

**Definition:** Retrieves records that have matching values in both tables.

**Query:**

SELECT e.employee\_id, e.name, d.department\_name

FROM employees e

JOIN departments d

ON e.department\_id = d.department\_id;

**Output:**

| **employee\_id** | **name** | **department\_name** |
| --- | --- | --- |
| 101 | Alice | HR |
| 102 | Bob | IT |

**2. Left Outer Join**

**Definition:** Returns all records from the left table, along with the matched records from the right table. If no match is found, NULL values are returned.

**Query:**

SELECT e.employee\_id, e.name, d.department\_name

FROM employees e

LEFT OUTER JOIN departments d

ON e.department\_id = d.department\_id;

**Output:**

| **employee\_id** | **name** | **department\_name** |
| --- | --- | --- |
| 101 | Alice | HR |
| 102 | Bob | IT |
| 103 | Eve | NULL |

**3. Right Outer Join**

**Definition:** Returns all records from the right table and matched records from the left table. If no match is found, NULL values are returned.

**Query:**

SELECT e.employee\_id, e.name, d.department\_name

FROM employees e

RIGHT OUTER JOIN departments d

ON e.department\_id = d.department\_id;

**Output:**

| **employee\_id** | **name** | **department\_name** |
| --- | --- | --- |
| 101 | Alice | HR |
| 102 | Bob | IT |
| NULL | NULL | Finance |

**4. Full Outer Join**

**Definition:** Returns all records when there is a match in either left or right table. If there is no match, NULL values are returned for missing data.

**Query:**

SELECT e.employee\_id, e.name, d.department\_name

FROM employees e

FULL OUTER JOIN departments d

ON e.department\_id = d.department\_id;

**Output:**

| **employee\_id** | **name** | **department\_name** |
| --- | --- | --- |
| 101 | Alice | HR |
| 102 | Bob | IT |
| 103 | Eve | NULL |
| NULL | NULL | Finance |

.

**5. Cross Join (Cartesian Product)**

**Definition:** Returns the Cartesian product of both tables.

**Query:**

SELECT e.employee\_id, e.name, d.department\_name

FROM employees e

CROSS JOIN departments d;

**Output:**

| **employee\_id** | **name** | **department\_name** |
| --- | --- | --- |
| 101 | Alice | HR |
| 101 | Alice | IT |
| 101 | Alice | Finance |
| 102 | Bob | HR |
| 102 | Bob | IT |
| 102 | Bob | Finance |
| 103 | Eve | HR |
| 103 | Eve | IT |
| 103 | Eve | Finance |

**Result**

By implementing different types of joins in Hive, we can retrieve data based on various conditions. Understanding how joins work helps in performing efficient data analysis and querying large datasets in distributed environments.