Select Query with Joins in Hive

Joins are used in a query to combine data from two or more tables based on the values of some columns.

We will see how to write queries using join in Hive.

Hive Tables

We have the following two tables in Hive.

Employee table containing data about Employees:-

0: jdbc:hive2://localhost:10000> **select \* from employee;**

+--------------+----------------+---------------+------------------+----------------------+--+

| employee.id  | employee.name  | employee.age  | employee.salary  | employee.department  |

+--------------+----------------+---------------+------------------+----------------------+--+

| 80001        | Aarti          | 25            | 37000            | BIGDATA              |

| 80003        | Rajesh         | 29            | 59000            | BIGDATA              |

| 70001        | Suresh         | 45            | 76000            | FINANCE              |

| 80002        | Neha           | 27            | 39000            | FINANCE              |

| 60001        | Sudip          | 34            | 62000            | HR                   |

| 80005        | Rahul          | 24            | 35000            | HR                   |

+--------------+----------------+---------------+------------------+----------------------+--+

Refunds table containing refunds information stored along-with employee id:-

0: jdbc:hive2://localhost:10000> **select \* from refunds;**

+--------------------+-----------------+-----------------+--+

| refunds.refund\_id  | refunds.emp\_id  | refunds.amount  |

+--------------------+-----------------+-----------------+--+

| 1                  | 60001           | 10000           |

| 2                  | 60001           | 15000           |

| 3                  | 70001           | 25000           |

| 4                  | 80001           | 12000           |

| 5                  | 70001           | 13000           |

| 6                  | 80003           | 17000           |

| 7                  | 80002           | 21000           |

| 8                  | 80001           | 31000           |

| 9                  | 80003           | 3000            |

| 10                 | 30001           | 30000           |

+--------------------+-----------------+-----------------+--+

Join

We will join data from both the tables, to see the employee name along-with the refunds information.

We can join data from the tables in two different ways.

**Using a Where clause**

We can Join data from both the tables by using a where clause as shown below:-

select

e.id as emp\_id, e.name as emp\_name, r.amount as refund\_amount

from employee e, refunds r

where e.id=r.emp\_id;

+---------+-----------+----------------+--+

| emp\_id  | emp\_name  | refund\_amount  |

+---------+-----------+----------------+--+

| 60001   | Sudip     | 10000          |

| 60001   | Sudip     | 15000          |

| 70001   | Suresh    | 25000          |

| 80001   | Aarti     | 12000          |

| 70001   | Suresh    | 13000          |

| 80003   | Rajesh    | 17000          |

| 80002   | Neha      | 21000          |

| 80001   | Aarti     | 31000          |

| 80003   | Rajesh    | 3000           |

+---------+-----------+----------------+--+

In the result, we have the employee names along-with their refunds information.

**Using a Join**

We can join data from different tables by using the join keyword in the query as shown below:-

#The default join is inner join

select

e.id as emp\_id, e.name as emp\_name, r.amount as refund\_amount

from employee e join refunds r

on e.id=r.emp\_id;

+---------+-----------+----------------+--+

| emp\_id  | emp\_name  | refund\_amount  |

+---------+-----------+----------------+--+

| 60001   | Sudip     | 10000          |

| 60001   | Sudip     | 15000          |

| 70001   | Suresh    | 25000          |

| 80001   | Aarti     | 12000          |

| 70001   | Suresh    | 13000          |

| 80003   | Rajesh    | 17000          |

| 80002   | Neha      | 21000          |

| 80001   | Aarti     | 31000          |

| 80003   | Rajesh    | 3000           |

+---------+-----------+----------------+--+

We can club join with group by to get information like – Department-wise refunds processed.

select e.department as department, sum(r.amount) as refund\_amount

from employee e join refunds r

on e.id=r.emp\_id

group by e.department;

+-------------+----------------+--+

| department  | refund\_amount  |

+-------------+----------------+--+

| BIGDATA     | 63000          |

| FINANCE     | 59000          |

| HR          | 25000          |

+-------------+----------------+--+

**Join types**

Following join types are available to be used in the query:-

1. **Inner join** – Keywords to be used in query(join/inner join).  Returns all the rows where there is match for join condition in both the tables. Default join used is the inner join.
2. **Left join** – Keywords to be used in query(left join/left outer join). Returns all the rows from the left table and only those rows from the right table for which there is a match for join condition.
3. **Right join**– Keywords to be used in query(right join/right outer join). Returns all the rows from the right table and only those rows from the left table for which there is a match for join condition.
4. **Full Join** – Keywords to be used in query(full join/full outer join). Returns all the rows from the both the tables with nulls in place where there is no match for the join condition.

#Left join, we can see below that all the rows from left table(Employee) are present

#in the result, with NULL values for unmatched rows.

select e.id as emp\_id, e.name as emp\_name, r.amount as refund\_amount

from employee e left join refunds r

on e.id=r.emp\_id;

+---------+-----------+----------------+--+

| emp\_id  | emp\_name  | refund\_amount  |

+---------+-----------+----------------+--+

| 80001   | Aarti     | 12000          |

| 80001   | Aarti     | 31000          |

| 80003   | Rajesh    | 17000          |

| 80003   | Rajesh    | 3000           |

| 70001   | Suresh    | 25000          |

| 70001   | Suresh    | 13000          |

| 80002   | Neha      | 21000          |

| 60001   | Sudip     | 10000          |

| 60001   | Sudip     | 15000          |

| 80005   | Rahul     | NULL           |

+---------+-----------+----------------+--+

#Right join, we can see below that all the rows from right table(Refunds) are present

#in the result, with NULL values for unmatched rows.

select e.id as emp\_id, e.name as emp\_name, r.amount as refund\_amount

from employee e right join refunds r

on e.id=r.emp\_id;

+---------+-----------+----------------+--+

| emp\_id  | emp\_name  | refund\_amount  |

+---------+-----------+----------------+--+

| 60001   | Sudip     | 10000          |

| 60001   | Sudip     | 15000          |

| 70001   | Suresh    | 25000          |

| 80001   | Aarti     | 12000          |

| 70001   | Suresh    | 13000          |

| 80003   | Rajesh    | 17000          |

| 80002   | Neha      | 21000          |

| 80001   | Aarti     | 31000          |

| 80003   | Rajesh    | 3000           |

| NULL    | NULL      | 30000          |

+---------+-----------+----------------+--+

#Full join

select e.id as emp\_id, e.name as emp\_name, r.amount as refund\_amount

from employee e full join refunds r

on e.id=r.emp\_id;

+---------+-----------+----------------+--+

| emp\_id  | emp\_name  | refund\_amount  |

+---------+-----------+----------------+--+

| NULL    | NULL      | 30000          |

| 60001   | Sudip     | 10000          |

| 60001   | Sudip     | 15000          |

| 70001   | Suresh    | 25000          |

| 70001   | Suresh    | 13000          |

| 80001   | Aarti     | 12000          |

| 80001   | Aarti     | 31000          |

| 80002   | Neha      | 21000          |

| 80003   | Rajesh    | 17000          |

| 80003   | Rajesh    | 3000           |

| 80005   | Rahul     | NULL           |

+---------+-----------+----------------+--+