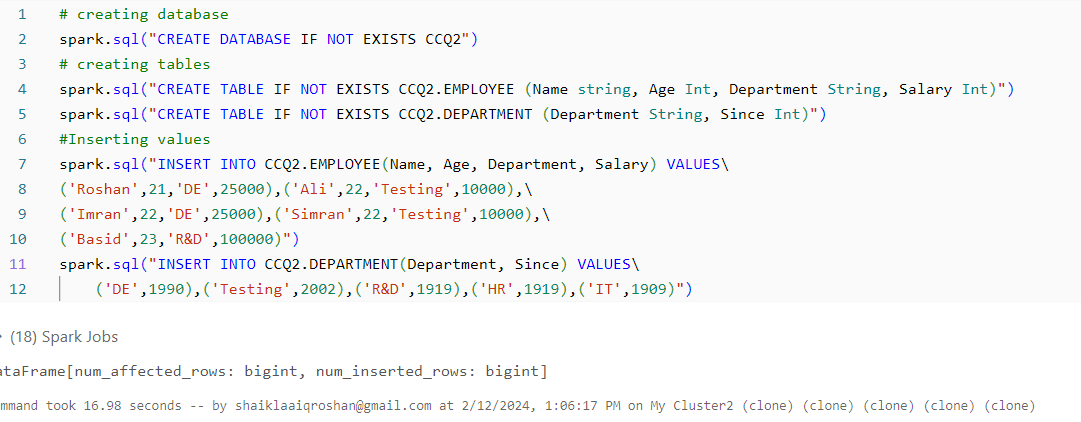
**Q. Execute Pyspark -sparksql joins & Applying Functions in a Pandas DataFrame**

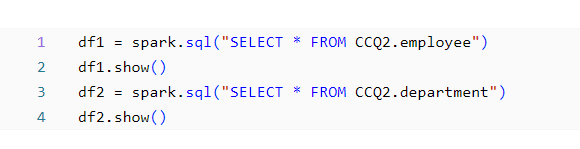
A. First we need to import the spark session module and create a spark session named ‘CCQ’



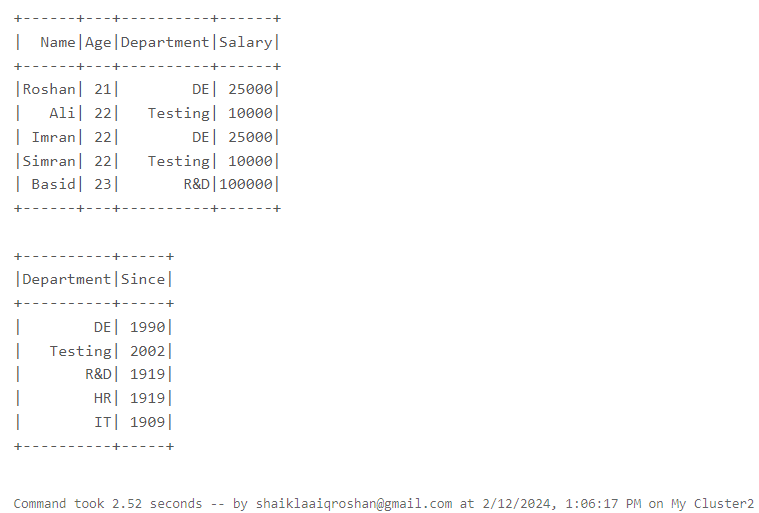
Then we create a database CCQ2 and created two tables Employee and Department and inserted the data using the required sparksql commands.



Then we created two dataframes df1 and df2 that represents employee and department table.



Output:



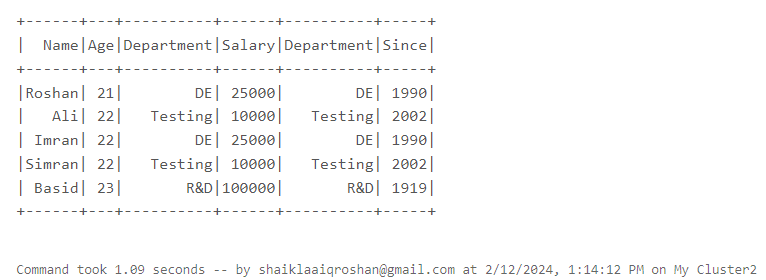
**Joins():**

PySpark Join is used to combine two DataFrames and by chaining these you can join multiple DataFrames; it supports all basic join type operations available in traditional SQL

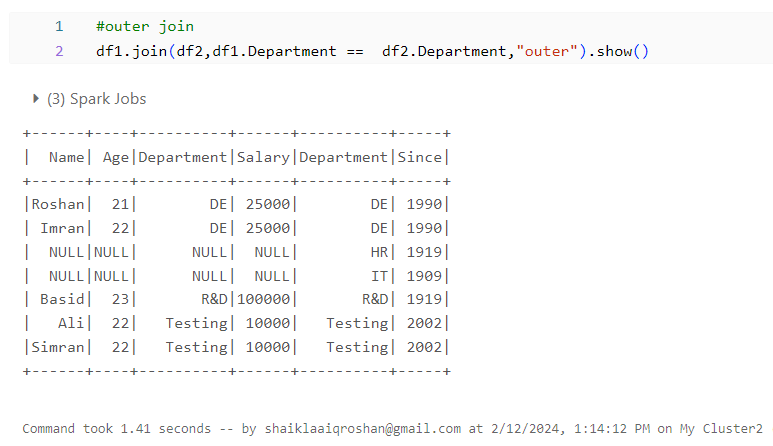
**Inner join** - Matches and retrieves common values from two tables



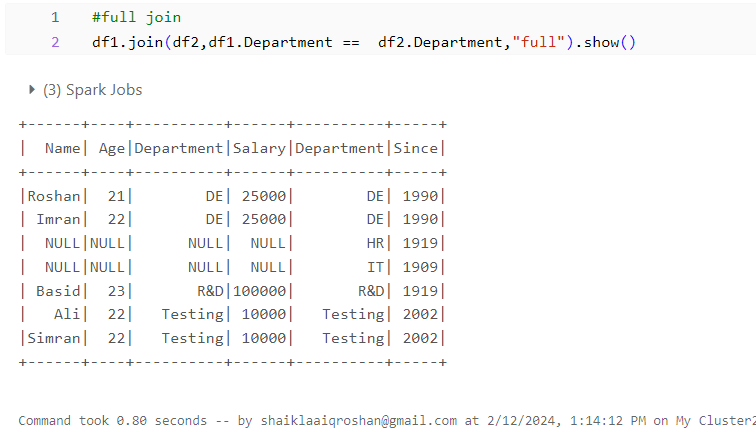
Output:



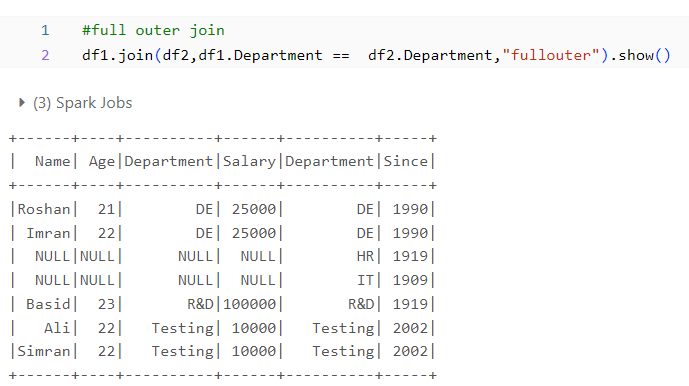
**Outer join** - Retrieves matched and unmatched values from two tables.



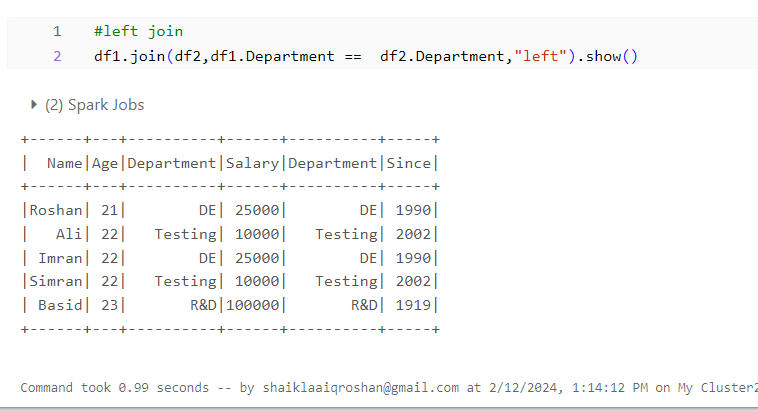
**Full join** - Retrieves all values from both tables, matching and non-matching



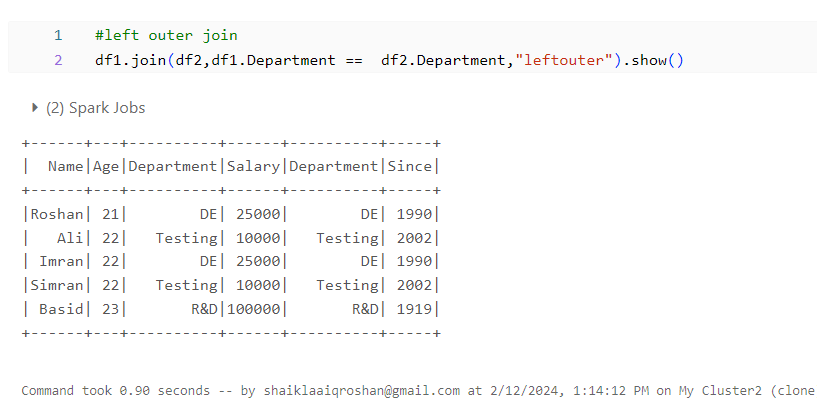
**Full outer join** - Retrieves all values, matching and non-matching, from both tables.



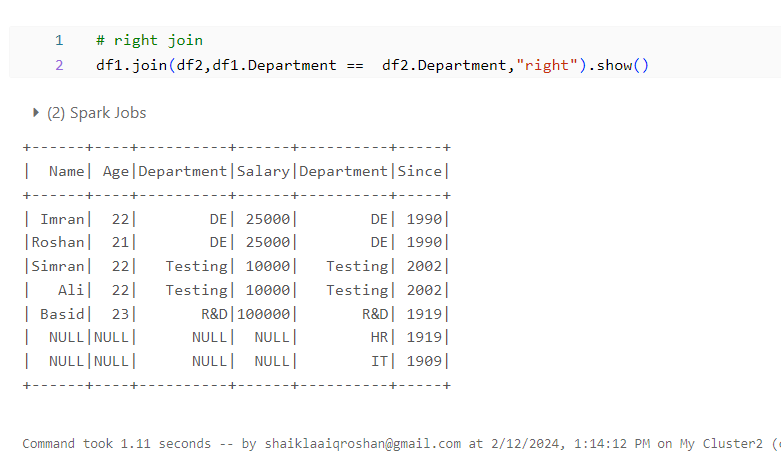
**Left join** - Retrieves all values from the left table and matching values.



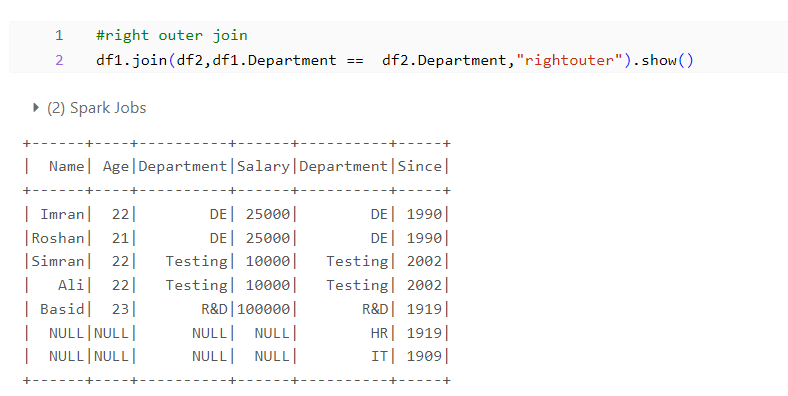
**Left outer join** - Retrieves all values from the left table and matching values.



**Right join** - Retrieves all values from the right table and matching values.



**Right outer join** - Retrieves all values from the right table and matching values.



**Left semi join** - Returns rows from the left table with matching right table.

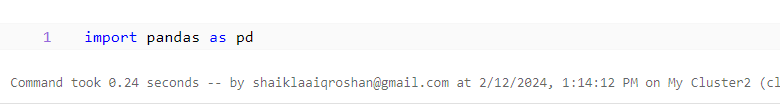


**Left anti join** - Returns rows from the left table with non-matching right table.

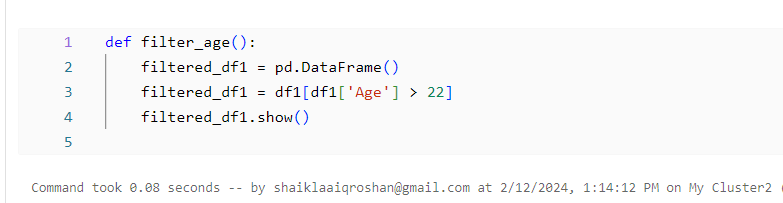


**Functions in pandas dataFrame:**

First we import pandas



Then we created a function called filtered\_age using pandas that filters the data records by selecting only the records which has age more than 22.



**Function call:**



Output:

