**Name -Monika Gariya**

**Email-** [**monikagariya2023@gmail.com**](mailto:monikagariya2023@gmail.com)

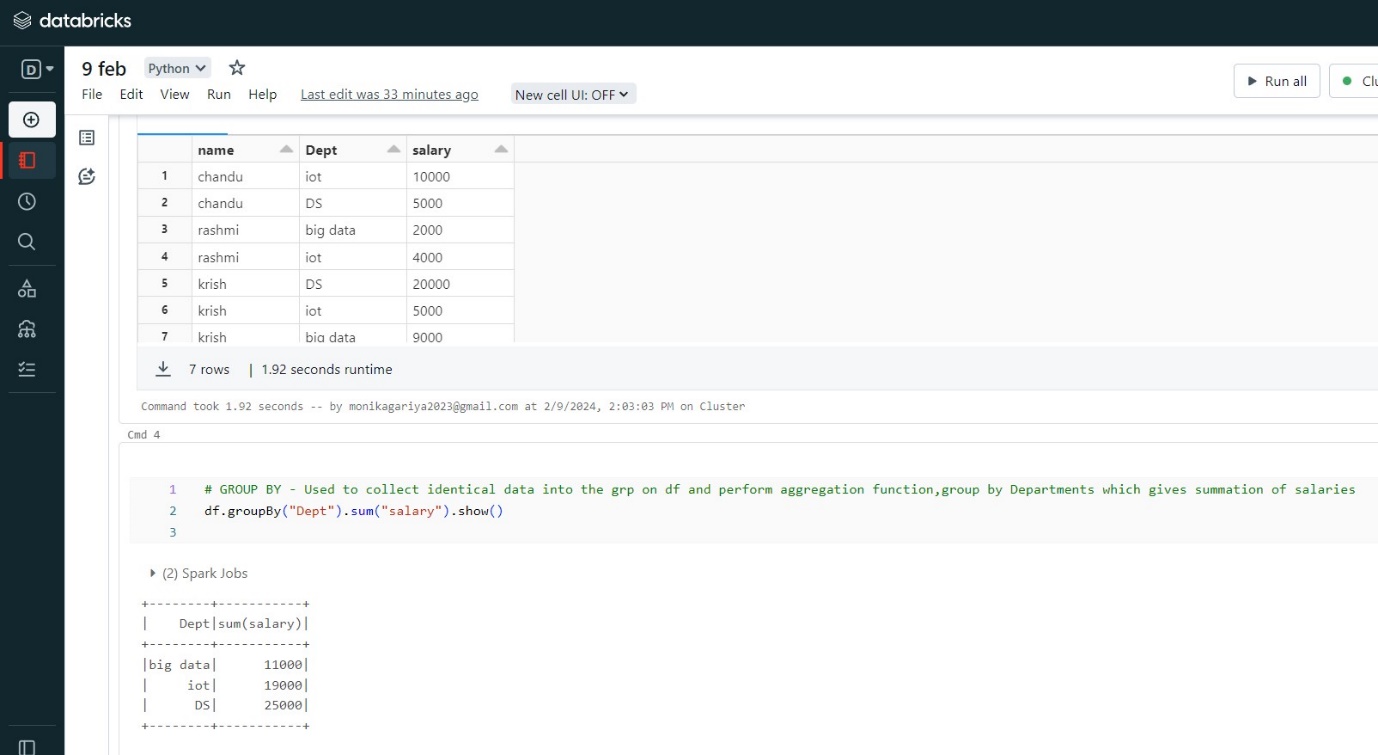
**Data Engineering Batch 1**

**Date – 09-02-2024**

**Topic-Manipulating,Droping,Sorting,Aggregation,Joins,GroupBy dataframes.**

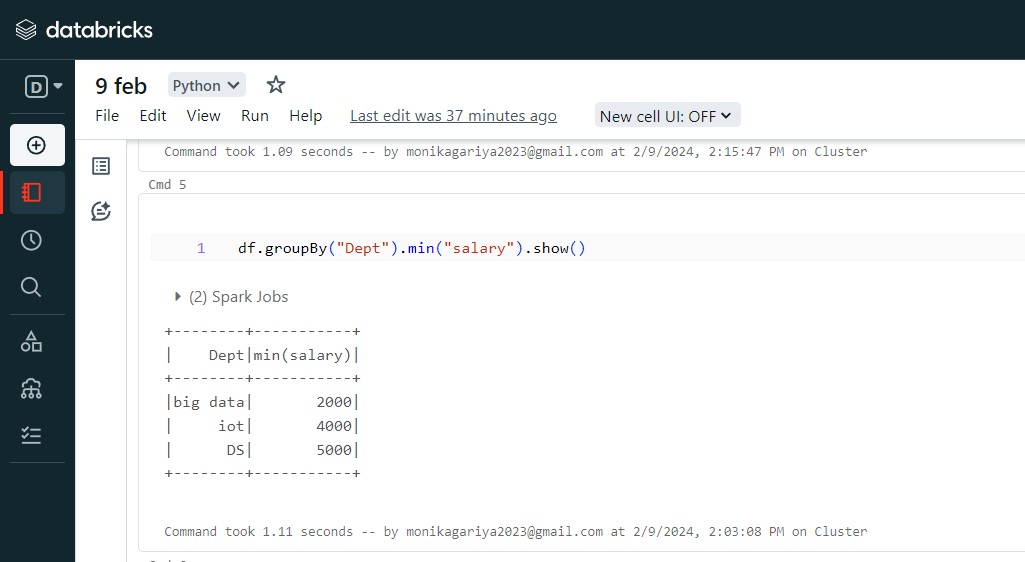
**GroupBy and Aggregate function**:

PySpark groupBy () function is used to collect the identical data into groups on DataFrame and perform count, sum, avg, min, and max functions on the grouped data.

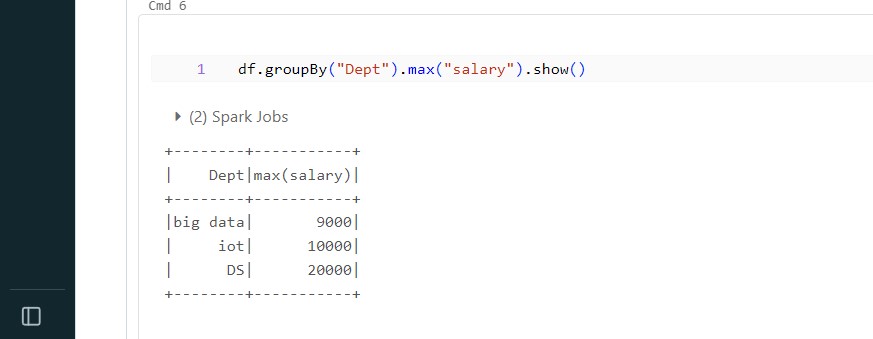


Similarly, we can perform min, max, mean, avg, and count using the groupBy function.

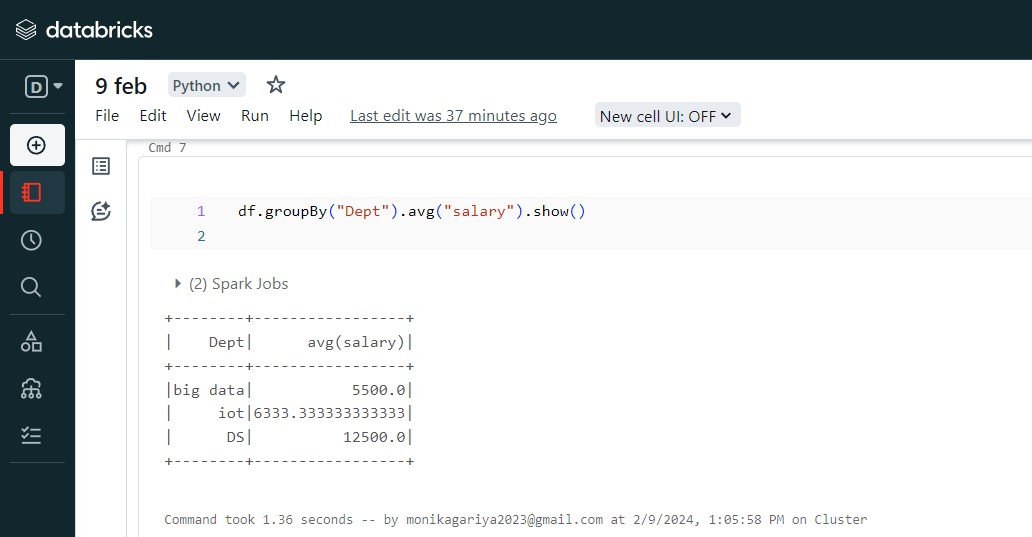
1)MIN:  return the lowest values in a particular column



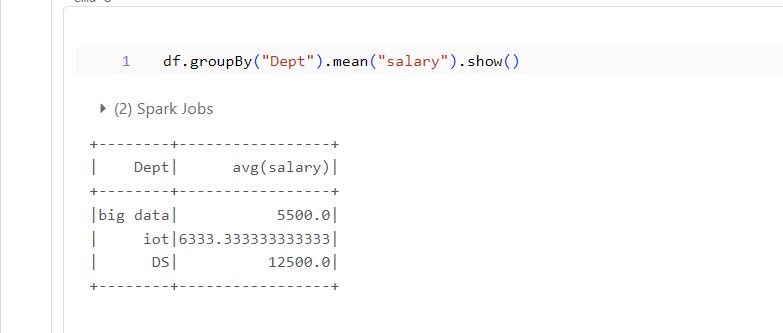
2) MAX:  return the highest values in a particular column



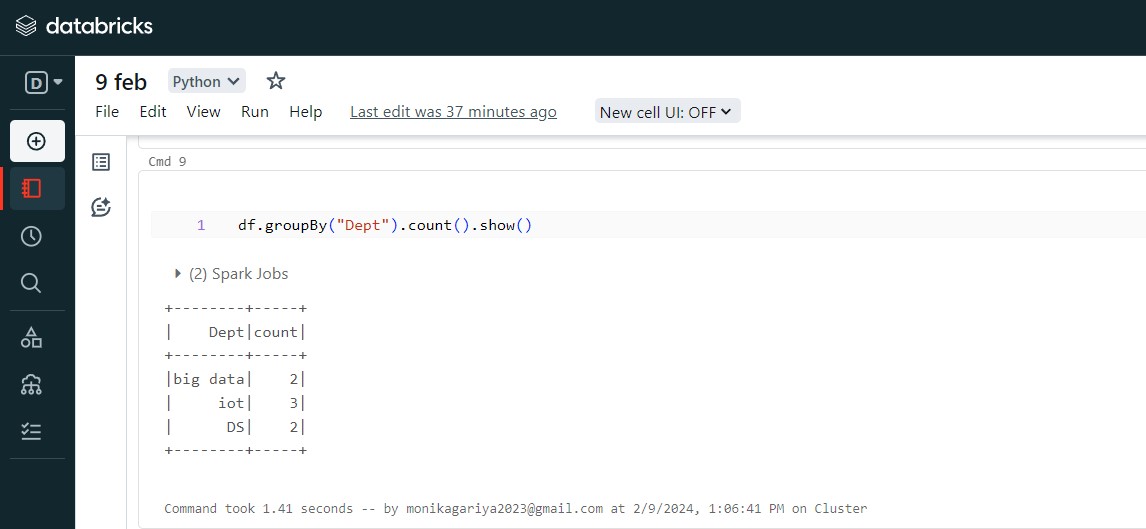
3) AVG: calculates the average of a group of selected values.



4) MEAN: get the average value from the dataframe.

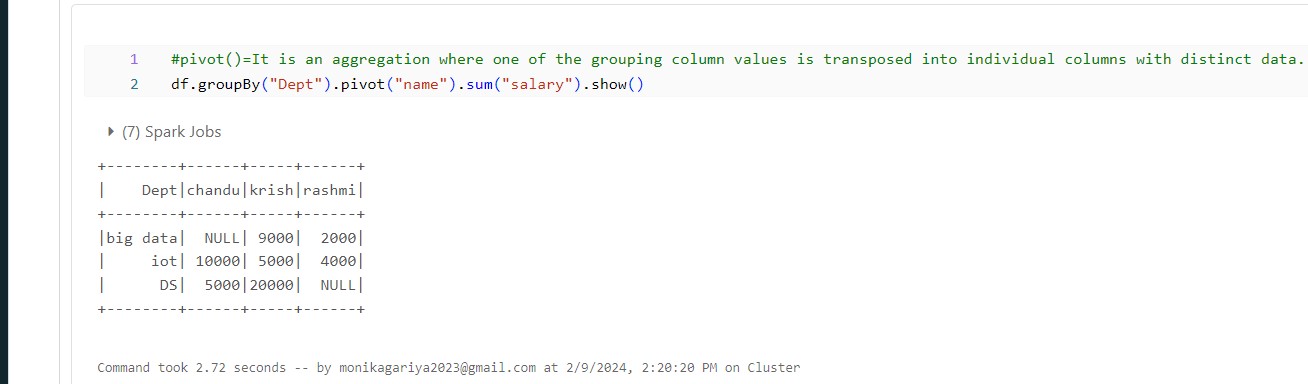


5)COUNT:  counts how many rows are in a particular column.



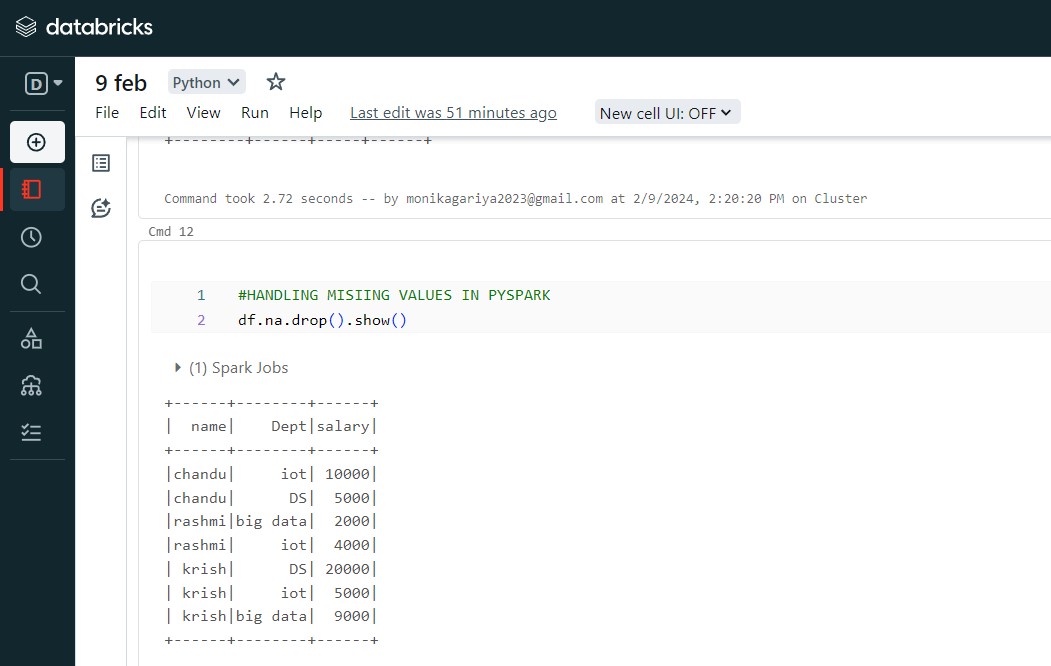
**PIVOT()**

Spark SQL provides pivot() function to rotate the data from one column into multiple columns (transpose row to column). It is an aggregation where one of the grouping column values is transposed into individual columns with distinct data.

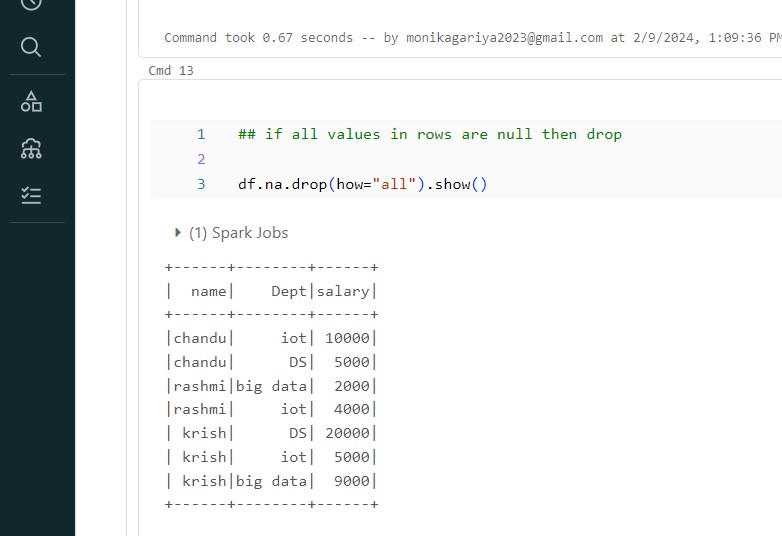


**Handling Missing Values Pyspark**

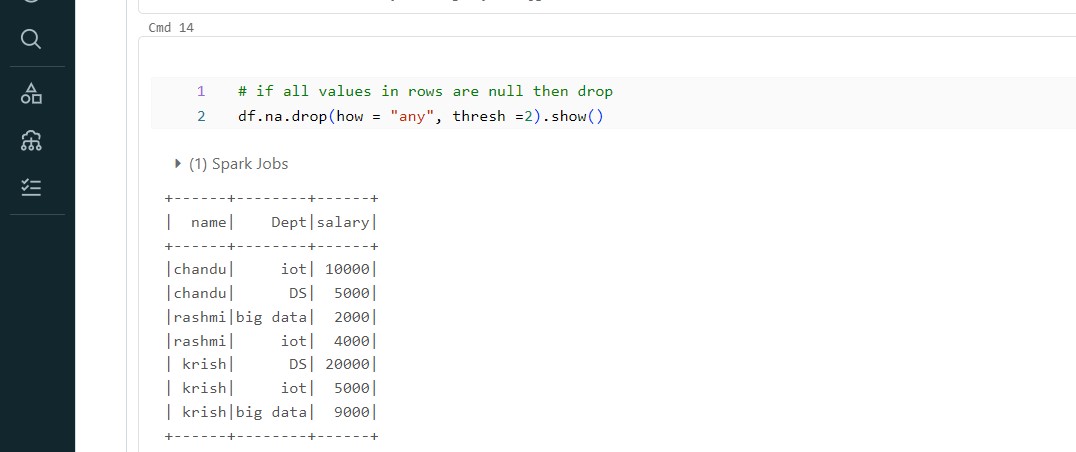
1. Dropping rows based on null values



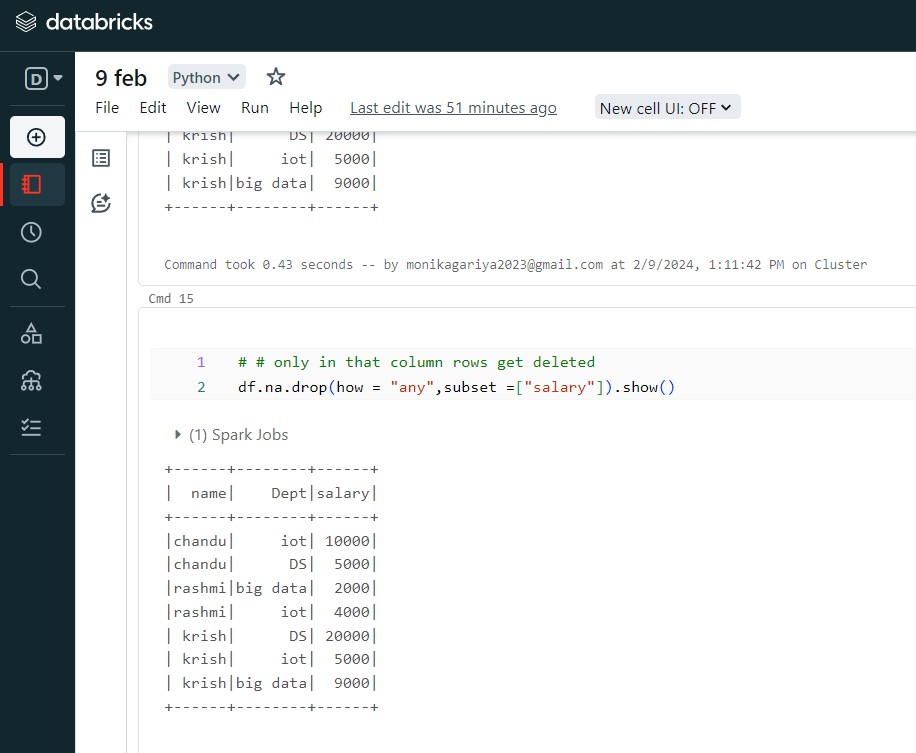
**2)** if all values in rows are null then drop



**3)** at least 2 non null values should be present.

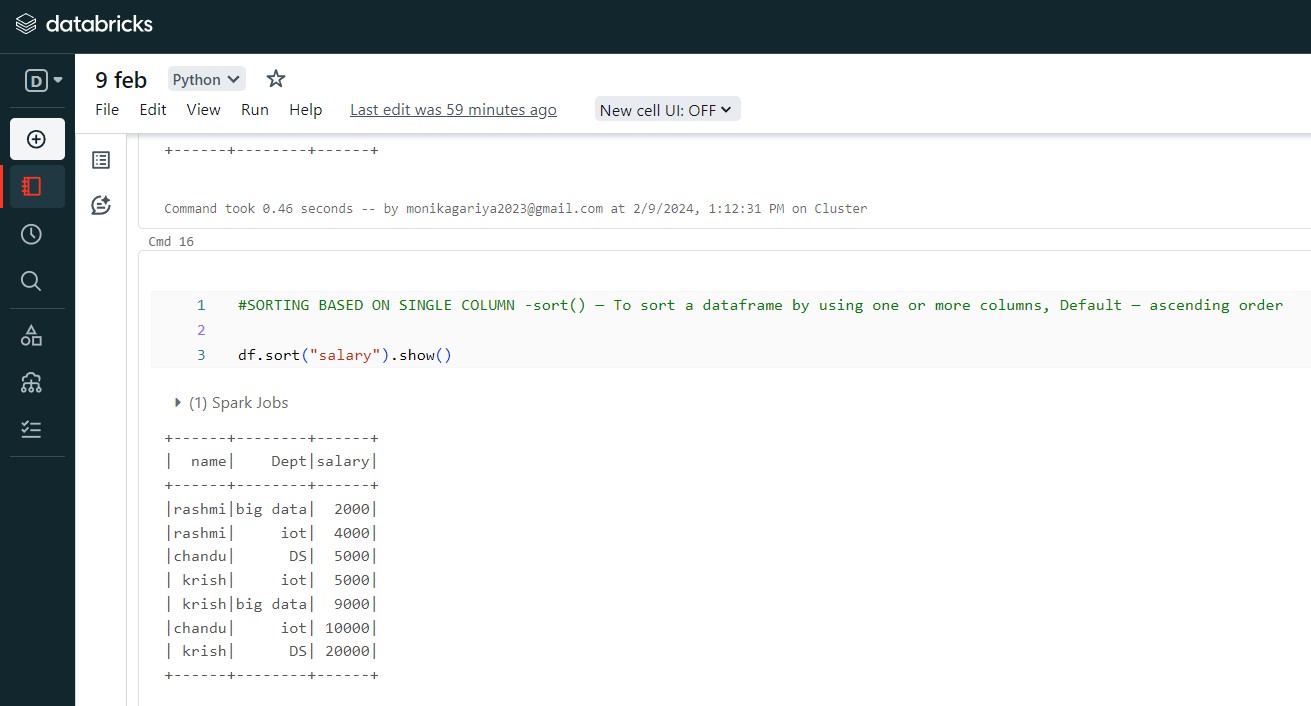


**4)** only in that column rows get deleted

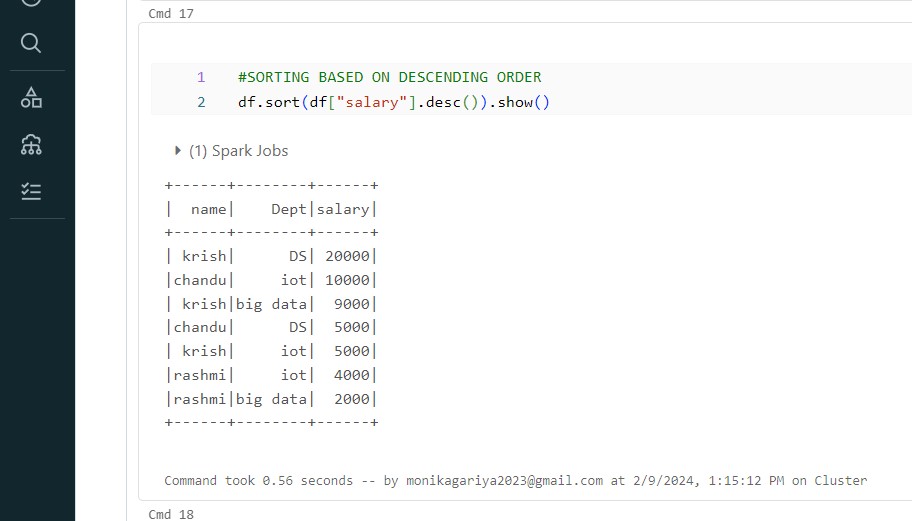


**orderBy() and sort() in Pyspark DataFrame**

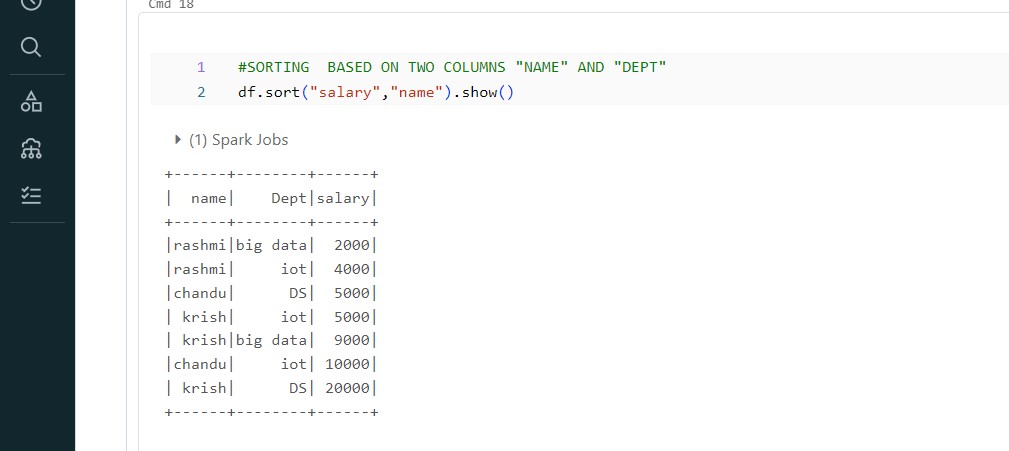
1)sort() — To sort a dataframe by using one or more columns, Default — ascending order.



2)Sorting based on descending order

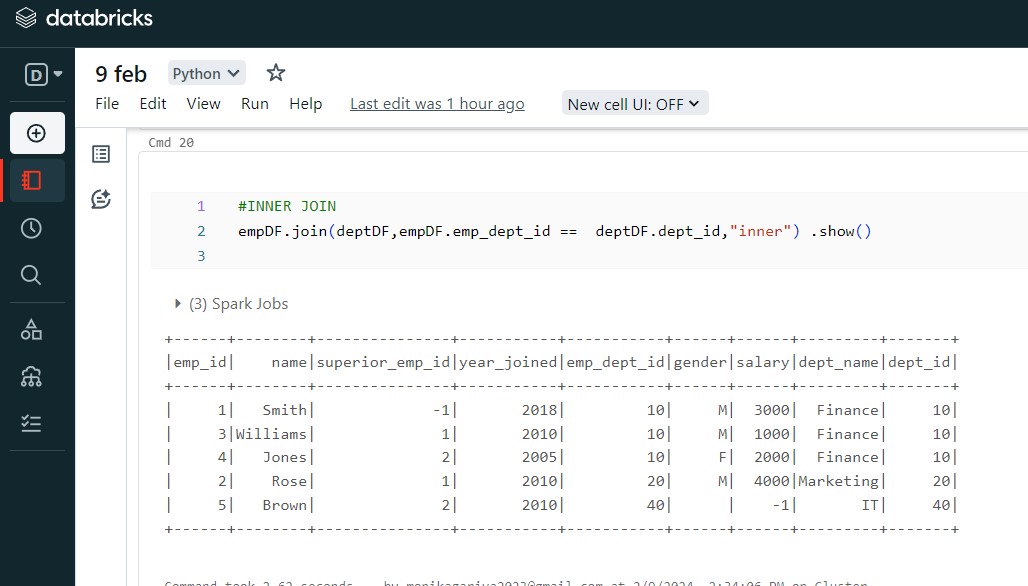


3)Sorting based on two columns “salary” and “name”.



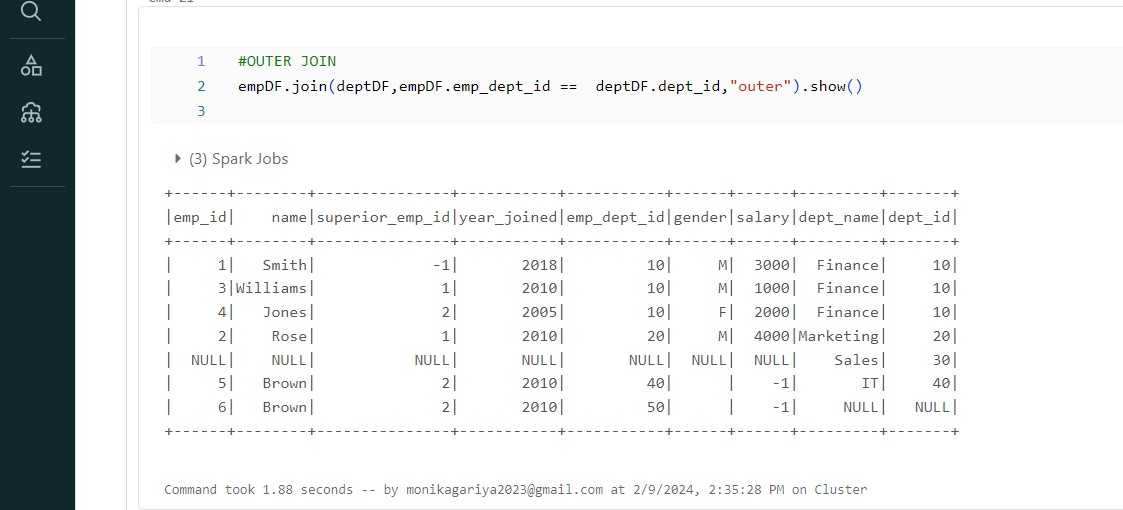
**Inner join**

This will join the two PySpark dataframes on key columns, which are common in both dataframes.



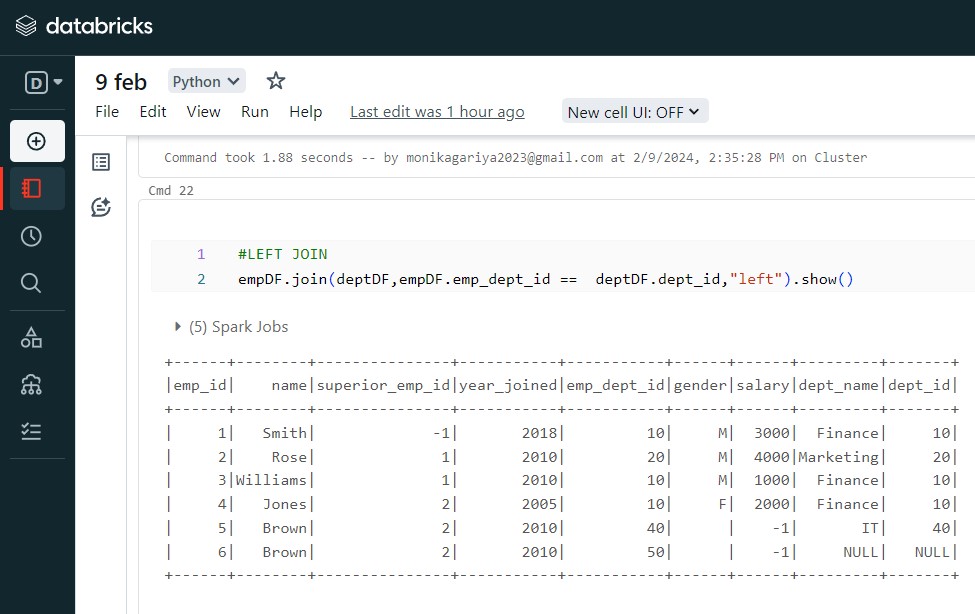
## Full Outer Join

This join joins the two dataframes with all matching and non-matching rows



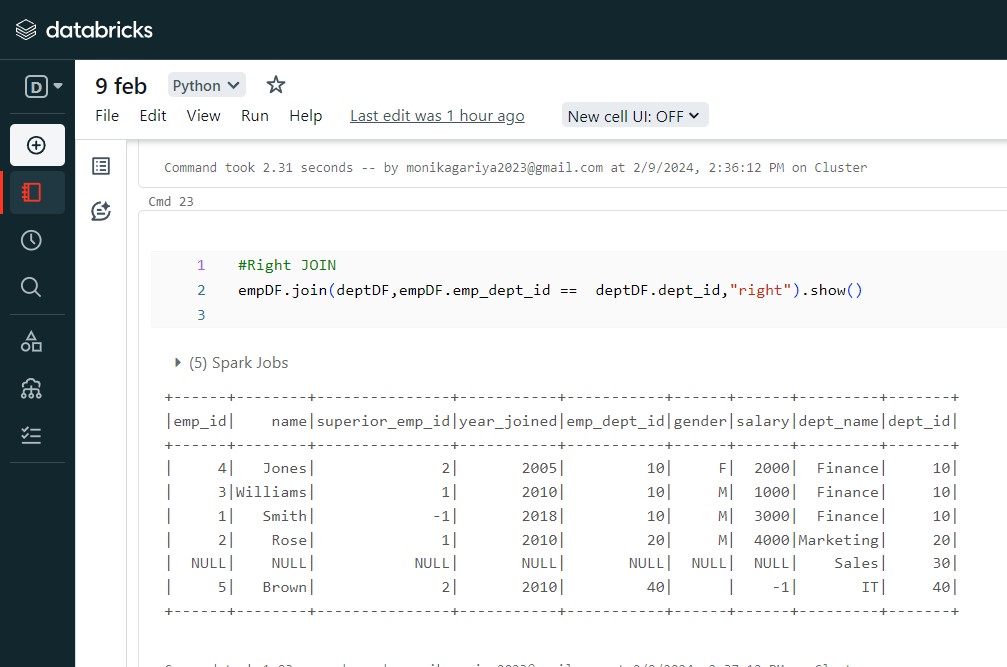
## Left Join

Here this join joins the dataframe by returning all rows from the first dataframe and only matched rows from the second dataframe with respect to the first dataframe.



## Right Join

This join joins the dataframe by returning all rows from the second dataframe and only matched rows from the first dataframe with respect to the second dataframe.



## Leftsemi join

This join will all rows from the first dataframe and return only matched rows from the second dataframe



## LeftAnti join

This join returns only columns from the first dataframe for non-matched records of the second dataframe

