PYSPARK ASSESSMENT

Q1) Execute Manipulating, Dropping, Sorting, Aggregations, Joining, Group By  DataFrames  
Execute Pyspark.

Pyspark: it is an open source distributed framework which is built on Apache Spark which can be used to design for large datasets.

It is helpful for developers to write, fault-tolerant and it is high-performance application in python.

Features of Pyspark :

* Fault-Tolerant
* Can be performed on many clusters
* Distributed processing
* In Memory computation etc

Steps to execute in Pyspark :

Step1 : Importing Pyspark into the cluster

Step2 : From pyspark.sql we need to import Spark Session and functions

Syntax For Spark Session:

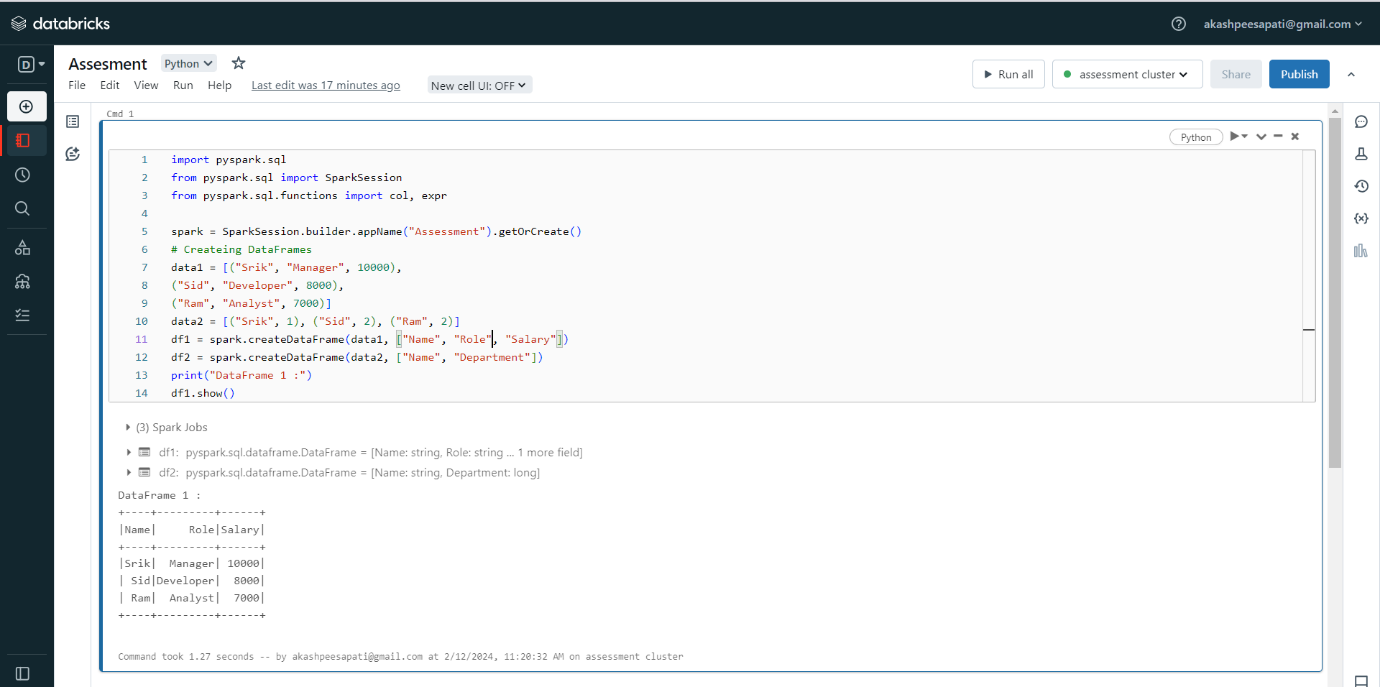
**SparkSession.builder.appName(“CHOOSE ANY NAME ”).getOrCreate()**

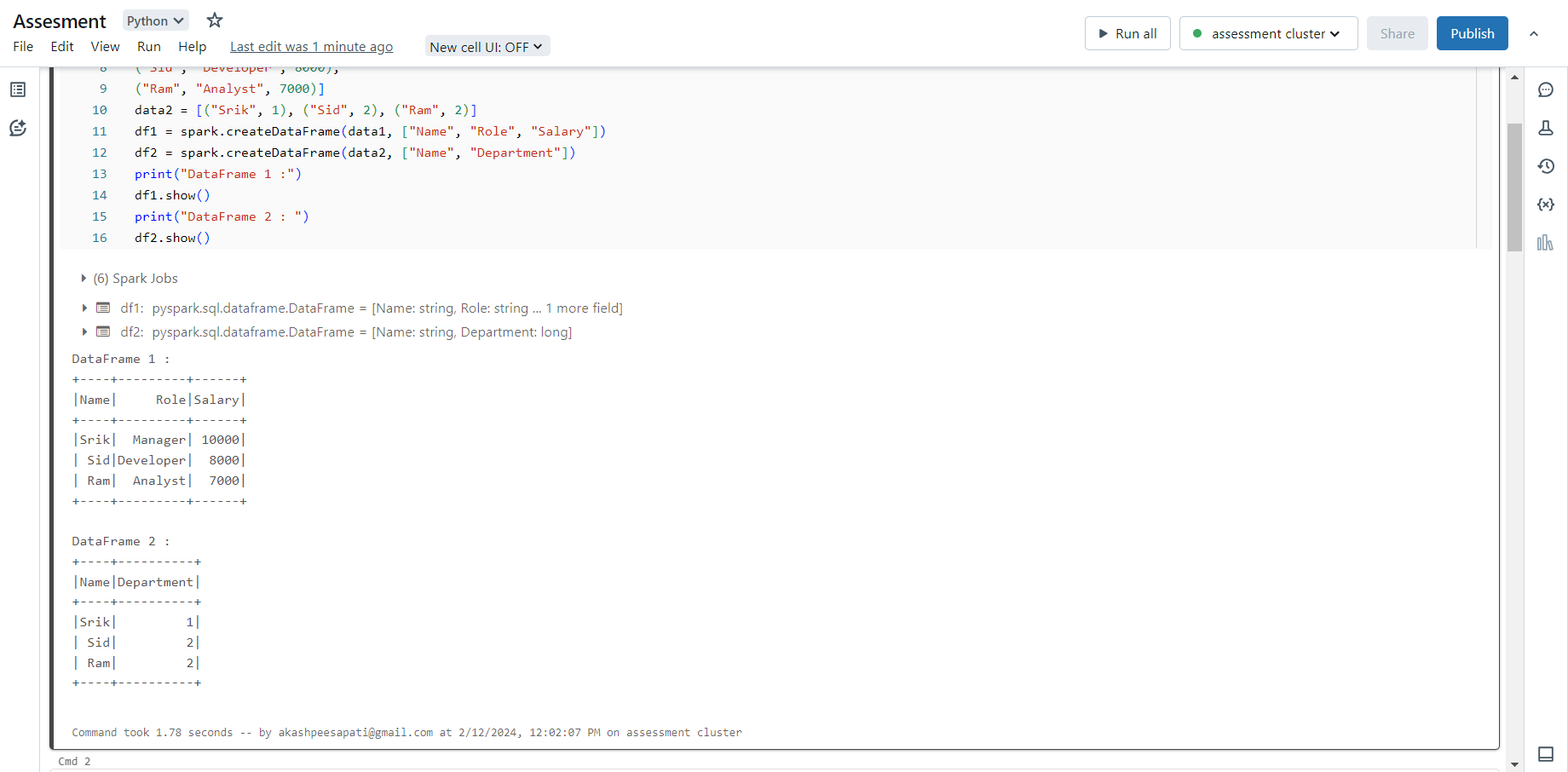
Step3 : Now creating a sample dataframes with data present In it.

We use “**createDataFrame()**” method for creating dataframe

Step4 : And lets print the data inserted into the dataframe by using this syntax below

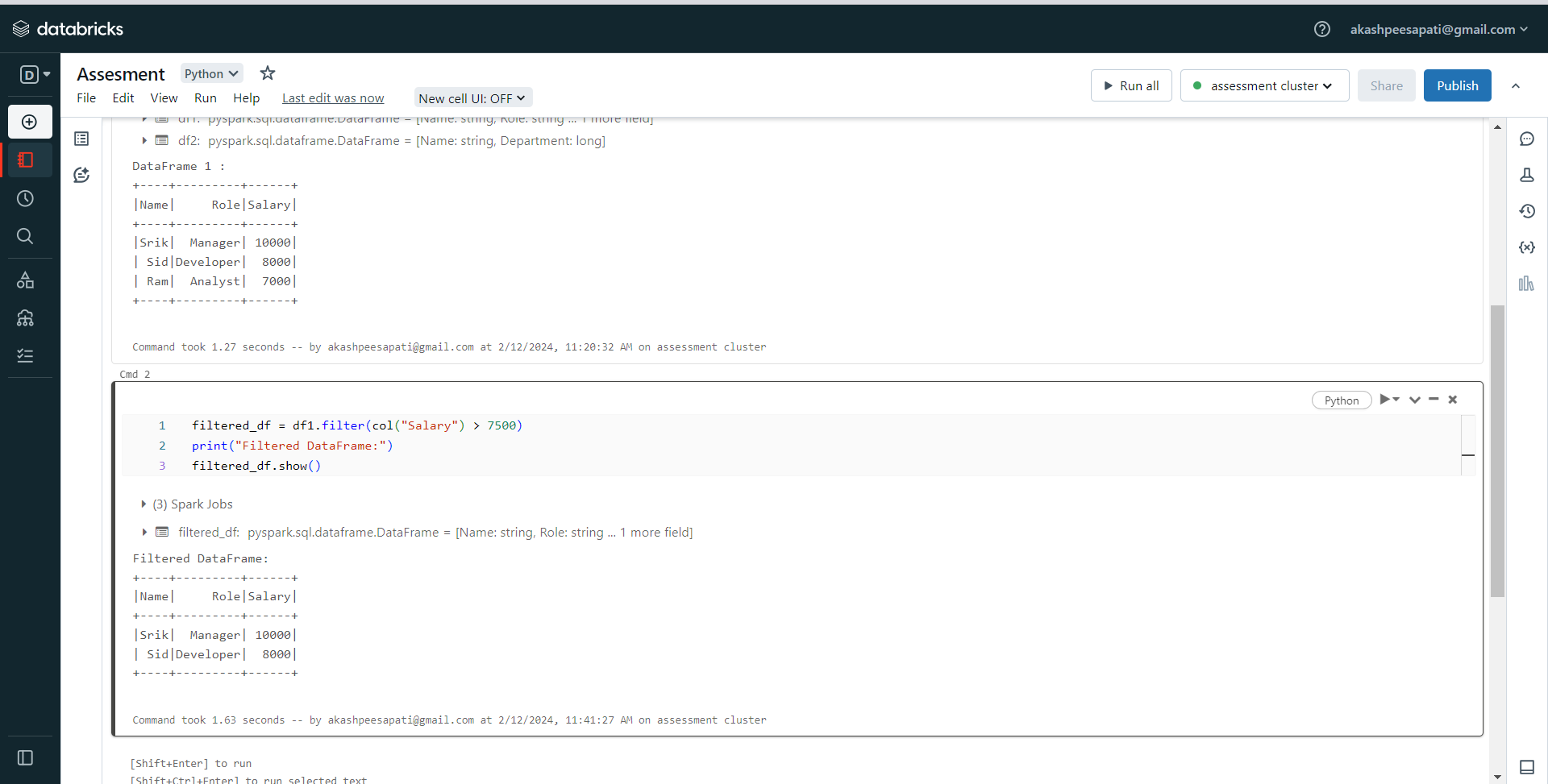
We use **“df.show()”** to show the values.



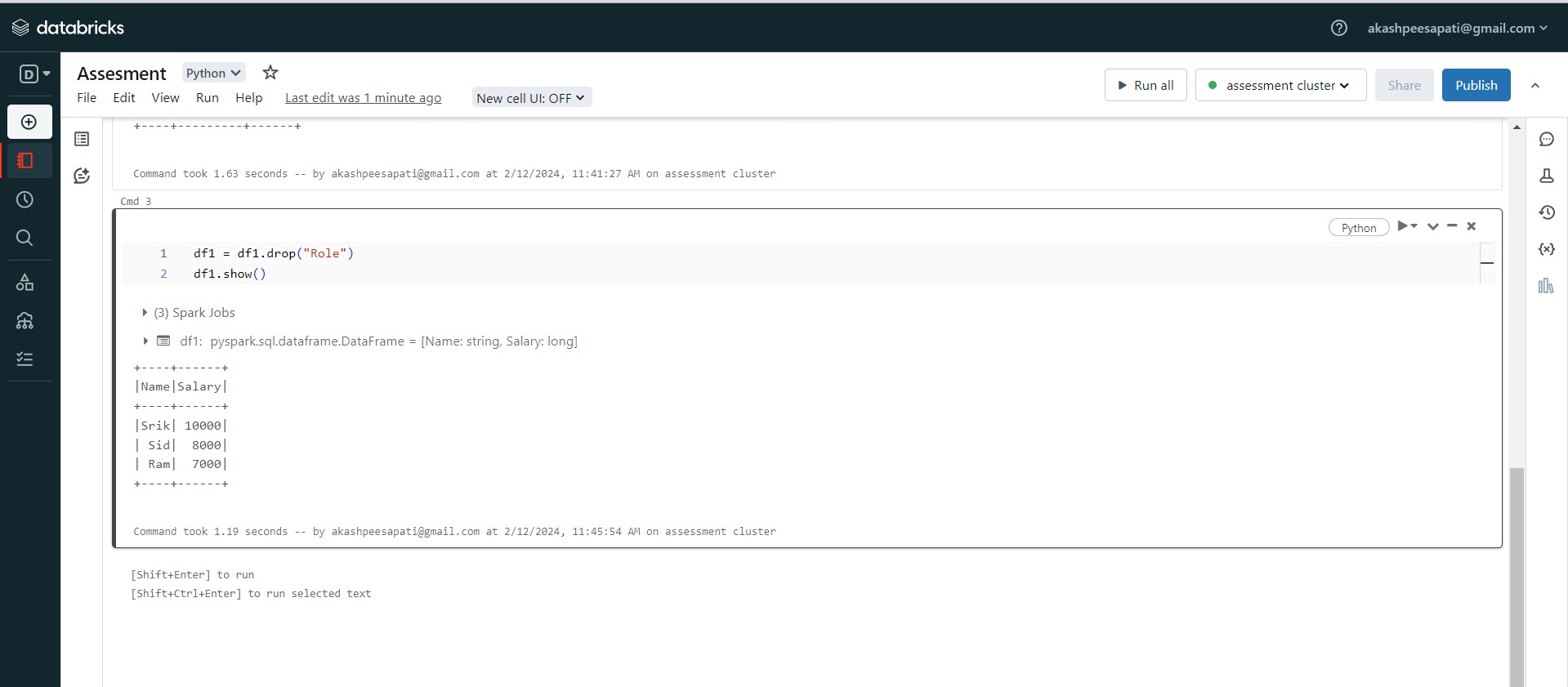


Now for Manipulation dataframe lets take an example below :

Here we will use “filter” for filtering the required data

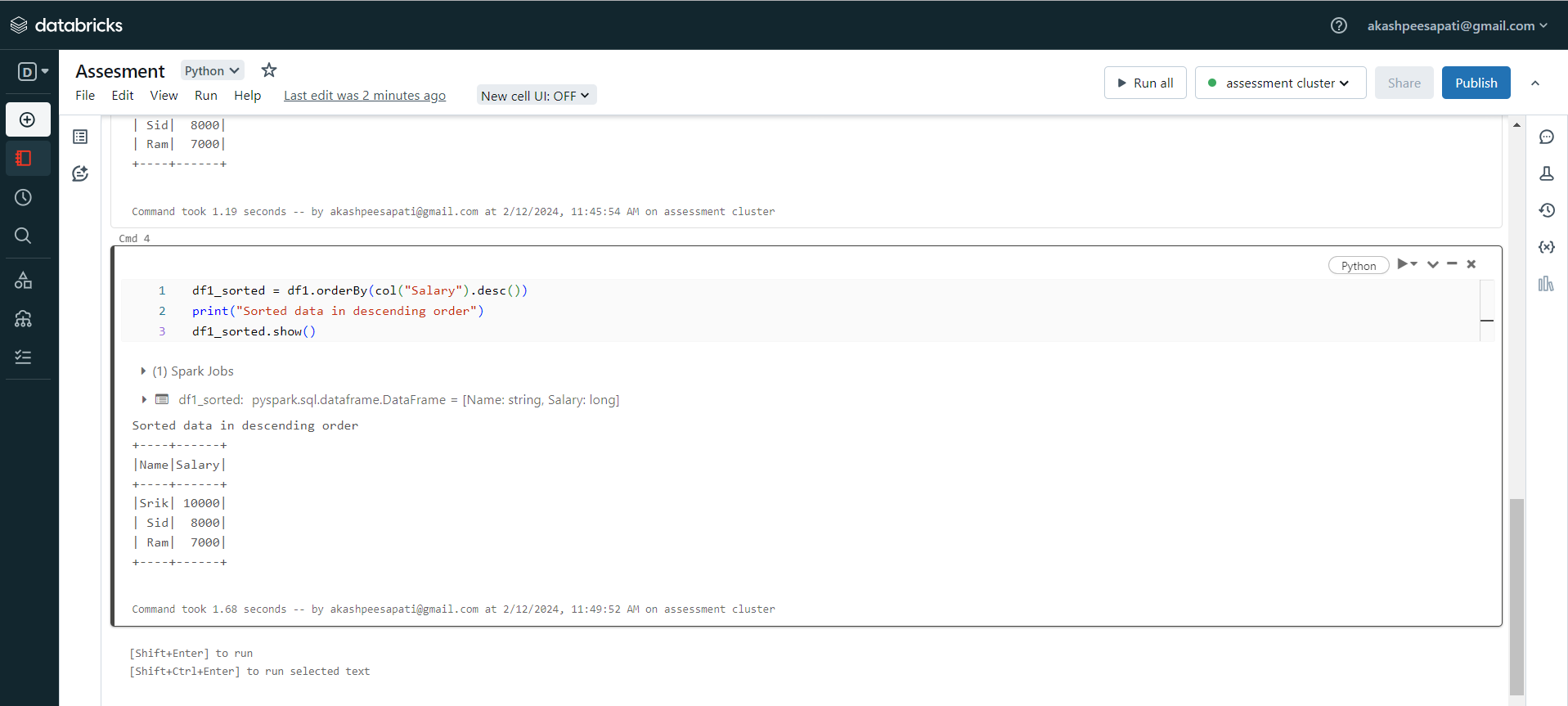


Now lets Drop the column from above Dataframe :  
We use “Drop” to simply all the required column

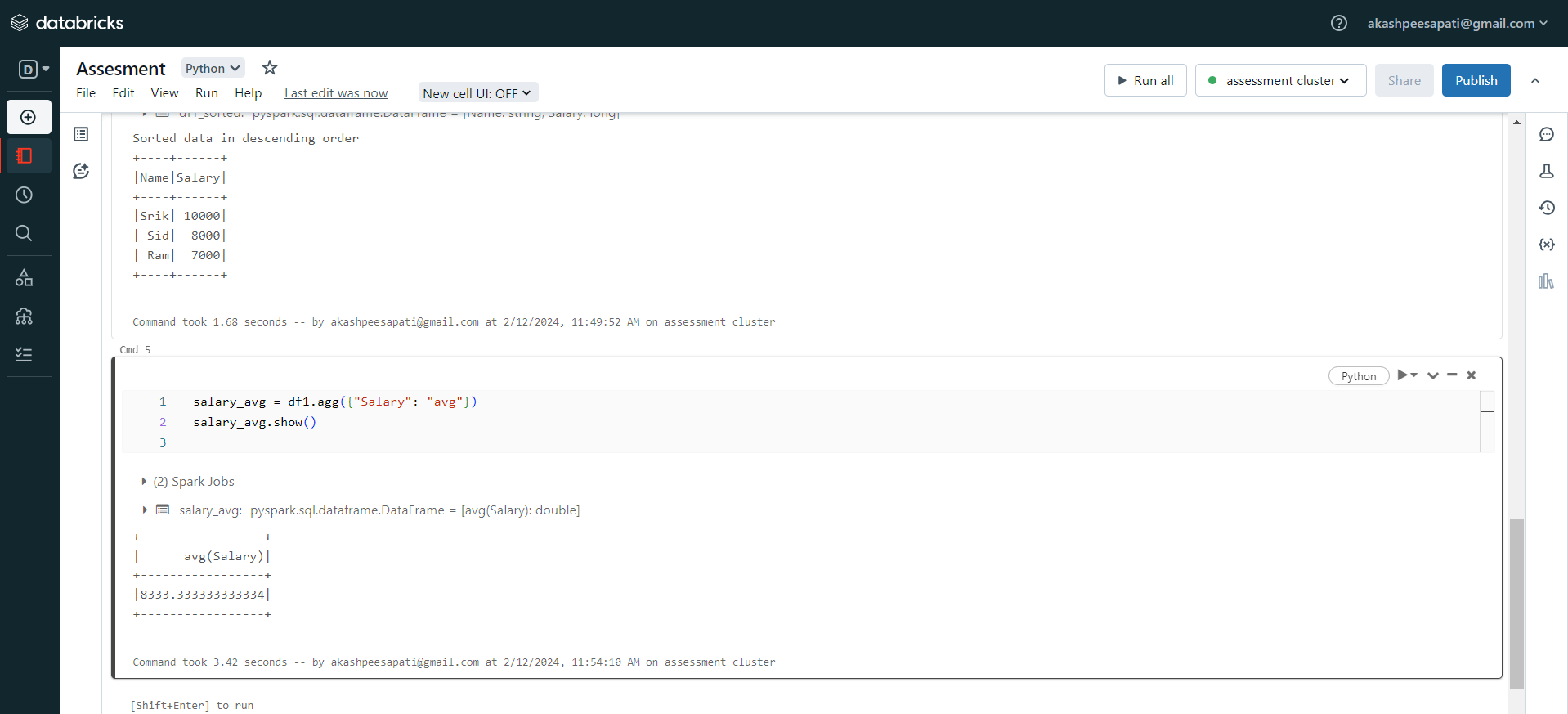


Now lets Sort the data from descending order :

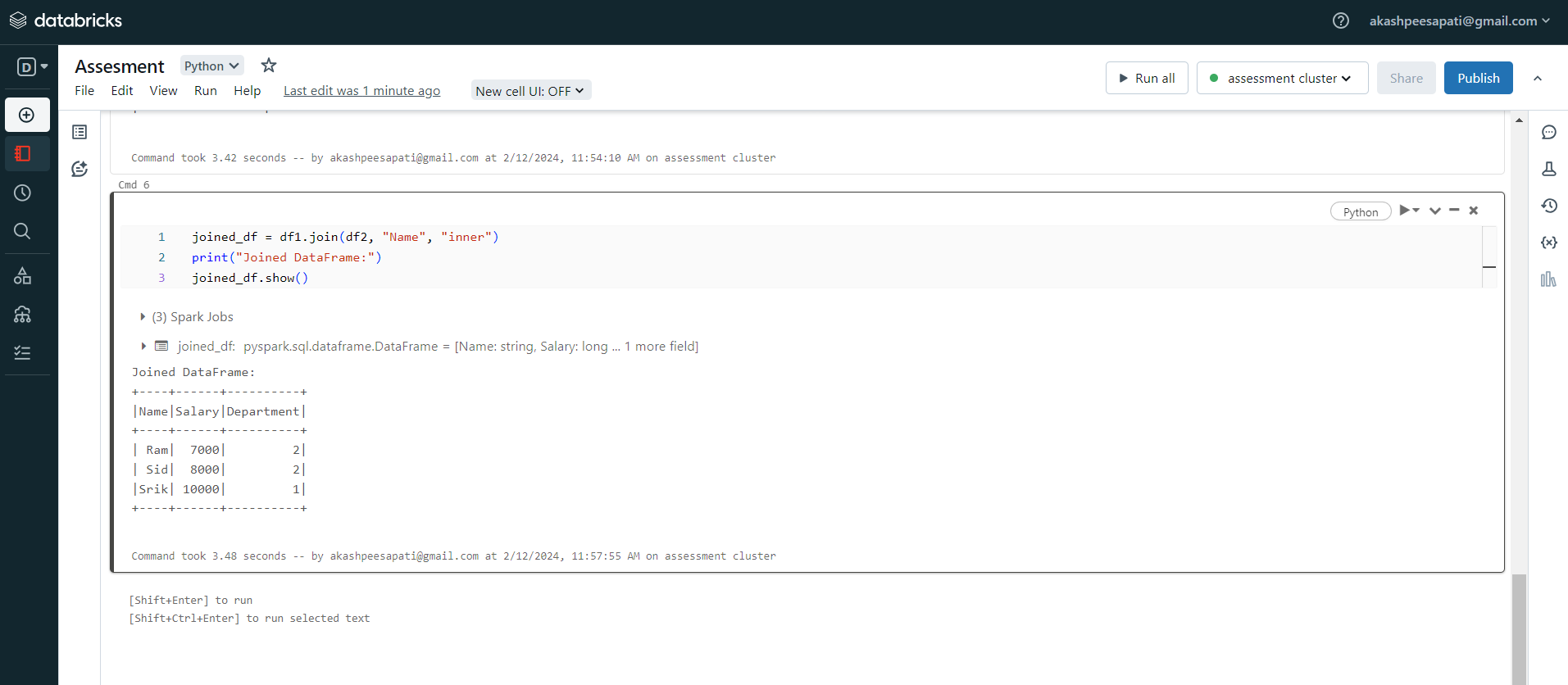
We used “orderBy” which is called on “df1\_sorterd” for salary column and hence after sorting it is stored on “df1\_sorted”



Now for aggregation lets consider “average” function for the salary column and lets find the average salaries from the data :

We used “avg”function for this 

Now lets join the dataframes using “join” function



Now lets perform GroupBy dataframes shown below :

We used “groupBy” function to group the DataFrame by the "Salary" column.

