****

**Big Data Tools and Techniques**

**NAME: ROHIT PASKARAN**

**STUDENT ID: @00649876**

**Abstract:**

The Course Work is Subjected to present the Learning Outcomes of Big Data Tools and Techniques of the past 11 weeks and including the AWS Implementation. Assuming as a Data Analyst Cleaned and Solved the Problems given with the help of Clinical Trial Data Set that is Downloaded from the Givens Links. As a part of Course Work Learned some new Functions for Implementing the Sorting, filtering.

**Data-Source:**

The files were compressed as GZIP and shared in the blackboard

1. clinicaltrial\_2021.csv.gz
2. Mesh and Parma was provided as .csv files

**Tools and Programming Language:**

Note-Book: Databricks (Run Time Version 10.4)

Cloud: AWS (S3 BUCKET)

Programming Language: PySpark Version 3.2.1

, Hive-SQL, Shell Commands, RDD’S

Plotting tools: Bokeh Version 2.4.2

and Matplotlib Version Version 3.4.2

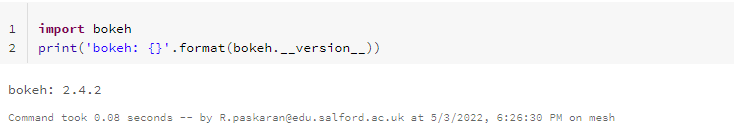
For Loading the Data, cleaning and querying the Databricks notebook was used with the university email login. For the Question 6 Visualization was made with the help of External python Library Bokeh and Matplotlib.

**School-Account:**



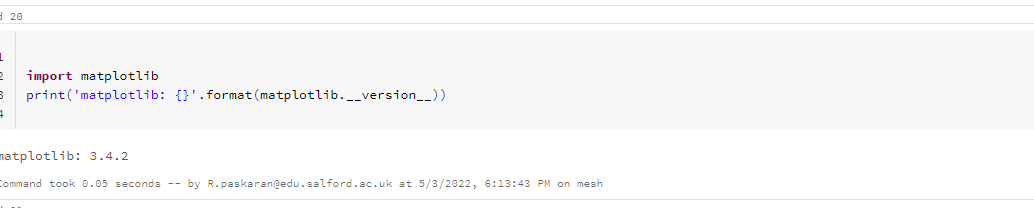
*Fig:1*

**Bokeh Version:**

****

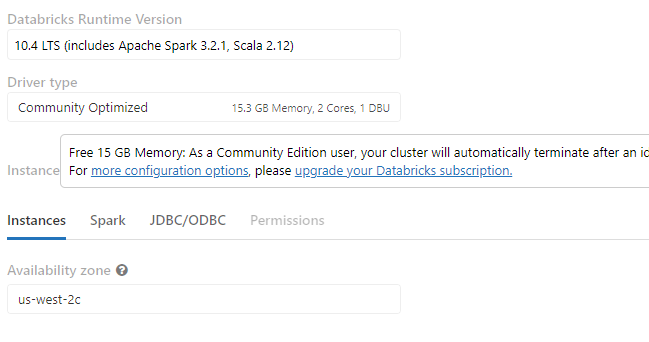
*Fig:2*

**Matplotlib Version:**

****

*Fig:3*

**Databricks and Spark Version:**

****

*Fig:4*

1. **Primary Run Down:**

The Data was Stored in a Common Directory for all the three Implementations. (/FileStore/tables…) which is the Default Directory. The Trial Data was 2019 was checked and then the Submission data(2021) was finally done**.**

* 1. **Cleaning the Data:**

The data as the header included while creating the table. Which was ignored with the help of different methods for all three implementations.

1. Data-Frames:

Using options by passing header value as True.

1. RDD:

Using mapPartitionWithIndex and iterating

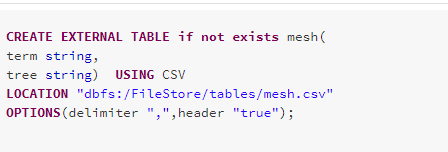
through all values in the form of list and get the data after the first index value.

1. Hive-SQL:

It Done in the same way as Data-Frame

Secondly, some Data’s has Null values But that was ignored with the isnull() Function.

The Data was not Extracted in the required name for the submission. The following Command were used to rename it as mentioned.



*Fig:1.1.1*



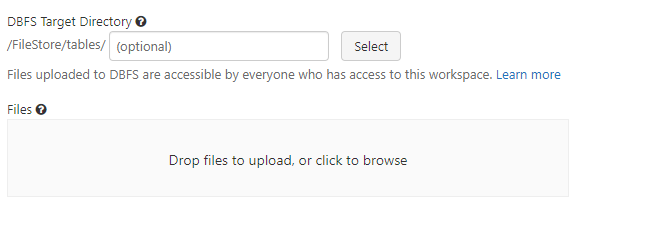
*Fig:1.1.2*

 *Fig:1.1.3*

* 1. **Data Preparation’s and Ingestion :**

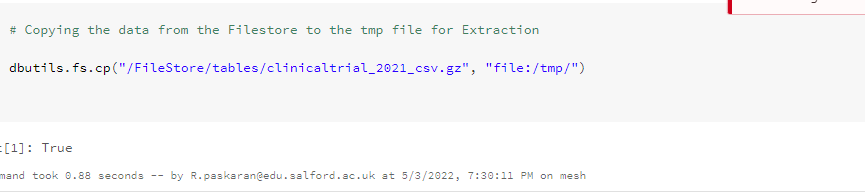
The data was stored in the DBFS file path Under that tables Directory to be access for all implementations. Imported the Zipped file of Clinical-Trial Data directly to the Databricks using the Import option available in the Databricks Notebook which stores the data by default in the Tables Directory.

**Default Path:**

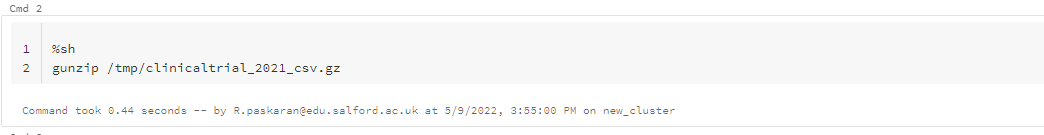
**

*Fig:1.2.1*

Since the data was in zipped format, shell commands were used to unzip it. The data were moved to the local temp directory from the Filestore to process further with shell commands.



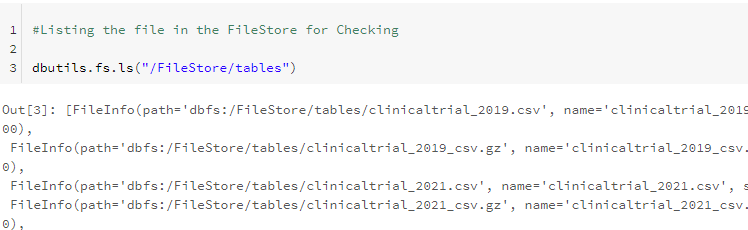
*Fig:1.2.2*

****

*Fig:1.2.3*

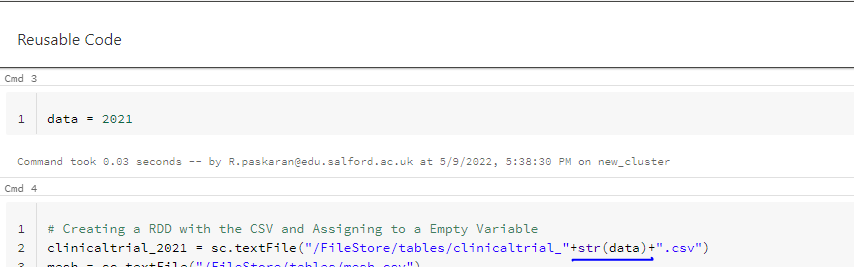
****

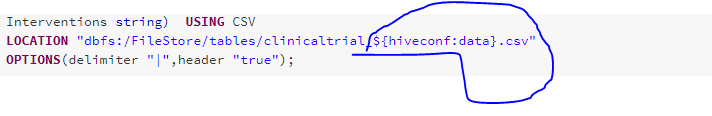
*Fig:1.2.4*

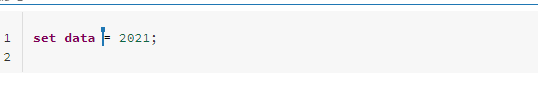
****

*Fig:1.2.6*

\*\*\*\*\*\*\*Since the Data was made into Reusable as per years with the help of creating a variable and assigning a specific years.\*\*\*\*\*\*\*\*\*\*





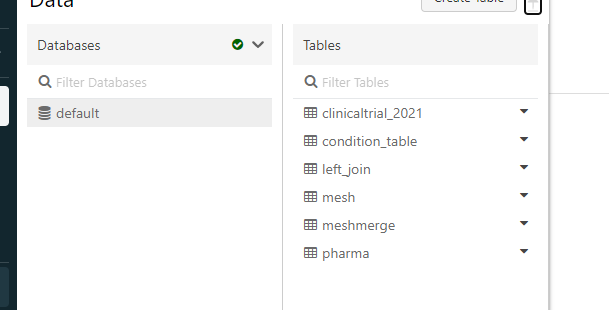


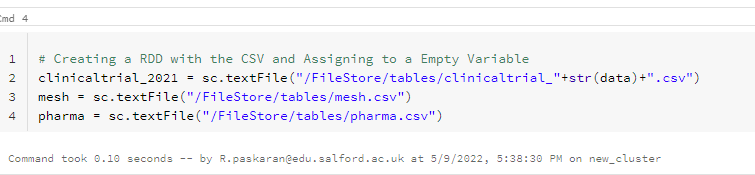
* 1. **Table Creation:**

The table is created separately for each Implementations, When creating for hive (if not exist commands) were used with options. For RDD’s (sc.textfile) was used. Finally for data frame’s (spark.read options) were used.

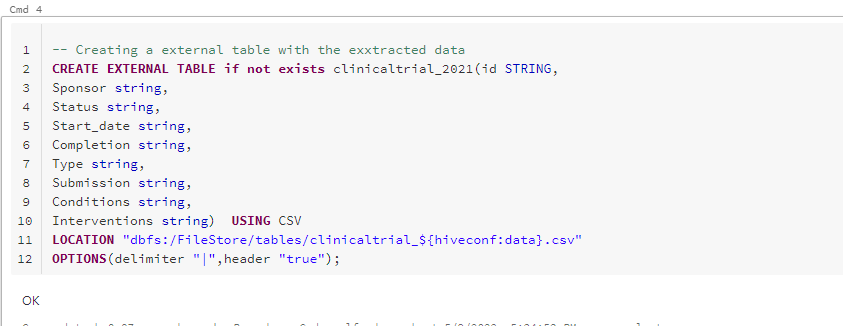
To answer the Six Questions using Hive SQL, Tables were created Separately for each data with the following names. This was same for the other three implementation.

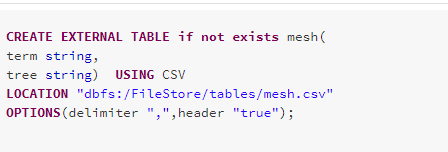
1. clinicaltrial\_2021
2. mesh
3. pharma

* Fig:1.3.1*



*Fig:1.3.2*

 *Fig:1.3.3*



*Fig:1.3.4*

* Fig:1.3.5*

1. **Statements of Problems and the Logics Used to Solve Them:**

As mentioned earlier the Six Problem Statements were Answered in the Following three languages

1. RDD.
2. HIVE SQL.
3. PYSPARK DATAFRAMES.

For Submission 2021 Data-Set were used.

1. **Problem Statement:** 
   * 1. **Problem 1:**

The number of studies in the dataset. You must ensure that you explicitly check distinct studies.

* + 1. **Problem 2:**

You should list all the types (as contained in the Type column) of studies in the dataset along with the frequencies of each type. These should be ordered from most frequent to least frequent.

* + 1. **Problem 3:**

The top 5 conditions (from Conditions) with their frequencies.

* + 1. **Problem 4:**

Each condition can be mapped to one or more hierarchy codes. The client wishes to know the 5 most frequent roots (i.e. the sequence of letters and numbers before the first full stop).

* + 1. **Problem 5:**

Find the 10 most common sponsors that are not pharmaceutical companies, along with the number of clinical trials they have sponsored.

* + 1. **Problem 6:**

Plot number of completed studies each month in a given year.

1. **Problems Solved.** 
   * 1. **Problem Statement 1:**

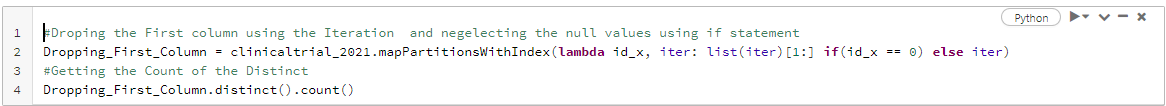
**Assumption of Problem:**

The problem is to know the count of the Individual Clinical Studies. The ClinicalTrial Data Was used at this time. Hence the Data Contains all the clinical Studies with the other data such as sponsor etc. To get the count of the individual Studies Conducted, ID of the studies can be used.

1. **Implementation by RDD:**

**Code Explanation:**

Count were used to get the count of all the ID’s Distinct were used to get the count of unique values.

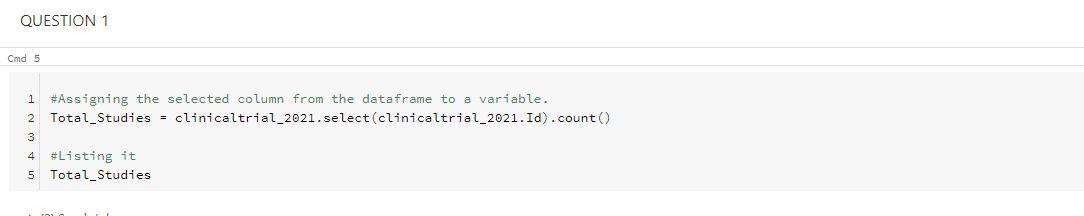


*Fig:3.1*

1. **Implementation by Data-Frames:**

**Code Explanation:**

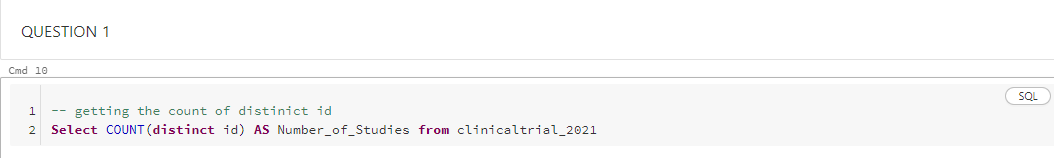
Selecting the particular column Id from the created table using select function and counting it using count function.

 *Fig:3.2*

1. **Implementation by HIVE-SQL:**

**Code Explanation:**

Using the General Hive SQL Select statement the count of the ID is taken to know the count of the Studies. The keyword Distinct is used for taking the unique values out of it.



*Fig:3.3*

**Discussion of Results:**

The result is obtained with the help of count function for all the three implementations and Distinict were used to get the unique values.The CliicalTrial Data was used has this contains the studies.ID was counted for obtaining the results.



*Fig:3.4*

* + 1. **Problem statement 2:**

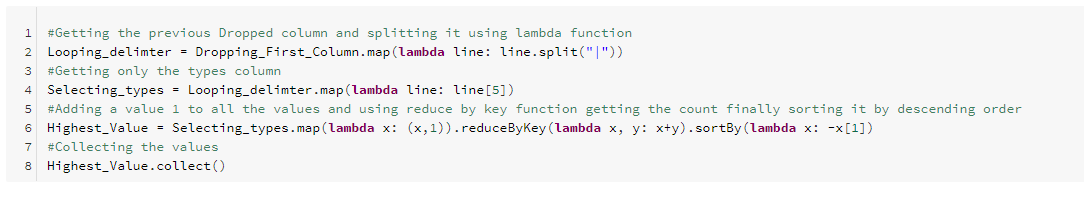
**Assumption of Problem:**

This problem is Specific about one Column of the ClinicalTrial Dataset. Where we need to get the count of the Type column. The types were repeated for each Study. So we need to get the count of each repeated types (Their Frequency). Atlast need to be sort the most highest frequency types first and the least one last

1. **Implementation by RDD:**

**Code Explanation:**

splitting the values by “|” delimeter and selecting only the Types Column. Adding one to each types and counting it using reducebykey function. Sorting it with the sortby function by taking the index and parsing it to a negative variable to populate the values in descending order.

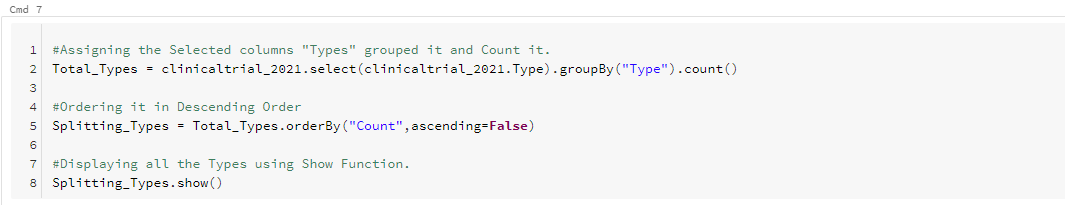


*Fig:2.1*

1. **Implementation by Data-Frames:**

**Code Explanation:**

Selecting the particular column Type from the created table using select function and counting and Grouping it using count and group function. Finally arrange it in descending order with the help of order-by function.

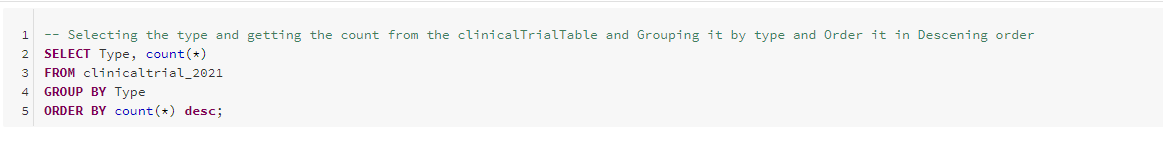


*Fig: 2.2*

1. **Implementation by Hive-SQL:**

**Code Explanation:**

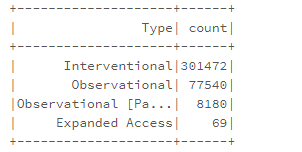
Using the General Hive SQL Select statement the count of the Type is taken from the Main Clinical Table and then Grouped it by type and order it by count in Descending order.



*Fig: 2.3*

**Discussion of Results:**

Group-By was used to get the frequency of the types and order by was used to get the result in descending order. There are only 4 types Among them **Interventional** was the highest repeated, it has frequency of 301472 and the lowest was **Expanded Access** with only 69.



*Fig: 2.4*

* + 1. **Problem statement 3:**

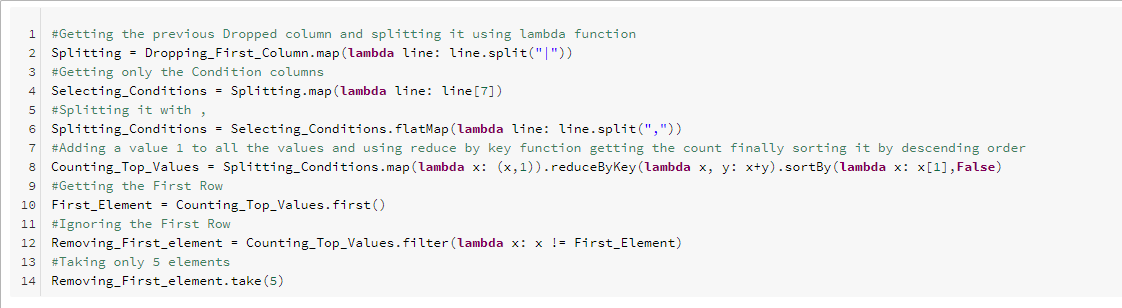
**Assumption of Problem:**

To get the frequency of the conditions and find the top 5 most repeated conditions from the Conditions table. Its same as the previous problem but the conditions column included the interventions and was separated with commas as mentioned in the Document from the BlackBoard, So cant apply the same procedure as the previous question. If do so, we get the count of the interventions too. The best solution would be getting the conditions alone in a separate table with the help of split method

**1. Implementation by RDD:**

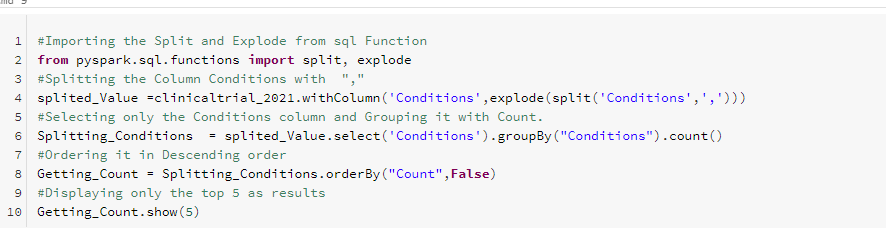
**Explanation:**

Selecting only the Conditions Column by using the lambda function. Splitting the interventions from the Conditions by comma. Later Sort it. Removing the First row and taking the other values to ignore the Column Names.



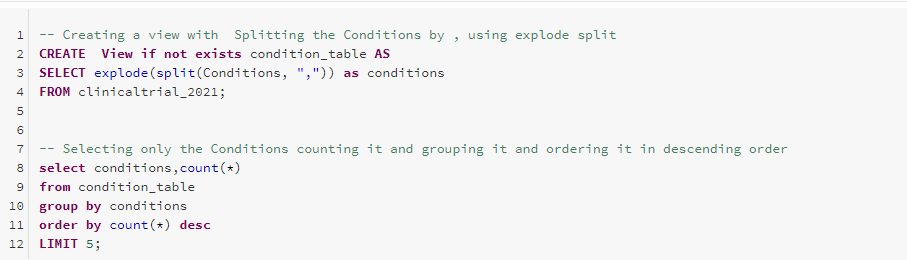
*Fig:3.1*

**2. Implementation by Data-Frames:**

 Split and Explode libraries were used to split it with comma as delimiter and then used Groupby by selecting only the condition column that is created new with the help of explode function and then order it by descending order.

*Fig:3.2*

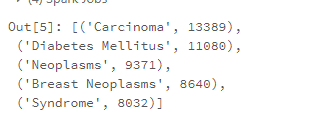
**3. Implementation by HiveSQL:**

This time a view is created and used a explode function to split it by “,” and created a new column for conditions. Then using the select statement taken the count of the conditions, then grouped it by conditions and the finally order it by Count in Descending order.

*Fig:3.3*

**Discussion of Results:**

Imported the explode and Split from the SQL function. After Splitting there are more than 1000 conditions. Then only top 5 results were taken from the results. Carcinoma was the top 1 Condition with a Count of 13389 were as the Syndrome was in the 5th with a Count of 8032.



*Fig:3.4*

* + 1. **Problem statement 4:**

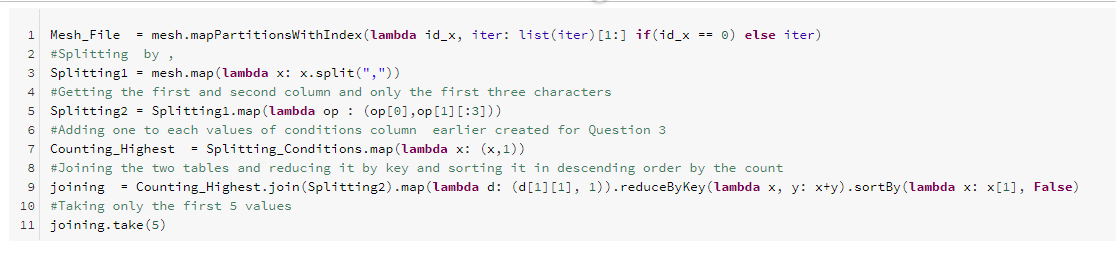
**Assumption of Problem:**

Finding the top root Frequency which is available in the Mesh dataset. But the roots column are interlinked with the Condition column of the clinicaltrial dataset. To find the Frequency of the roots, need to join the two tables and get the Frequency of the roots based on the conditions. As a part of the problem need to get only the first three digit of the roots in the mesh

1. **Implementation by RDD:**

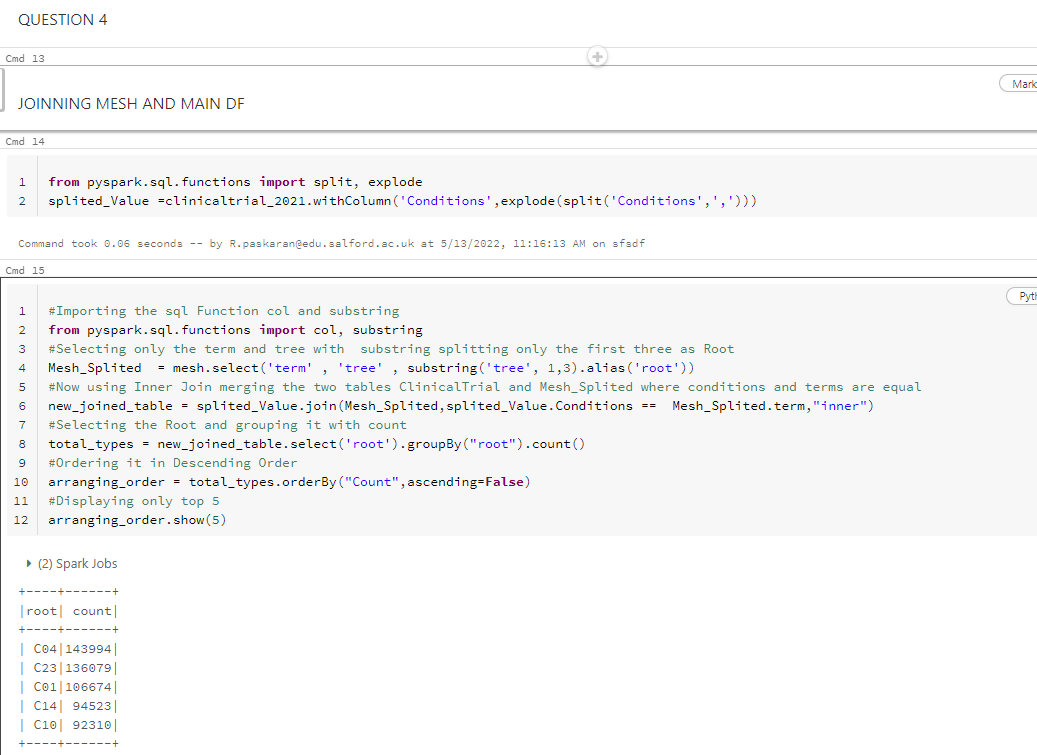
**Explanation:**

splitting it with “,”.Selecting the first three value with mapping the index. Condition table which was edited earlier was joined with the mesh table. Finally after sorting and Filtering the results are obtained.

 *Fig: 4.1*

1. **Implementation by Data-Frames:**

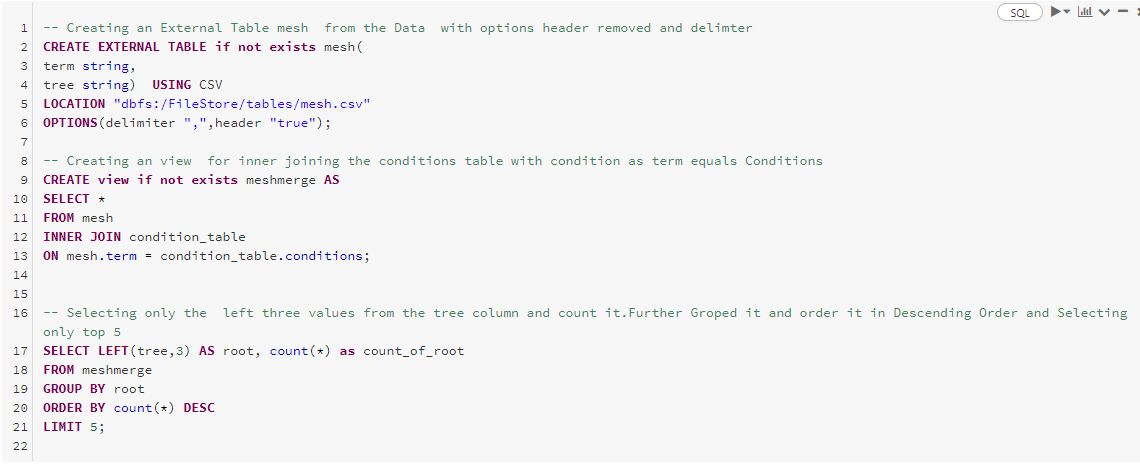
From the mesh table selecting the columns and using substring spilited the first 3 values from the tree column and named it has root. Using the previous selection of conditions that is splited with “,” as used to Inner join with the Conditions equals the term columns. Further Selecting only the root from the table and count it. Finally ordering it with the order-by function to make it in descending order.



*Fig: 4.2*

1. **Implementing using Hive-SQL:**

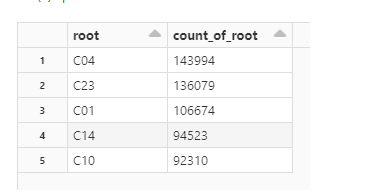
The mesh table is Inner joined with the Conditions table by parsing the conditions term equals conditions, where it is already splitted by commas in the earlier step. Now using the Left function selecting the first three values in the left and getting the count of it. Finally arranging it in Descending order.



*Fig: 4.3*

**Discussion of Results:**

The mesh data was joined with clinical-Trial Data for getting the outcomes. The root is hierarchy of the Condition’ so it has named with more alphanumeric values.C04 has the highest value of 143994 and the 10 was in top 5 92310 values.



*Fig: 4.4*

* + 1. **Problem 5:**

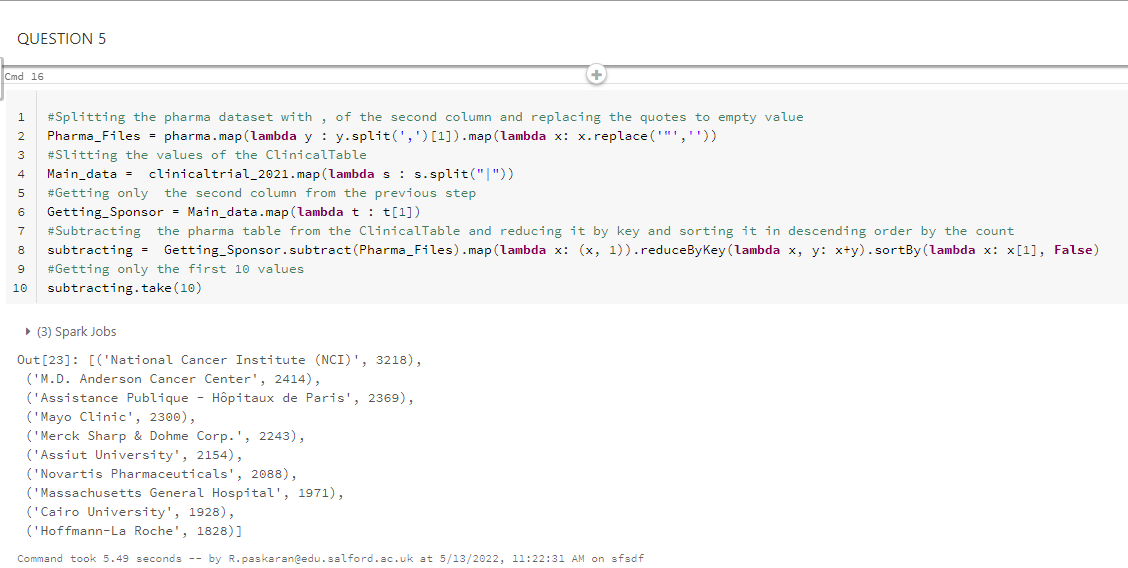
**Assumption of Problem:**

The sponsors column were in the ClinicalTrial Table. But the major problem is to find out the sponsor list which is not a pharma companies. The hint clearly states that the parent companies in the pharma data are all pharmaceutical companies. So as an initiative to compare the columns and neglect the values in the pharma data and follow the regular steps to get the count. Once again we need to compare the data’s.

1. **Implementation by RDD:**

**Explanation:**

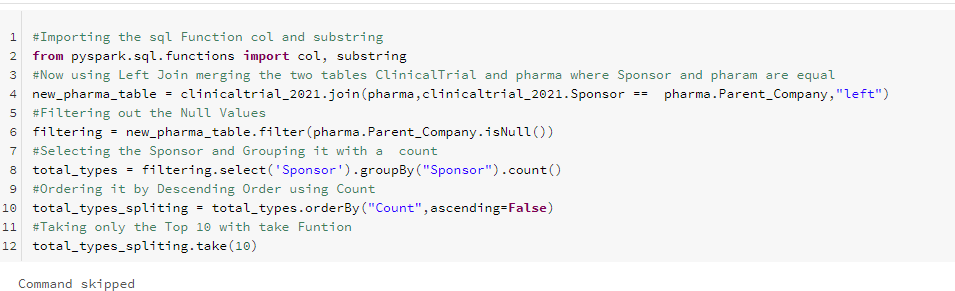
Step 1, splitting the parent company value from parent data with the array index [1] and replacing the empty space with “,”.The clinical Data was splited by “|”.Getting only the sponsors from the clinical trial table. Using subtracting method removing the values from the clinical trial data from the pharma datasets. As of the other Steps Filtering and ordering was used to get the Desired Results.

****

*Fig: 5.1*

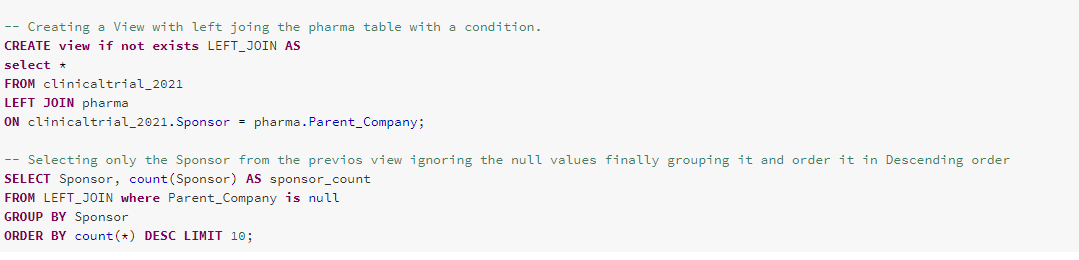
1. **Implementation by Data-Frame:**

The clinicaltrial dataset is joined with pharma dataset where the sponsor equals to the parent company in pharma using left join. It will get only the values in the left table which is not in the right table. Null values are ignored with the help of is Null Function. Further Grouping the table with the sponsor column and getting the count of it. Finally arranged it in descending order with the help of order by.

 *Fig: 5.2*

1. **Implementation by Hive-SQL:**

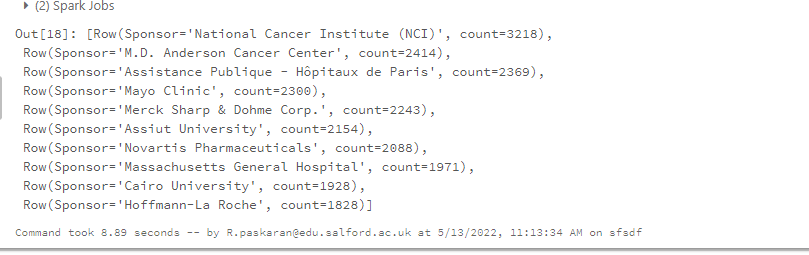
Creating a view to left join the two tables with a condition that clinicaltrial sponsor is equal to the parent company in pharma table. Selecting the Sponsor and get the count of the joined table and order it by descending without the null values using order by.



*Fig: 5.3*

**Discussion of Results:**

The pharma data has all the list of pharmaceutical companies. According to the result National Cancer Institute (NCI) was sponsored for more Trials with a count of 3218.

****

*Fig: 5.4*

**Problem 6:**

**Assumption of Problem:**

To check the number of completed studies, in the given year. As for the Submission 2021 has been initiated as the completed year. The ClinicalTrial has the list of completed year and the status states that its Completed or not. Its to take two columns and visualize it for better understanding.

1. **Implementation by RDD:**

**Explanation:**

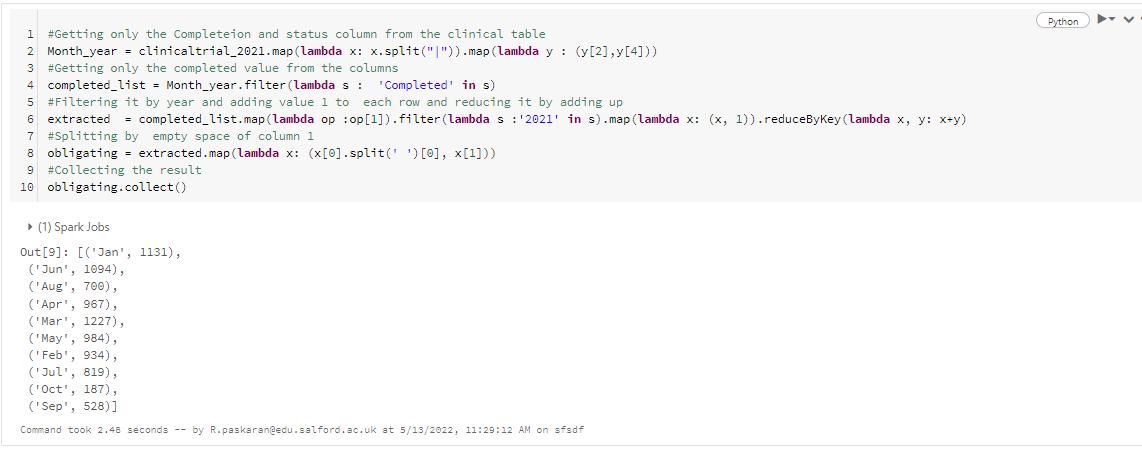
Need to get only the two columns ‘Completed’ ,‘Status’ from ClinicalTrial data.Filtering the columns whose value = ‘Completed’.Filtering it further for getting the values of a particular year (2021) in the completed column

Removing the year from the completed.

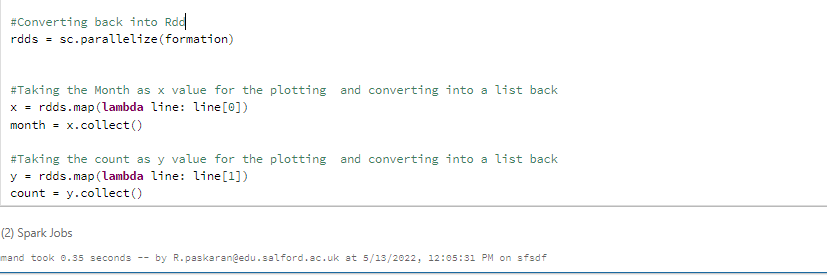
To make it look same as the answer expected, it was arranged in month wise order.

**Coding Explanation:**

Calender library was Imported from python and enumerated by month by parsing the list of 1 and taking the second row. Finally using sort by key sorted it by month wise were the keyfun was used to implement the lambda function and collecting as an list. The sorted data was in the list type so, it was later converted back to rdd using sc.parallelize for plotting with x and y values.

 *Fig: 6.1*

* Fig: 6.2*



*Fig: 6.3*

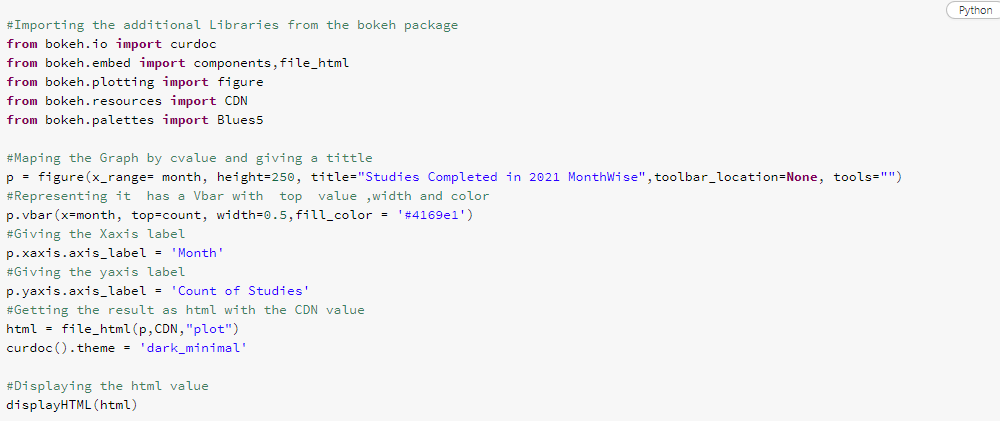
**Visualization:**

**Explanation:**

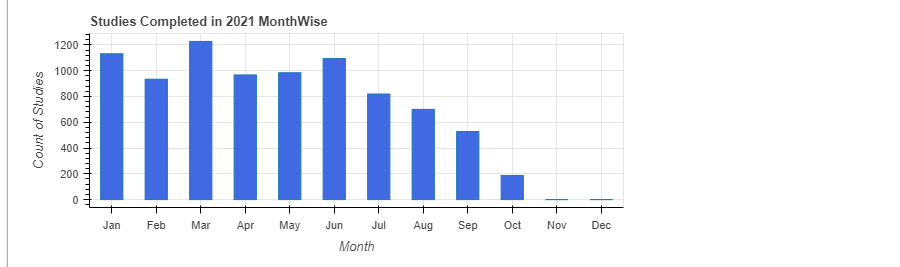
Bokeh was first installed using the pip install Command

Then Import the various additional libraries was imported Such as CDN, Blues5 from Bokeh.

The figure is created by plotting the x range, height and the tittle.Then the graph is represented as vbar which stands to implement the bar chart with x value and the other properties such as color and the width. Then used label to make the x-axis ,y-axis. Finally the result was taken as an html file.



*Fig: 6.4*



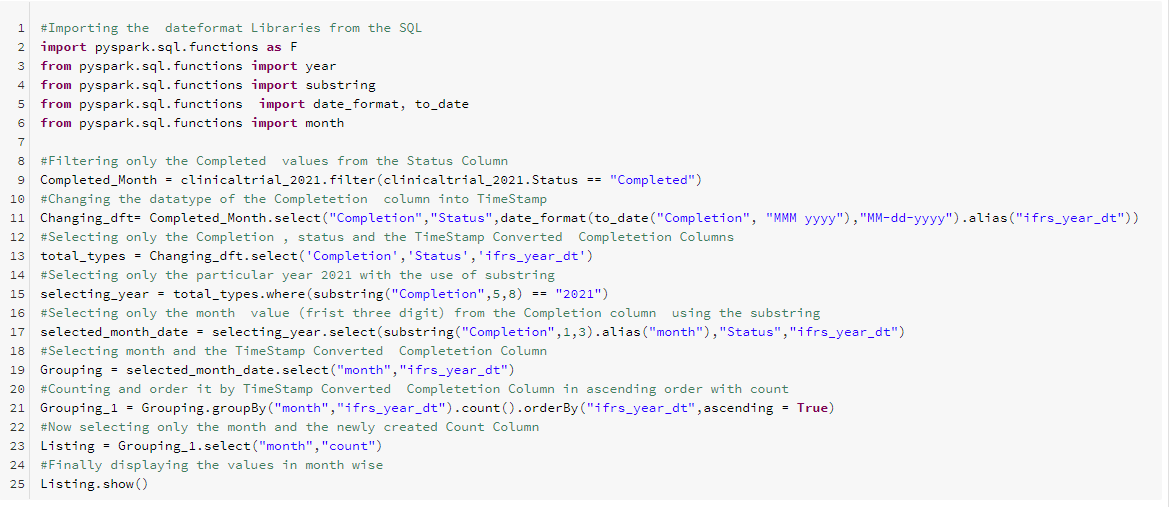
*Fig: 6.5*

1. **Implementation by Data-Frames:**

**Code Explanation:**

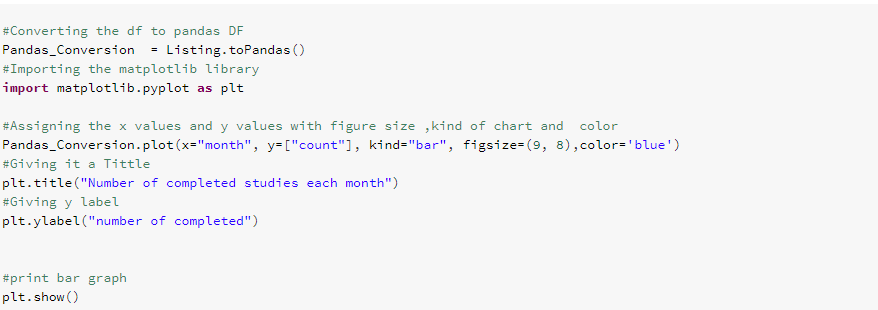
Getting the Completed value of the Status Column using filter method. changed the Completion column value to date format using the SQL data format function. Then selecting only the completion, status and the converted completion column. Further filtering the completion column only for a

particular year using substring and specify the particular index. Then selecting only the month and completion column for grouping, counting and ordering it in ascending order month wise. Finally selecting only the month and the count.

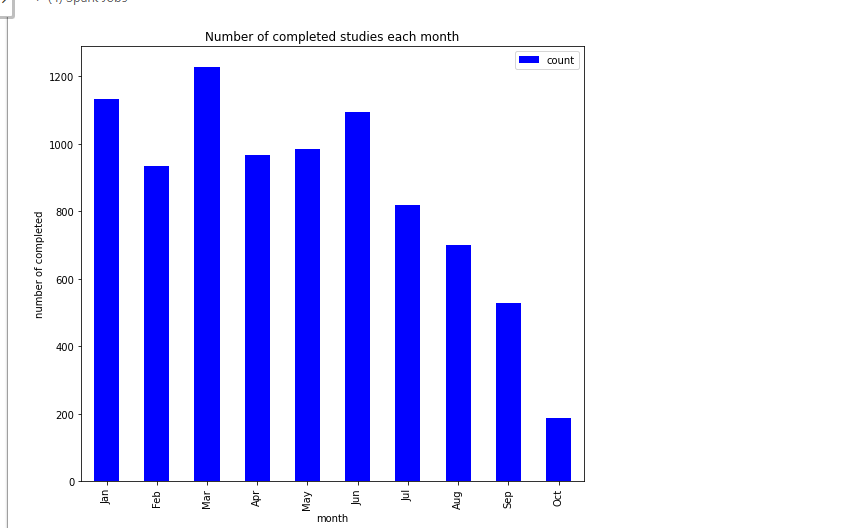


*Fig: 6.6*

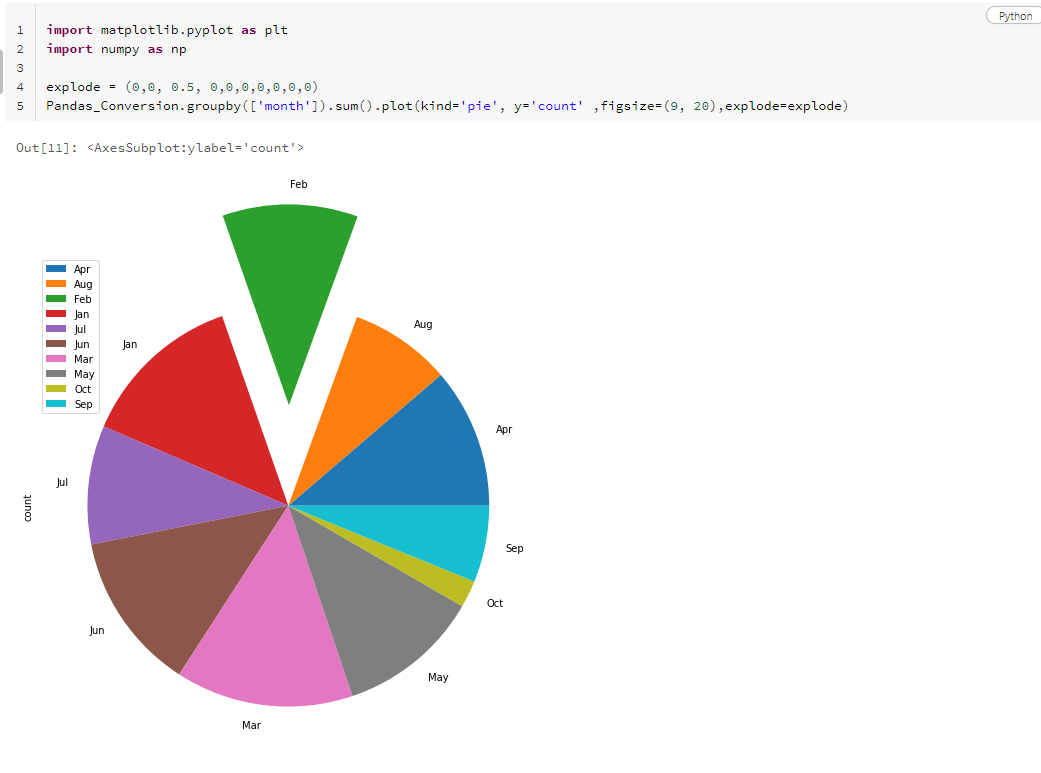
Matplotlib is used. The pyspark Dataframe is converted in pandas dataframes to plot the x and y values in matplotlib along with the tittle and ylabel.

****

*Fig.6.7*



*Fig.6.8*

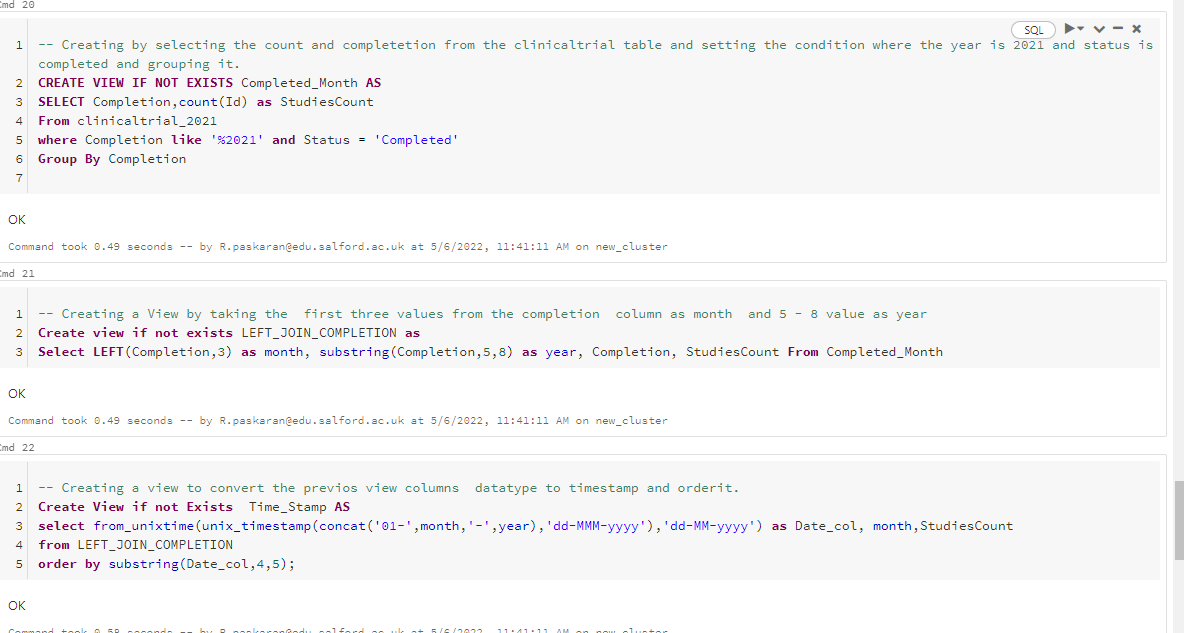


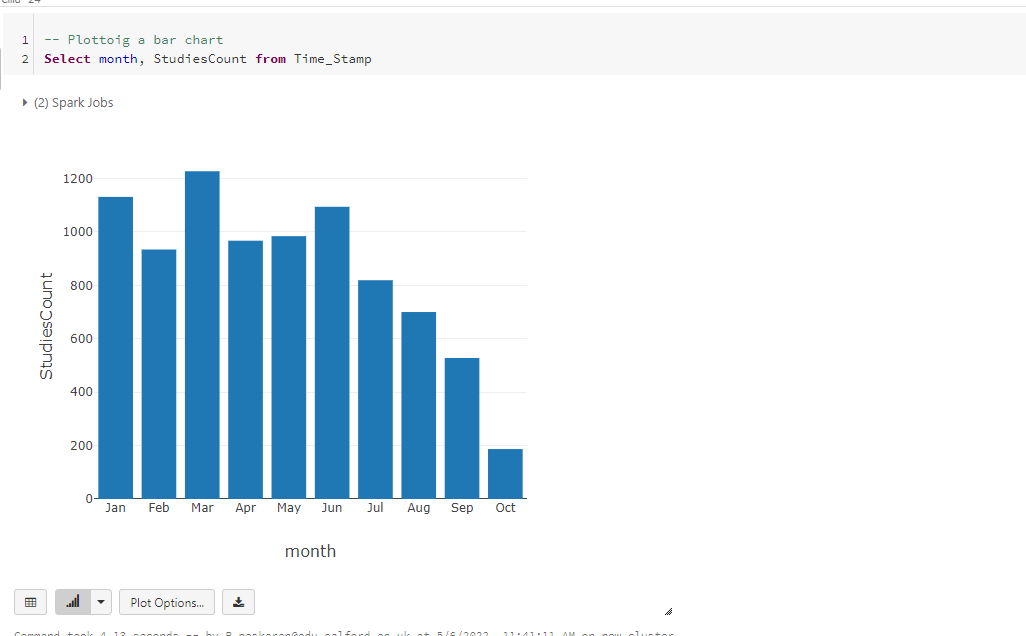
*Fig.6.9*

1. **Problem Solved with Hive SQL:**

Created a new view which contains the completion and the count from the clinicaltrial table and passed into a where conditions which filter out by particular year and the status equals completed and finally grouped by completion. Created a other View that splits the completion month and year in separate columns using substring. For arranging the values by month order unix timestamp is used to change the month column to timestamp value and then order it by the timestamp converted column.

Since there was a default plotting available in hive-sql that is used for plotting as bar graph.

 *Fig.6.10*

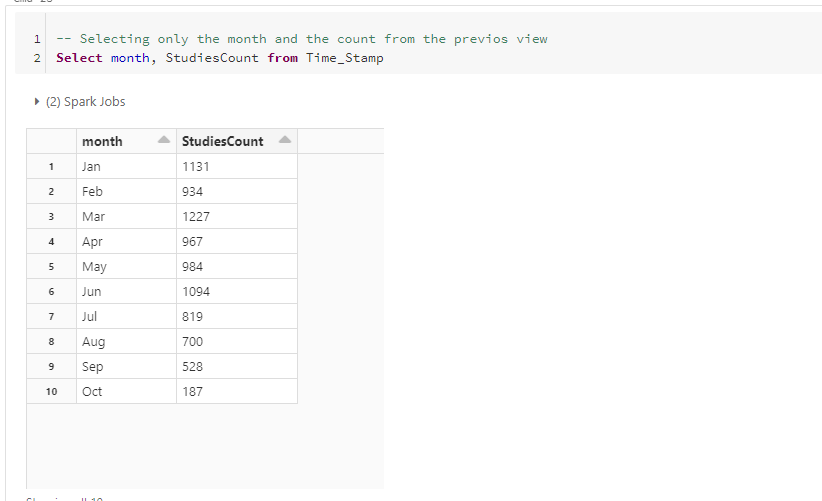


*Fig.6.11*

**Discussion of Results:**

The Completion data was taken for the year 2021. The Studies were only completed from Jan to Oct. The remaining two months were not recorded.

The March month has the highest Completed Studies Recorded. The Oct Month has the Lowest Studies Completed. For Better Visualization Graphs were plotted. (6.12)



*Fig.6.12.*

**Summary:**

The Implementation was done by all the three method and the output was tested with the sample data given with the past results. And this Submission includes only the year 2021 with all the Visualisation parts. Its reusable for the other codes too with the change of only one variable at the end.

**Further Analysis:**

For Each Implementation the Further Analysis was made on Different Questions.

**RDD:**

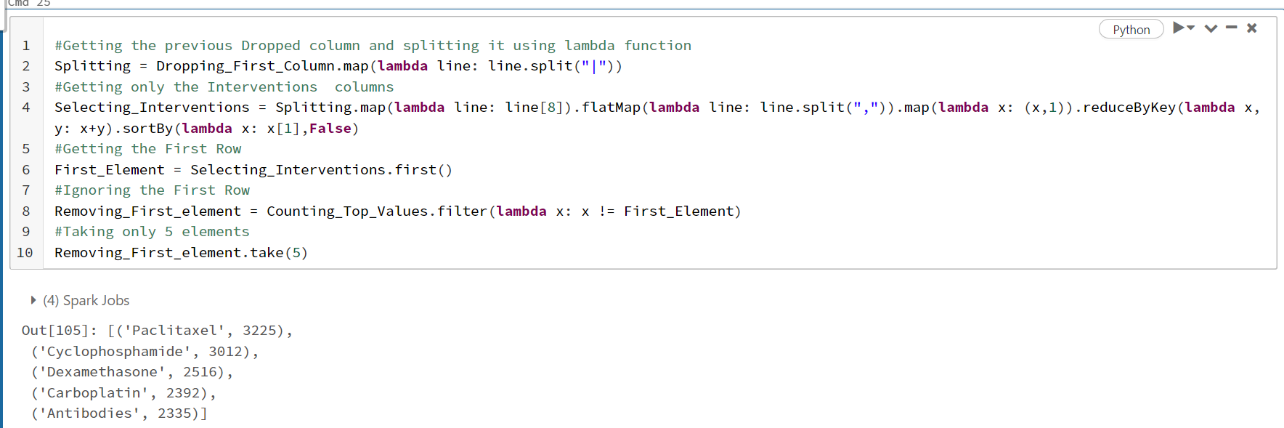
**To find out top 5 interventions.**

**Analysis Explanation:**

This was Done to check the Interventions which repeated more times to cure the Diseases. The top 5 was consider has the Best Solution for Curing the Diseases.

**Coding Explanation:**

The same method of the Question 3 was used but this time The logics Was changed to get the results for the Interventions.



**Fig. 7.1**

**Result Discussion:**

As per the Out-Come , The Intervention Paclitaxel has the highest number of Frequency compared to all the other Interventions.

**Data Frames:**

Companies that have made the highest Number of Offences related to health.

**Problem Discussion:**

This was Carried out to find out the Companies that have made the highest Issues on the HealthRelated offence.

**Code Explanation:**

The Company and the Offense Group was selected and filtered the offence group with healthrelated offence. Later used group by count to find the frequency and order it using the order by command.



**Fig. 7.2**

Result Discussion :

As per the Result the Pharma Companies which makes the Highest offence.”Jhonsons and jhonson ” was made the highest offence.

**HIVE-SQL :**

Finding the top 10 Sponsors count who failed in the Studies.

**Problem Discussion:**

To find the count of the number of Studies Failed by the Sponsors.

**Code Explanation:**

Selecting the sponsor and the Count of status column from the ClinicalTrial Datasets. Filtering it with the Status conditions. Grouping it with the Sponsor column and ordering it with the status count columns.

****

**Fig. 7.3**

**Result Discussion:**

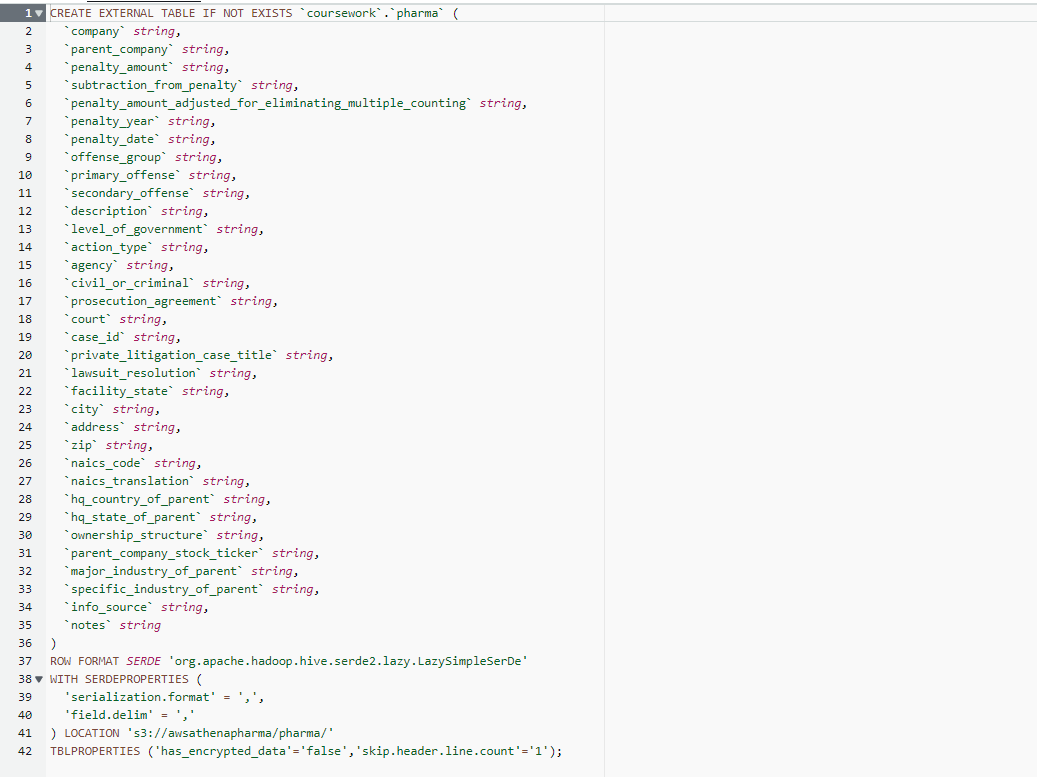
The Highest Number of Failed Studies was made by the NCI.

**Athena Implementations:**

The problems were also Queryed in AWS Athena. The files were uploaded to the AWS s3 Buckets and then created Database and tables with the uploaded files. For the format .**csv** was used. This was Done with the help of AWS Tutorials Lab 2.

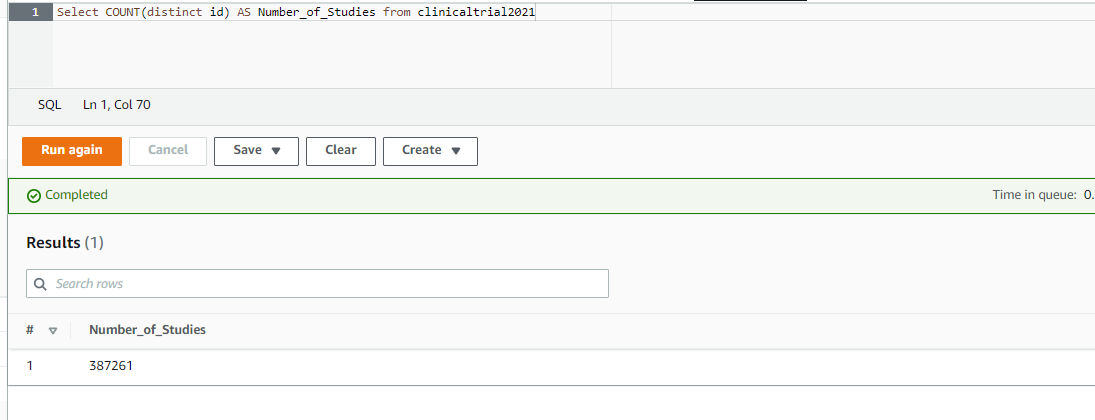
**Table Creations:**

****

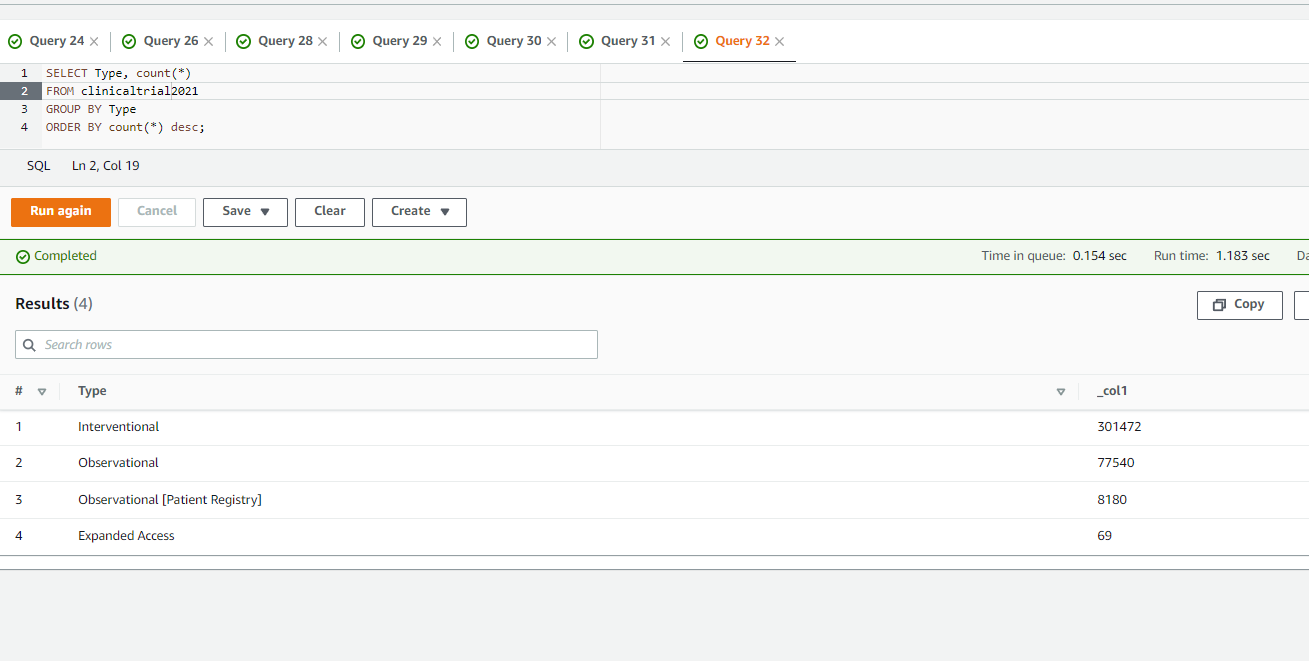
****

**Problems Implementations:**

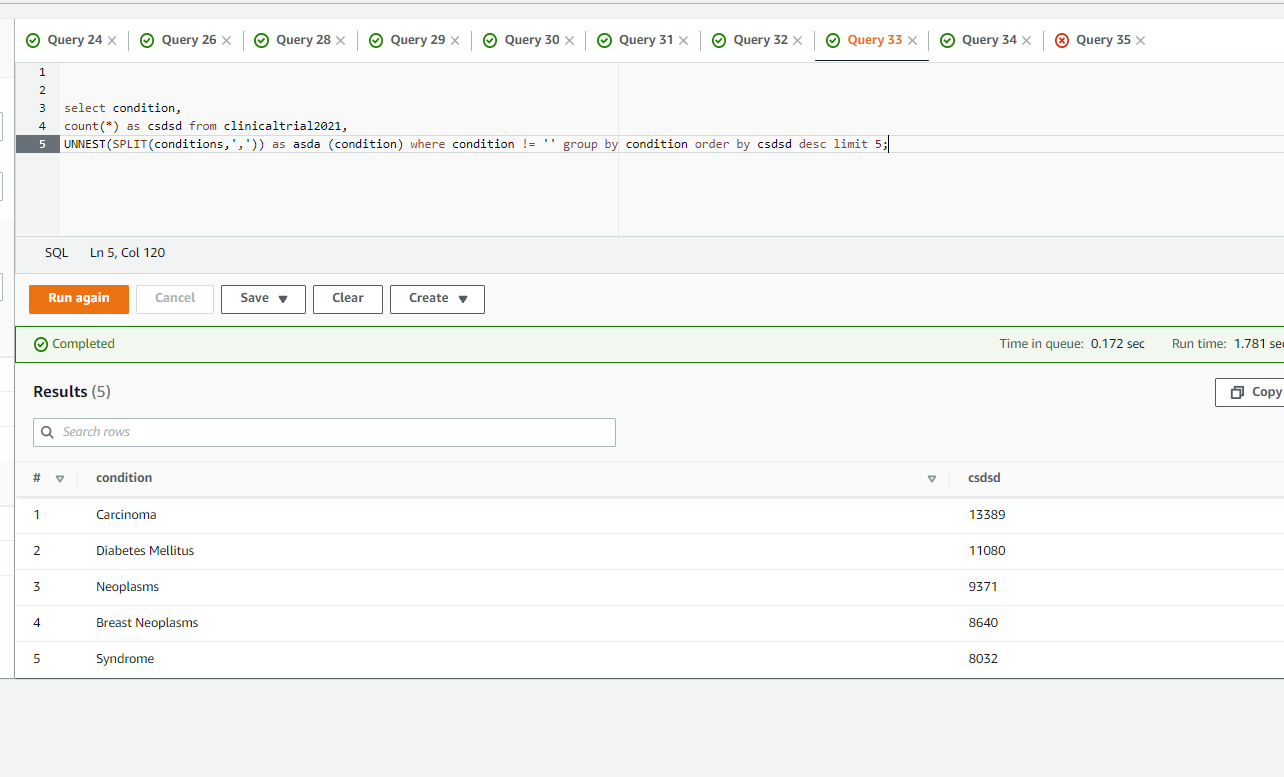
**Problem 1:**

****

**Problem 2:**

****

**Problem 3:**

****