Cognizant Technology Solutions



A Project Report on

“WEB SERVER LOG ANALYSIS”

By

|  |  |
| --- | --- |
| Adarsh P | 2320771 |
| Amritha Narayanankutty | 2320932 |
| Digvijay Patil | 2320451 |
| Naguru Mounika | 2320670 |
| Naga Phani Narina | 2320859 |
| Prajwal C Mannangi | 2320570 |
| Pratik Patil | 2320814 |

**Contents**

1.Introduction --------------------------------------------------------------------- 03

a) Pyspark

2.Business Requirements -------------------------------------------------------- 04

3.Overview of the project ------------------------------------------------------- 05

4.Implementation -----------------------------------------------------------------06

5.Visualisation --------------------------------------------------------------------14

6.Conclusion ----------------------------------------------------------------------17

**INTRODUCTION**

Server log analysis is an ideal use case for Spark. It's a very large, common data source and contains a rich set of information. Spark allows you to store your logs in files on disk cheaply, while still providing a quick and simple way to perform data analysis on them. This homework will show you how to use Apache Spark on real-world text-based production logs and fully harness the power of that data.

Log data comes from many sources, such as web, file, and compute servers, application logs, user-generated content, and can be used for monitoring servers, improving business and customer intelligence, building recommendation systems, fraud detection, and much more.

**Pyspark:**

PySpark is a Python API to support Python with Apache Spark. PySpark provides **Py4j library,** with the help of this library, Python can be easily integrated with Apache Spark. PySpark plays an essential role when it needs to work with a vast dataset or analyse them. This feature of PySpark makes it a very demanding tool among data engineers.



A large amount of data is generated offline and online. It is necessary to extract valuable information from the raw data. We require a more efficient tool to perform different types of operations on the big data. There are various tools to perform the multiple tasks on the huge dataset, but these tools are not so appealing anymore. It is needed some scalable and flexible tools to crack big data and gain benefit from it.

**BUSINESS REQUIREMENTS**

**Description:**    
Analyse and process web server logs to extract insights about website traffic, user behaviour, and performance.

**REQUIREMENTS:**

1. **Data Ingestion:** Ingest web server logs from various sources or file systems.
2. **Data Transformation:** Parse and process log data to extract relevant information like IP addresses, URLs, response codes, etc.
3. **Analysis:** Perform analysis to identify popular pages, detect errors or anomalies, and measure website performance metrics.
4. **Visualization:** Visualize the analysis results using charts, graphs, or dashboards.

**IMPLEMENTATION**

**Objective:** The primary objective of web server log analysis is to extract meaningful insights from server logs. These insights help improve website performance, enhance security, and optimize user experience. Here are the key goals:

* **Understand Website Traffic:** Analyse log data to identify patterns in user visits, peak traffic times, and popular pages.
* **Monitor User Behaviour:** Detect anomalies, track user sessions, and observe navigation paths.
* **Identify Errors:** Spot issues like broken links (404 errors), server errors (500 errors), or unauthorized access attempts.
* **Measure Performance Metrics:** Evaluate response times, page load speeds, and server resource utilization.
* **REQUIREMENT 1:**

**DATA INGESTION**

Data ingestion is the process of importing and loading data into a system. It’s a critical step in any data-centric workflow, ensuring that the correct information is available at the right time for analysis and decision-making.

To ingest a dataset into Databricks, you can follow these steps:

1. Create a Compute engine:

* Start by creating a Databricks cluster.
* This cluster will provide the compute resources needed to run your commands.

2. Enable DBFS in Databricks:

* Use Databricks features to explore your raw dataset.

3. Ingest the Raw Data:

* Load the raw data into a table to make it available for further processing.

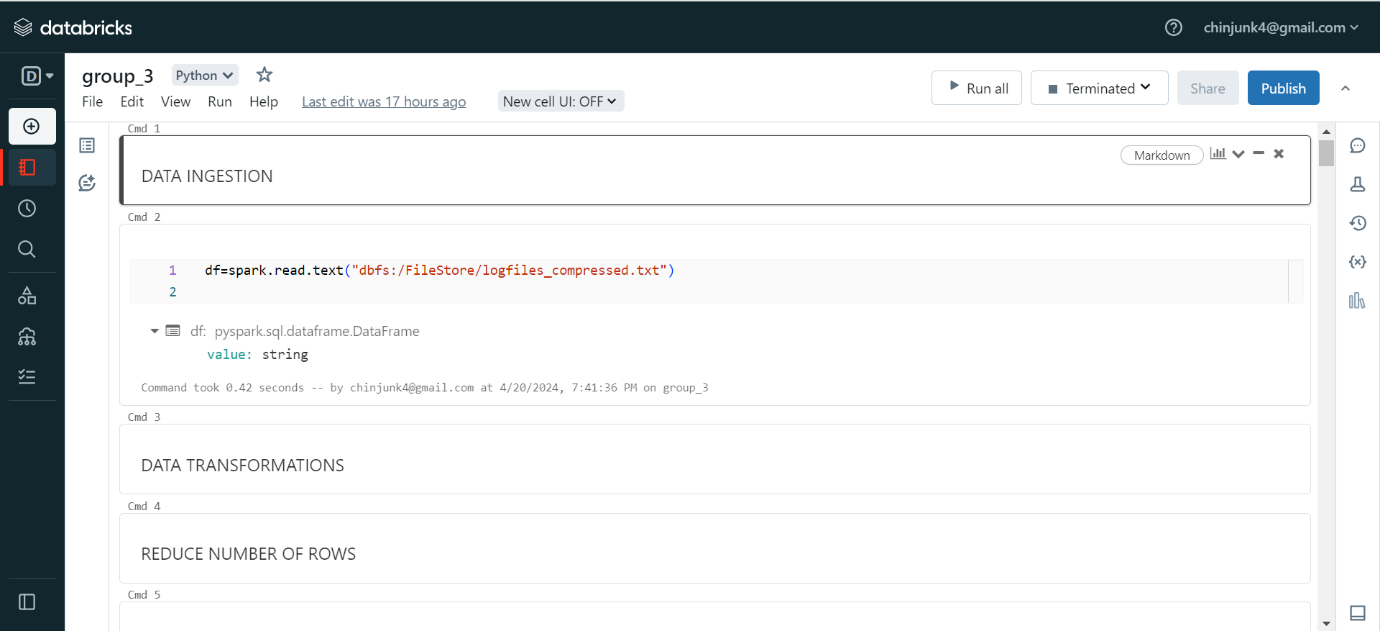


Fig1: Reading data from text file to dataframe.

**Steps:**

1. Open Databricks community and login to your account.
2. Create a compute engine for analysis.
3. Create new notebook in workspace and write the above syntax.

* **REQUIREMENT 2:**

**DATA TRANSFORMATION**

1. Prepare the Raw Data:

* Transform the raw source data as needed.
* This step ensures data quality, deduplicates records, and prepares it for analysis.

1. Query the Transformed Data:

* Create a Databricks notebook to query the transformed data.

A screenshot of a computer

Description automatically generated

Fig2: Adding an index column to dataframe.

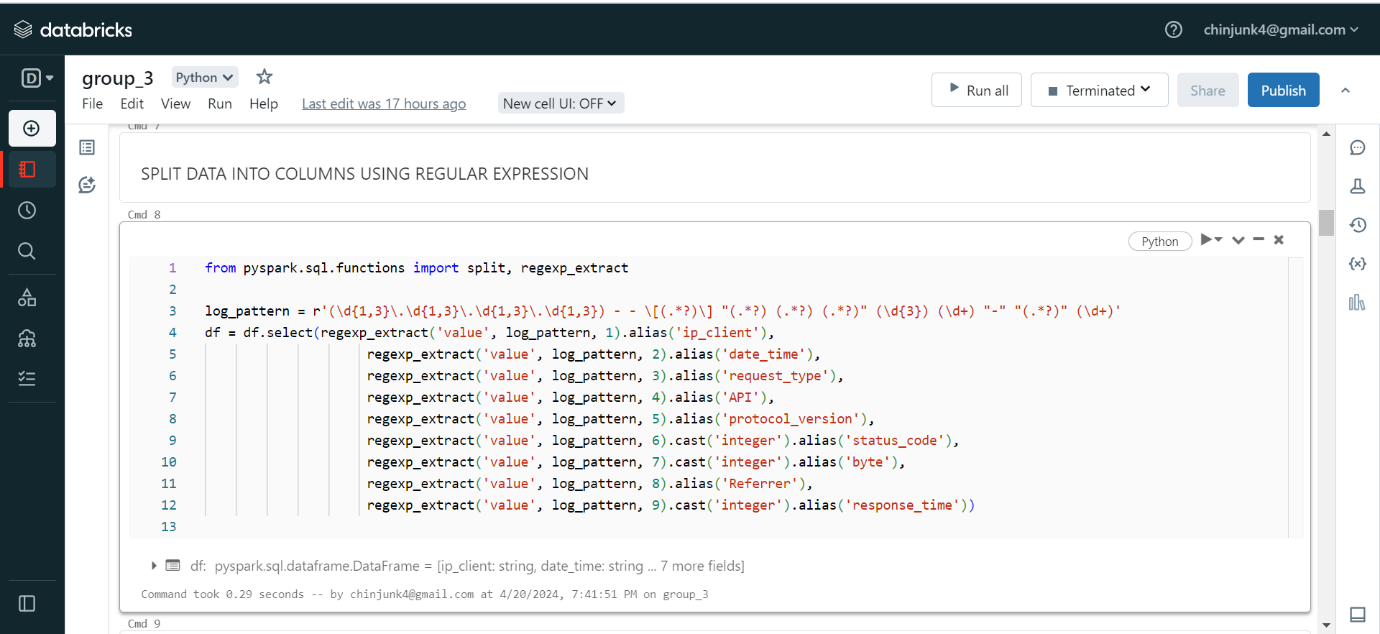


Fig3: Extraction of data in required format from dataframe.

A screenshot of a computer

Description automatically generatedFig4: dataframe table result

In pyspark, using show () method we can retrieve data from dataframe .by default show ()

will give first 20 values from dataframe.

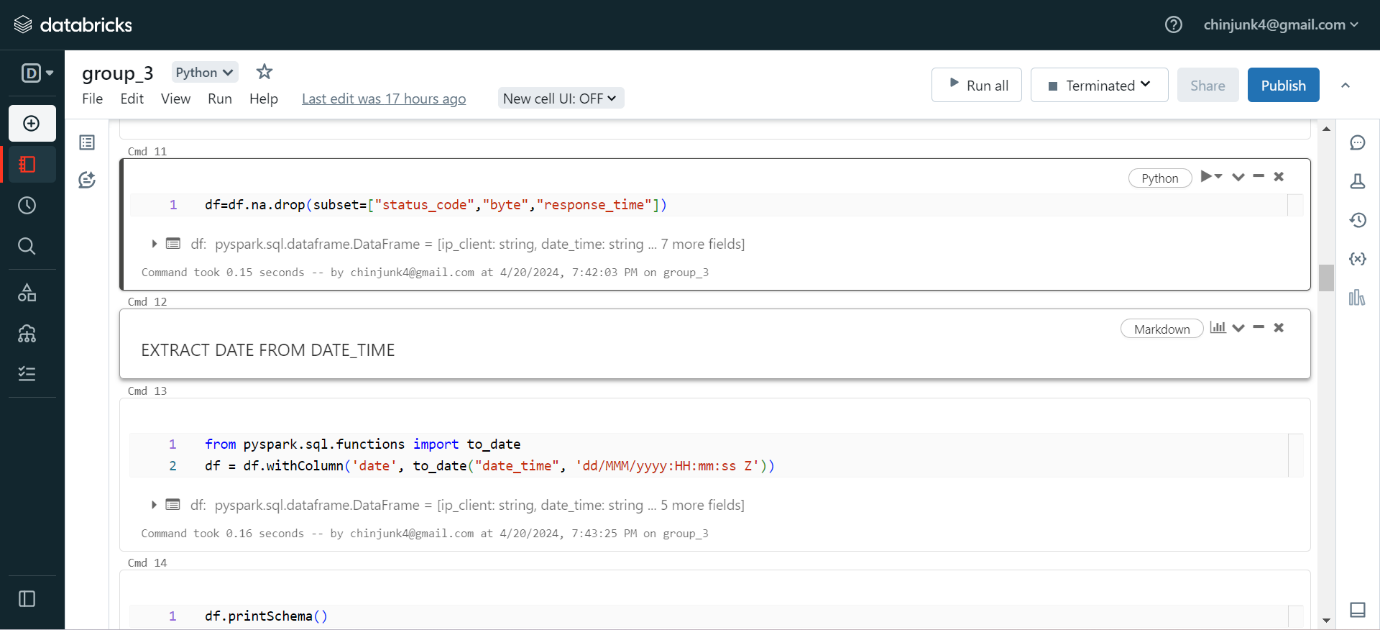


Fig5: Transformation on dataframe.

1.Using drop, removing the null values in table related columns like status code, byte, response time.

2. using withColumn method extract date from timestamp.

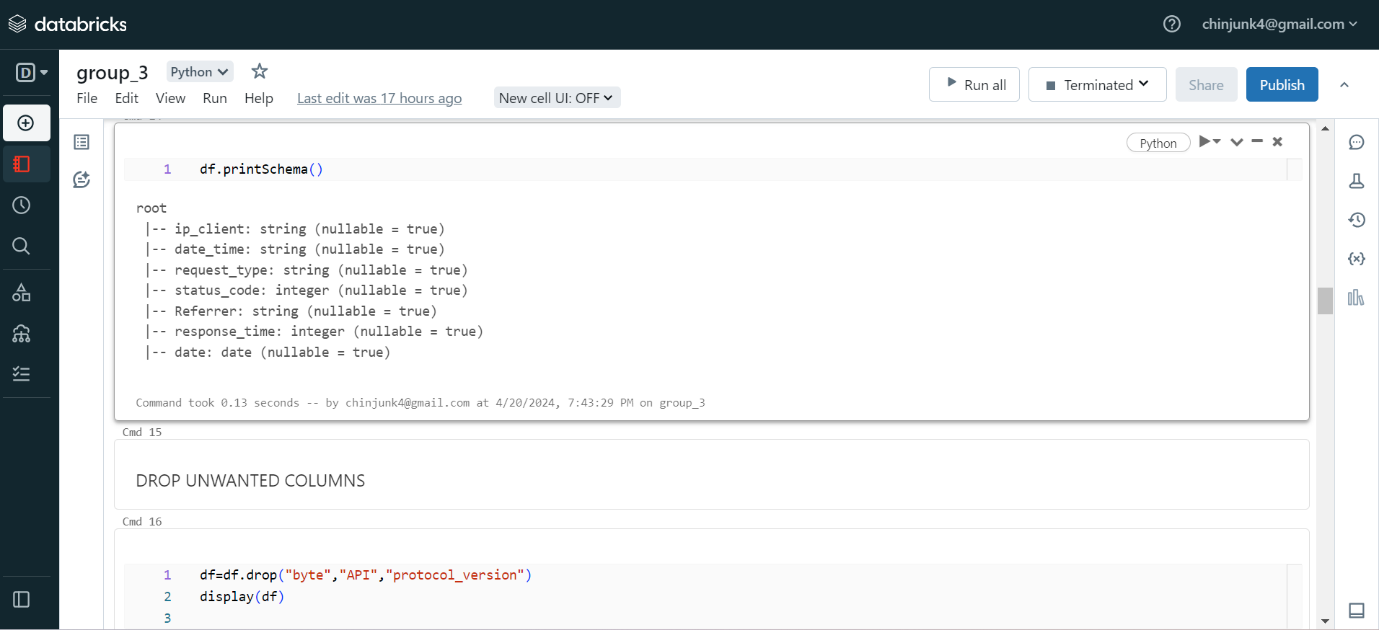


Fig6: description of dataframe.

The description of dataframe, column names and datatypes of respective columns will be shown using printSchema () method.

A screenshot of a computer

Description automatically generated

Fig7: Table data after dropping Null values.

Display () method is same as Show method to display records of dataframe.

* **REQUIREMENT 3:**

**DATA ANALYSIS**

**1.Popular Pages based on referrer:**

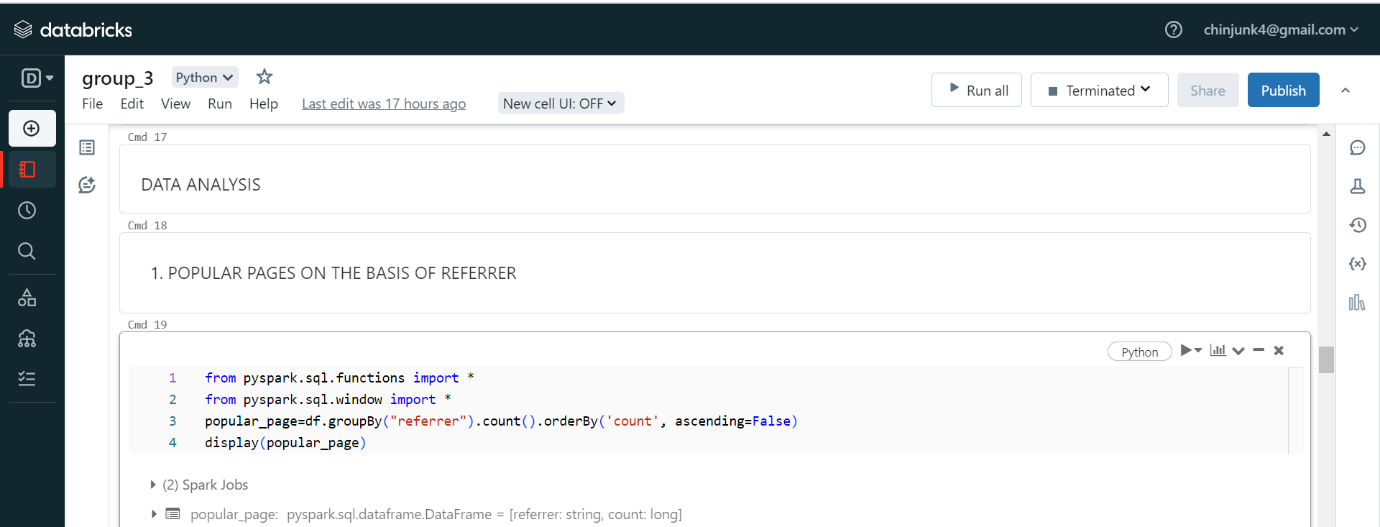


Fig7: Calculating popular pages from above dataframe.

For this, first the count of views is calculated for that referrer using groupby referrer and count function.

From above calculation we got know about visitors count and by count we found the pages popularity.

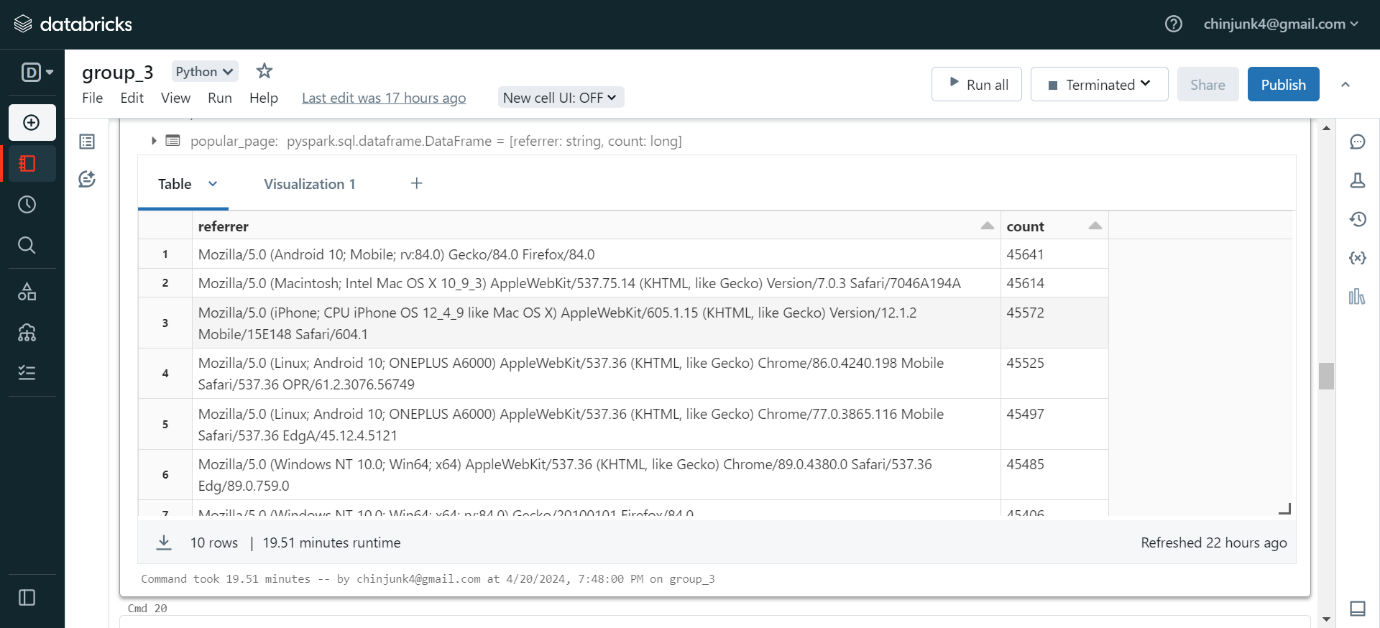


Fig8: Result of query calculating popular pages.

**2. Analysis:**

**a) Number of Unique Hosts:**

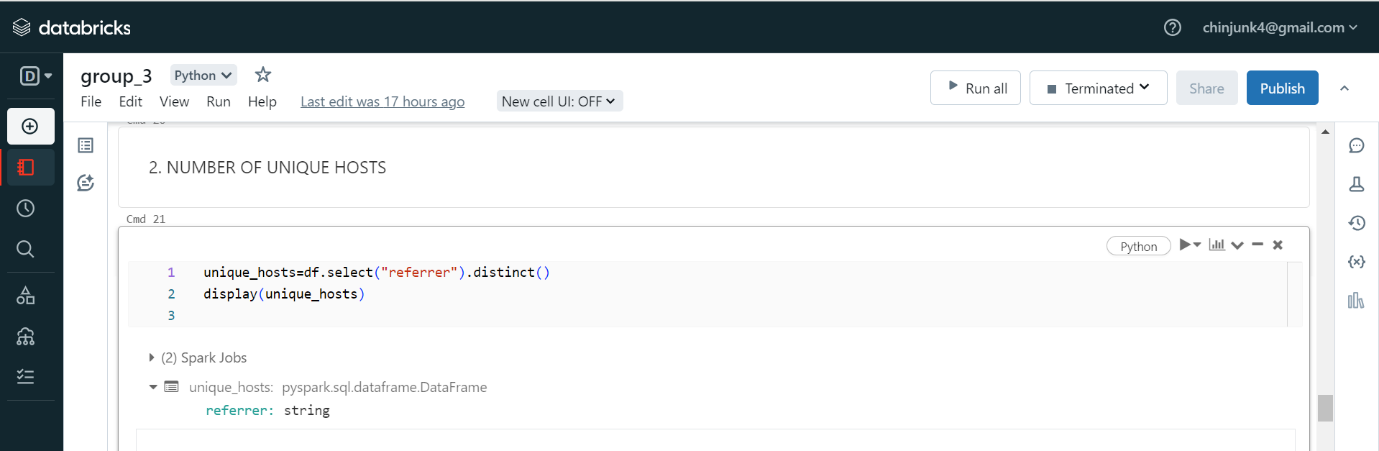


Fig9: Calculating Unique hosts for each page.

Distinct (), using the method distinct () unique hosts calculation is done for each page or referrer.

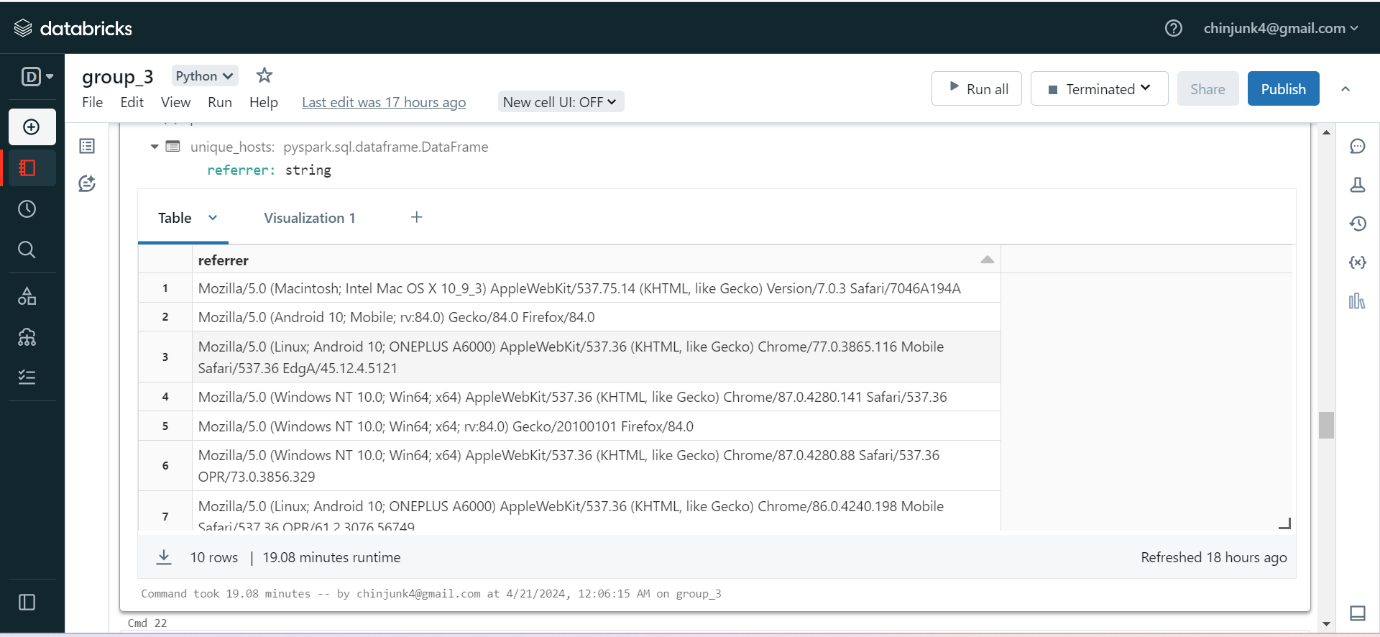


Fig10: Result of unique host for each referrer.

**b) Unique hosts on daily basis:**

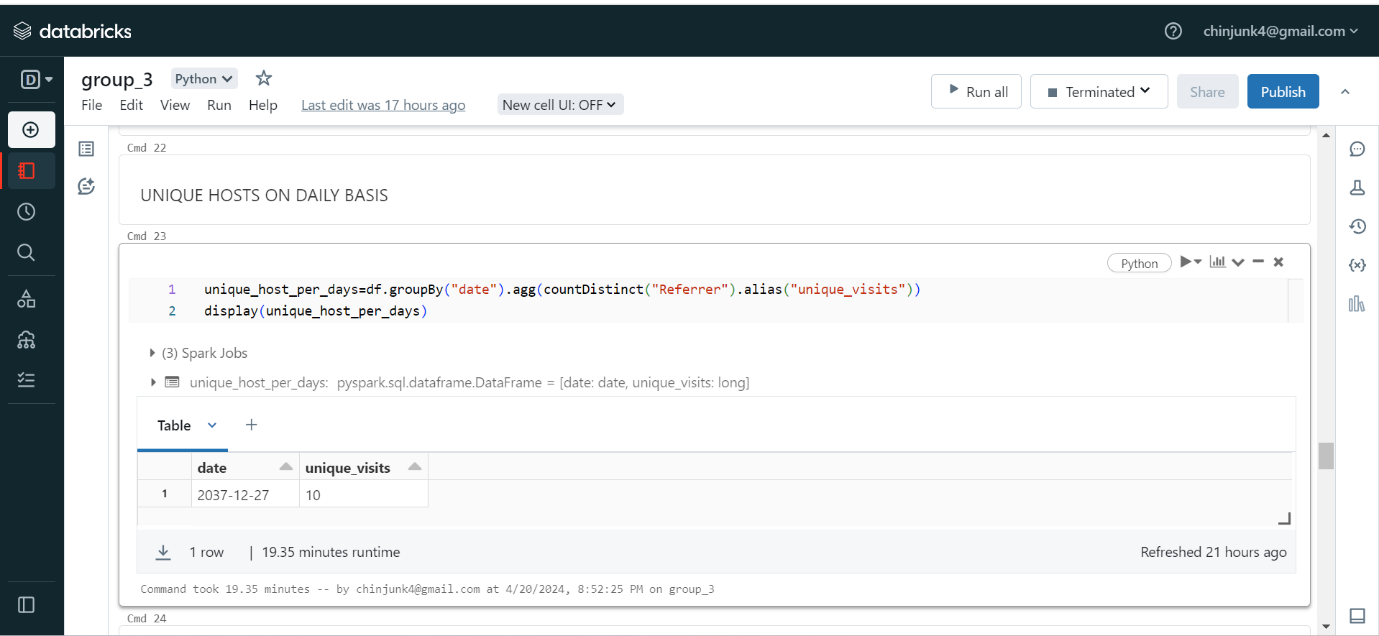


Fig11:Unique hosts on daily basis.

1. **Average response time for referrer:**

A screenshot of a computer

Description automatically generated

Fig12:Response Time.

Here in data, we have response time, time taken to load a page for referrer is response time using that column calculation is done.

From this we calculated the response time, and which is giving fast response from the all referrer.

A screenshot of a computer

Description automatically generated

Fig13:Average Response time for each Referrer.

1. **Maximum response time for each referrer:**

A screenshot of a computer

Description automatically generated

Fig14:Maximum time for each Referrer.

Here in this, the page with maximum response time, which is referrer is taking maximum response by taking all rows into consideration we calculated the maximum value for each referrer.

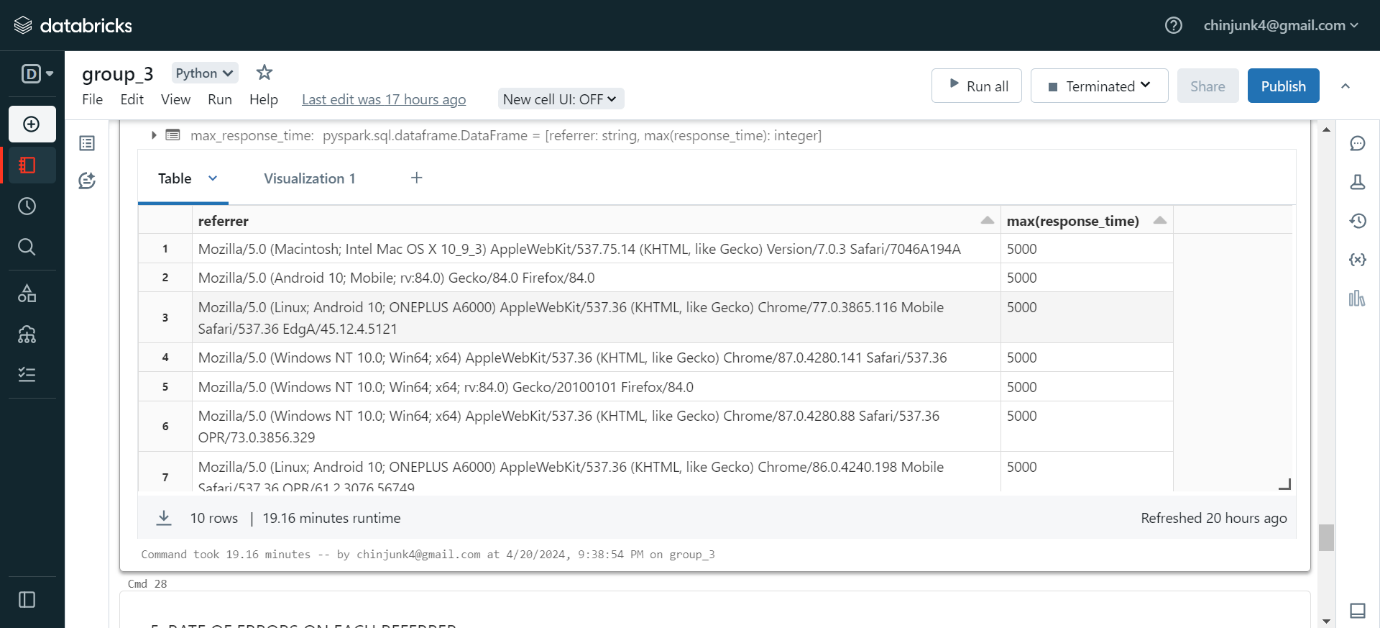


Fig15:Result of maximum response time.

1. **Rate of Errors on each referrer:**

A screenshot of a computer

Description automatically generated

Fig16:Error code Calculation.

A screenshot of a computer

Description automatically generated

Fig17:Result for error code count.

* **REQUIREMENT 4:**

**VISUALIZATION:**

**Visualization for all the data analysis is shown below.**

1. For Popular pages

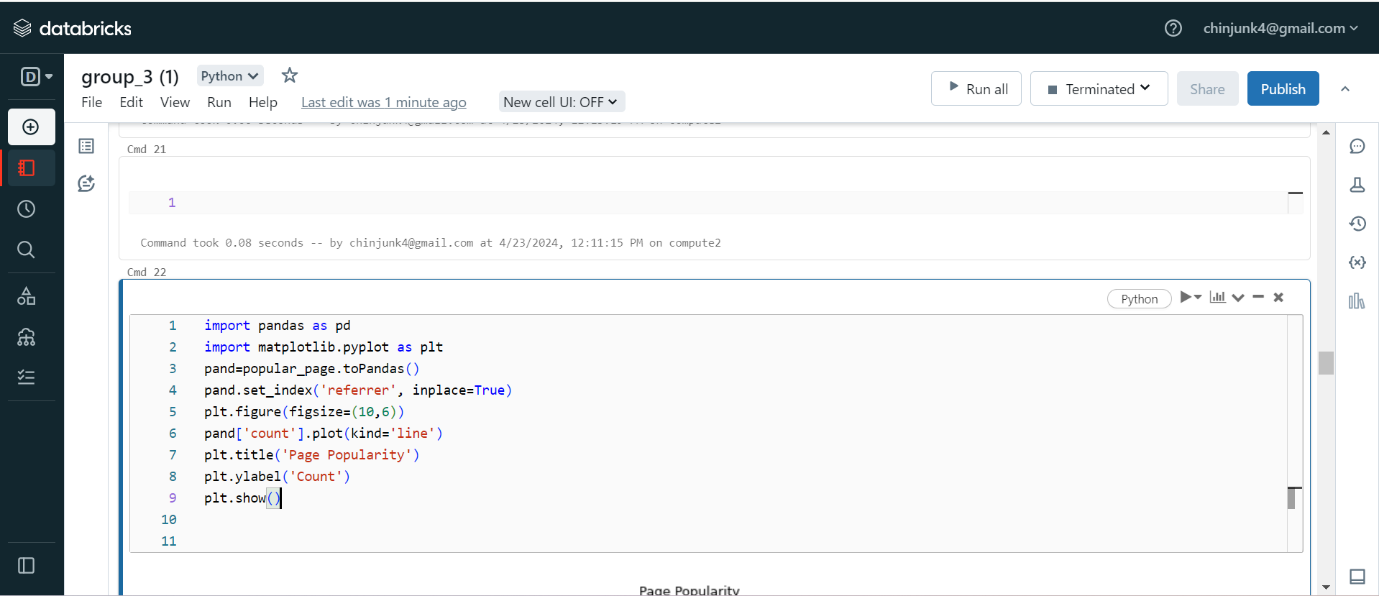


Fig18: Plot using pandas for popular page.

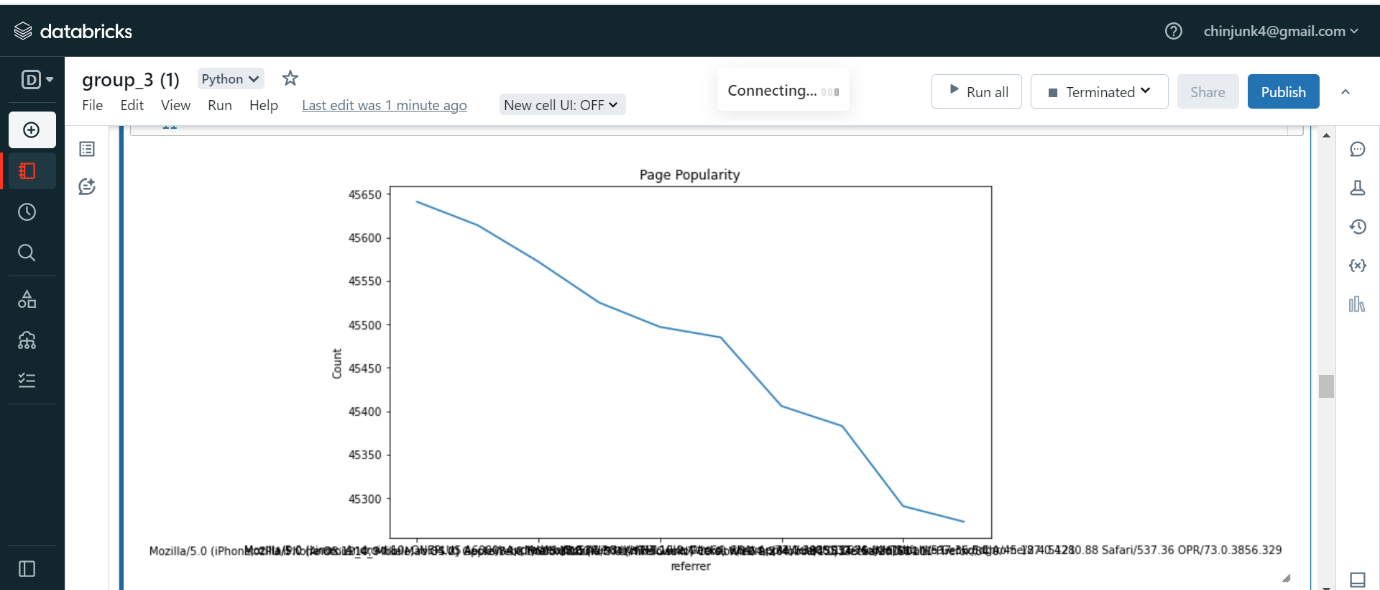


Fig19: Visualization for popular pages.

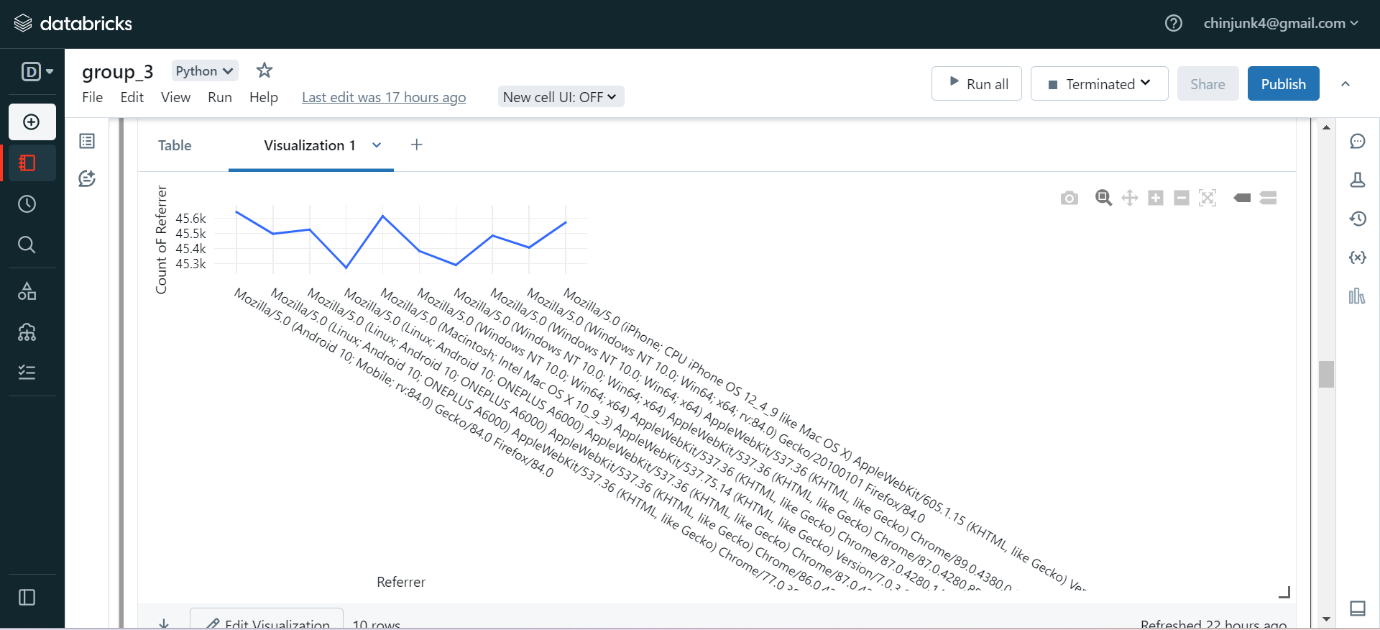


Fig20: Visualization for popular pages.

2.Visualization for unique hosts using word cloud.

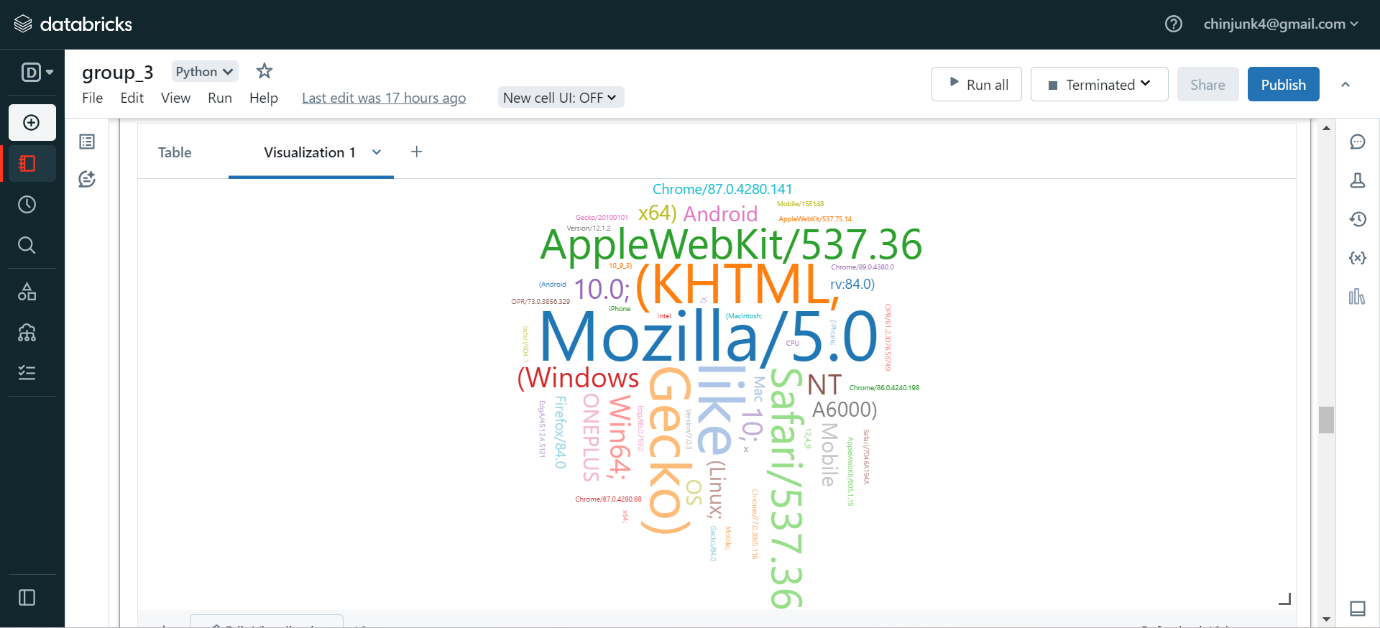


Fig21: Visualization for unique hosts.

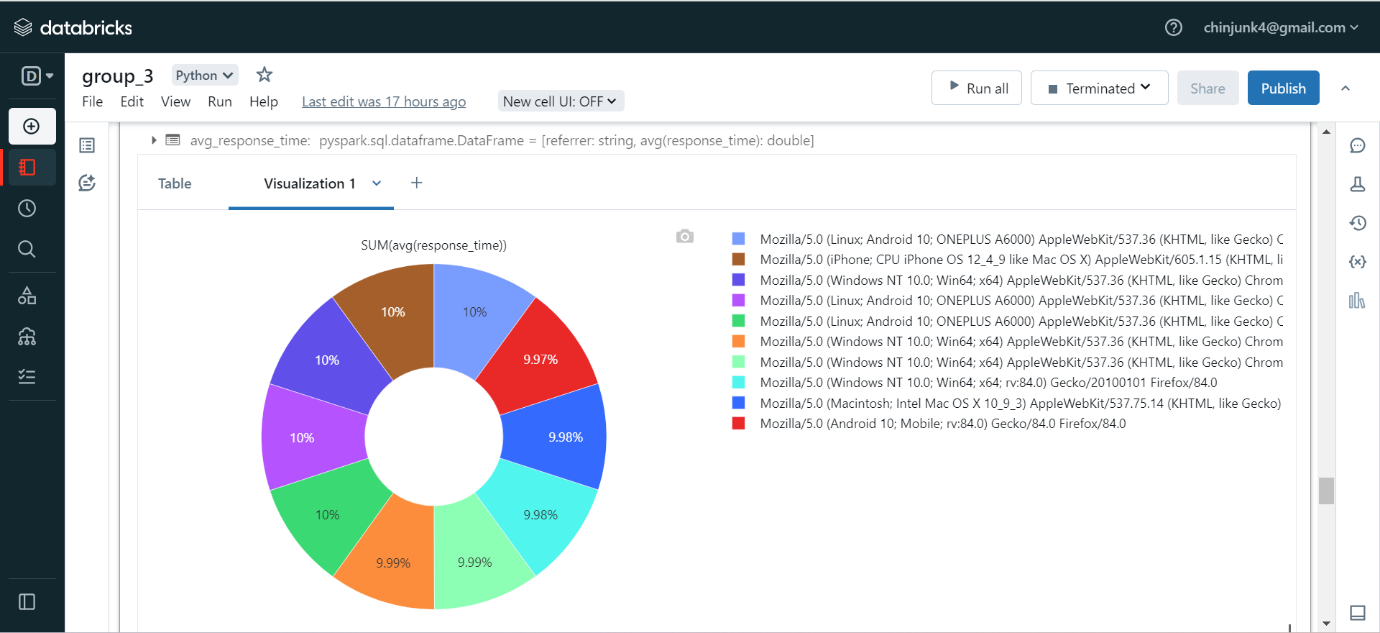
3.Visualization for average response time

Fig22: Visualization for Average Response time.

4. Visualization for Maximum response time.

A screenshot of a computer

Description automatically generated

Fig23: Visualization for Maximum response time.

A screenshot of a computer

Description automatically generated5.Visualization for count of errors for each referrer using area graph.

Fig24: Visualization for count of errors.

**CONCLUSION:**

Web server log analysis is crucial for understanding system behavior, identifying issues, and enhancing security. Here we found out about the popular page, count of unique visits, average response time, maximum response time and rate of errors. This analysis gave us a valuable insights about user interaction, performance of website and threats.