

REPORT ON DATA TRANSPARENCY IN MOTOR VEHICLE CRASH REPORTING

**Prepared for
Dr. Vishnu Pendyala
San Jose State University**

Prepared by

SINDHU NAGESHA	017419987
SYED FARAAZ AHMED LNU	017428619
ARJUN RAI	016980899
PRAYAG NIKUL PURAN	017416737

November 25, 2023

TABLE OF CONTENTS

	Page
ABSTRACT	03
MOTIVATION	04
INTRODUCTION	05
LITERATURE SURVEY	06
INNOVATION AND IMPACT OF THE PROJECT	08
SIGNIFICANT TO REAL WORLD	08
PROJECT WORKFLOW	09
ANALYTICS AND DATA DRIVEN DECISIONS	29
TECHNICAL DIFFICULTY	43
KEY LEARNINGS	45
CREDIT TAXONOMY	46
CONCLUSION	46
REFERENCES	47
LINKS TO FILES	47
APPENDIX	47

Abstract:- *The objective of the project is to source and extract the insights from the motor collision dataset which tell us about the date,time, location and casualty involved in the accident, and also the involvement of the types of vehicles and the number of people injured or died in the given particular collision. This project delves into a comprehensive analysis of motor vehicle collision data to gain insights into their causes, patterns, and consequences of the collision. The study will help to know the risk analysis, road safety rules and the safety of pedestrians. We will be using SQL queries, Python, warehouse to feed the data after that we will be using a data visualization tool like Power BI. We will summarize the use of your data to improve traffic rules, road safety and penalty enforcement in order to control the number of accidents.*

Keyword:- *Data Analytics, DBMS, RDBMS, Python, Data Visualization, SQL, MongoDB, AWS S3, AWS Glue, AWS Athena, Postgre, ETL .*

I. MOTIVATION

The motivation of the study is to generate awareness in the community by knowing the stats and the outcome. It will be easy to bring a spotlight on the public safety, urban planning, policy development and the relation between the accident and the peak hours. This also will motive to focus more on technological developments, resource allocation and it will also provide more opportunities for the researchers and students who are willing to contribute in this field. Apart from this talking about the business logic many insurance companies can be benefited from the collision risk factors and the trends affecting it. This information can inform underwriting practices and premium calculation.

II. INTRODUCTION

Motor vehicle crash reporting is a crucial aspect of traffic safety management and law enforcement. When a motor vehicle crash occurs, it is essential to document and analyze various details surrounding the incident to understand the causes, identify contributing factors, and develop strategies to prevent future accidents. Crash reports serve as a valuable source of information for government agencies, insurance companies, researchers, and other stakeholders involved in road safety. Emphasize the significance of crash reporting in enhancing road safety. Highlight how the information collected can be used to identify trends, evaluate the effectiveness of safety measures, and implement targeted interventions. Determine the important parties—law enforcement, insurance, transportation, and emergency services—that are involved in reporting crashes. Emphasize how these organizations' cooperation advances a more thorough comprehension of road safety-related concerns. Describe in brief the different kinds of data that are gathered for a crash report. This might contain information on the cars involved, the people impacted, the incident's location and timing, the state of the weather, the features of the road, and any contributing variables. Bring up the usage of standardized reporting forms as a means of promoting uniform data collecting. Usually, these forms have designated sections for pertinent data, making sure that important specifics are not missed. In this report, we delve into a comprehensive exploration of motor vehicle crash data, leveraging a robust toolkit that aligns with industry standards and academic teachings. Our approach intertwines key software requirements: AWS as our data warehouse, Python and SQL Workbench for meticulous data cleaning, and Power BI for vivid visualizations.

By harnessing MySQL Workbench, we implement data management strategies and employ normalization techniques, enriching data quality and structure—a fundamental aspect of our curriculum. Embracing MongoDB ensures flexible and scalable data storage, aligning seamlessly with our course's content objectives. Moreover, AWS takes center stage, facilitating API delivery and dynamic resource adjustments through autoscaling, showcasing practical applications from our syllabus in real-world contexts. This project becomes a journey navigating various data management tools and methodologies. Our engagement with a data warehouse tool enhances our ability to select optimal data organization methods for diverse tasks. Emphasize that raising road safety is the ultimate purpose of collision reporting. Talk about how the creation of efficient safety measures, upgrades to the infrastructure, and educational programs may all result from the examination of crash data and help to prevent such events in the future. Respond to worries about the confidentiality and privacy of the data gathered. Describe the statistical analysis of accident reports that takes into account the protection of people's personal data.

Finally, the introduction to reporting motor vehicle crashes should emphasize the significance of gathering precise and thorough data to improve road safety, adhere to legal requirements, and include a range of stakeholders in cooperative efforts to prevent and lessen the effects of motor vehicle crashes.

III. LITERATURE SURVEY

- **Exploratory analysis of automated vehicle crashes in California: A text analytics & hierarchical Bayesian heterogeneity-based approach**

The literature review of this study delves into the evolving landscape of research on automated vehicle (AV) crashes, with a focus on understanding contributing factors and outcomes. One foundational premise is the potential of AVs to reduce crash frequency by eliminating human error, a primary contributor to conventional crashes. To provide context, the study draws on safety studies emphasizing the predominant role of human error in the majority of crashes.

The review highlights the innovative approach taken by the current study in creating a unique database from manufacturer-reported Traffic Collision Reports and integrating detailed data on roadway and built-environment attributes. This approach allows for a more nuanced examination of AV crashes beyond traditional analyses.

Notably, the study employs a novel text analysis method to extract valuable information from crash report narratives, introducing a qualitative dimension to the quantitative examination of crash data. The emphasis on understanding AV crashes in real-world settings is underscored by the geocoding of crashes and linking them with detailed contextual information.

The literature review sets the stage by pointing out that AV crash studies are relatively nascent, with a limited body of knowledge compared to conventional crash studies. The study acknowledges the small sample size and challenges posed by the novelty of AV technology, driving the need for innovative methodologies and in-depth analyses.

Building on this foundation, the review emphasizes the study's contribution to the understanding of AV crash dynamics, specifically focusing on rear-end collisions, injury outcomes, and associations with key roadway and environmental variables. The Bayesian statistical models employed in the analysis underscore the methodological rigor applied to investigate the nuanced relationships between AV crashes and various contributing factors.

In summary, the literature review positions the study within the broader context of AV crash research, underlining the innovative methodologies employed and the significance of gaining insights into the complex interactions of AVs in urban environments. The review provides a comprehensive backdrop for the study's empirical analysis and contributes to the evolving discourse on AV safety.

- **A note on modeling vehicle accident frequencies with random-parameters count models**

The literature review charts the evolution of methodologies employed in the analysis of accident frequencies on roadway segments. Traditionally, count data models such as Poisson regression, negative binomial, and zero-inflated models have served as the bedrock for such analyses. However, the inherent assumption of a uniform accident rate across all segments has prompted researchers to explore alternative approaches. In response to this need, recent studies have introduced random-parameters count models as a progressive methodological alternative. These innovative models recognize the dynamic nature of accident rates, allowing

for variations across different road segments. This evolution not only challenges the limitations of traditional models but also provides a more nuanced understanding of the multifaceted factors influencing accident frequencies. The literature review critically examines the strengths and limitations of established models while emphasizing the transformative potential of random-parameters models in reshaping the landscape of accident frequency analysis, fostering a more comprehensive and adaptable framework for traffic safety research.

the basic Poisson model, the probability $P(n_i)$ is $P(n_i) = \frac{\exp(-\lambda_i) \lambda_i^{n_i}}{n_i!}$, where λ_i is the...

● ASSESSMENT OF SAFER ROAD USER BEHAVIOUR

In the realm of transportation, road users stand as the linchpin of a sustainable system. Recognizing their pivotal role, this research delves into the core of road safety, identifying road user behavior as the predominant factor influencing accidents. The study meticulously evaluates the efficacy of strategic road safety plans, pinpointing seven critical risk factors: drinking-drivers, seatbelts, child restraints, speed management, helmet usage, mobile phone use while driving, and driving under the influence. Aligned with international directives, the research accentuates the role of law legislation and enforcement as paramount in shaping road user conduct. The methodology employs the creation of an index for each risk factor, amalgamating them into a comprehensive road user assessment index. This index becomes a tool for tracking the implementation of safety measures at the national level and facilitates cross-country comparisons. Surprising findings emerge, revealing disparities in the incorporation of risk factors into road safety laws among nations. Moreover, the ranking based on the newly developed index challenges the conventional ranking based on road fatalities for a significant portion of countries. The study recommends an expanded focus on additional road safety factors and the adoption of successful strategies from countries that have achieved substantial declines in road fatalities. In essence, this research not only scrutinizes the global landscape of road user behavior but also advocates for a nuanced and comprehensive approach to enhancing road safety on a global scale.

IV. INNOVATION AND IMPACT OF THE PROJECT

Our project stands at the forefront of reshaping how we comprehend motor vehicle crash data. By weaving together advanced tools and methodologies in alignment with industry standards and real-world applications, we're revolutionizing the analysis and utilization of crash data. Through meticulous data management strategies and dynamic visualizations, we strive to metamorphose raw data into actionable intelligence, unraveling intricate insights into the causes and patterns of vehicle crashes. This innovative approach promises to empower stakeholders with a comprehensive understanding of road safety challenges, fostering a proactive approach to mitigate risks and bolster a safer road environment.

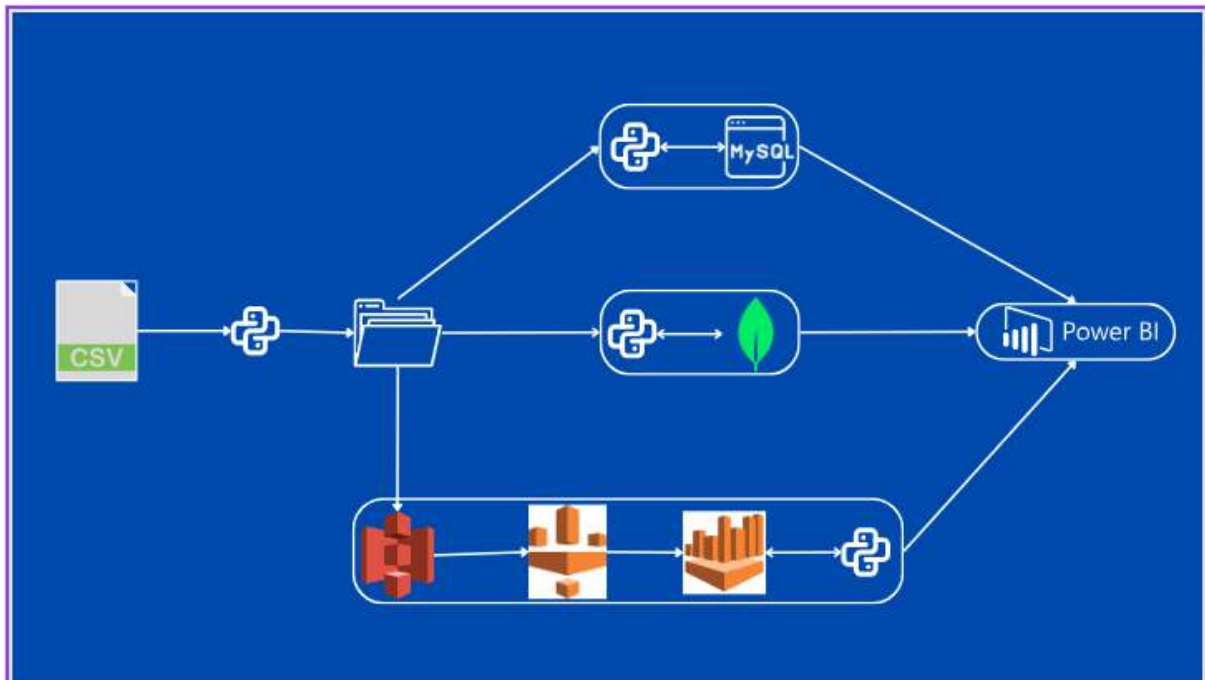
In addition to our prowess in evaluating and implementing new data systems, effective communication of insights through clear reports or presentations is a foundational skill. Our aspiration extends beyond exploration; we're driven to uncover the primary causes of vehicle crashes within traffic regulations and propose tangible solutions. We're committed to aiding individuals and families impacted by these incidents through impactful initiatives like Public Awareness Campaigns, bolstered Emergency Response Systems, and enriched Post-Crash Care and Rehabilitation programs. This innovative leap in data-driven road safety initiatives aims to not only contribute meaningfully to ongoing efforts but also pave the way for a tangible transformation in road safety for all road users.

V. SIGNIFICANT TO REAL WORLD

In the real world, motor vehicle collision reporting is very important for a number of reasons that affect public safety, policymaking, and numerous facets of society. Crash reporting is important in the real world in the following ways.

Data gathered from crash reports is used to find trends and patterns in the accidents that occur. Having this data is essential for putting targeted safety measures into place—like traffic laws, infrastructure upgrades, and public awareness campaigns—that will ultimately lower the number and severity of collisions. Law enforcement relies heavily on crash reports to investigate events and assign blame. For legal procedures such as insurance claims, court cases, and settlements, they offer a factual basis. Precise and comprehensive crash reports facilitate equitable and effective settlement of related legal disputes. Crash reports are used by insurance firms to evaluate claims precisely. These reports' contents aid in determining responsibility and calculating damages for impacted parties. Ensuring that accident victims receive just compensation in a timely manner is essential. Crash data is used by government organizations and local government bodies to efficiently distribute resources. Authorities may prioritize locations for safety improvements, undertake targeted traffic enforcement, and design infrastructure upgrades that address specific safety problems by analyzing where and why accidents occur. Crash reports are used by emergency services to evaluate response times, pinpoint locations with greater occurrence rates, and streamline operations. This data is essential for optimizing emergency response plans and effectively allocating resources.

VI. PROJECT WORKFLOW



Data Cleaning:

The main and the important step for the project is to clean the data and the data set which we have got is very dirty and the data in it sparse and the spatial data as we have nearly 10 lakhs of rows and in it the data also is not filled properly and we can't directly delete the rows as it will cause the meaningful data to lose too.

Importing Dataset

```
In [2]: motor = pd.read_csv('Motor.csv')
```

```
In [3]: motor.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1048575 entries, 0 to 1048574
Data columns (total 29 columns):
 #   Column                                     Non-Null Count  Dtype
---  -
 0   CRASH DATE                               1048575 non-null object
 1   CRASH TIME                               1048575 non-null object
 2   BOROUGH                                  671910 non-null  object
 3   ZIP CODE                                 671726 non-null  float64
 4   LATITUDE                                973773 non-null  float64
 5   LONGITUDE                               973773 non-null  float64
 6   LOCATION                                973773 non-null  object
 7   ON STREET NAME                           791561 non-null  object
 8   CROSS STREET NAME                       505581 non-null  object
 9   OFF STREET NAME                         255686 non-null  object
10   NUMBER OF PERSONS INJURED               1048558 non-null float64
11   NUMBER OF PERSONS KILLED               1048545 non-null float64
12   NUMBER OF PEDESTRIANS INJURED          1048575 non-null int64
13   NUMBER OF PEDESTRIANS KILLED          1048575 non-null int64
14   NUMBER OF CYCLIST INJURED              1048575 non-null int64
15   NUMBER OF CYCLIST KILLED              1048575 non-null int64
16   NUMBER OF MOTORIST INJURED            1048575 non-null int64
17   NUMBER OF MOTORIST KILLED            1048575 non-null int64
18   CONTRIBUTING FACTOR VEHICLE 1         1044829 non-null object
19   CONTRIBUTING FACTOR VEHICLE 2         870100 non-null  object
20   CONTRIBUTING FACTOR VEHICLE 3         77589 non-null   object
21   CONTRIBUTING FACTOR VEHICLE 4         18048 non-null   object
22   CONTRIBUTING FACTOR VEHICLE 5         5036 non-null    object
23   COLLISION_ID                          1048575 non-null int64
24   VEHICLE TYPE CODE 1                   1039916 non-null object
25   VEHICLE TYPE CODE 2                   799608 non-null  object
26   VEHICLE TYPE CODE 3                   73175 non-null   object
27   VEHICLE TYPE CODE 4                   17146 non-null   object
28   VEHICLE TYPE CODE 5                   4834 non-null    object
dtypes: float64(5), int64(7), object(17)
memory usage: 232.0+ MB
```

So, from the above information we can say that the data set contains 4 themes in one table so, we separated the tables into 4 table location, vehicle, contributing factor and casualty.

Cleaning of location df

```
In [13]: # So in Location as we have Latitde and Longitude so we don't need borough,zip code,street name,cross street name and off street
location = location.drop(columns=['BOROUGH', 'ZIP CODE', 'ON STREET NAME', 'CROSS STREET NAME', 'OFF STREET NAME', 'LOCATION'])
```

So, from the location table we will be removing zip code, street name, cross street name, off street name as these are strings and we can't perform any analytics on them so they are just the attributes not required for the analytics purpose.

```

18 CONTRIBUTING FACTOR VEHICLE 1 1044829 non-null object
19 CONTRIBUTING FACTOR VEHICLE 2 870100 non-null object
20 CONTRIBUTING FACTOR VEHICLE 3 77589 non-null object
21 CONTRIBUTING FACTOR VEHICLE 4 18048 non-null object
22 CONTRIBUTING FACTOR VEHICLE 5 5036 non-null object
23 COLLISION_ID 1048575 non-null int64
24 VEHICLE TYPE CODE 1 1039916 non-null object
25 VEHICLE TYPE CODE 2 799608 non-null object
26 VEHICLE TYPE CODE 3 73175 non-null object
27 VEHICLE TYPE CODE 4 17146 non-null object
28 VEHICLE TYPE CODE 5 4834 non-null object
dtypes: float64(5), int64(7), object(17)
memory usage: 232.0+ MB

```

So, from the above screenshot we can see that contributing factor 1 and vehicle type 1 has no missing values so they are good to go but after that we can see that the number of missing values keep on increasing as we go down and so, we can't remove those row that will lead us to huge loss in the data that the reason why we have added Unknown and Unspecified to the columns and zero if its the numerical column so, to maintain the quality and quantity of the data set.

SCENARIO 1: Using RDMS

Now we will connect the python with MySQL using the mysql connector.

```

In [1]: import mysql.connector
        from getpass import getpass

In [2]: username = input("Enter your MySQL username: ")
        password = getpass("Enter your MySQL password: ")

        Enter your MySQL username: root
        Enter your MySQL password: .....

In [3]: host = "localhost" # Change this to your MySQL server hostname or IP address
        database = "mysql" # Change this to the name of your data

```

```
In [4]: try:
        # Establish a connection to the MySQL server
        connection = mysql.connector.connect(
            host=host,
            user=username,
            password=password,
        )

        if connection.is_connected():
            print("Connected to the MySQL server")
    except mysql.connector.Error as e:
        print(f"Error: {e}")
```

Connected to the MySQL server

```
In [5]: mycursor = connection.cursor()
```

```
In [6]: mycursor.execute("use mysql")
```

```
In [17]: mycursor.execute("create table location(crash_date date,crash_time varchar(5),latitude varchar(20),longitude varchar(20),collisio
< [REDACTED] >
```

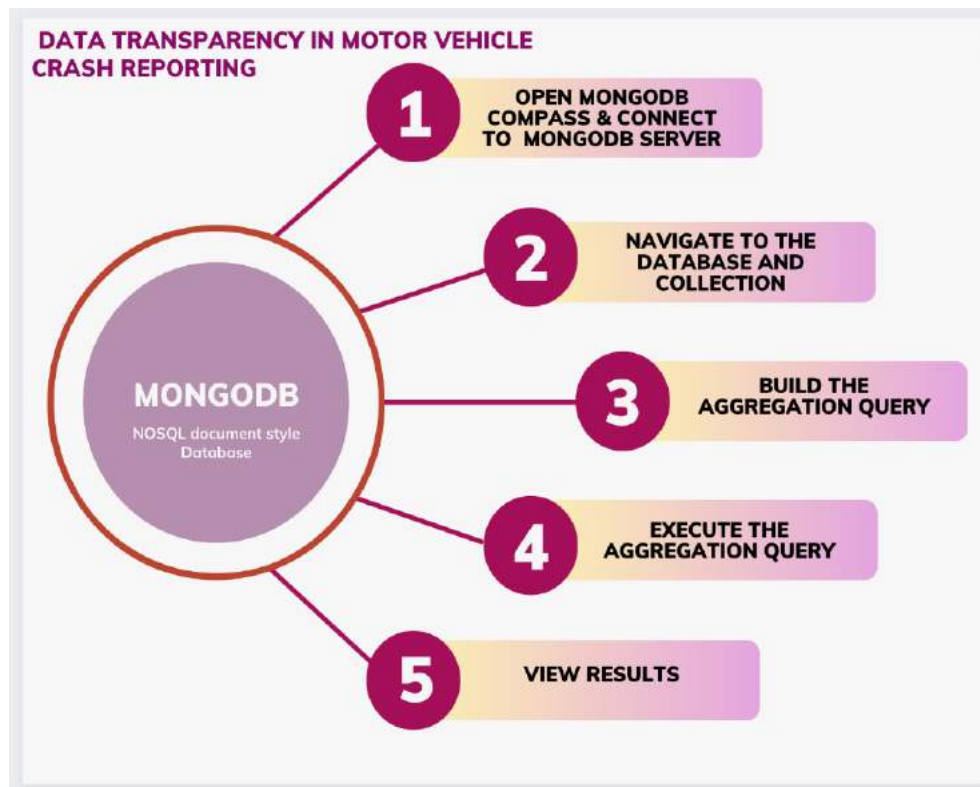
```
In [18]: mycursor.execute("create table vehicles(VEHICLE_TYPE_CODE_1 varchar(20), VEHICLE_TYPE_CODE_2 varchar(20), VEHICLE_TYPE_CODE_3 var
< [REDACTED] >
```

```
In [19]: mycursor.execute("create table casulty(NUMBER_OF_PERSONS_INJURED FLOAT(3), NUMBER_OF_PERSONS_KILLED FLOAT(3),NUMBER_OF_PEDESTRIAN
< [REDACTED] >
```

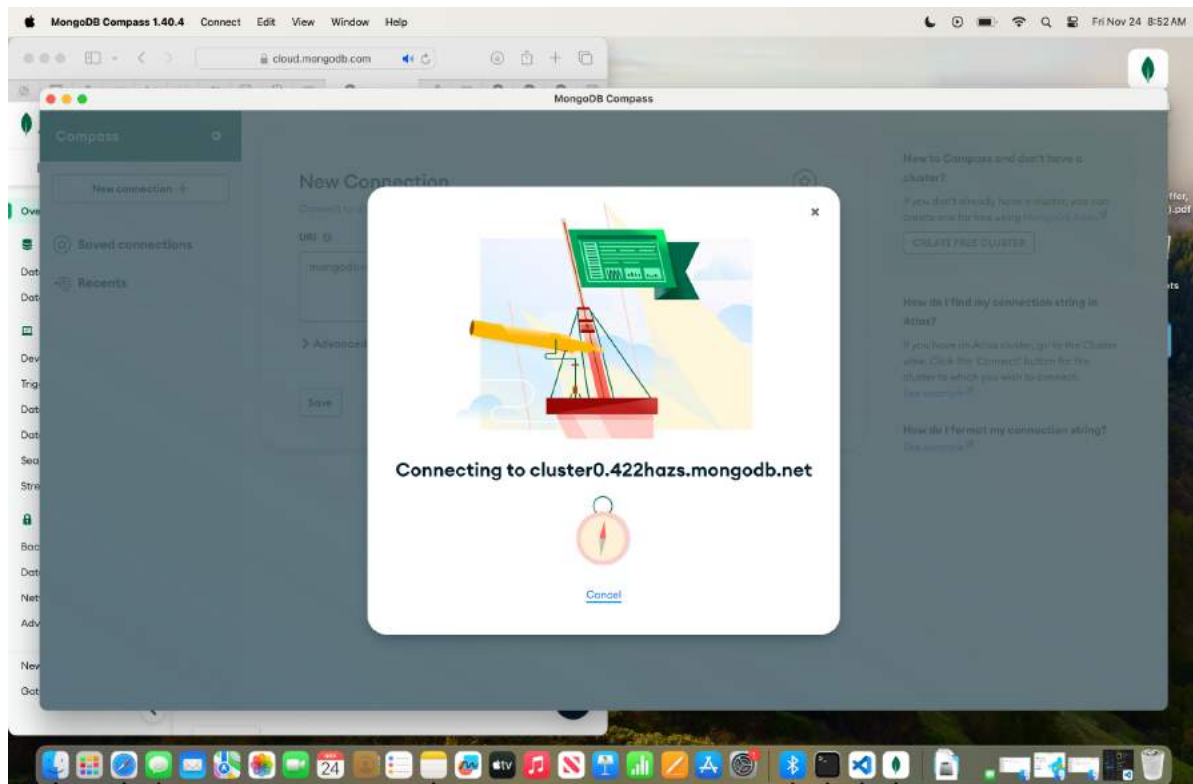
```
In [21]: mycursor.execute("create table contributing(CONTRIBUTING_FACTOR_VEHICLE_1 varchar(50), CONTRIBUTING_FACTOR_VEHICLE_2 varchar(50),
< [REDACTED] >
```

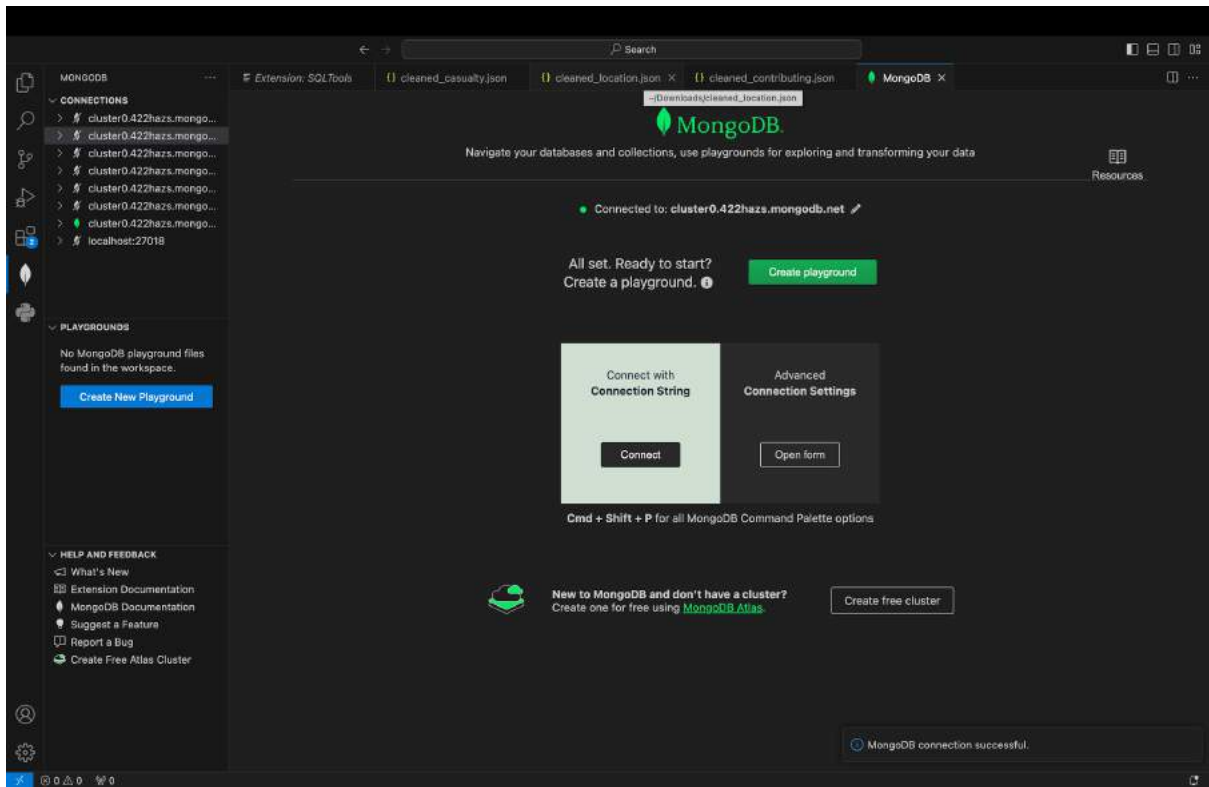
SCENARIO 2: Using MongoDB

Workflow

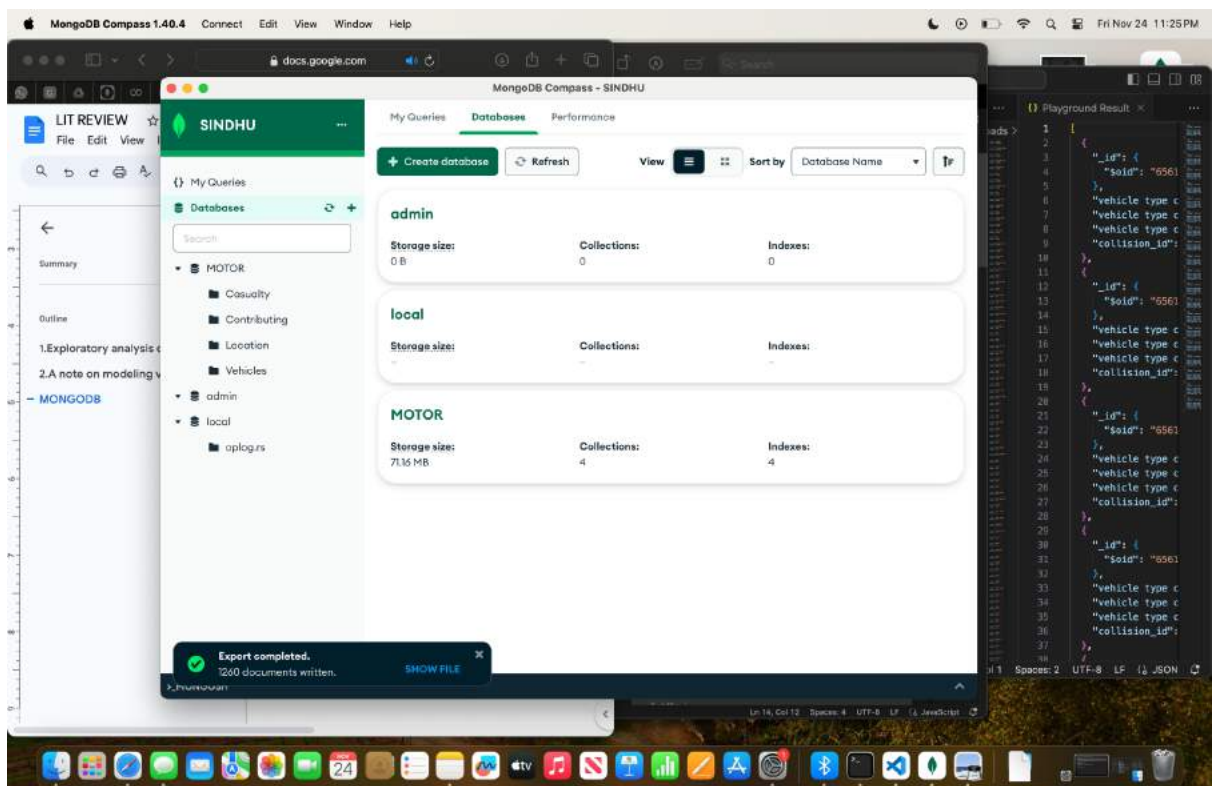


1. Creating Cluster and Connecting to MONGODB server

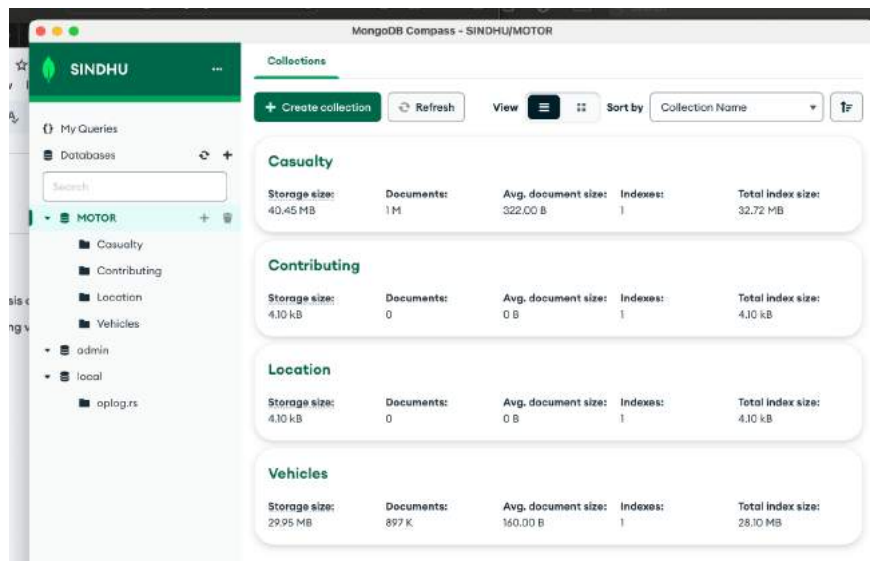




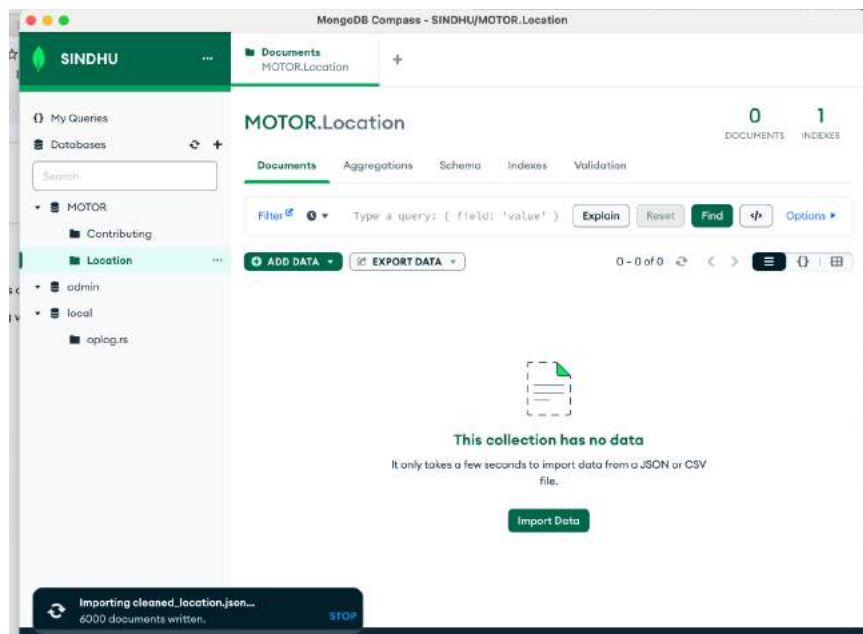
2. Database



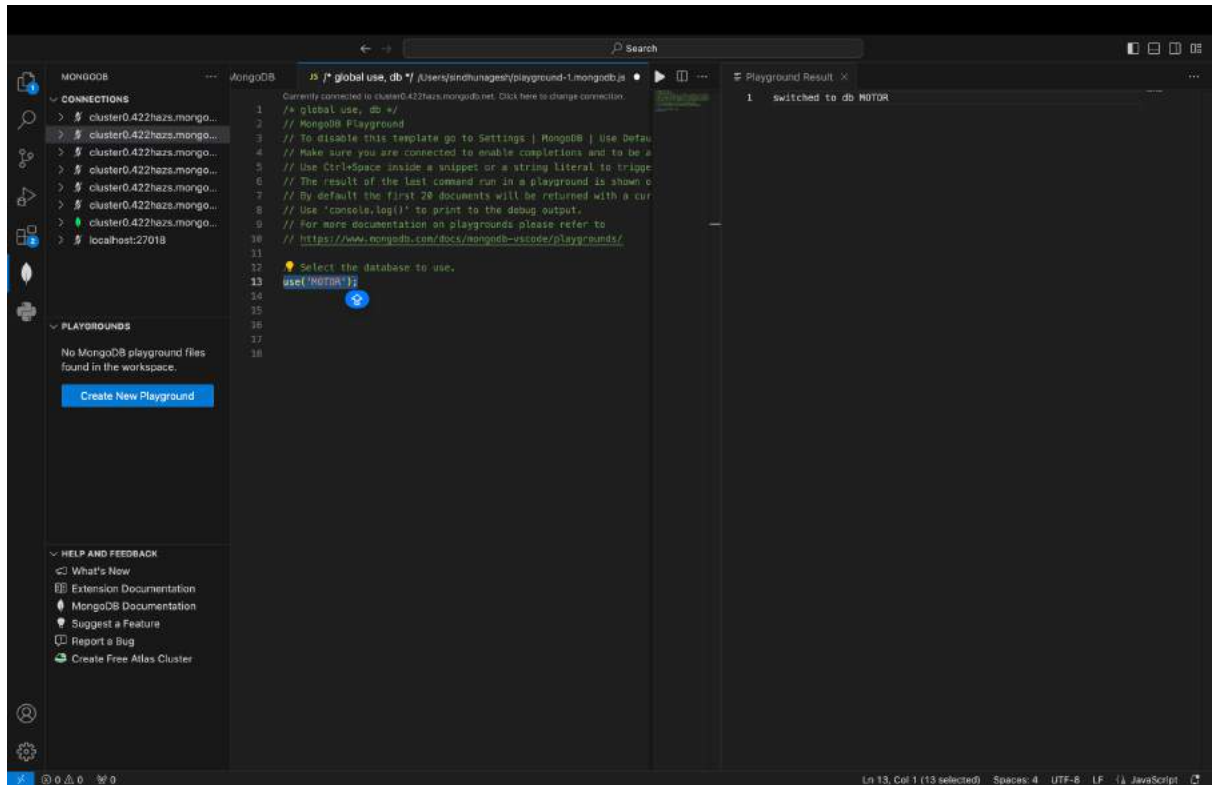
3. Collections



4. Importing Data into Location Collection

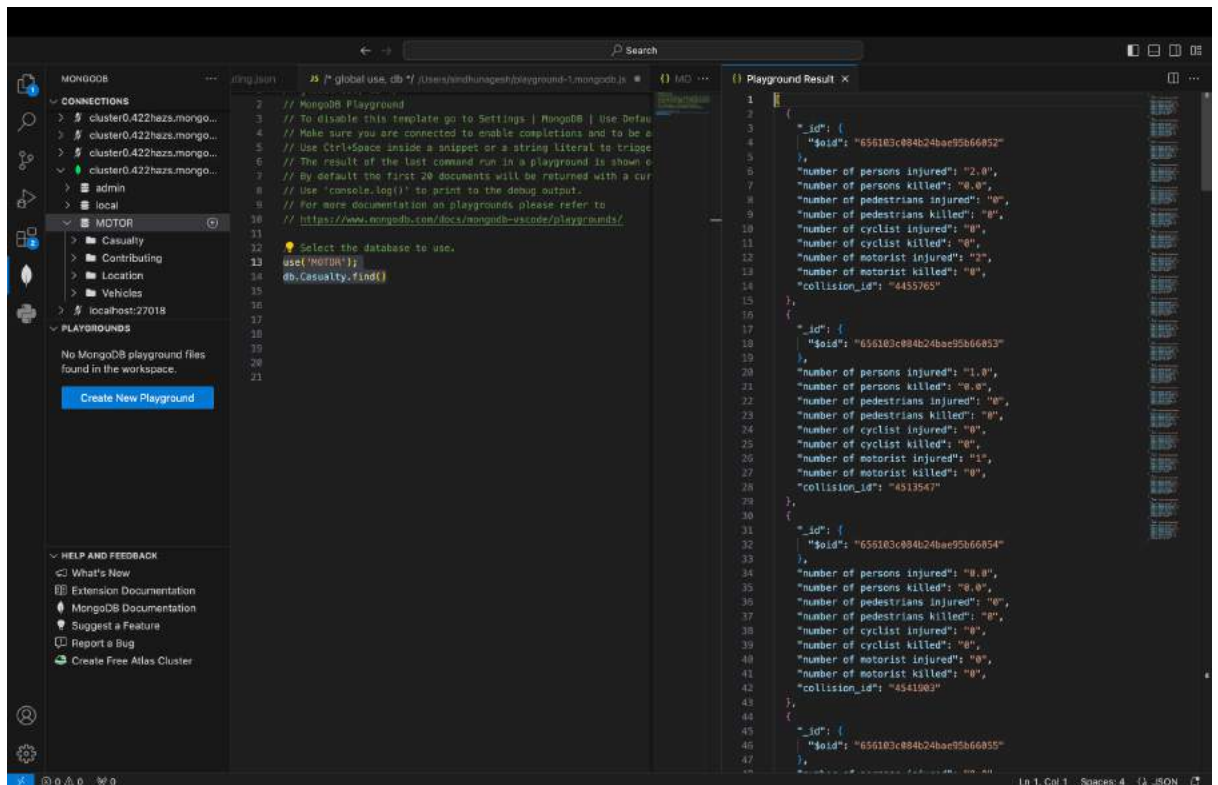


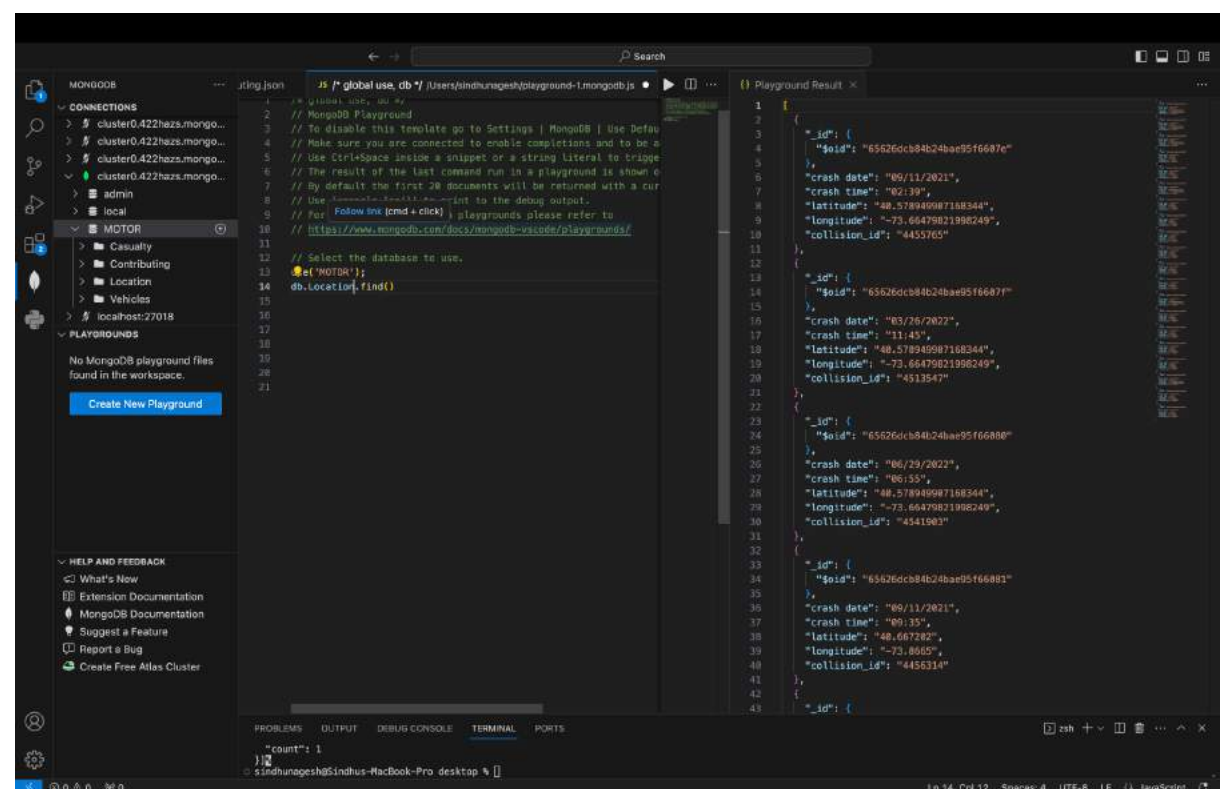
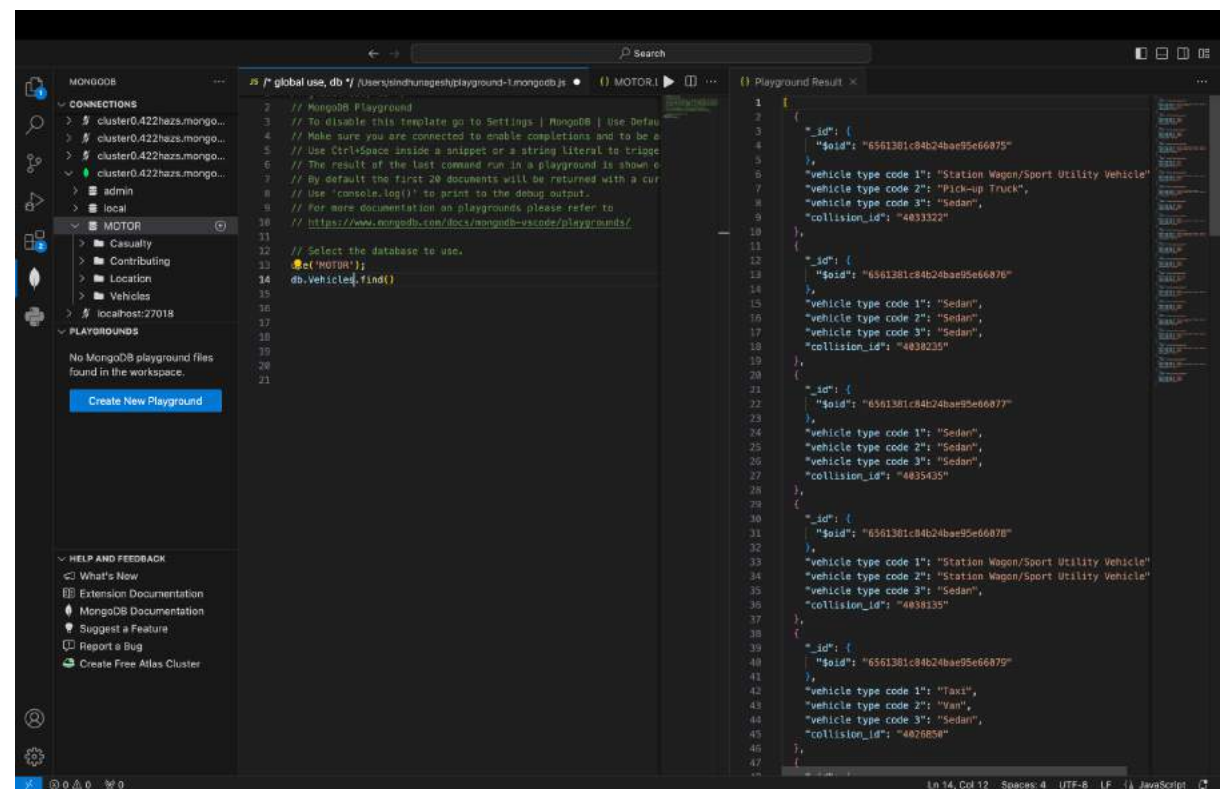
Switching to our database



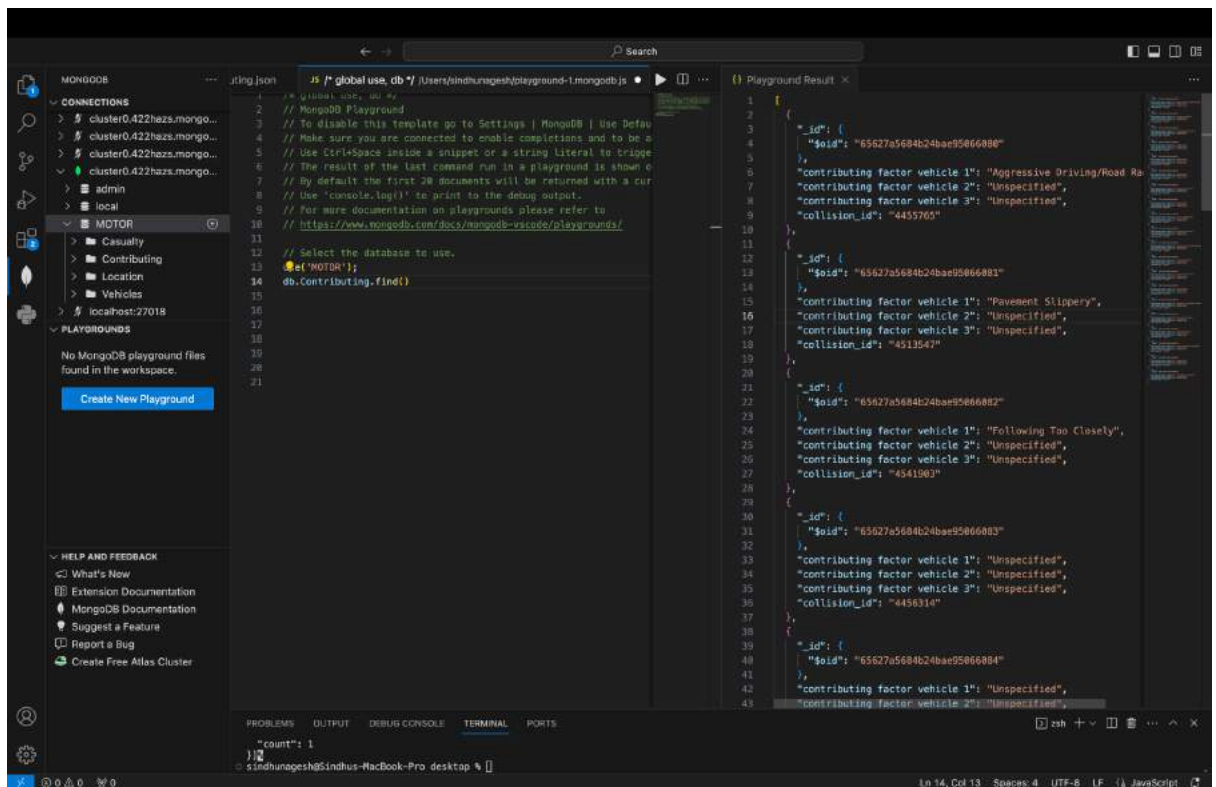
5. Sample data

Collection : Casualty





Collection: Contributing



6. Our sample aggregate

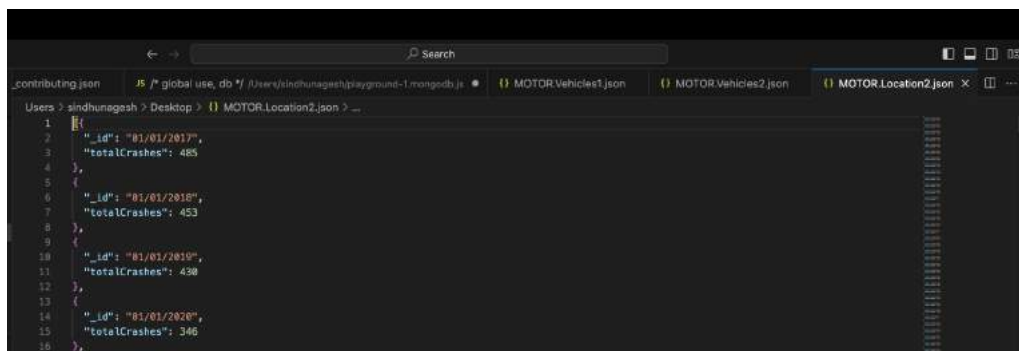
```

db.getCollection('Location').aggregate(
  [{ $sort: { 'crash time': -1 } }],
  { maxTimeMS: 60000, allowDiskUse: true }
);
  
```

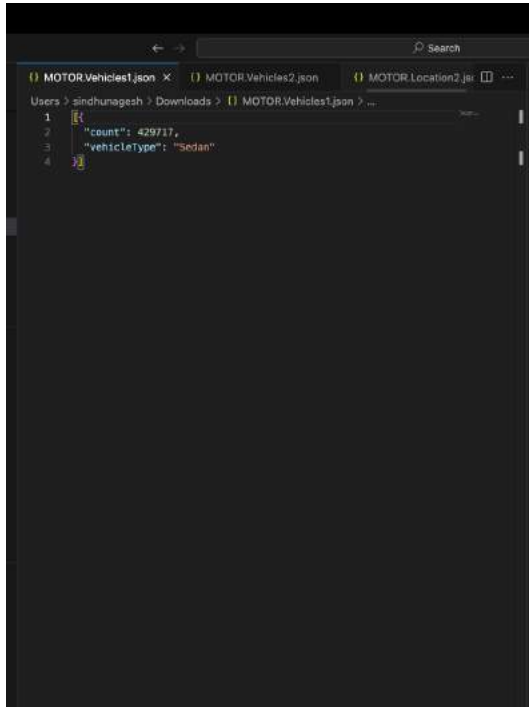
7. QUESTIONS

1.To analyze accidents on a monthly and yearly basis and identify when the maximum number of accidents occurred.

CODE:

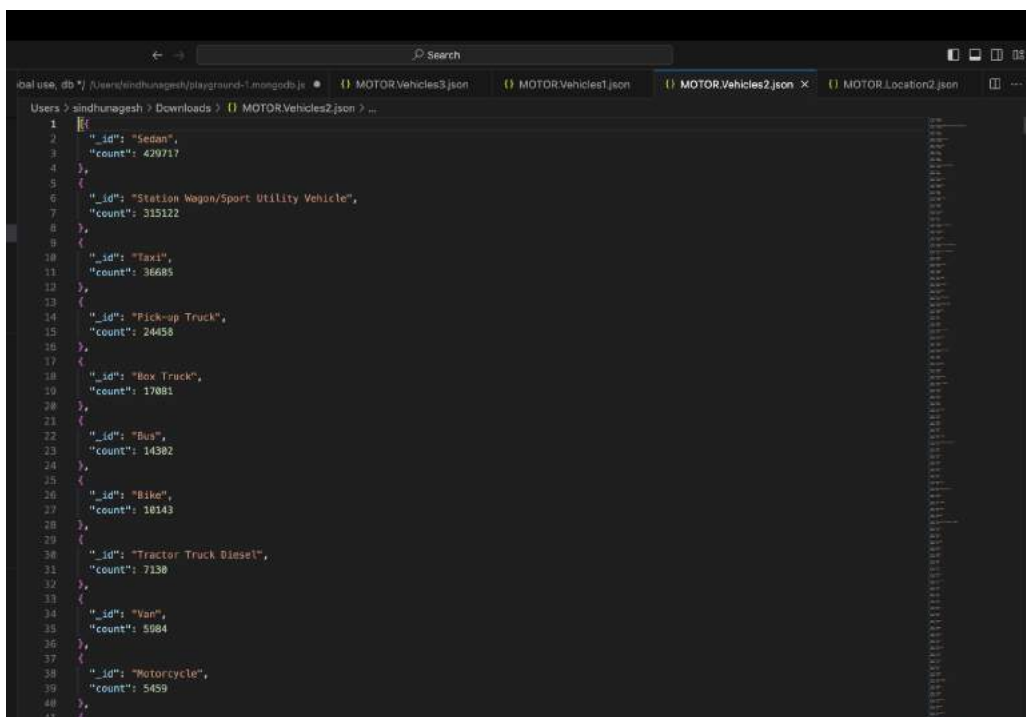


2.To understand which company's motor vehicles have undergone more crashes.
CODE:



```
1 {
2   "count": 429717,
3   "vehicleType": "Sedan"
4 }
```

3.To understand which company's motor vehicles have undergone more crashes in descending order.



```
1 {
2   "_id": "Sedan",
3   "count": 429717
4 },
5 {
6   "_id": "Station Wagon/Sport Utility Vehicle",
7   "count": 315122
8 },
9 {
10  "_id": "Taxi",
11  "count": 36685
12 },
13 {
14  "_id": "Pick-up Truck",
15  "count": 24458
16 },
17 {
18  "_id": "Box Truck",
19  "count": 17081
20 },
21 {
22  "_id": "Bus",
23  "count": 14382
24 },
25 {
26  "_id": "Bike",
27  "count": 10143
28 },
29 {
30  "_id": "Tractor Truck Diesel",
31  "count": 7130
32 },
33 {
34  "_id": "Van",
35  "count": 5984
36 },
37 {
38  "_id": "Motorcycle",
39  "count": 5459
40 },
41 {
```

4. To analyze most, least contributing factors for crashes.

The screenshot shows the SINDHU Aggregations interface for the 'MOTOR.Contributing' collection. The pipeline is configured with a single stage: `$group`. The aggregation query is: `{ "most contributing factor for crashes"`. The interface displays the output after the `$group` stage, showing a sample of 10 documents. The output is a list of documents, each representing a contributing factor and its count.

```
1 {
2   _id: {
3     $concat: [
4       {
5         $ifNull: [
6           "$contributing factor vehicle 1",
7           ""
8         ],
9       },
10      {
11        $ifNull: [
12          "$contributing factor vehicle 2",
13          ""
14        ],
15      },
16      {
17        $ifNull: [
18          "$contributing factor vehicle 3",
19          ""
20        ],
21      },
22    ],
23    count: {
24      count: 1,
25    },
26  },
27 }
```

Output after `$group` stage (Sample of 10 documents):

- `_id: "Turning Improperly Backing Unsafely Unspecified"`, `count: 19`
- `_id: "Passing or Lane Usage Improper/Unsafe Lane Changing/Unspecified"`, `count: 28`

The screenshot shows the SINDHU Aggregations interface for the 'MOTOR.Contributing' collection. The pipeline is configured with three stages: `$group`, `$sort`, and `$limit`. The aggregation query is: `{ "most contributing factor for crashes"`. The interface displays the output after the `$sort` stage, showing a sample of 10 documents. The output is a list of documents, each representing a contributing factor and its count.

```
1 {
2   count: 1,
3 }
```

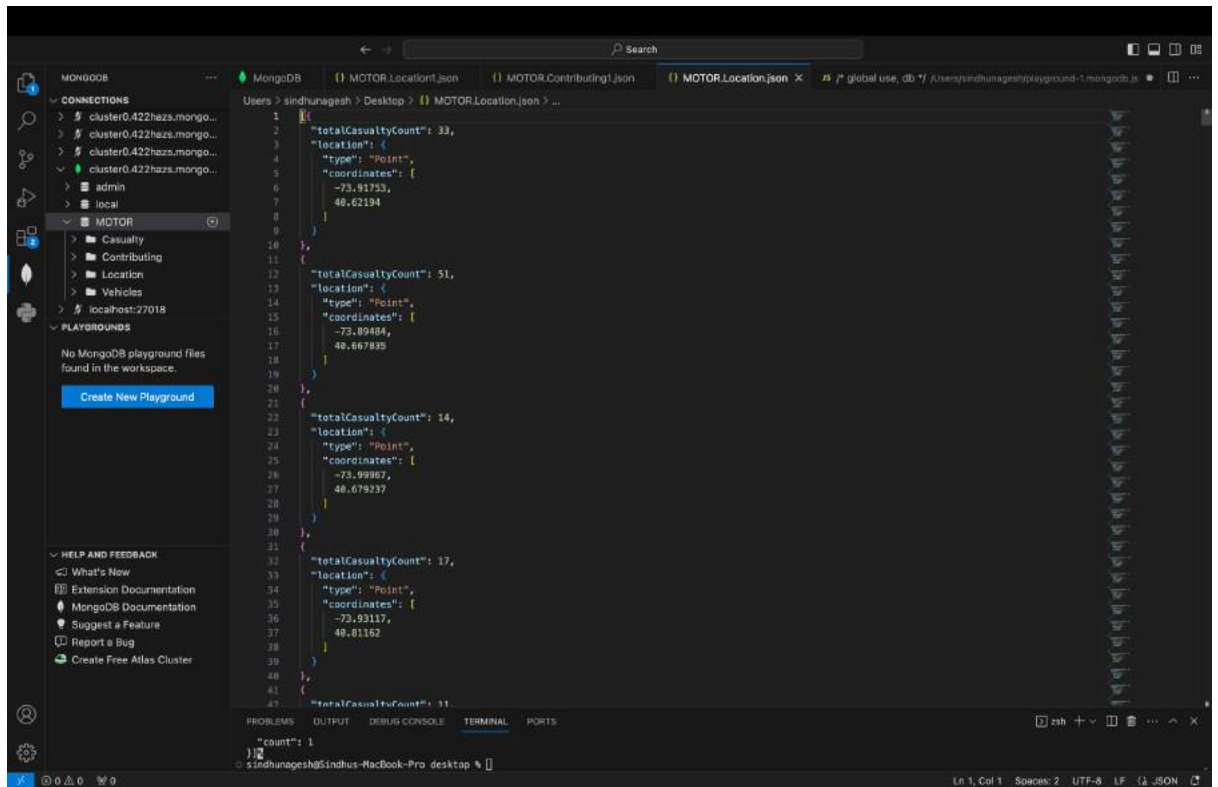
Output after `$sort` stage (Sample of 10 documents):

- `_id: "Unspecified/Unspecified/Unspecified"`, `count: 23824`
- `_id: "Driver Inattention/Distracted/Unspecified/Unspecified"`, `count: 20840`

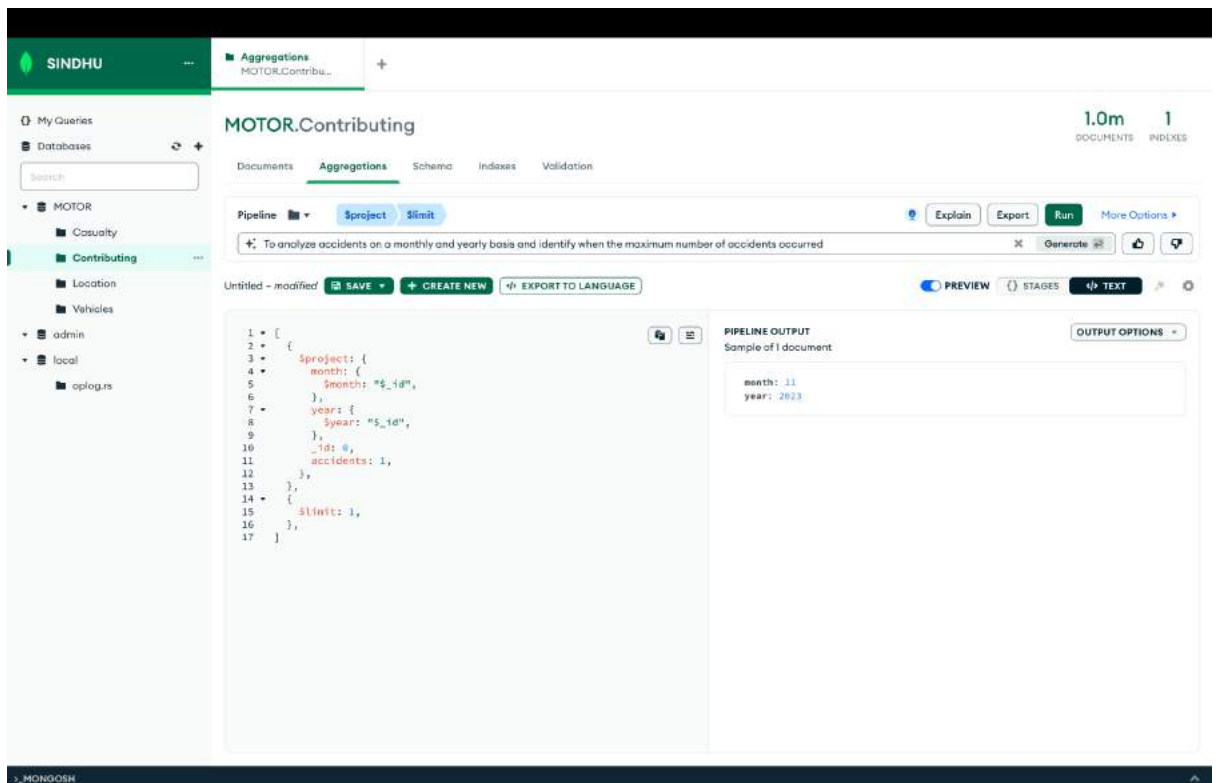
Output after `$limit` stage (Sample of 1 document):

- `_id: "Unspecified/Unspecified/Unspecified"`, `count: 23824`

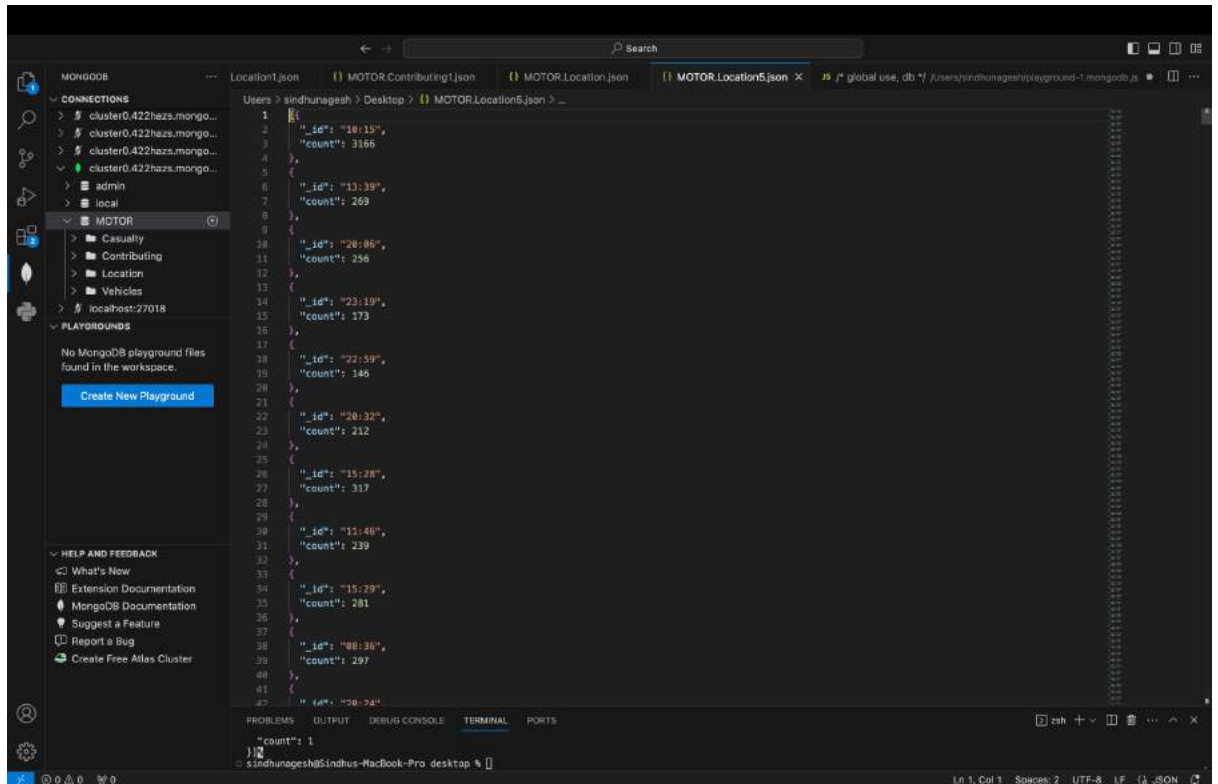
5. To analyze in which location we have total casualty count.



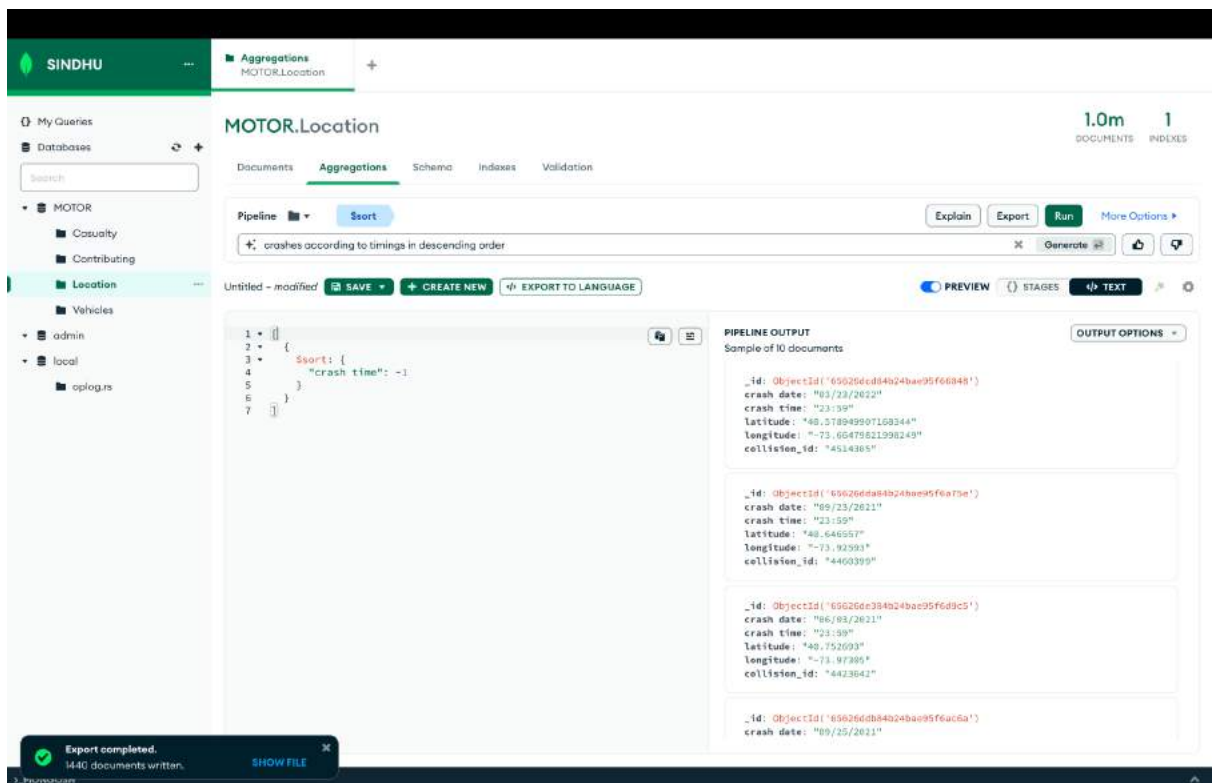
6. To analyze Year and month among the data where max crashes happened.



7. To analyze Crashes according to timings



To Check the most recently occurred crash.

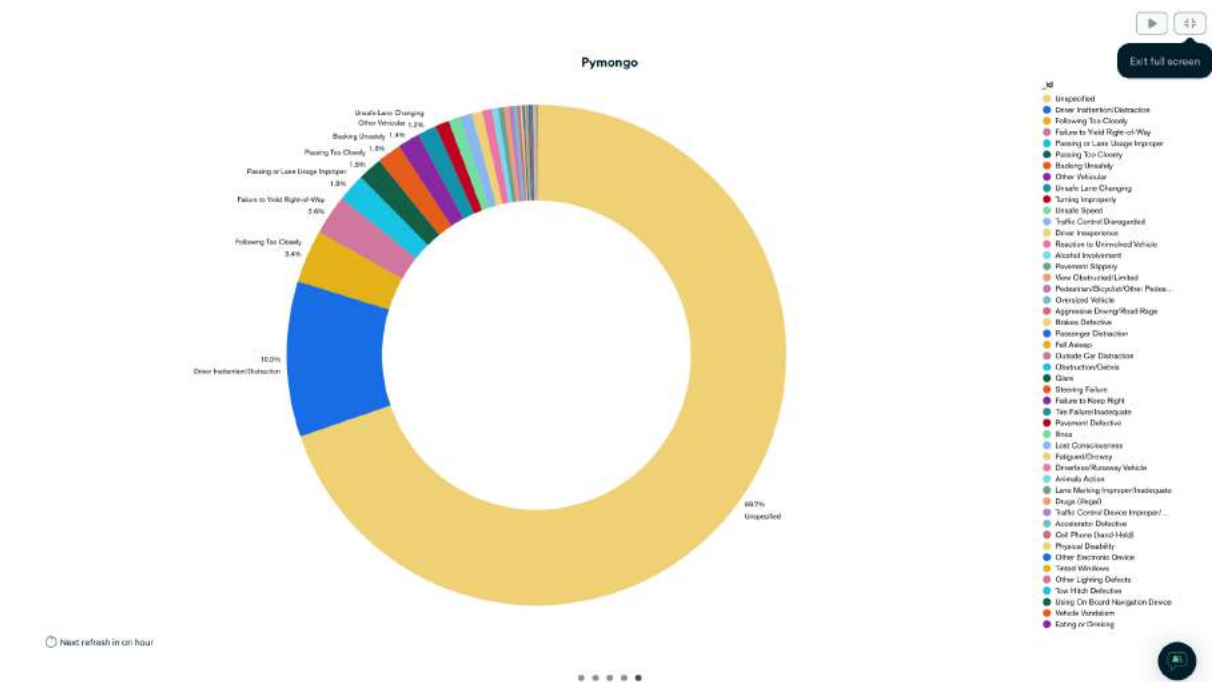


8. Pymongo

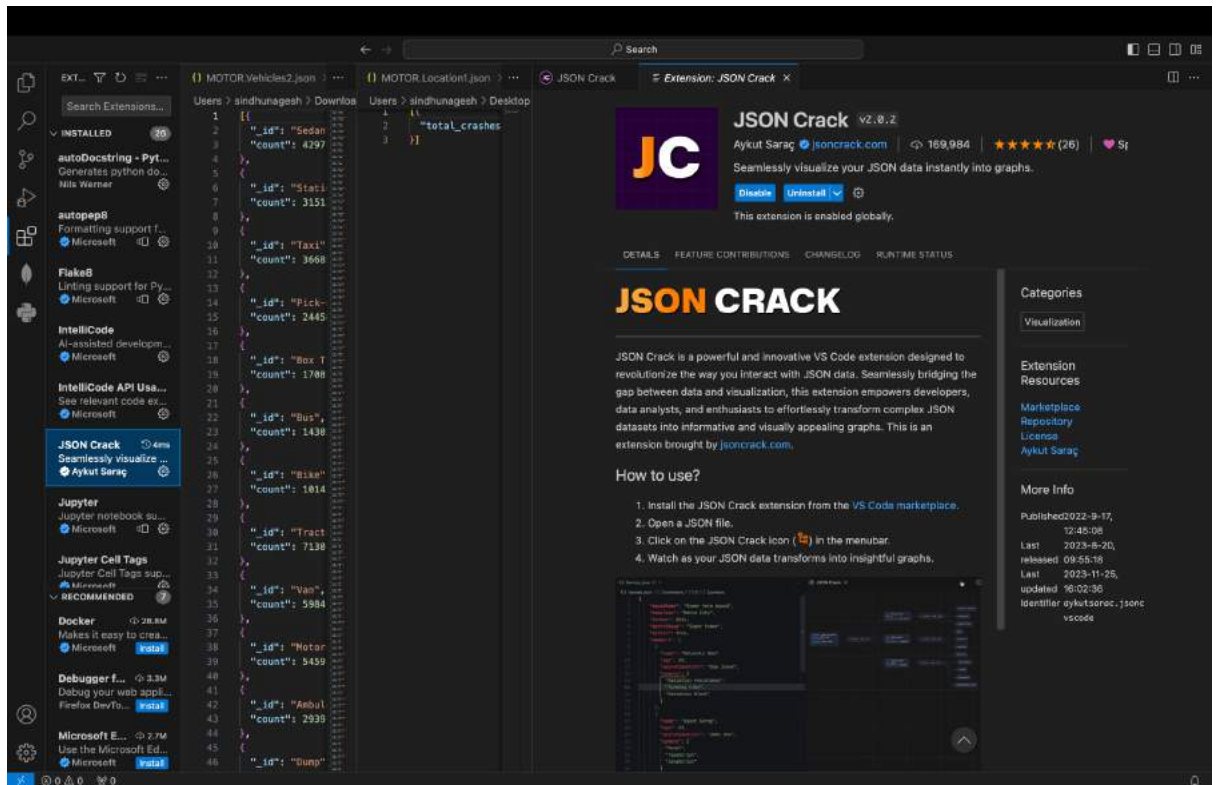
```

1 [{"_id": "Unspecified",
2   "count": 2191070},
3 ],
4
5 [{"_id": "Driver Inattention/Distracted",
6   "count": 315287},
7 ],
8
9 [{"_id": "Following Too Closely",
10  "count": 187915},
11 ],
12
13 [{"_id": "Failure to Yield Right-of-Way",
14  "count": 88792},
15 ],
16
17 [{"_id": "Passing or Lane Usage Improper",
18  "count": 57852},
19 ],
20
21 [{"_id": "Passing Too Closely",
22  "count": 49485},
23 ],
24
25 [{"_id": "Backing Unsafely",
26  "count": 49248},
27 ],
28
29 [{"_id": "Other Vehicular",
30  "count": 44518},
31 ],
32
33 [{"_id": "Unsafe Lane Changing",
34  "count": 38981},
35 ],
36
37 [{"_id": "Turning Improperly",
38  "count": 27789},
39 ],
40
41 [{"_id": "Unsafe Speed",
42  "count": 24601},
43 ],
44
45 [{"_id": "Traffic Control Disregarded",
46  "count": 24601},
47 ],
48 ]

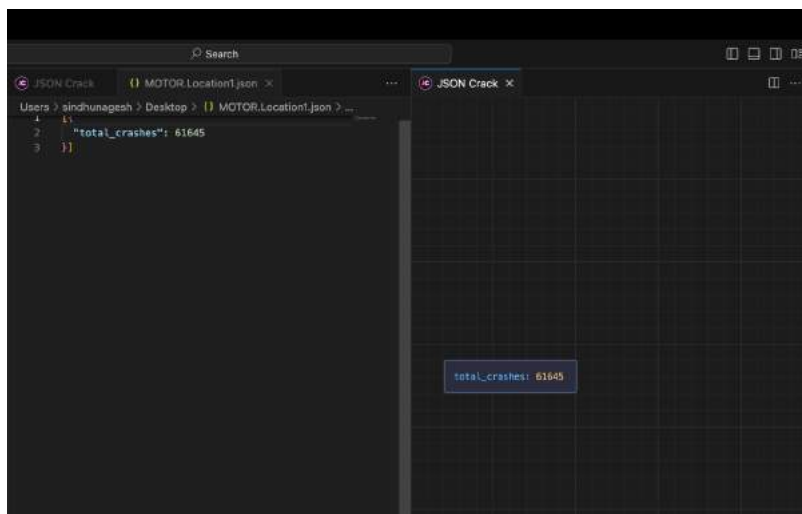
```



9. Visualizing through JSON CRACK



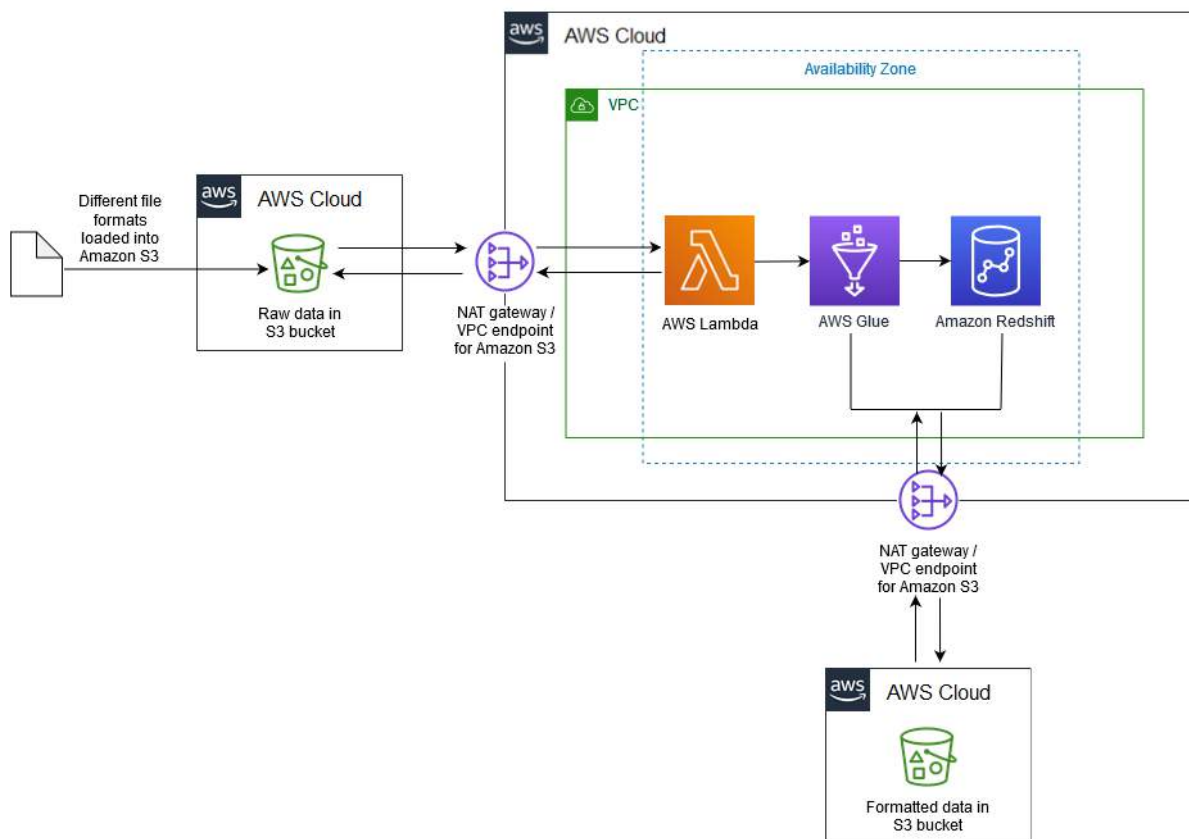
Here we are trying to use JSON CRACK to show how our output can be displayed as graphs.



Here we can see all the counts of crashes that happened on March 22nd in all the years from 2016 to 2022.

		<code>_id: "03/22/2016"</code> totalCrashes: 8
		<code>_id: "03/22/2017"</code> totalCrashes: 617
		<code>_id: "03/22/2018"</code> totalCrashes: 671
		<code>_id: "03/22/2019"</code> totalCrashes: 718
		<code>_id: "03/22/2020"</code> totalCrashes: 139
		<code>_id: "03/22/2021"</code> totalCrashes: 4
		<code>_id: "03/22/2022"</code> totalCrashes: 17

SCENARIO 3: Using AWS Environment

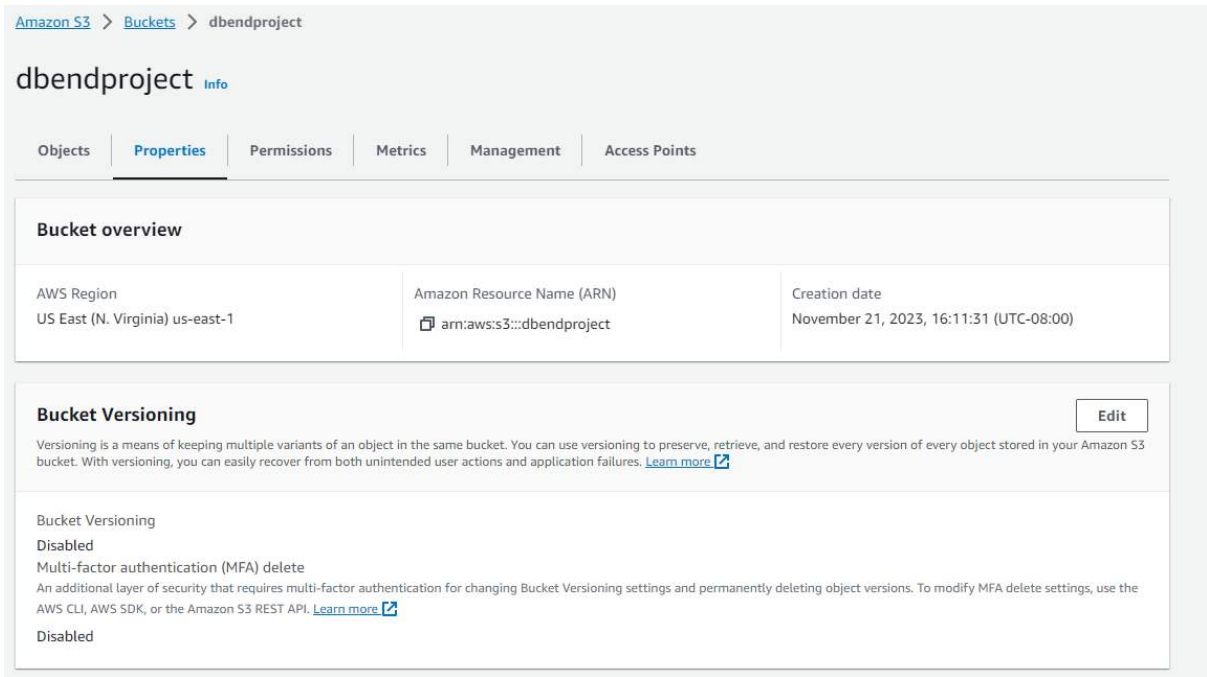


AWS internal workflow ([Source](#))

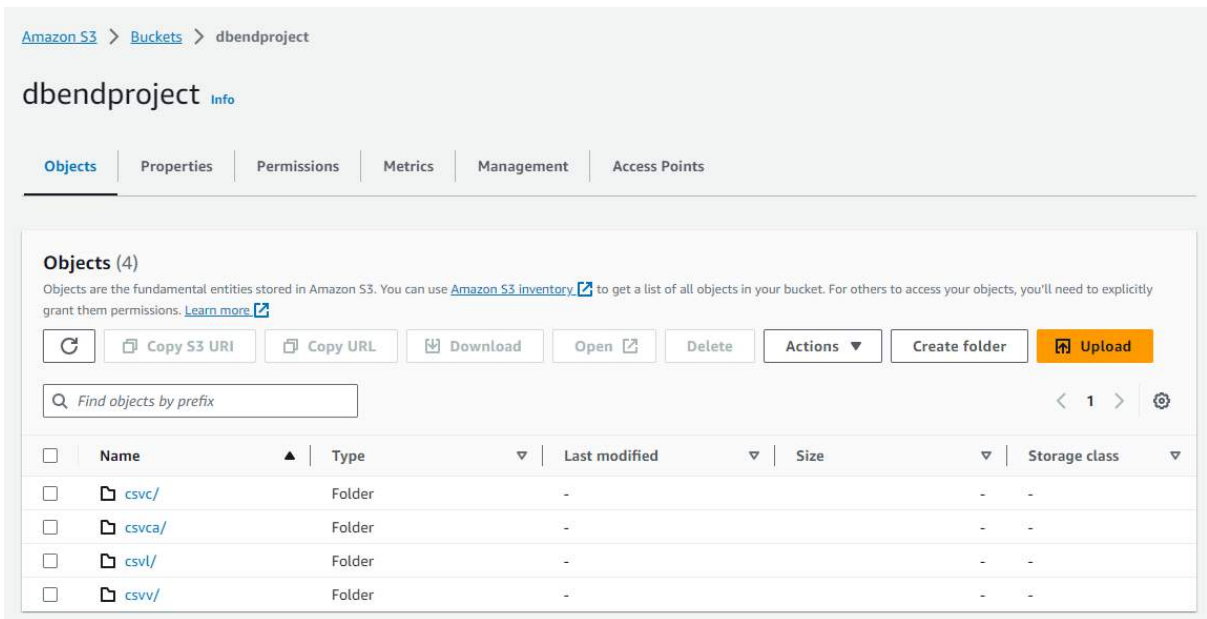
STEP 1: Create S3 bucket

With Amazon S3 (Simple Storage Service), a cloud-based object storage service, a user may store and retrieve unlimited amount of data from any location on the internet. It is widely used for big data analytics, data lakes, content distribution, and website hosting.

An S3 bucket refers to a storage container in Amazon Web Services (AWS) Simple Storage Service (S3), which is designed for storing and retrieving any amount of data from anywhere on the web. These buckets can store various types of objects, including files, images, videos, databases, and backups. They provide scalability, security, and durability, making them a fundamental component for data storage and management in the AWS cloud infrastructure. You can control access to your S3 buckets through various means such as bucket policies, access control lists (ACLs), and IAM roles to ensure security and compliance with your requirements.



S3 bucket creation

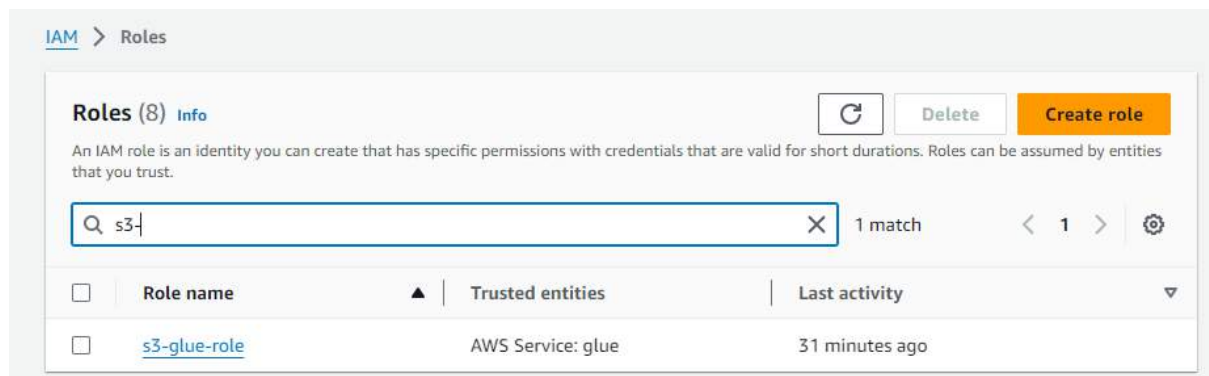


Folders for S3

STEP 2: IAM (We created IAM roles for the Crawlers access)

The Identity and Access Management (IAM) service from AWS enables a user to securely control access to AWS services. IAM positions ensure only authorized users may access your AWS account resources is made feasible by the development and control over groups, users, and permissions.

IAM stands for Identity and Access Management. It's a service provided by AWS (Amazon Web Services) that helps manage users, groups, roles, and their corresponding level of access to AWS services and resources. With IAM, you can control who has permission to do what within your AWS environment.

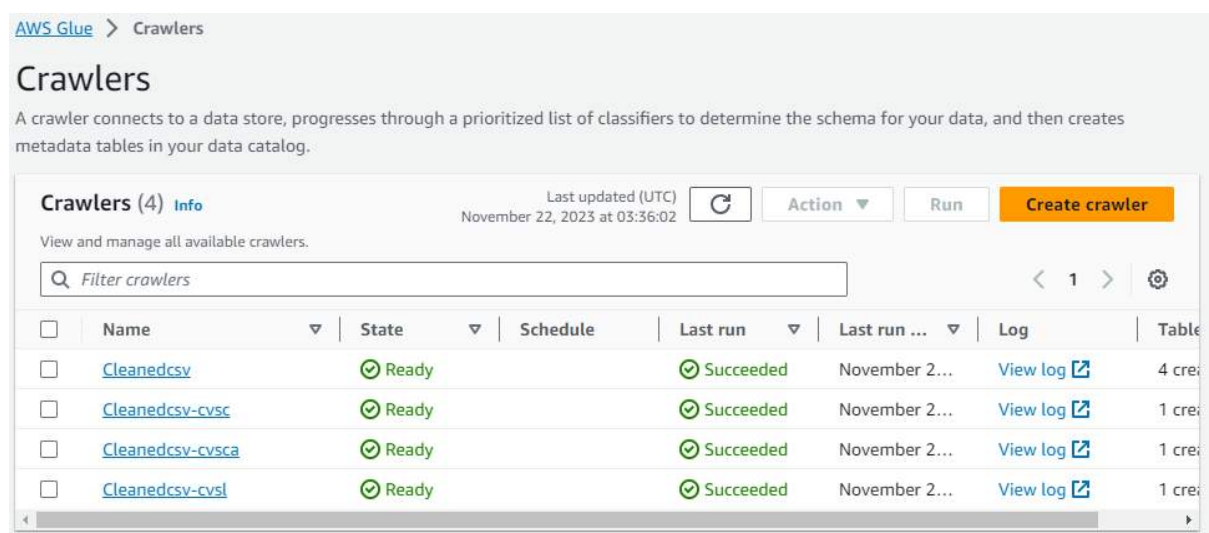


Roles for the crawler to import

STEP 3: AWS Glue (Crawlers)

With the fully-managed extract, transform, and load (ETL) service AWS Glue, moving data across data storage is easy. It simplifies the process for preparing and adapting data for machine learning, analytics, and education as well as other uses. The act of producing and data pipeline management is automated, making it possible to expand to manage enormous datasets.

A crawler connects to a data store, progresses through a prioritized list of classifiers to determine the schema for your data, and then creates metadata tables in your data catalog.



AWS Glue for ETL process

STEP 4: Amazon Athena:

AWS Athena is an interactive query service that allows you to analyze data directly in Amazon S3 using standard SQL. It enables you to run queries on data stored in S3 without needing to set up complex infrastructure or manage servers.

Step 5: Connection of Athena with Python

Verify that your computer is configured with your AWS login information. These credentials are used by Boto3 to verify your requests. Using the Python script's environment variables or the AWS Command Line Interface (CLI), you may configure your login credentials. To build an Amazon Athena client, use Boto3. Using this client, you may use your Python script to communicate with Athena services. You may use the client to run SQL queries on Athena once you have it. The Boto3 methods allow you to submit SQL queries and receive the results. After running a query, Athena can deliver the results in CSV, JSON, or Parquet formats. Depending on your needs, Boto3 can handle and process these findings.

```
In [1]: import boto3
import pandas as pd
from io import StringIO
import time
```

Libraries Required For Connection

```
In [2]: AWS_ACCESS_KEY = "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
AWS_SECRET_KEY = "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
AWS_REGION = "us-east-1"
SCHEMA_NAME = "crash_dataset"
S3_STAGING_DIR = "s3://project-db-bucket/output/"
S3_BUCKET = "project-db-bucket"
S3_OUTPUT_DIRECTORY="output"
```

Some Keys And Location Required

```
In [3]: # conect to Athena
athena_client = boto3.client(
    "athena",
    aws_access_key_id = AWS_ACCESS_KEY,
    aws_secret_access_key = AWS_SECRET_KEY,
    region_name = AWS_REGION,
)
```

Connection with Athena

```
In [5]: response = athena_client.start_query_execution (
    QueryString="SELECT * FROM csvv",
    QueryExecutionContext={"Database": SCHEMA_NAME},
    ResultConfiguration={
        "OutputLocation": S3_STAGING_DIR,
        "EncryptionConfiguration": {"EncryptionOption": "SSE_S3"},
    },
)
```

Queries Using Python To Athena

```
In [6]: response
```

```
Out[6]: {'QueryExecutionId': 'fa8a524f-b8f2-44e9-85cc-5d3cd13b78f0',
'ResponseMetadata': {'RequestId': '450cc216-df4d-48e0-8f3e-a78e72e512fc',
'HTTPStatusCode': 200,
'HTTPHeaders': {'date': 'Thu, 23 Nov 2023 18:09:09 GMT',
'content-type': 'application/x-amz-json-1.1',
'content-length': '59',
'connection': 'keep-alive',
'x-amzn-requestid': '450cc216-df4d-48e0-8f3e-a78e72e512fc'},
'RetryAttempts': 0}}
```

```
In [7]: ## vehical table
csvv = download_and_load_query_results(athena_client, response)
```

```
In [8]: csvv.head()
```

```
Out[8]:
```

	vehicle type code 1	vehicle type code 2	vehicle type code 3	collision_id
0	Station Wagon/Sport Utility Vehicle	Pick-up Truck	Sedan	4033322.0
1	Sedan	Sedan	Sedan	4030235.0
2	Sedan	Sedan	Sedan	4035435.0
3	Station Wagon/Sport Utility Vehicle	Station Wagon/Sport Utility Vehicle	Sedan	4038135.0
4	Taxi	Van	Sedan	4026850.0

Executing the basic query to check

VII. ANALYTICS AND DATA DRIVEN DECISIONS

Queries Analysis using python and Athena:

```
In [21]: #Query 1
```

```
In [22]: athena_query = """
SELECT
    collision_id,
    COUNT(*) AS total_crashes,
    SUM("number of persons killed") AS total_fatalities
FROM csvca
GROUP BY collision_id
ORDER BY total_fatalities DESC
.....
```

```
In [23]: response = athena_client.start_query_execution (
QueryString= athena_query,
QueryExecutionContext={"Database": SCHEMA_NAME},
ResultConfiguration={
    "OutputLocation": S3_STAGING_DIR,
    "EncryptionConfiguration": {"EncryptionOption": "SSE_S3"},
},
)
```

```
In [24]: rs1 = download_and_load_query_results(athena_client, response)
rs1.head()
```

```
Out[24]:
```

	collision_id	total_crashes	total_fatalities
0	3782508	1	8.0
1	4278634	1	4.0
2	3752786	1	4.0
3	4355333	1	3.0
4	4419561	1	3.0

Query 1 using python and Athena Connector(boto3)

In [25]: # Query 2 Identify the top 5 contributing factors for crashes:

```
In [26]: athena_query = """
SELECT
    contributing_factor,
    COUNT(*) AS total_crashes
FROM (
    SELECT
        all(ARRAY["contributing factor vehicle 1", "contributing factor vehicle 2", "contributing factor vehicle 3"]) AS flattened_factors
    FROM csvc
) as flattened_factors
WHERE contributing_factor IS NOT NULL
GROUP BY contributing_factor
ORDER BY total_crashes DESC
LIMIT 5
"""
response = athena_client.start_query_execution (
    QueryString= athena_query,
    QueryExecutionContext={"Database": SCHEMA_NAME},
    ResultConfiguration={
        "OutputLocation": S3_STAGING_DIR,
        "EncryptionConfiguration": {"EncryptionOption": "SSE_S3"},
    },
)
rs1 = download_and_load_query_results(athena_client, response)
rs1.head()
```

Out[26]:

	contributing_factor	total_crashes
0	[Driver Inattention/Distracted, Unspecified,]	165797
1	[Unspecified, Unspecified,]	134045
2	[Unspecified, ,]	97810
3	[Following Too Closely, Unspecified,]	73155
4	[Failure to Yield Right-of-Way, Unspecified,]	47469

Query 2 using python and Athena Connector(boto3)

In [27]: # Query 3 Calculate the average number of persons injured and killed per crash:

```
In [28]: athena_query = """
SELECT
    AVG("number of persons injured") AS avg_persons_injured,
    AVG("number of persons killed") AS avg_persons_killed
FROM csvca;
"""
response = athena_client.start_query_execution (
    QueryString= athena_query,
    QueryExecutionContext={"Database": SCHEMA_NAME},
    ResultConfiguration={
        "OutputLocation": S3_STAGING_DIR,
        "EncryptionConfiguration": {"EncryptionOption": "SSE_S3"},
    },
)
rs1 = download_and_load_query_results(athena_client, response)
rs1.head()
```

Out[28]:

	avg_persons_injured	avg_persons_killed
0	0.320919	0.00148

Query 3 using python and Athena Connector(boto3)

In [29]: # Query 4 Find the most common vehicle types involved in crashes:

```
In [30]: athena_query = """
SELECT
    vehicle_type,
    COUNT(*) AS total_crashes
FROM (
    SELECT
        ALL(ARRAY["vehicle type code 1","vehicle type code 2", "vehicle type code 3"]) AS vehicle_type
    FROM csvv
) as flattened_types
WHERE vehicle_type IS NOT NULL
GROUP BY vehicle_type
ORDER BY total_crashes DESC;
"""
response = athena_client.start_query_execution (
    QueryString= athena_query,
    QueryExecutionContext={"Database": SCHEMA_NAME},
    ResultConfiguration={
        "OutputLocation": S3_STAGING_DIR,
        "EncryptionConfiguration": {"EncryptionOption": "SSE_S3"},
    },
)
rs1 = download_and_load_query_results(athena_client, response)
rs1.head()
```

Out[30]:

	vehicle_type	total_crashes
0	[Sedan, Sedan, Sedan]	301214
1	[Station Wagon/Sport Utility Vehicle, Sedan, S...	191953
2	[Station Wagon/Sport Utility Vehicle, Station ...	107644
3	[Sedan, Station Wagon/Sport Utility Vehicle, S...	107265
4	[Taxi, Sedan, Sedan]	19291

Query 4 using python and Athena Connector(boto3)

In [31]: # Query 5 Calculate the ratio of pedestrian injuries to total injuries:

```
In [32]: athena_query = """
SELECT
    SUM("number of pedestrians injured") / (SUM("number of persons injured")) AS pedestrian_injury_ratio
FROM csvca
"""
response = athena_client.start_query_execution (
    QueryString= athena_query,
    QueryExecutionContext={"Database": SCHEMA_NAME},
    ResultConfiguration={
        "OutputLocation": S3_STAGING_DIR,
        "EncryptionConfiguration": {"EncryptionOption": "SSE_S3"},
    },
)
rs1 = download_and_load_query_results(athena_client, response)
rs1.head()
```

Out[32]:

	pedestrian_injury_ratio
0	0.168787

Query 5 using python and Athena Connector(boto3)

In [33]: `# Query 6 Find the most common contributing factor for each v`

```
In [34]: athena_query = """
        SELECT
            csvc.collusion_id,
            "vehicle type code 1",
            "vehicle type code 2",
            "vehicle type code 3",
            MAX(CASE WHEN "vehicle type code 1" = 'Car' THEN "contribut
            MAX(CASE WHEN "vehicle type code 2" = 'Car' THEN "contribut
            MAX(CASE WHEN "vehicle type code 3" = 'Car' THEN "contribut
        FROM csvv
        LEFT JOIN csvc ON csvv.collusion_id = csvc.collusion_id;
        """
        response = athena_client.start_query_execution (
            QueryString= athena_query,
            QueryExecutionContext={"Database": SCHEMA_NAME},
            ResultConfiguration={
                "OutputLocation": S3_STAGING_DIR,
                "EncryptionConfiguration": {"EncryptionOption": "SSE_
            },
        )
        rs1 = download_and_load_query_results(athena_client, response)
        rs1.head()
```

Out[34]:

	collision_id	vehicle type code 1	vehicle type code 2	vehicle type code 3
0	3405265.0	4 dr sedan	Sedan	Sedan
1	3406511.0	Sedan	Sedan	Sedan
2	3406526.0	4 dr sedan	Taxi	Sedan
3	3406877.0	Taxi	Taxi	Sedan
4	3407782.0	4 dr sedan	4 dr sedan	Sedan

Query 6 using python and Athena Connector(boto3)

In [35]: `# Query 7 Calculate the percentage of crashes involving multiple vehicles:`

```
In [36]: athena_query = """
        SELECT
            COUNT(*) AS total_crashes,
            COUNT(*) * 100 / (SUM(CASE WHEN "vehicle type code 2" IS NOT NULL OR "vehicle type code 3" IS NOT NULL THEN 1 ELSE 0 END))
        FROM csvv
        """
        response = athena_client.start_query_execution (
            QueryString= athena_query,
            QueryExecutionContext={"Database": SCHEMA_NAME},
            ResultConfiguration={
                "OutputLocation": S3_STAGING_DIR,
                "EncryptionConfiguration": {"EncryptionOption": "SSE_S3"},
            },
        )
        rs1 = download_and_load_query_results(athena_client, response)
        rs1.head()
```

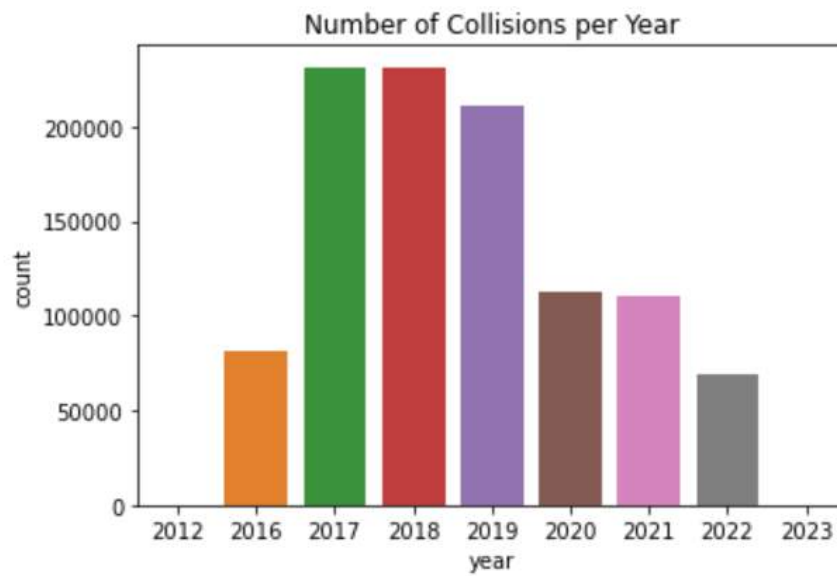
Out[36]:

	total_crashes	multi_vehicle_percentage
0	1048575	100

Query 7 using python and Athena Connector(boto3)

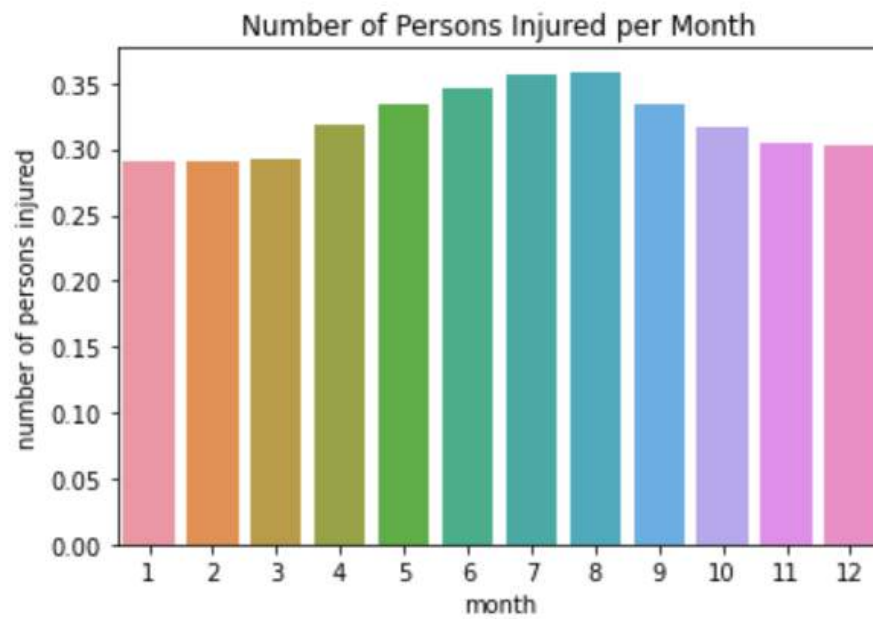
Visualizations using Python after getting data from athena:

Out[46]: Text(0.5, 1.0, 'Number of Collisions per Year')



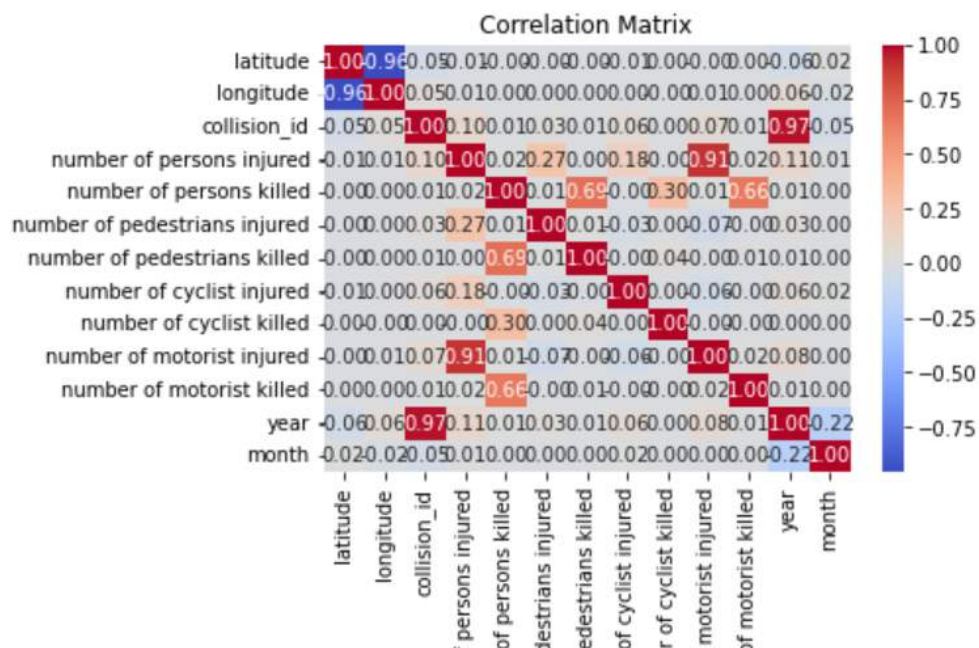
Crashes in each year

Out[49]: Text(0.5, 1.0, 'Number of Persons Injured per Month')

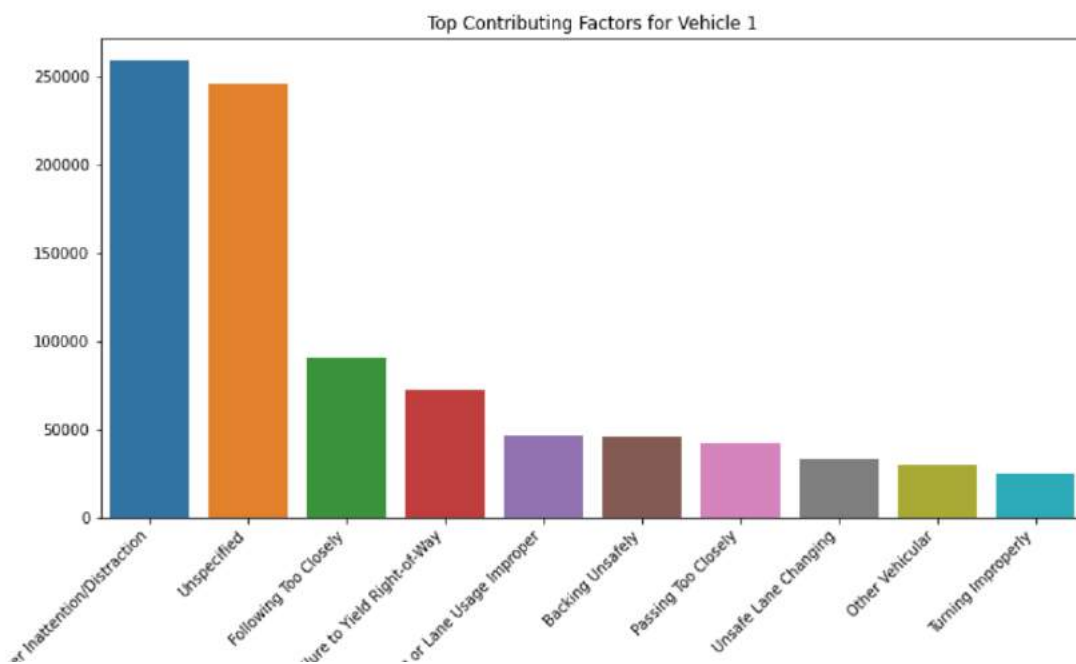


Crashes in month of the year in last 10 years

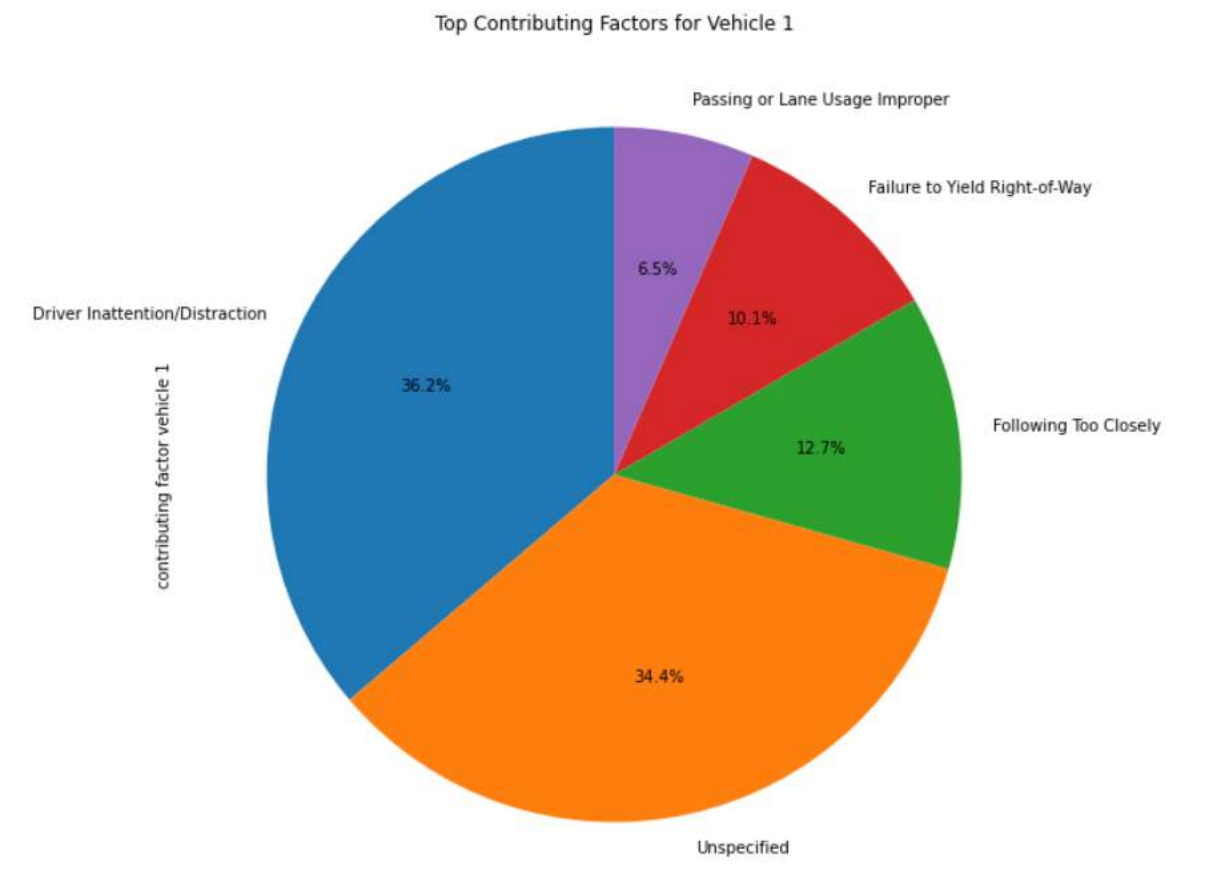
Out[50]: Text(0.5, 1.0, 'Correlation Matrix')



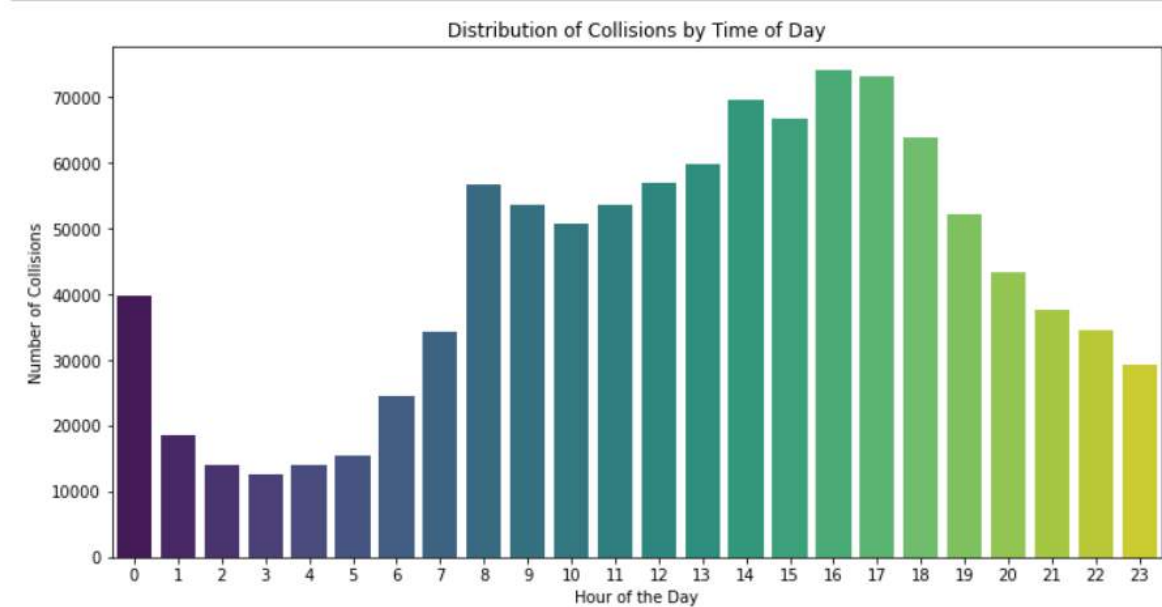
Heatmap / Correlation matrix with all the numeric data-set



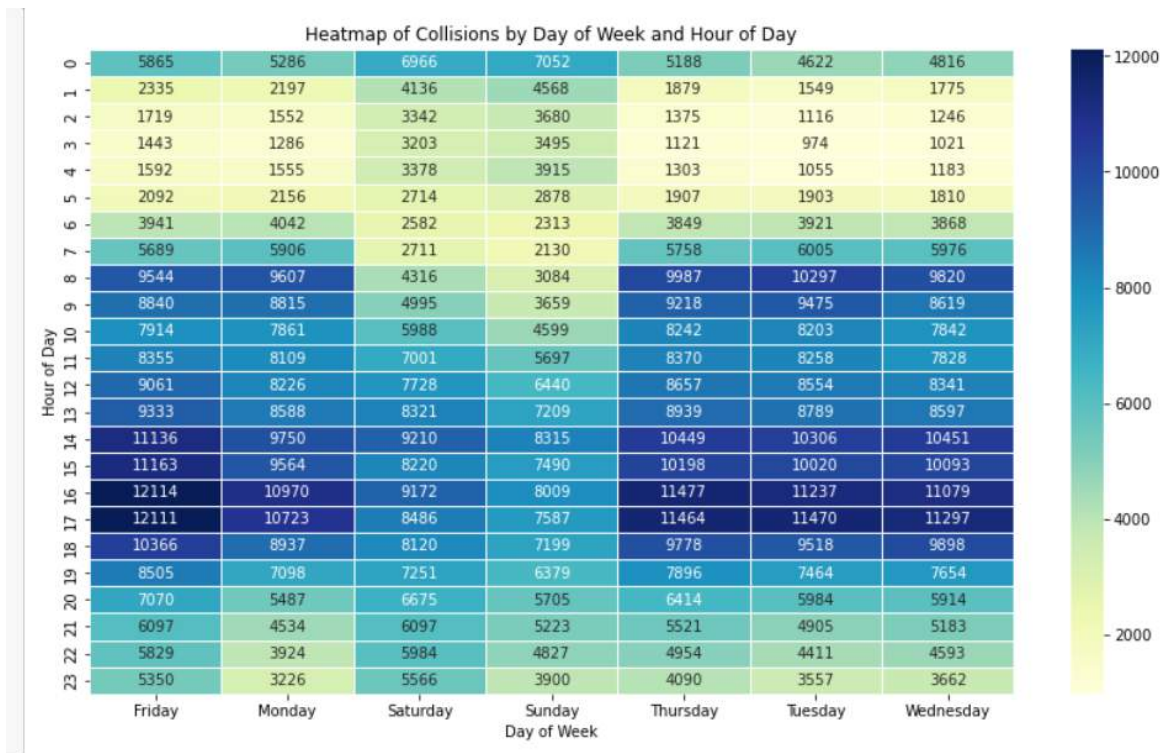
Top 10 factors responsible for crashes



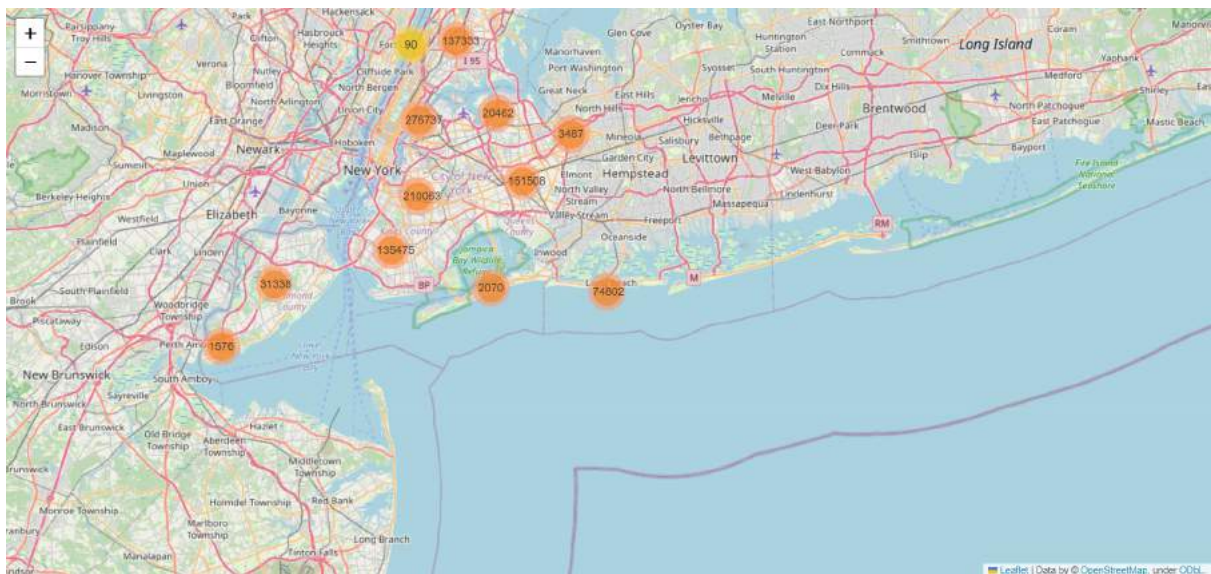
Top 10 factors responsible in crashes in form of pie chart



Number of crashes depending on the

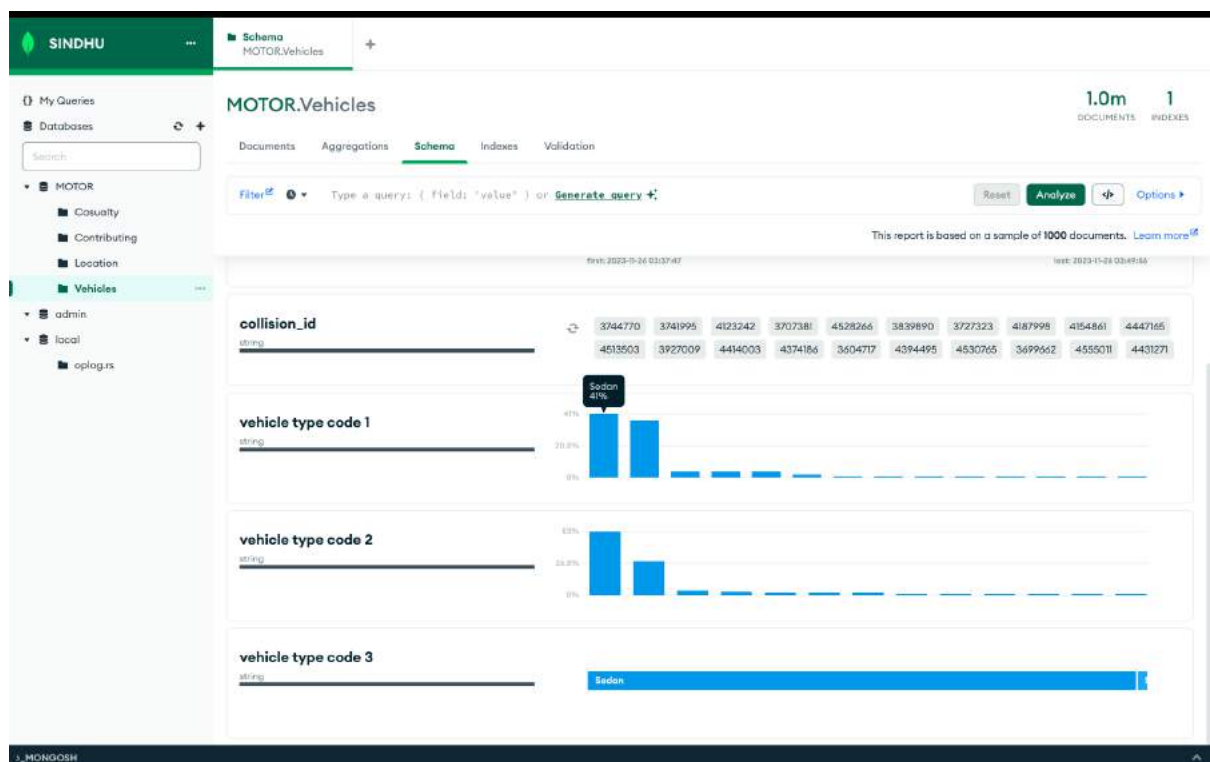
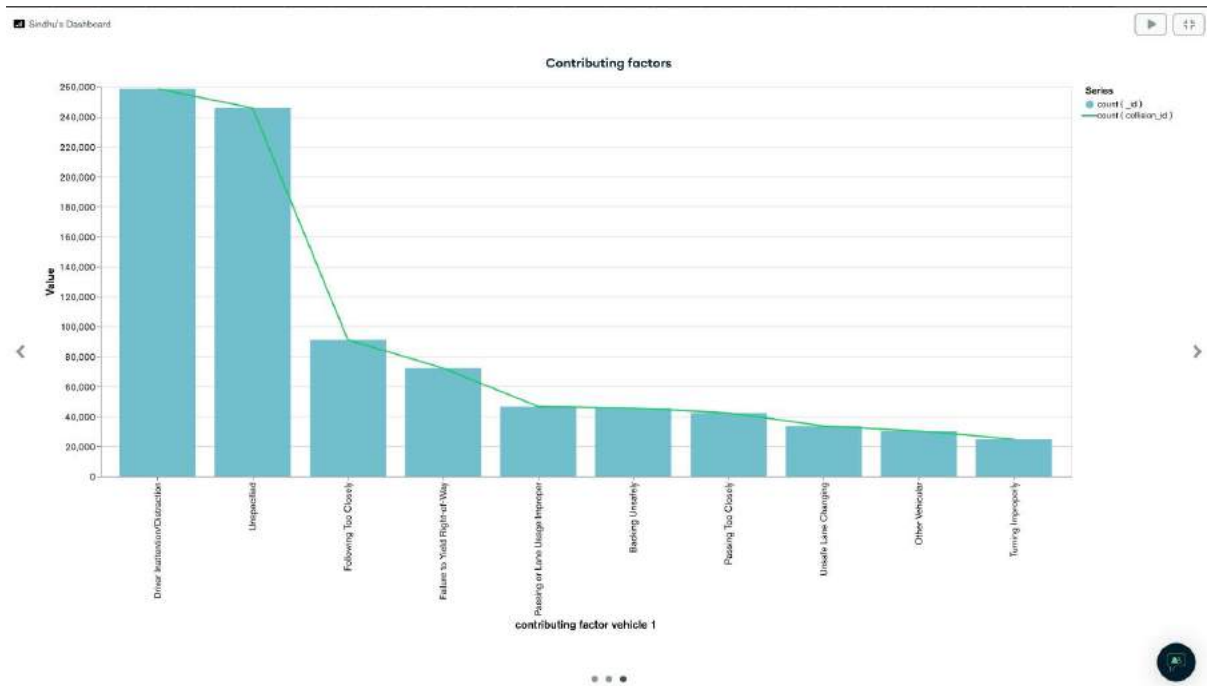


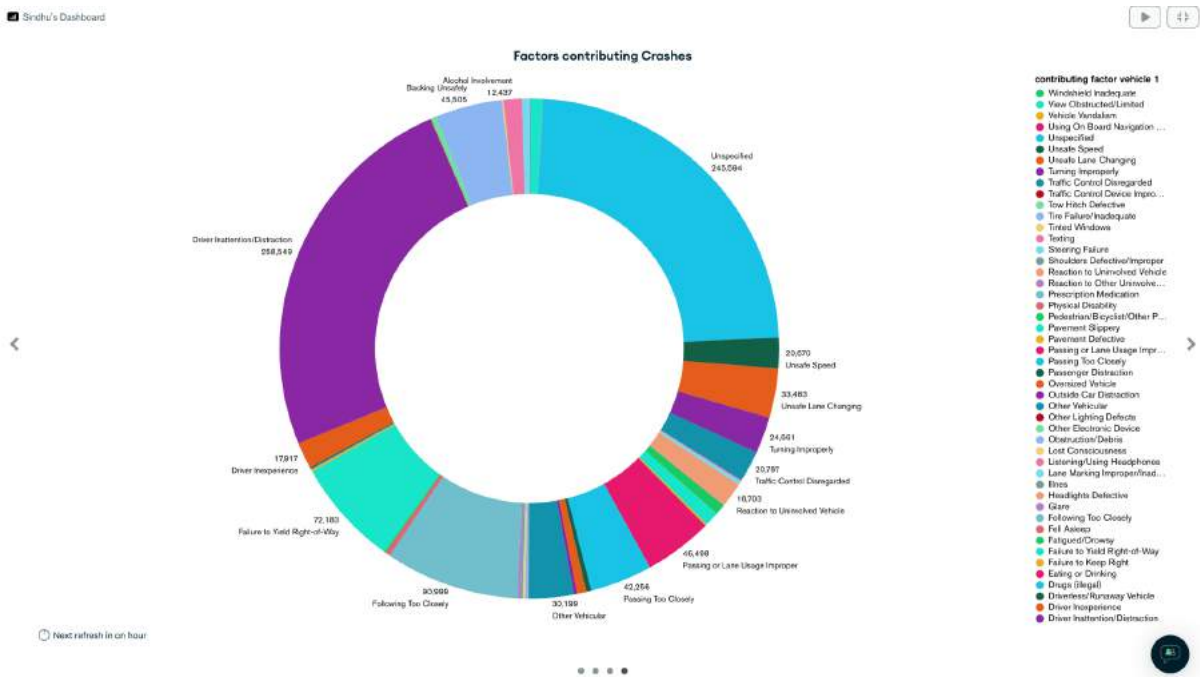
Heatmap for the above pie for further details



Map for the position and location of crashes in new york

Visualizations using MongoDB:





Visualizations using Power BI:

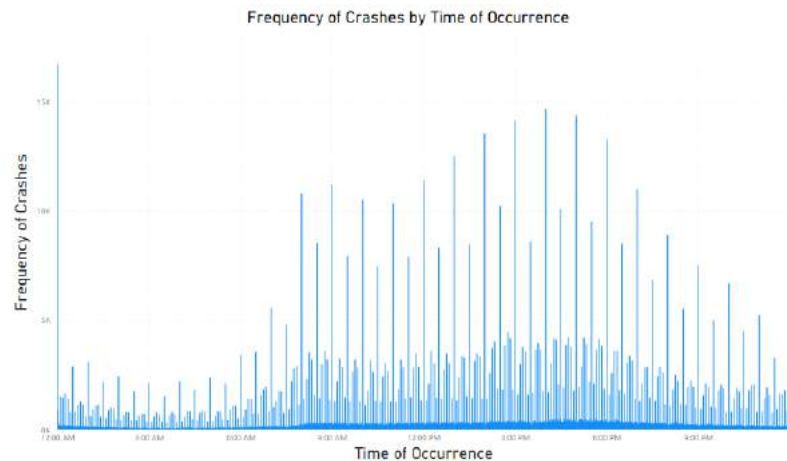
- Analysis of Crash Frequency by Time of Occurrence

In our comprehensive project analysis, we utilized Power BI to generate a detailed report on the frequency of crashes concerning their time of occurrence. This investigation sheds light on critical patterns and insights regarding when accidents are most prevalent within a given timeframe.

Key Findings:

Prevalence of Accidents: Through our Power BI analysis, we observed a consistent trend indicating that accidents occur more frequently between the hours of 9 am to 10 pm. This duration encompasses the bulk of recorded crash incidents.

Peak Hours: Within this timeframe, our data particularly highlights the period from 3 pm to 6 pm as the peak hours for accidents. This timeframe demonstrates a heightened occurrence of crashes compared to other times throughout the day.



- Analysis of Fatalities by Category and Year

In our project's comprehensive analysis facilitated by Power BI, we meticulously examined fatalities categorized by type and year, providing a comprehensive view of the trends and fluctuations in these incidents over time.

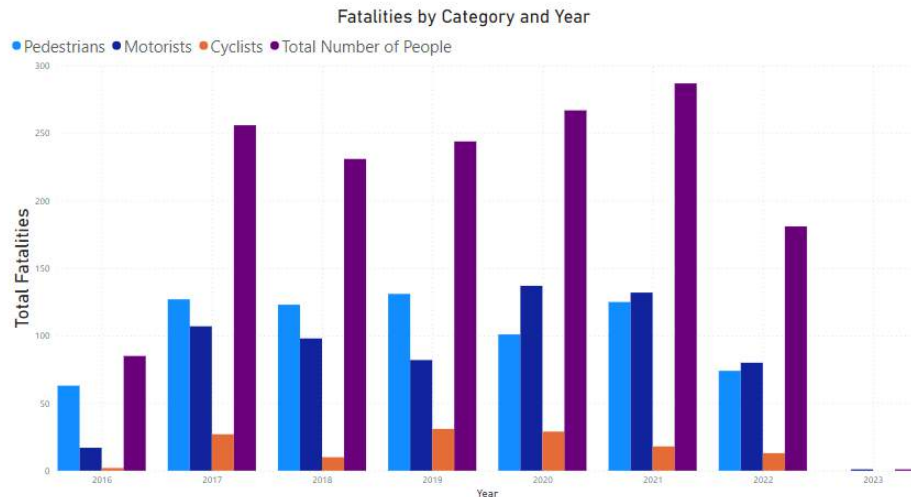
Key Insights:

Pedestrian Fatalities: Our analysis reveals a consistent trend in pedestrian fatalities, maintaining a relatively stable range between 100 to 150 fatalities annually from 2017 to 2022. This consistency highlights a persistent concern in pedestrian safety.

Motorcyclist Fatalities: The data notably points to a peak in motorcyclist fatalities during the years 2020 and 2021. This period witnessed the highest recorded incidents of motorcyclist fatalities, emphasizing a concerning rise in these specific accidents during this timeframe.

Cyclist Fatalities: Over the years under consideration, an average of 20 to 30 cyclist fatalities occurred annually. This consistent but relatively lower rate compared to other categories signifies a noteworthy yet comparatively steady occurrence of cyclist fatalities.

Overall Fatalities: The comprehensive analysis underscores 2021 as the year with the highest overall fatalities. This year witnessed an alarming surge in total fatalities across all categories, emphasizing the urgency for comprehensive safety measures and intervention strategies.



- **Analysis of Injuries by Category and Year**

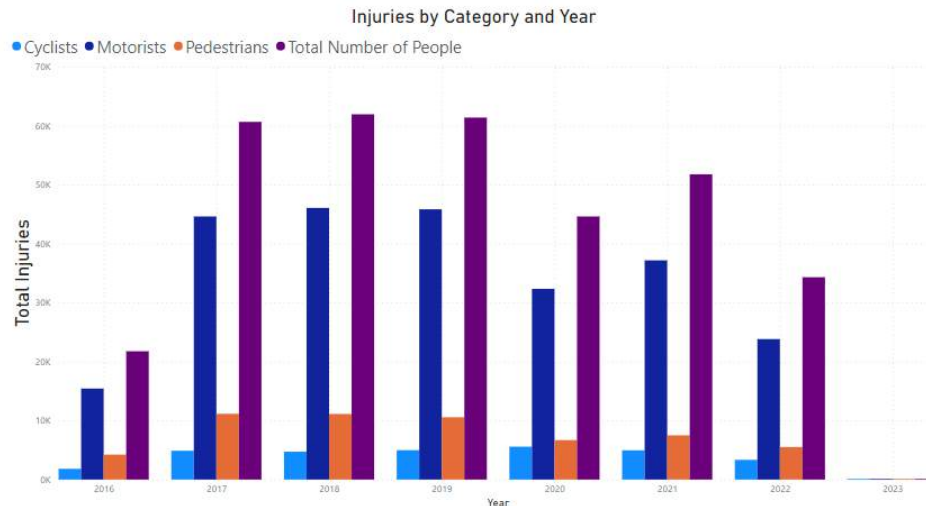
Our detailed analysis, conducted through Power BI, delved into the injuries sustained by different road user categories over the years, unveiling crucial insights into the severity and patterns of injuries.

Key Findings:

Motorcyclist Injuries: The data vividly showcases a substantial prevalence of injuries among motorcyclists compared to pedestrians and cyclists. Motorcyclists consistently experienced a notably higher rate of injuries across the years, emphasizing the vulnerability of this road user group.

Overall Injury Trends: The years 2017, 2018, and 2019 emerge as pivotal periods, witnessing the highest numbers of overall injuries. These years recorded an alarming surge in injuries, peaking at figures as high as 60,000 to 65,000 injuries, underscoring a concerning pattern of heightened risk during this timeframe.

Pedestrian and Cyclist Injuries: In contrast to motorcyclists, pedestrians and cyclists, while affected, exhibited comparatively lower rates of injuries. However, the analysis indicates that these road users are not exempt from injury risks, albeit at a lower frequency compared to motorcyclists.



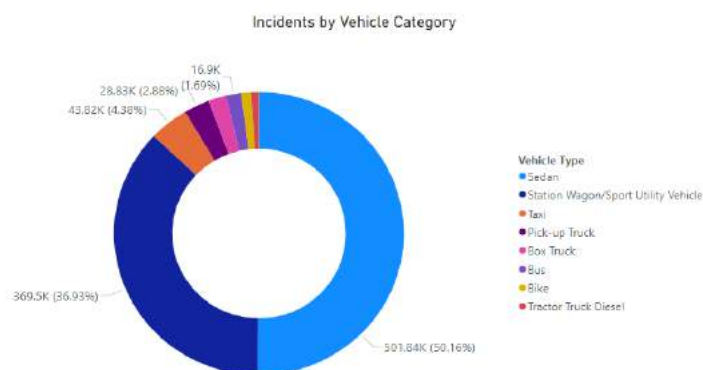
- Analysis of Incidents by Vehicle Categories

Utilizing a donut chart within Power BI, we've effectively visualized the distribution of incidents across various vehicle categories, offering a clear perspective on the involvement of different vehicle types in reported incidents.

Key Insights:

Sedan and SUV Incidents: Our analysis demonstrates a dominant involvement of sedans, constituting approximately 50% of the total incidents. Additionally, SUVs account for about 37% of the reported incidents, collectively making up a significant portion of the incidents analyzed.

Minor Categories: Beyond sedans and SUVs, smaller but notable contributions to incidents include taxis at approximately 4%, pickups at 3%, and buses at 2%. The residual percentage comprises a diverse range of vehicles such as box trucks, bikes, tractor trucks, diesel vehicles, ambulances, among others.



- Analysis of Incident Causes via Donut Chart

Our analysis, facilitated by a donut chart in Power BI, meticulously categorized incident causes, providing a comprehensive breakdown of the primary factors contributing to reported incidents.

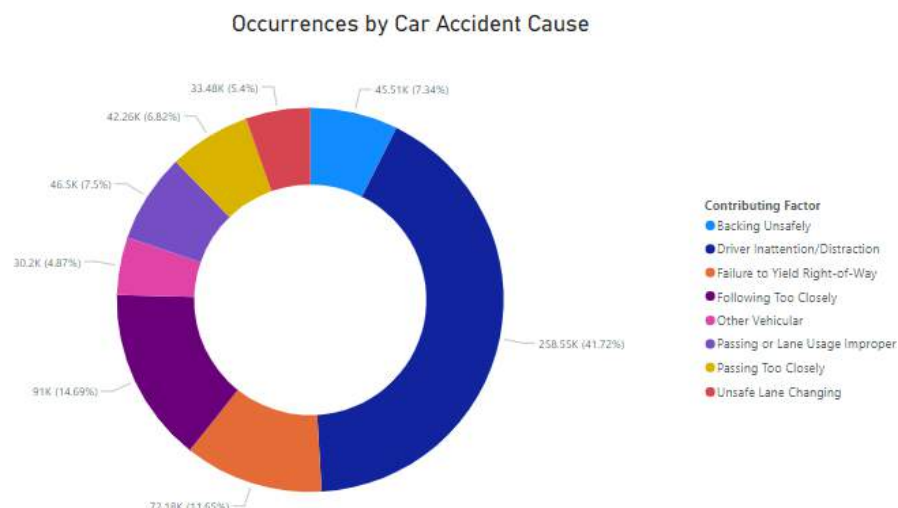
Key Findings:

Driver Inattention and Distraction: A substantial 41.72% of reported incidents stem from driver inattention and distraction, marking it as the most prevalent cause identified in our analysis. This emphasizes the critical role of attentiveness in preventing accidents.

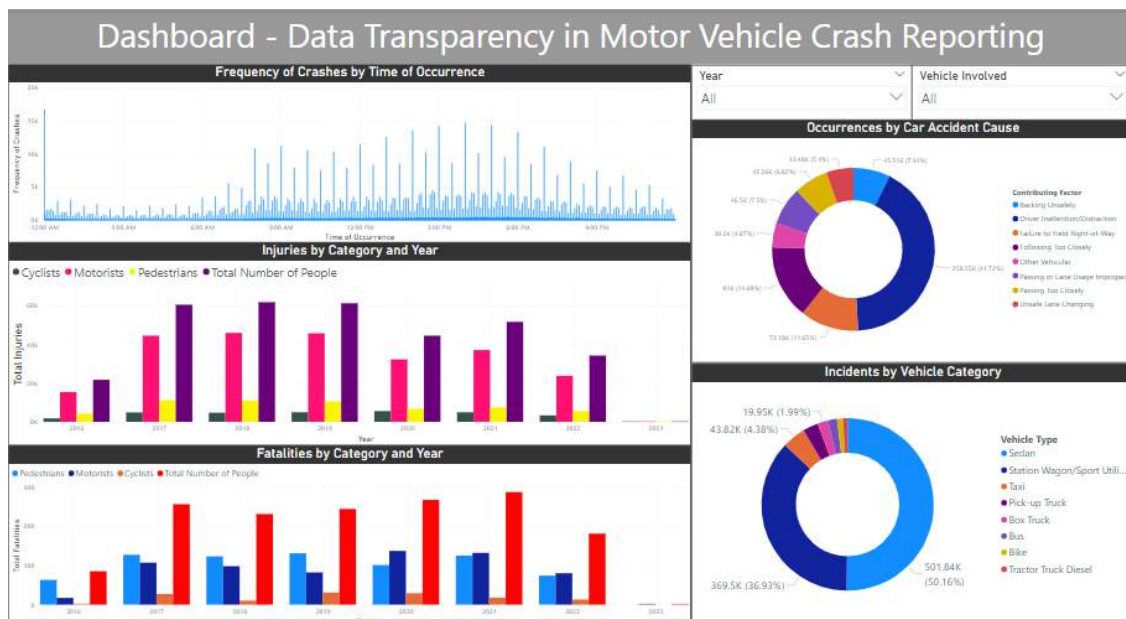
Following Too Closely: Approximately 15% of incidents result from vehicles following too closely, underscoring the significance of maintaining safe distances between vehicles to avert collisions.

Failure to Yield Right of Way: Another notable cause, contributing to 12% of incidents, is the failure to yield the right of way. This highlights the importance of adhering to traffic regulations to prevent accidents.

Other Significant Causes: Our analysis also identifies various significant causes contributing to incidents, including improper lane usage, passing too closely, and unsafe lane changing. While individually these causes may represent smaller percentages, collectively they constitute a substantial portion of incident triggers.



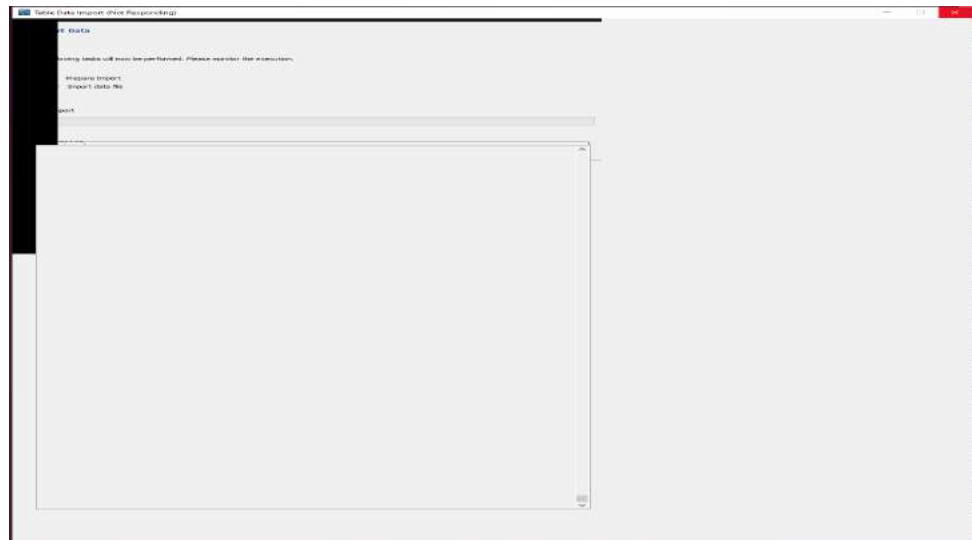
Our Power BI dashboard amalgamates five key visualizations capturing critical road safety trends. It delineates peak accident hours, stable pedestrian fatalities but alarming spikes in motorcyclist incidents, particularly in 2020-2021, and an overall surge in 2021 fatalities. Sedans and SUVs dominate incidents, alongside significant but smaller involvements from taxis, pickups, and buses. Primary causes, notably driver inattention and tailgating, are highlighted. Key performance indicators for Year and Vehicle Involved facilitate strategic decision-making, empowering targeted safety interventions for a safer road environment.



VIII. TECHNICAL DIFFICULTY

1. While using RDBMS

The aim of this project at the first was to use MySQL for the first part of the project but while importing the data to MySQL the python was showing the error and then we tried to import the data using workbench but at that point also MySQL was crashing and showing error for the 4 million of data. So, there was a change in the plan and then we needed to go for the postgres PgAdmin and the results from that were so great that it was able to import the data within seconds so, the first part was later done using Postgre PgAdmin.



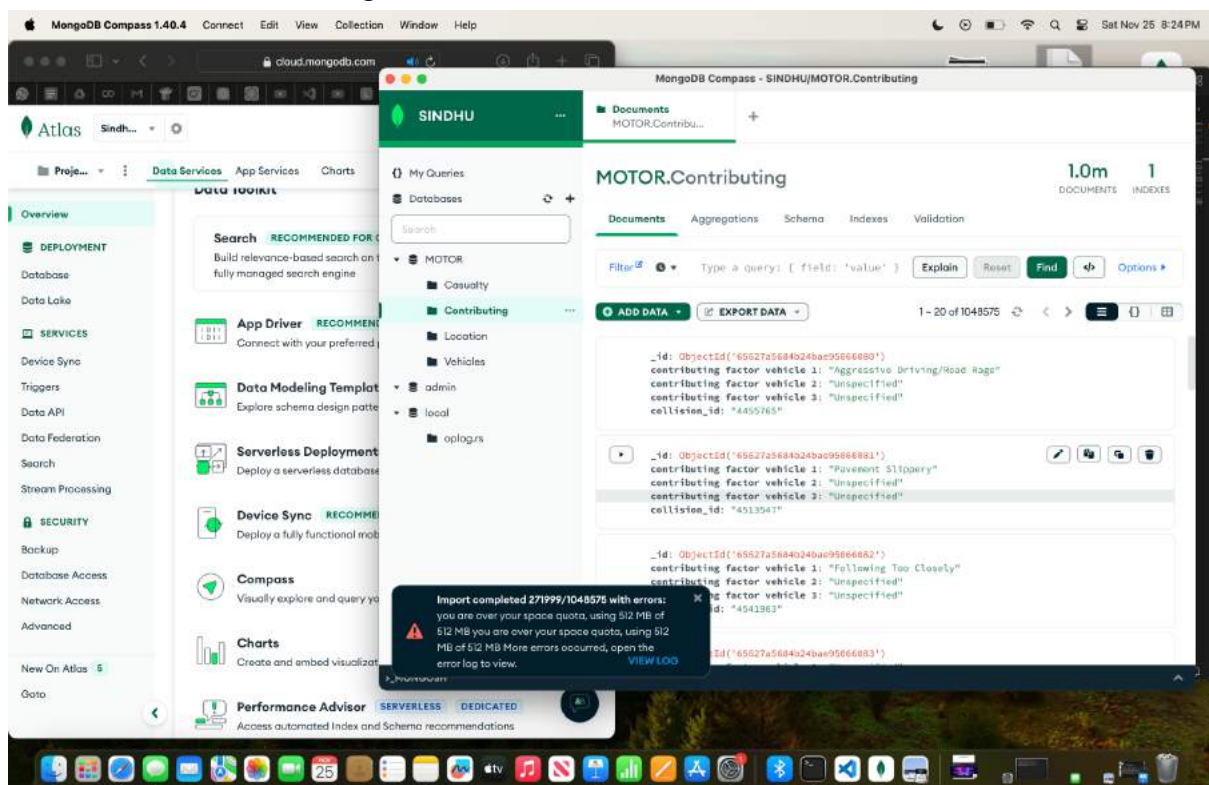
Importing to MySQL resulted in crashing of the workbench.

2. While Using MongoDB

Faced challenges in identifying the optimal set of indexes based on query patterns.

The working set (the portion of our data and indexes that fit in RAM) exceeded available RAM, it resulted in increased disk I/O and decreased performance.

Sharding in MONGODB was not available in the free version which made us face many issues as our data set has huge collections.



3. While using AWS environment

→ S3

Problem: Data access problems may arise from erroneous or inadequate access permissions. Problems with data consistency may arise from overwrites or incomplete uploads. Performance can be affected by high latency or slow data transport.

→ Glue

Data problems or improper setups might cause glue tasks to fail. Issues may arise from incomplete or inaccurate metadata in the Amazon Glue Data Catalog. Resource limitations or sluggish work performance. Examine the logs, confirm the data type and schema, and check the task settings. Configurations should be changed as necessary.

→ Athena

Use relevant data formats (like Parquet), partition your data, and optimize your searches. To assess and enhance query performance, think about utilizing Athena query execution strategies. Next, main challenge was the connection of Athena with Python so, it so, difficult to solve that problem because libraries and the establishing the connection between them and if that completed the problem we faced was how to pass queries from python too athena and after that the main problem was how to find the correct function to convert the data to dataframe.

IX. KEY LEARNINGS

- This project covers key data management aspects, exploring both normalization and denormalization techniques for database optimization. Python is utilized for its versatility in data manipulation, analysis, and scripting, providing a powerful tool for handling diverse data scenarios.
- For effective data visualization, the project delves into choosing the right visualization type based on the nature of the data. It leverages popular Python libraries like Matplotlib, Seaborn, and explores tools like Tableau.
- The study extends to NoSQL concepts, particularly focusing on document-oriented databases like MongoDB. It covers CRUD operations in MongoDB, delves into the Aggregation Framework for complex queries, and addresses storing and retrieving data in AWS S3 buckets with access controls and permissions, integrating seamlessly with other AWS services.
- Further, the project explores data crawling and cataloging in various formats, creating ETL jobs for data transformation, and implementing serverless ETL workflows using AWS Glue. It emphasizes querying data in Amazon S3 using standard SQL, managing named queries, and understanding query performance considerations.
- Additionally, the project involves the installation and configuration of PostgreSQL, covering SQL queries and data manipulation. It delves into advanced features such as stored procedures and triggers, providing a holistic understanding of database management and optimization.

X. CREDIT TAXONOMY

SJSU ID	NAME	RESPONSIBILITIES
017416737	Prayag Nikul Purani	Data Preprocessing, Athena Glue, Query Analysis, Data Visualization, Intermediate Report Draft, PPT preparation
016980899	Arjun Amarnath Rai	Data Import, Intermediate Report Draft, Pre-analysis, Final Data Analysis and Data Visualization
017428619	Syed Faraaz Ahmed	Data Cleaning, AWS, IAM connections, Visualization, Intermediate Draft Report, PPT preparation
017419987	Sindhu Nagesha	Intermediate Report draft, PPT preparation, MongoDB CRUD operations, aggregate pipelines, Query Analysis, visualization and Pymongo execution.

XI. CONCLUSION

This project leverages various technical tools to conduct an in-depth analysis of motor vehicle crash data. By utilizing SQL, Python, PostgreSQL, MongoDB, AWS services, and more, the authors are able to dive deep into the intricacies of these incidents. Ultimately, the goal extends far beyond data science to drive awareness, inform urban planning, guide policy, and underscore the connection between crashes and peak transit times. At its core, this project aims to revolutionize how we view crash data - transforming raw numbers into actionable insights that can empower stakeholders to improve public safety. The real-world implications of comprehensively understanding crash data simply cannot be overstated. Accurate reporting lays the foundation for data-driven policymaking, enforcement, emergency response optimization, and more. By gleaning crucial insights from diligent analysis, targeted and effective safety initiatives can be implemented. This project offers an inspiring case study of how technical prowess and social impact can synergize. It highlights the critical need for precise crash data collection and underscores how multi-stakeholder collaboration is key to leveraging these insights to prevent and mitigate motor vehicle collisions. This impactful undertaking serves as a springboard for ongoing work to enhance road safety through improved data gathering, strategic planning, and technological innovation. The fusion of state-of-the-art tools and prosocial objectives provides a model for future efforts aimed at fostering a safer transit environment for all.

XII. REFERENCES

<https://www.sciencedirect.com/science/article/abs/pii/S0001457519308735>
<https://www.sciencedirect.com/science/article/abs/pii/S0001457508001954>
<https://www.witpress.com/Secure/elibrary/papers/SDP18/SDP18064FU1.pdf>
<https://dev.mysql.com/doc/connector-python/en/connector-python-example-connecting.html>
<https://stackoverflow.com/questions/372885/how-do-i-connect-to-a-mysql-database-in-python>
https://www.w3schools.com/python/python_mysql_getstarted.asp
https://www.tutorialspoint.com/python_data_access/python_postgresql_database_connection.htm
<https://www.postgresqltutorial.com/postgresql-python/connect/>
<https://www.datacamp.com/tutorial/tutorial-postgresql-python>
<https://www.dataquest.io/blog/tutorial-connect-install-and-query-postgresql-in-python/>
https://youtu.be/gFWu-SSzRzc?si=9QKcZgFt6OnJ8CK_
https://youtu.be/tW6g_WbwAl0?si=TegHxknawvC6hEUz
<https://youtu.be/qBocgdMGEWs?si=T7IO3aGpm0fk8Y3U>

XIII. LINKS TO FILES

<https://drive.google.com/drive/folders/0ALIZY42yBRCxUk9PVA>
https://www.canva.com/design/DAF1Oov74_w/lVomvbwN9WXEcmHoBX1Oxg/view?utm_content=DAF1Oov74_w&utm_campaign=share_your_design&utm_medium=link&utm_source=shareyourdesignpanel

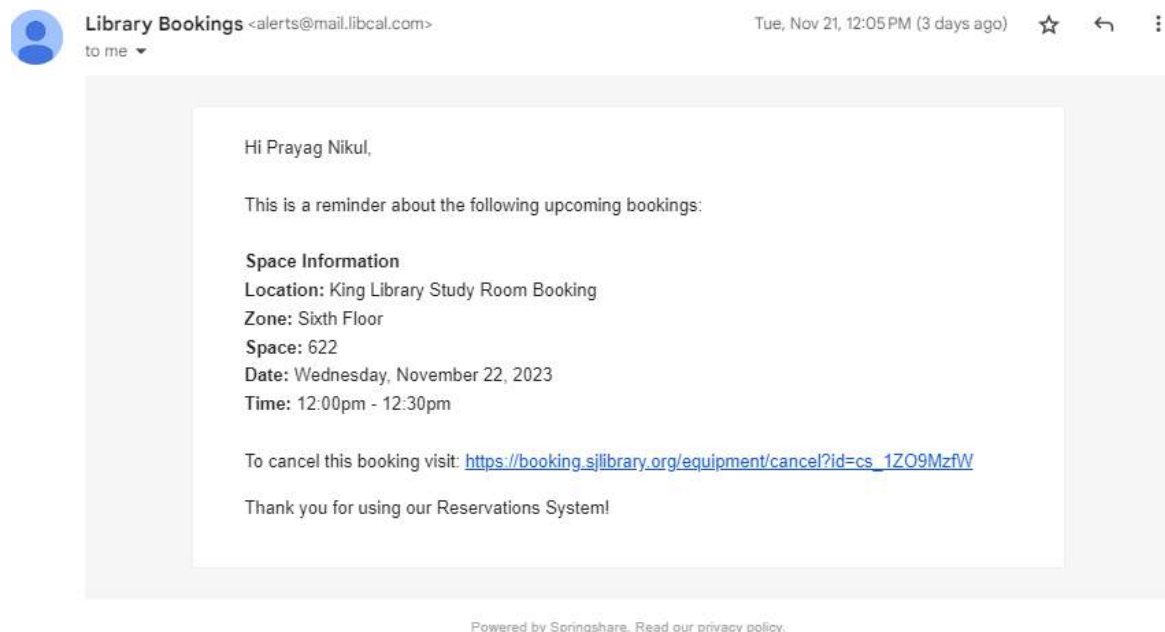
XIV. APPENDIX

Criteria	Pts	Explanation
This criterion is linked to a Learning Outcome Presentation Skills Includes time management	5 pts	
This criterion is linked to a Learning Outcome Code Walkthrough	3 pts	Connectors, NOSQL, Datawarehouse, ETL tool
This criterion is linked to a Learning Outcome Discussion / Q&A	4 pts	
This criterion is linked to a Learning Outcome Demo	3 pts	
This criterion is linked to a Learning Outcome Version Control	3 pts	We have added the links to all the in the above section

Use of Git / GitHub or equivalent; must be publicly accessible		
This criterion is linked to a Learning Outcome Significance to the real world	5 pts	Implemented all the tools and technologies used in Industry
This criterion is linked to a Learning Outcome Lessons learned Included in the report and presentation? How substantial and unique are they?	5 pts	The connection of python with Athena is the unique this project
This criterion is linked to a Learning Outcome Innovation	5 pts	The innovation of the project is that how to reduce the casualty using the data
This criterion is linked to a Learning Outcome Teamwork	5 pts	We have attached table telling the work distribution of the project
This criterion is linked to a Learning Outcome Technical difficulty	4 pts	This section is mentioned in above section of the report
This criterion is linked to a Learning Outcome Practiced pair programming? See: https://en.wikipedia.org/wiki/Pair_programming Links to an external site.	2 pts	This part is well covered in the references section of the report
This criterion is linked to a Learning Outcome Practiced agile / scrum (1-week sprints)? Submit evidence on Canvas - meeting minutes, other artifacts	3 pts	Weekly follow ups and meetings schedules to track updates, backlogs. Screenshots have been attached for the same.
This criterion is linked to a Learning Outcome Used Grammarly / other tools for language? Grammarly free version is sufficient; can use other tools as well. Submit report screenshot on Canvas.	2 pts	For formal language processing we have utilized grammarly as a chrome extension.
This criterion is linked to a Learning Outcome Slides	5 pts	

<p>This criterion is linked to a Learning Outcome Report</p> <p>Format, completeness, language, plagiarism, whether turnItIn could process it (no unnecessary screenshots), etc</p>	7 pts	We have aimed for a clean Report, emphasizing format, indentation, precise language, referencing, thoroughness, and originality while avoiding plagiarism.
<p>This criterion is linked to a Learning Outcome Used unique tools</p> <p>E.g.: LaTeX for writing report (submit .tex that is not generated from another format such as .docx; generating from .lyx and similar LaTeX editor outputs is fine. Also checkout https://www.overleaf.com/LinksLinksLinks Links to an external site.</p> <p>Links to an external site. to an external site. to an external site. to an external site.)</p> <p>Unique features of Prezi or powerpoint, etc</p>	5 pts	
<p>This criterion is linked to a Learning Outcome Performed substantial analysis using database techniques</p> <p>Project must include an analytics component</p>	3 pts	We utilized SQL queries to extract, transform, and analyze database data. Power BI dashboards visually presented insightful project analyses.
<p>This criterion is linked to a Learning Outcome Used a new database or data warehouse tool not covered in the HW or class</p>	3 pts	We have used Postgre and Athena and its connector
<p>This criterion is linked to a Learning Outcome Used appropriate data modeling techniques</p>	5 pts	We tried your best to cover all the section if the course
<p>This criterion is linked to a Learning Outcome Used ETL tool</p>	1 pts	How important is it so make your work easy
<p>This criterion is linked to a Learning Outcome Demonstrated how Analytics support business decisions</p>	3 pts	Glue for ETL
<p>This criterion is linked to a Learning Outcome Used RDBMS</p> <p>Idea is to exercise as many topics from the course as possible</p>	1 pts	MongoDB part helped us to learn about stage clustering and other things

This criterion is linked to a Learning Outcome Used Data Warehouse Idea is to exercise as many topics from the course as possible	1 pts	We have used AWS environment like S3, Glue, Athena to learn new things
This criterion is linked to a Learning Outcome Includes DB Connectivity / API calls Possibly using Python	1 pts	We have used connector for MySQL and Athena
This criterion is linked to a Learning Outcome Used NOSQL	1 pts	We have used MongoDB as a part of your project



Library room booking details

You are invited to a Zoom meeting now.

Join from PC, Mac, Linux, iOS or Android: [https://sjsu.zoom.us/j/81172996563?](https://sjsu.zoom.us/j/81172996563?pwd=SFd2L1IHdzBLNUZzZWovcTI6Q2JEQT09)
pwd=SFd2L1IHdzBLNUZzZWovcTI6Q2JEQT09
Password: 980659

Or iPhone one-tap:

US: +16699006833,,81172996563# or +16694449171,,81172996563#

Or Telephone:

Dial(for higher quality, dial a number based on your current location) :

US: +1 669 900 6833 or +1 669 444 9171 or +1 253 215 8782 or +1 346 248 7799
or +1 719 359 4580 or +1 253 205 0468 or +1 305 224 1968 or +1 309 205 3325 or +1
312 626 6799 or +1 360 209 5623 or +1 386 347 5053 or +1 507 473 4847 or +1 564
217 2000 or +1 646 876 9923 or +1 646 931 3860 or +1 689 278 1000 or +1 301 715
8592

Meeting ID: 811 7299 6563

International numbers available: <https://sjsu.zoom.us/j/kcKOyBlc7r>

01:38 PM ✓

Wednesday, November 21 · 1:30 – 2:30pm

Time zone: America/Los_Angeles

Google Meet joining info

Video call link: <https://meet.google.com/rde-fawu-tkk>

10:37 PM ✓

