TRIP DATA ANALYSIS

**Topics covered:**

1. **Mongo DB Installation**
2. **Python Packages Installation:**
3. **Postman**
4. **Performance metrics**

**Installation Steps:**

**Mongo DB Installation:**

1. Install Mongo DB Community edition. Use below link.

<https://www.mongodb.com/try/download/community>

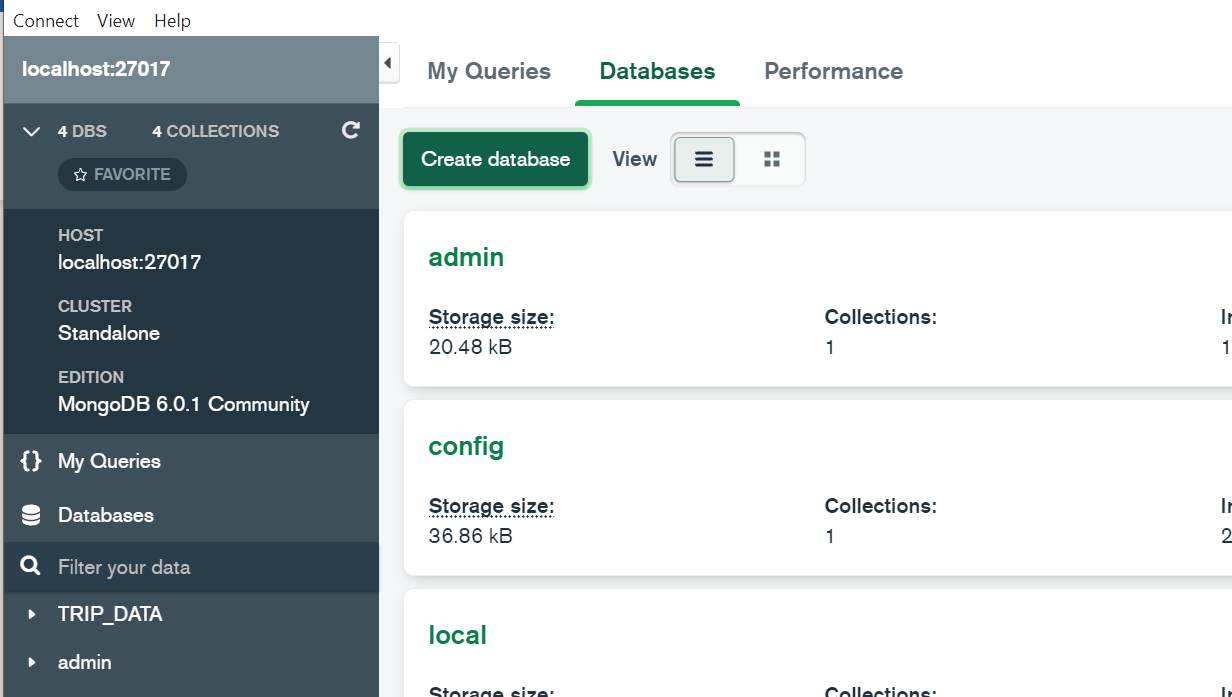
1. MongoDBCompass will be installed as part of step 1. Launch Mongo Compass and use it.

Here MongoDBCopass is used to interact with MongoDB and query development

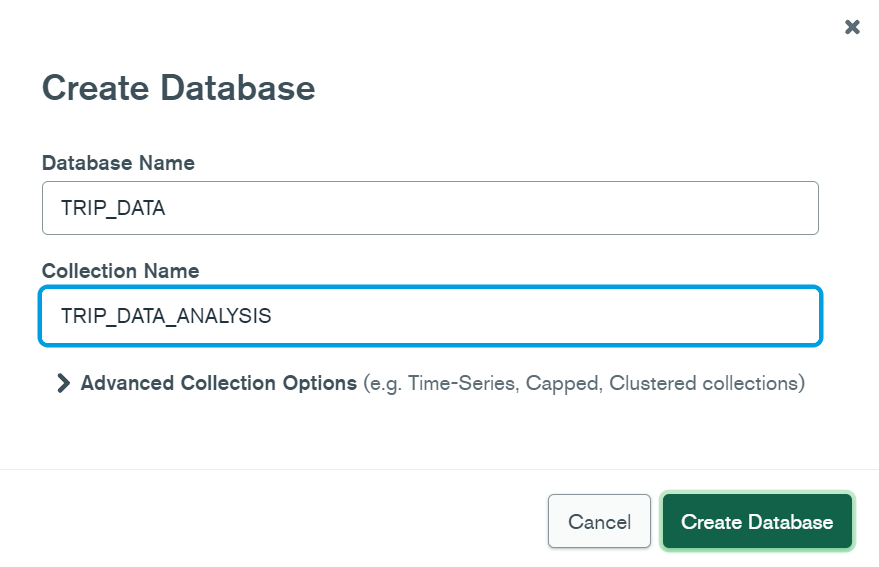
Connection string to connect to locally hosted MongoDB - mongodb://localhost:27017

1. Once connection is established create a database. Click on create database and collection

Screenshot (1)



Screenshot (2)



1. Once database & collection is created, insert data to collection.

Click ADD DATA. Then Import File. Browse trip\_data.csv file from local and import it.

Please refer below screenshot

**Note:** Please make sure to change datatype of tpep\_pickup\_datetime to DATE, tip\_amount to double, total\_amount to DOUBLE datatype. By default everything will be loaded as string datatype

Graphical user interface, text, application, email

Description automatically generated

**Python Packages Installation:**

1. Download TripData\_Analysis folder.
2. cd TripData\_Analysis. pip install -r requirements.txt
3. Once package are installed run trip\_data\_analysis.py –
4. Now Navigate to code folder. cd code
5. python trip\_data\_analysis.py

This is the main script. Once script is launched successfully you will see below output.

Text

Description automatically generated

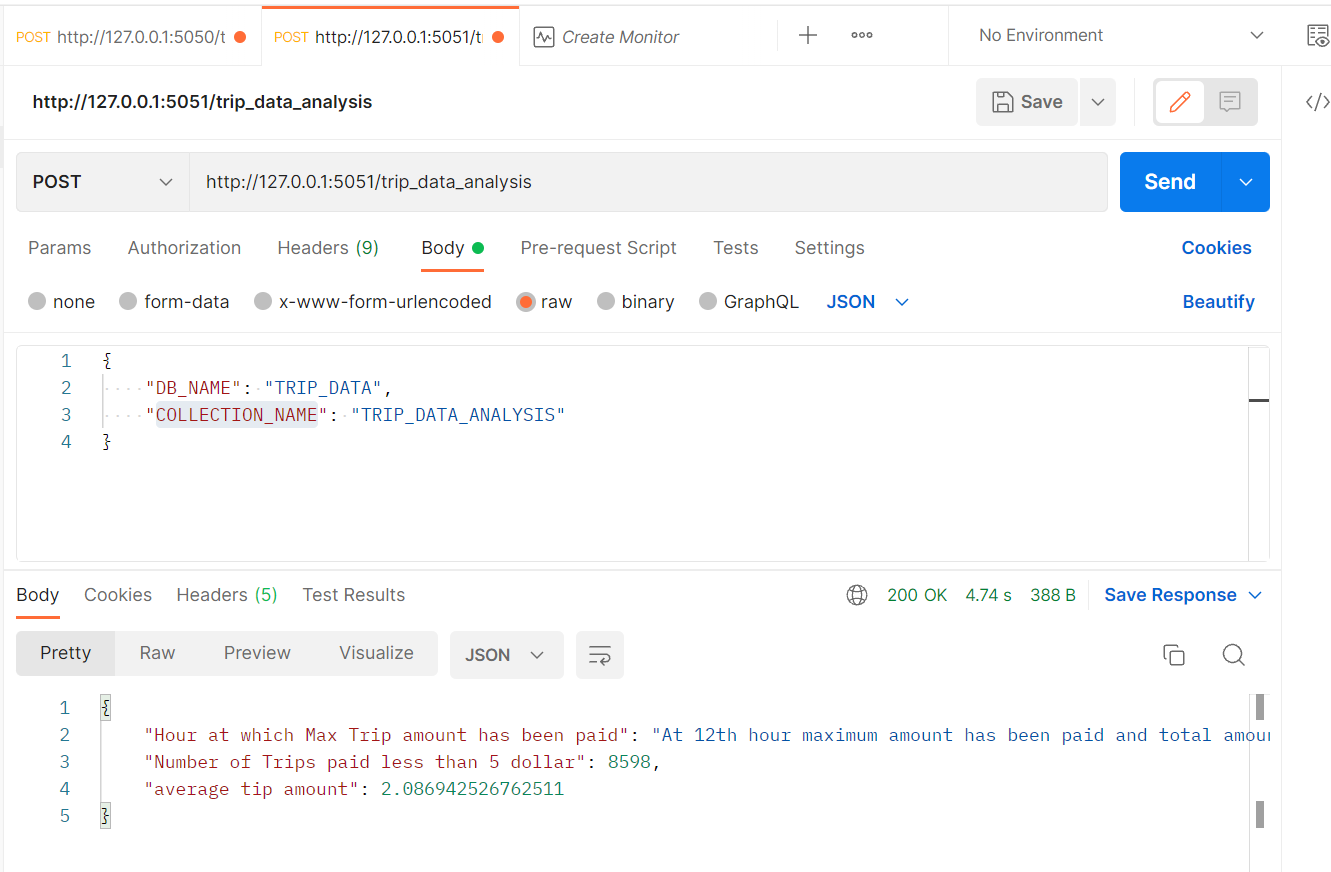
1. Now application is up and running in port 5051 and it is hosted in local

**Postman:**

In postman use below curl command to hit our application.

curl --location --request POST 'http://127.0.0.1:5051/trip\_data\_analysis' \  
--header 'Content-Type: application/json' \  
--data-raw '{  
 "DB\_NAME": "TRIP\_DATA",  
 "COLLECTION\_NAME": "TRIP\_DATA\_ANALYSIS"  
}'

Below is the postman response



**Performance Metrics:**

Approximately it takes 3.4seconds to query 1million documents and retrieve below response.

|  |  |
| --- | --- |
| Run | Time Taken |
| 1st Run | 4.09 |
| 2nd Run | 2.94 |
| 3rd Run | 4.40 |
| 4th Run | 2.94 |
| 5th Run | 2.91 |

On an average it takes 3.4 seconds to get response