Team -06: DATA ENCODERS

Version-01

Dataset from the Website

https://m.cricbuzz.com/

Excel Sheet

(For importing the data)

Unstructured

daa

Python for further conversions in the dataset to visualize

Power BI for visualization

Structured data

SQL or Excel (yet to be decided)

**Work Division:**

1. Implementation of Python : Raghunandan
2. Data prep : Deepthi and Nikitha Kethireddy
3. Visualization : Jeevan , Dheeraj

Version-02

Using a package called (pycricbuzz) for generating the live data

A new file with data is generated in the form of JSON file.

We are using python to append the data to the generated file every time the data is updated in the website

Whenever we update the file in power BI, the graphs/bars etc will be updated dynamically.

We use that generated file in Power BI and we have an option to update that file by setting time

**Work Division:**

1. Implementation of Python : Raghunandan
2. Data prep : Deepthi and Nikitha Kethireddy
3. Visualization : Jeevan , Dheeraj

Version-03

Using a package called (pycricbuzz) for generating the live data

We are using python to get the data onto the local system every time the data is updated in the website.

Data is appended to MongoDB as we get live score from the package.

Reports in Power BI will be refreshed as we update the data in MongoDB

We connect to MongoDB to fetch the data and will use Power BI to generate the graphs

**Work Division:**

1. Implementation of Python: Raghunandan
2. Data prep: Deepthi and Nikitha Kethireddy
3. Visualization: Jeevan
4. Commentary Sentiment Analysis: Dheeraj

Version-04

Using a package called (pycricbuzz) for generating the live data

We are using python to get the data onto the local system every time the data is updated in the website.

Data will be in format of JSON and this will be sent to KAFKA

JSON

Fetch from KAFKA

Reports in Power BI will be refreshed as we update the data in KAFKA

We connect to KAFKA to Power BI

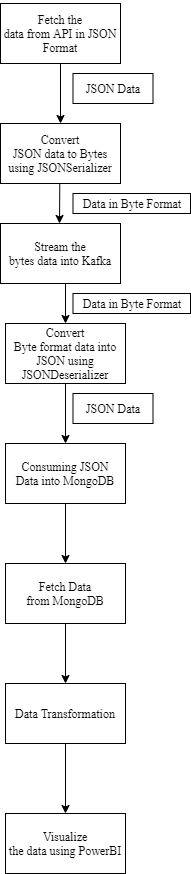
Reporting

**Work Division:**

1. Implementation of Python: Raghunandan
2. Data prep: Deepthi and Nikitha Kethireddy
3. Visualization: Jeevan
4. Commentary Sentiment Analysis: Dheeraj

**Version-05**

**Score Estimation:**



**Data Source:** We are fetching the data from the API provided by [www.crickbuzz.com](http://www.crickbuzz.com) to perform data visualization.

**Steps for Data Extraction:**

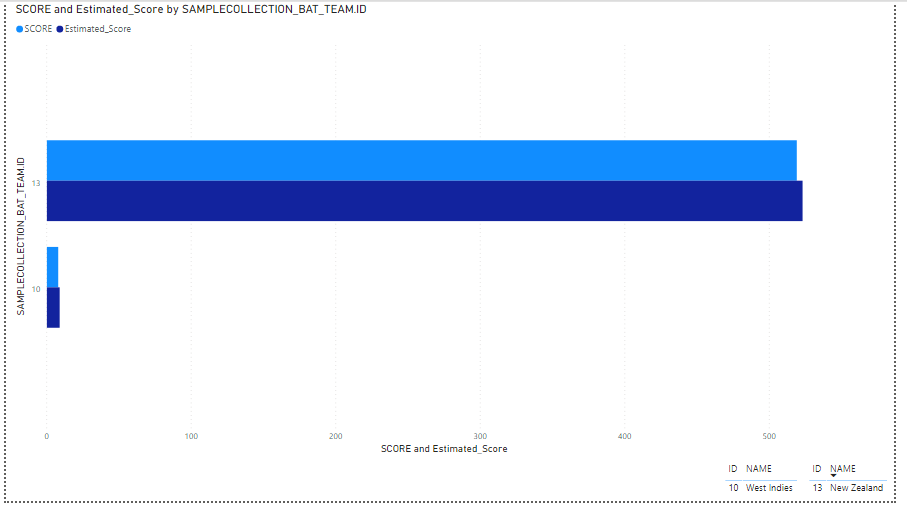
* Run zookeeper and kafka by using the following commands:
  + .\bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties
  + .\bin\windows\kafka-server-start.bat .\config\server.properties
* Run local MongoDB server and connect it PowerBI using JDBC connections.
* Build the project using *mvn clean compile assembly:single* command
* Run the producer and consumer by using any Java IDE using the following commands:
  + java -cp target/KafkaLiveScoreStream-1.0-SNAPSHOT-jar-with-dependencies.jar edu/nwmissouri/KafkaLiveScoreStream/LiveScoreKafkaProducer
  + java -cp target/KafkaLiveScoreStream-1.0-SNAPSHOT-jar-with-dependencies.jar edu/nwmissouri/KafkaLiveScoreStream/LiveScoreKafkaConsumer
* Here the producer gets the data from the API of the data source as mentioned above by using Java.
* The reason for using java is because of Serialization and Deserialization of the JSON data is easy.
* Consumer consumes the data by using Java and stores the end-result using MongoDB.
* Reason for using MongoDB is that, it’s used to store the streamed data and also it acts as a middleware between Kafka and PowerBI.

**Data Analyzing and transformation:**

* Establishment connection from MongoDB to Power BI
* Click on transform data option. A new tab will be opened wherein we will perform the following operations:
  + We need to perform group-by operation upon the unique id which is generated for every individual teams.
  + We should be finding most recent overs and recent score by calculating the maximum of overs and maximum score so as to obtain the most recent overs and runs scored by each team at that point of time.
  + Using score and total number of overs played, we will be run rate of the team. This is done by dividing score with overs. The value thus obtained will be rounded-up.
* By adding the run rate to total score we can obtain the estimated score.
* The estimated score thus obtained will be used to visualize the data accordingly.

**Data Visualizing:**

We will be visualizing the data using Power BI.

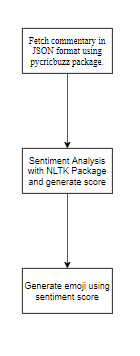


The image shown above is the final visualized output for the streaming data for a cricket match.\

**Work Division:**

* Implementation of Kafka and Data Extraction: Raghunandan Kumar Naishadam
* Connecting MongoDB to PowerBI : Nikitha Kethireddy
* Data Transformations: Deepthi Chokka
* Data Visualization: Jeevan

**Commentary Sentiment Analysis:**



**Work Division:**

* Data Extraction: Raghunandan Kumar Naishadam
* Analysis of Commentary: Dheeraj

**Data Source:** We are fetching the data from the API provided by [www.crickbuzz.com](http://www.crickbuzz.com) to perform data visualization.

**Data Extraction:**

Commentary data will be extracted using a python package called “pycricbuzz”

**Data Analyzing:**

* Data Analysis is done by using a python package called as NLTK which results in score.
* Based on the score emojis will be generated.

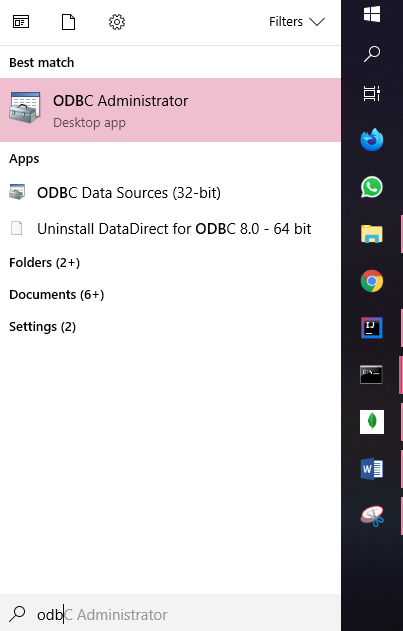
**Data Visualization:**

The data thus obtained will be visualized by using python programming.

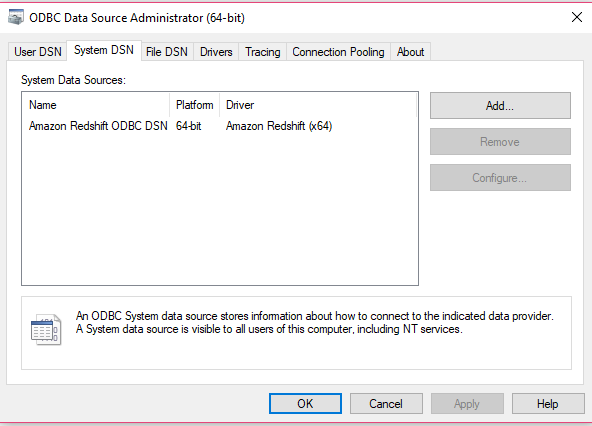
**Steps for Data Connection for Score Estimation:**

Prerequisites:

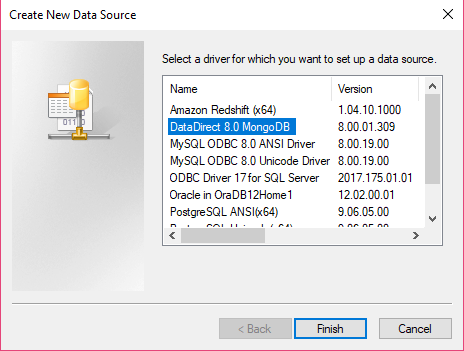
* We need to make sure to download MongoDB, PowerBI and ProgressDataDirect.
* After running IntelliJ commands, we need to continue the following steps:
* We need to type ODBC Administrator In search box and select ODBC Administrator option(32 or 64 bit depending upon the system)



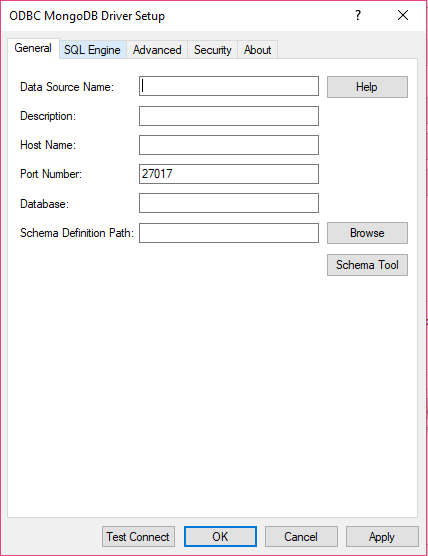
* **Step2 :** After selecting ODBC data source administrator, a pop up will be opened where we need to select System DSN and choose the option called ADD



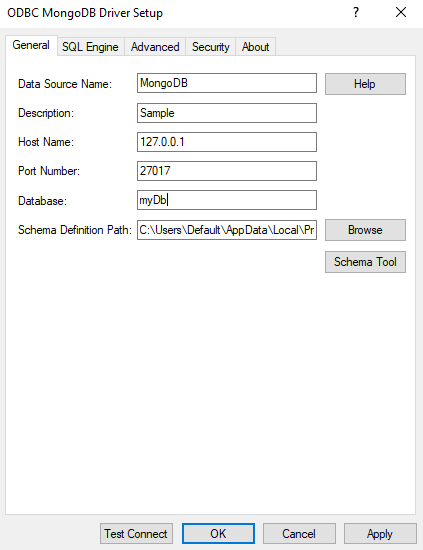
* **Step 3:** When we click ADD button, a pop will be opened where we need to select the option called Data Direct 8.0 MongoDB and select the option called FINISH



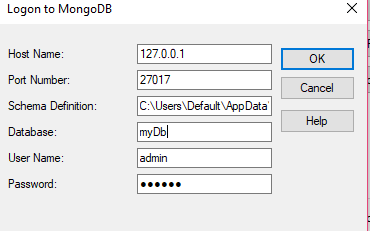
* **Step 4:** Once you click finish, a pop will be opened with the following details



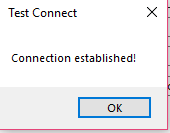
* **Step 5:** We need to fill all the details depending upon your requirement and we need to choose the option called TEST CONNECT and a popup will be opened. In our case, I have used myDB because it is the data which we get when we run our Kafka Commands



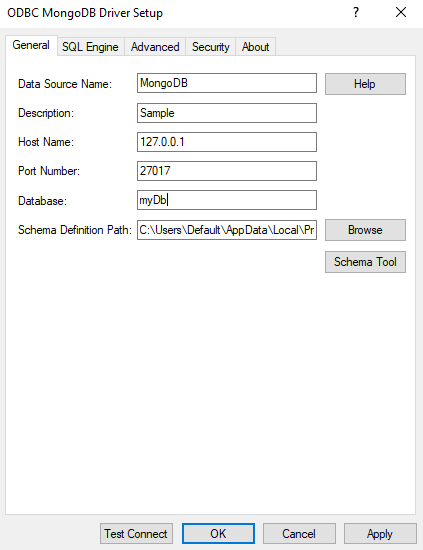
* **Step 6:** Once the pop up is opened we need to enter the user name and password related to our MongoDB database



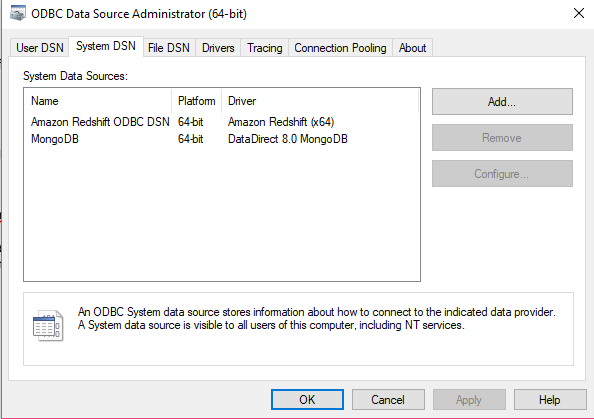
* **Step 7:** Once we press the OK option, if the credentials and database details are correct, we will get an alert that a connection is established, we need to click OK button.



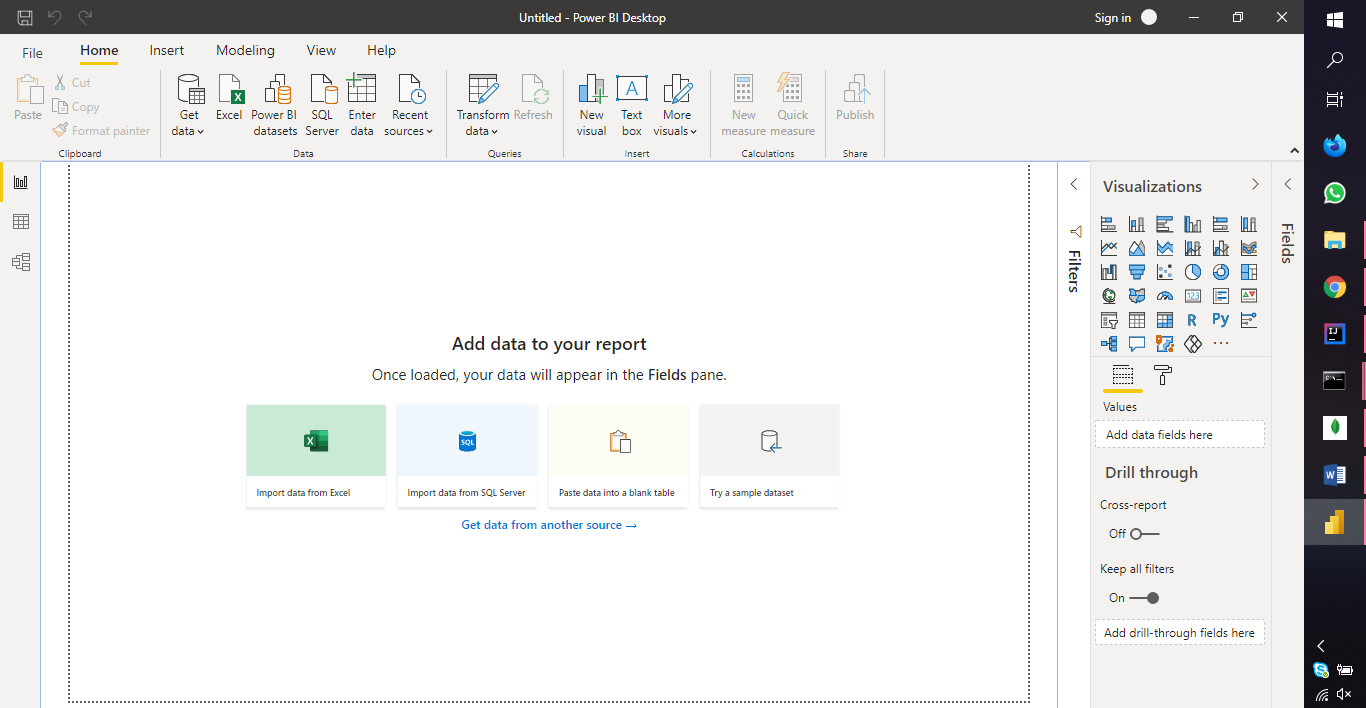
* **Step 8:** Once the connection is established, we need to make sure to click on APPLY and OK buttons in order add the DSN to our System DSN



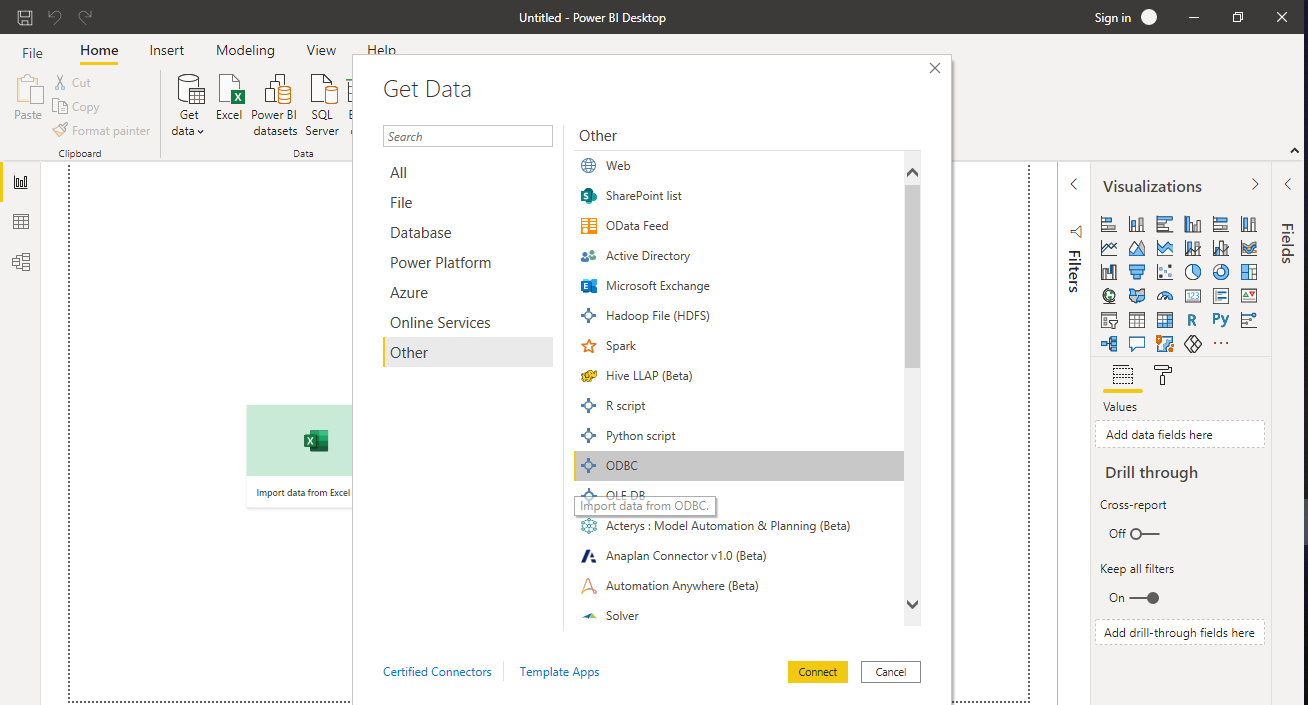
* **Step 9:** Once everything is done, the DSN which we added is displayed in the System DSN of the ODBC Data Source Administrator.



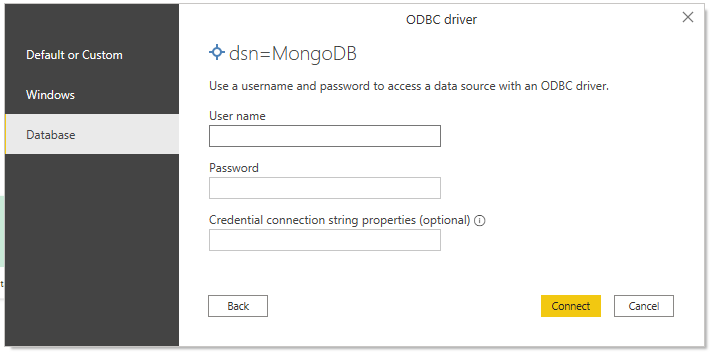
* **Step 10:** Once we are done, we can now open PowerBI desktop and choose the option called GETDATA in order to get the data from MongoDb.



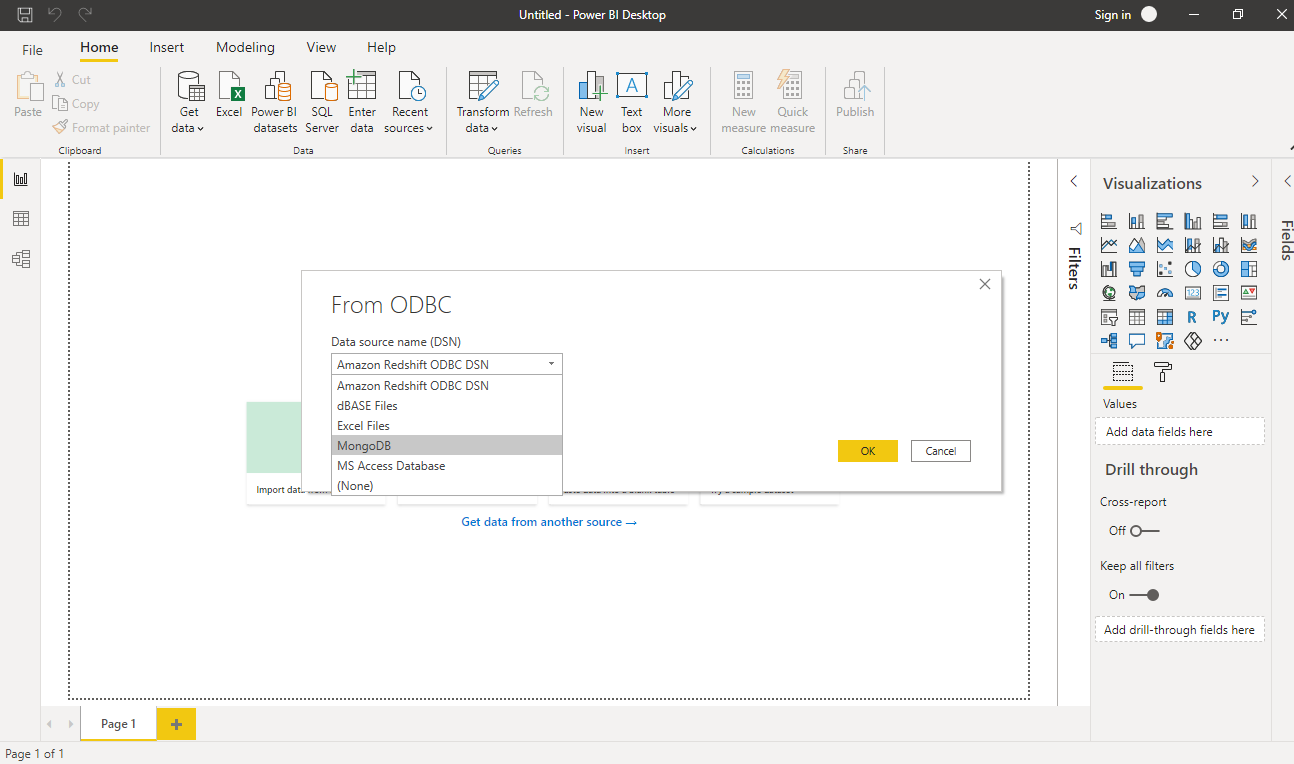
* **Step 11:** When we choose the option GETDATA, the following options are displayed and we need to select the option OTHER and ODBC and we need to press CONNECT button



* **Step 12:** Once we click on CONNECT button, an alert dialog is displayed where we need to enter our username and password of MongoDB and click on CONNECT option



* **Step 13:** Once we connect, another pop up will be displayed where we need to select the Data source name which we created in the early steps and we need to click OK



* **Step 14:** Once we are connected, the tables present in the MongoDB are displayed as in the following Screenshot, later on we need to press the option called LOAD in order to LOAD the data.

