## 

Capstone Project Report

01.11.2021

**─**

Group Name: MyKingdom

Group Members: N Venkata Surya Bharadwaj, Geethanadh Sunkara

# Overview of the Project

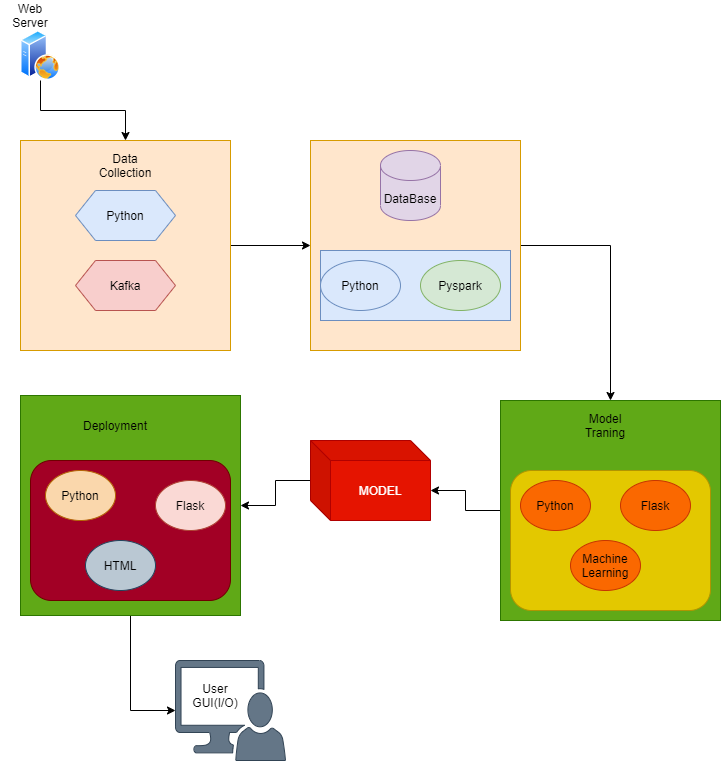
Design a system which can classify incoming news articles and appropriately tag

the corresponding category and understanding of all software engineering of a machine learning project lifecycle such as Data streaming, data cleansing/pre-processing, preparing dataset for model training and finally use the model and building application to classify.

Tools/programming languages used in the Project :

1. Python3
2. Rest API’s
3. Kafka (Zookeeper, producer, consumer/sink)
4. MongoDB
5. PySpark(Spark Mongo Connector)
6. Jupyter
7. Flask API
8. VS Code/Pycharm IDE
9. Libraries ( NLTK, SKlearn,Pandas)
10. Pickle
11. HTML
12. CSS
13. JavaScript
14. Docker
15. Docker\_Compose

# Architecture diagram of the Project :



**Week 1:**

For Week 1: Data Ingestion

Project Name: Data-Ingestion Service

* We are Supposed to Create the Entire Plan and Architecture of the project.
* The objective of this project is to source new data to re-train the model.

# Goals for Week 1 Milestone

1. Extracting Data from News feed API (Rapid news API) by using Python and passing it through Apache Kafka pipeline (Producer, Broker (Zookeeper, Topic), Consumer/Sink)
2. Finally, data is dumped to sink Database here we used mongo db.

# Environment Details required/Used:

1. OS with Zookeeper instance configured
2. Kafka instance configured
3. MongoSinkConnector properties
4. MongoDB installed and configured to localhost.

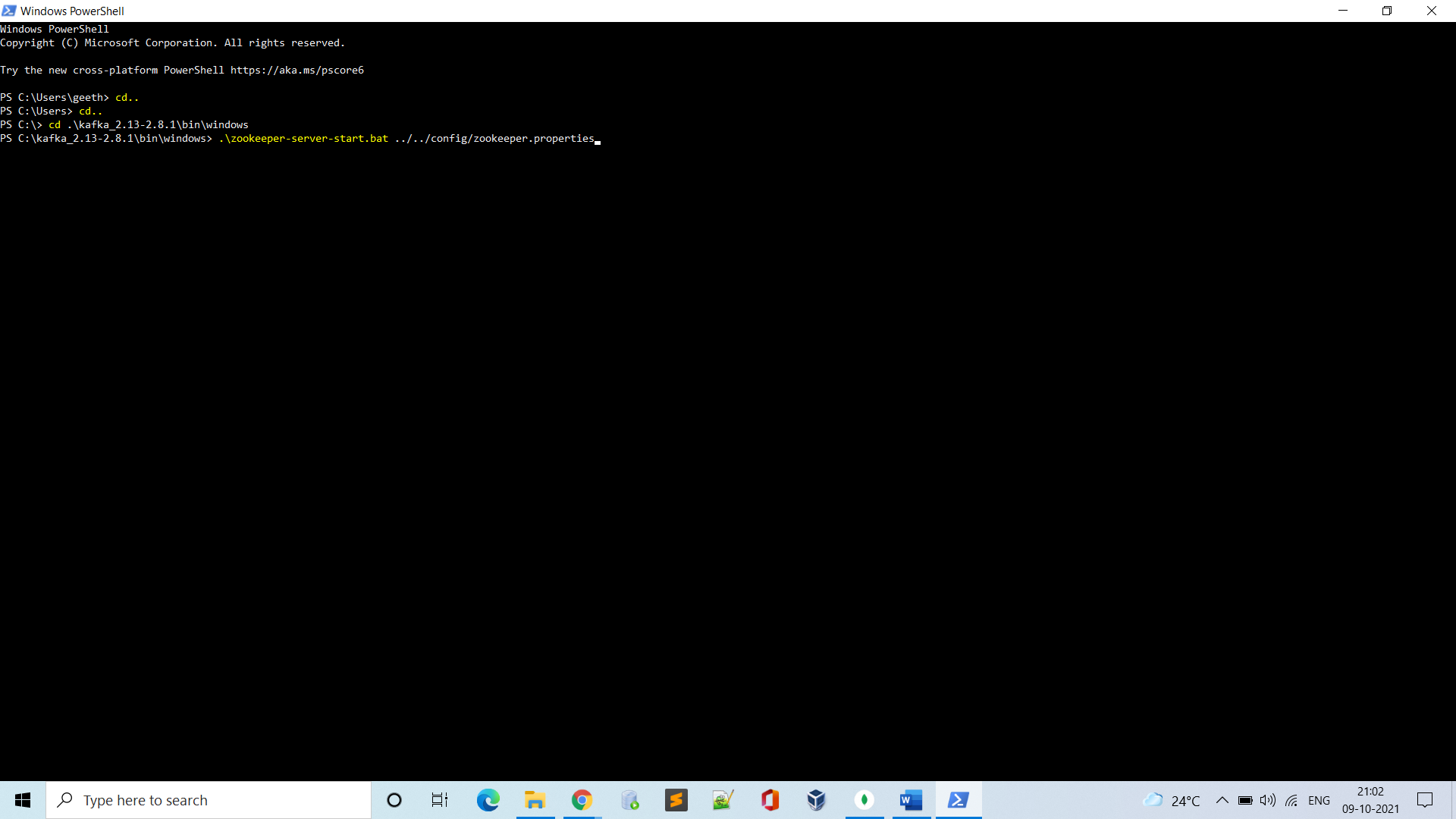
# Process of Input/commands and Output:

# Input/commands

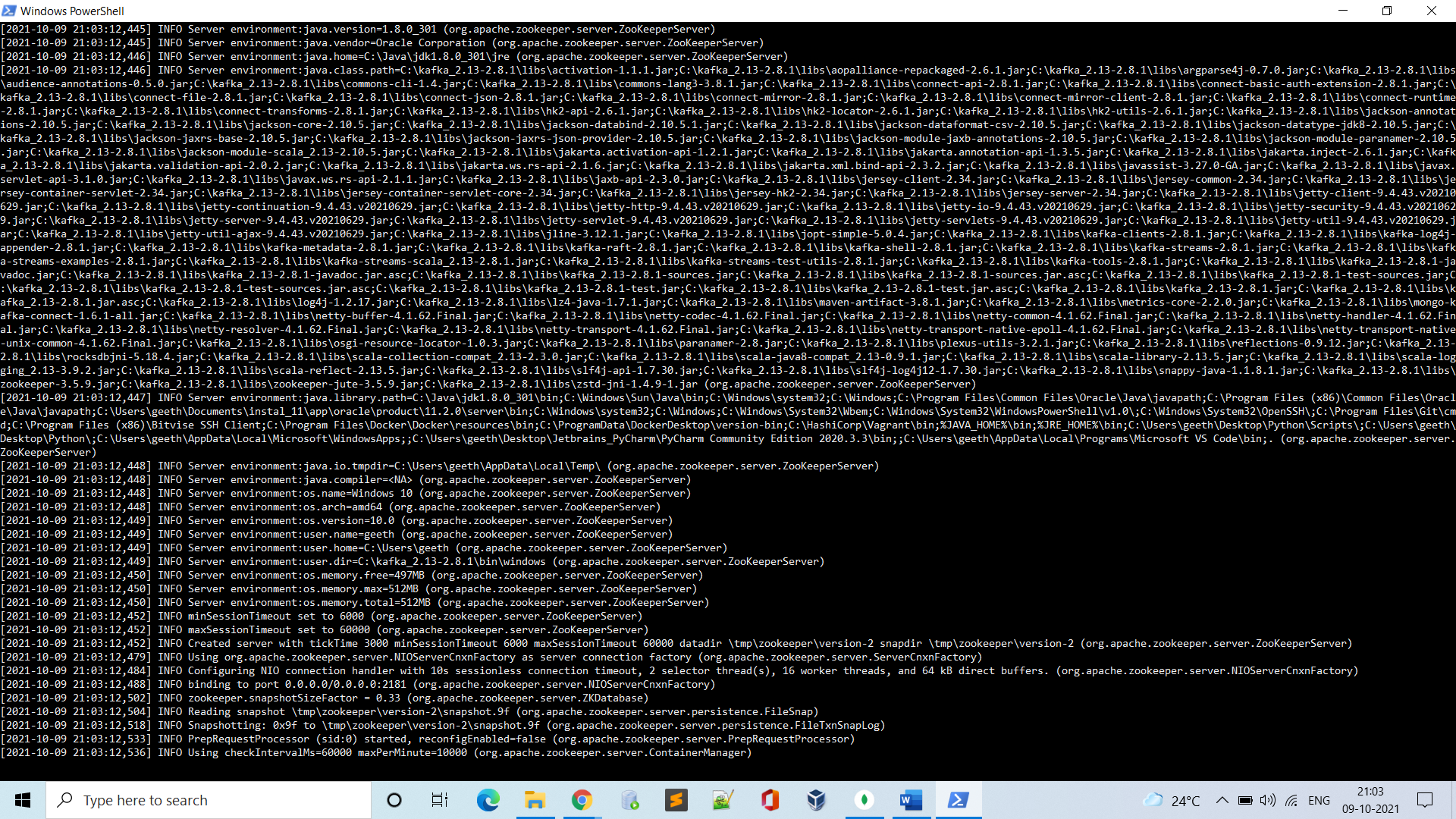
Here In This Week 1: Data Ingestion Need to be done, for that After setting up the above environment we need to follow the below steps-

We need to start the zookeeper. Using the command-

* .\zookeeper-server-start.bat ../../config/zookeeper.properties



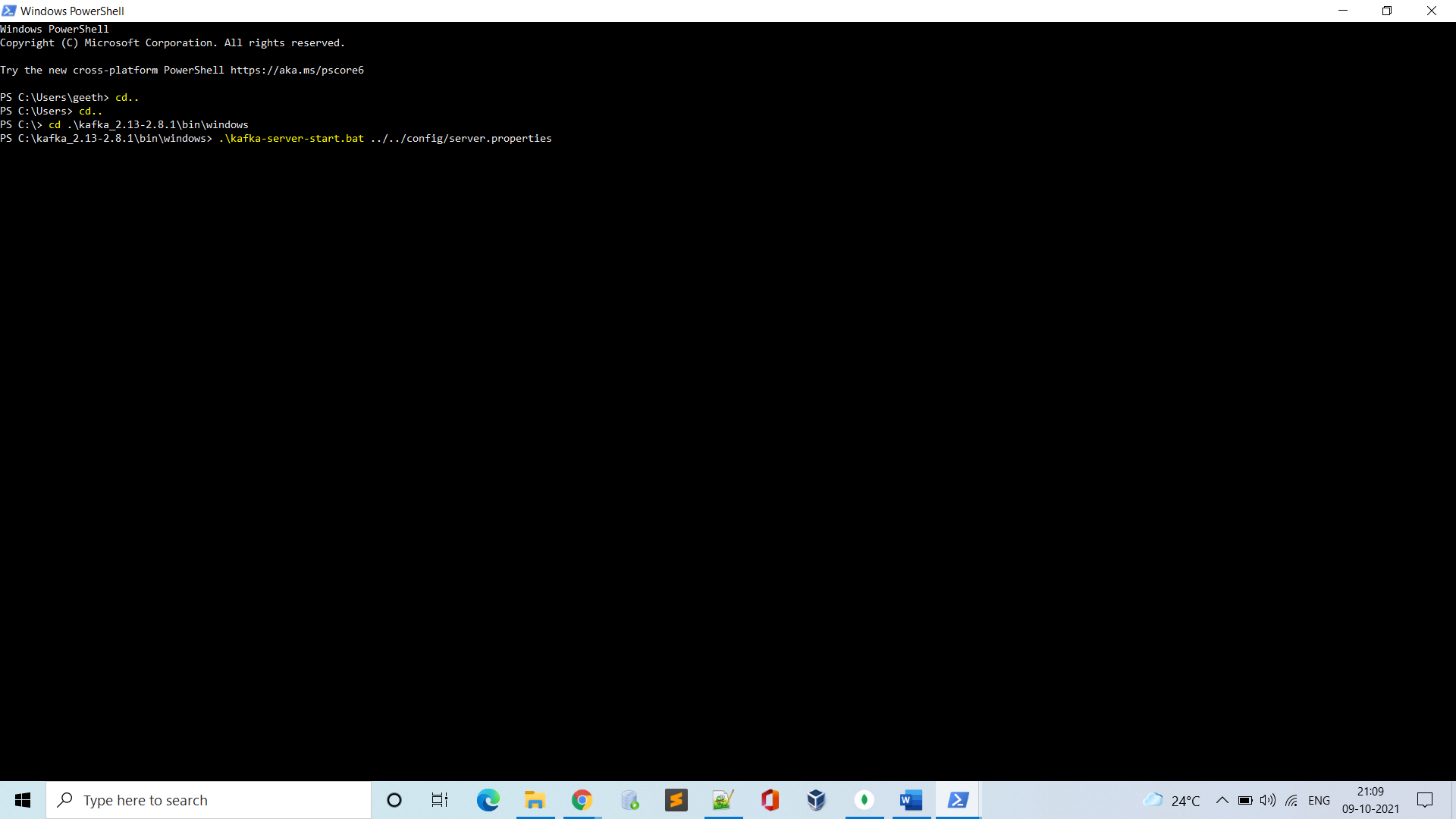
* Zookeeper Starts running



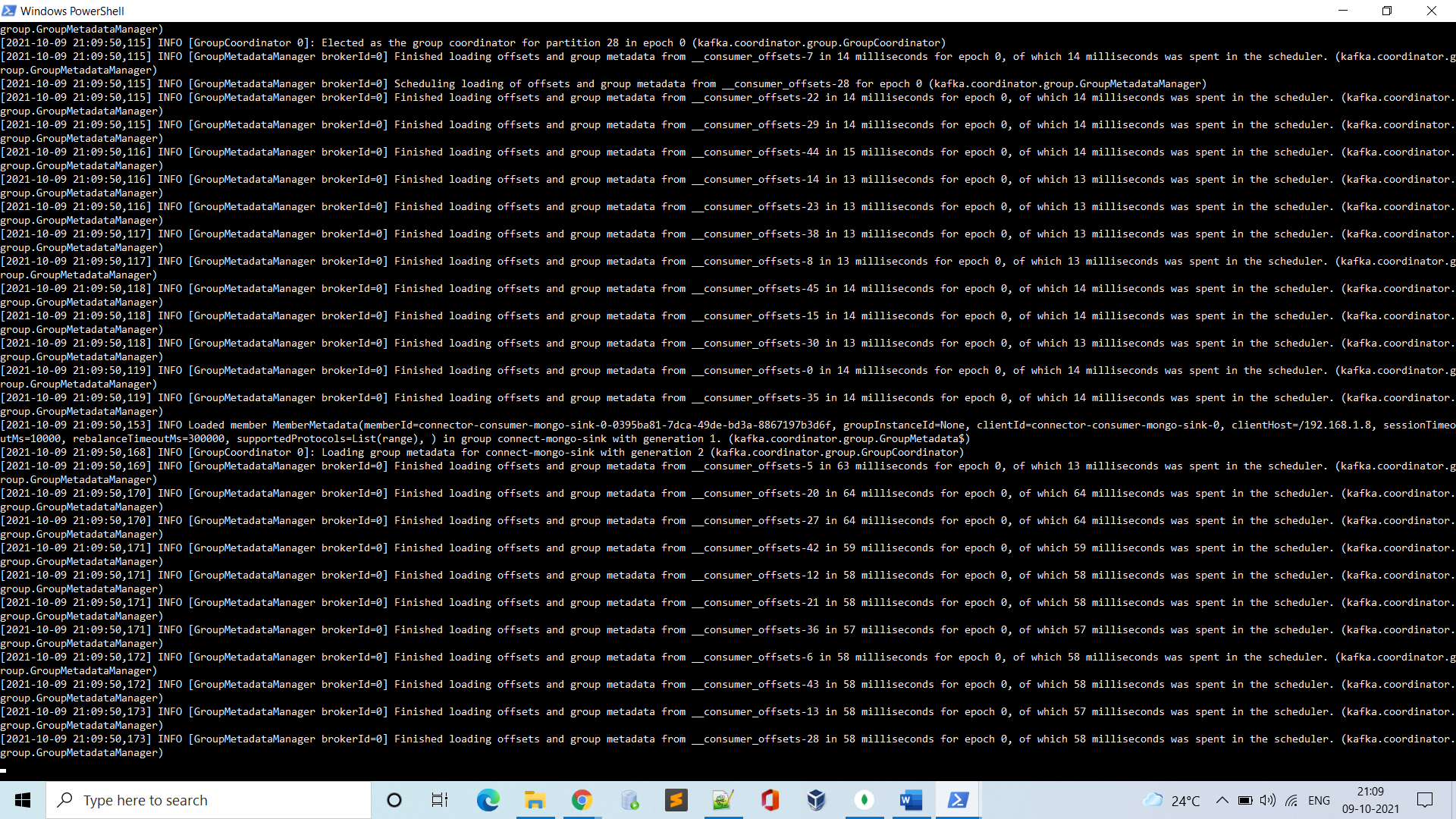
Now Open another terminal to start Kafka server-

Run the command-

* .\kafka-server-start.bat ../../config/server.properties

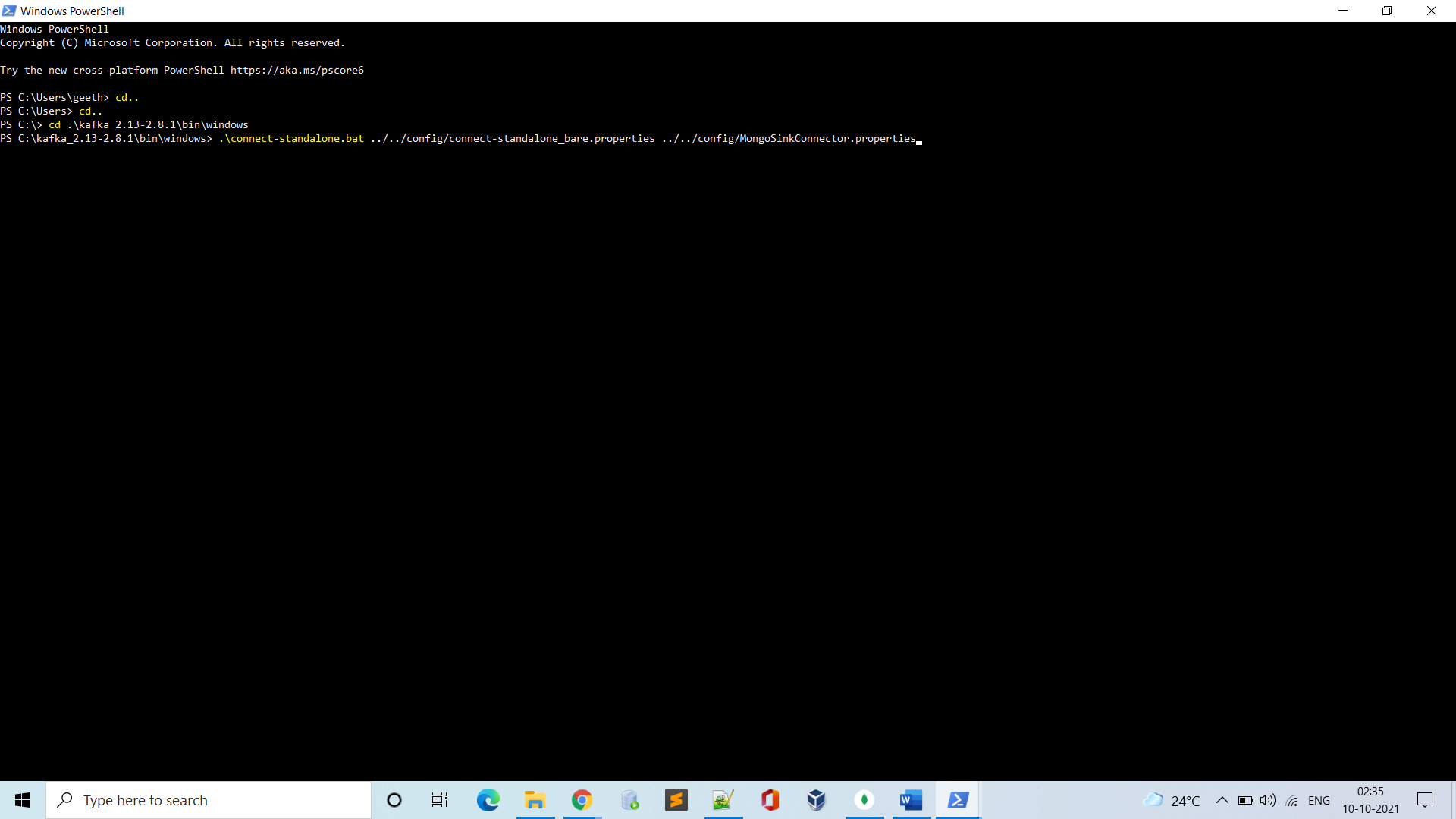


Kafka instance is now up-

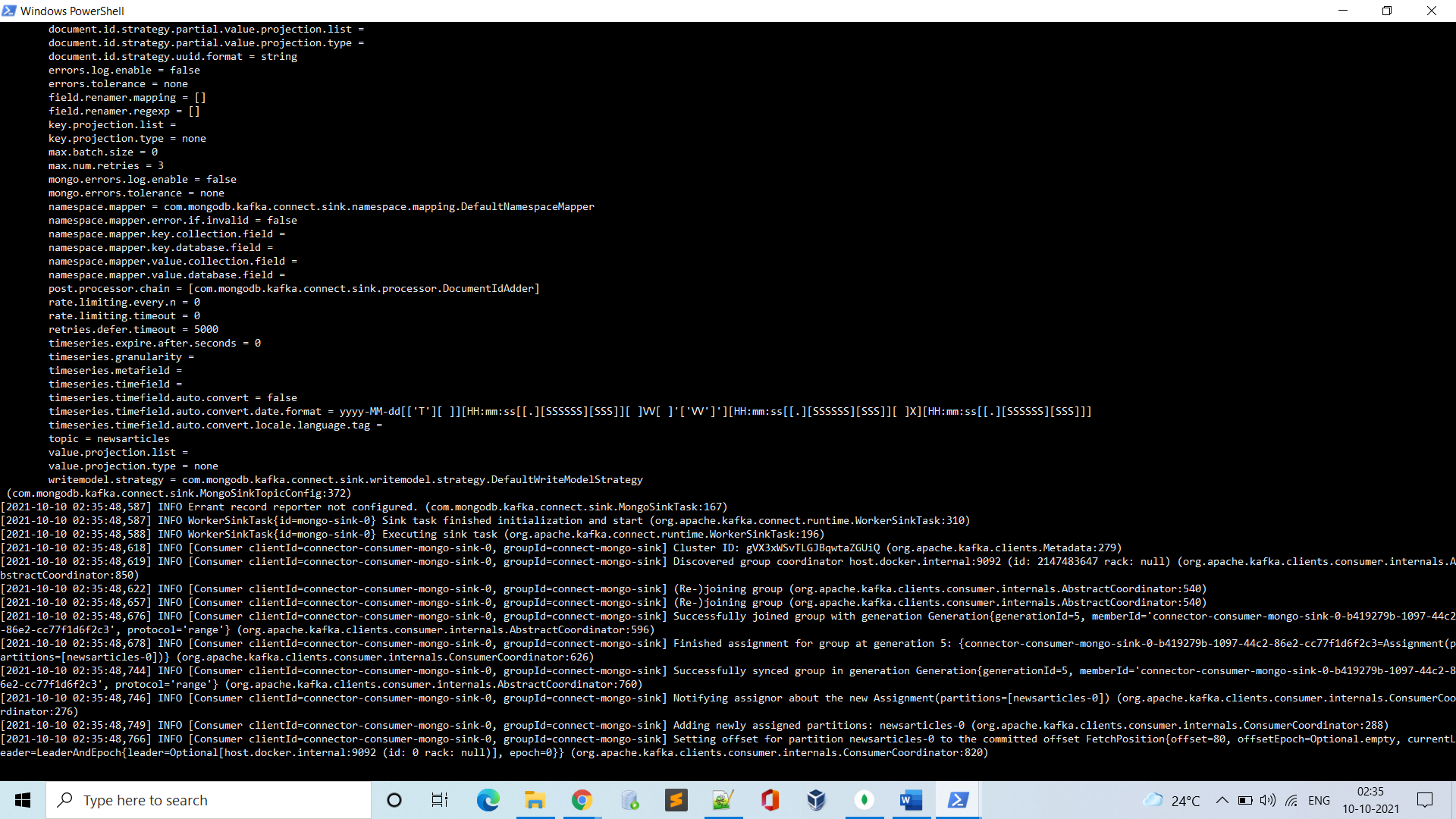


In a new Terminal Now start the mongo server by using the config files, by executing the command-

* .\connect-standalone.bat ../../config/connect-standalone\_bare.properties ../../config/MongoSinkConnector.properties



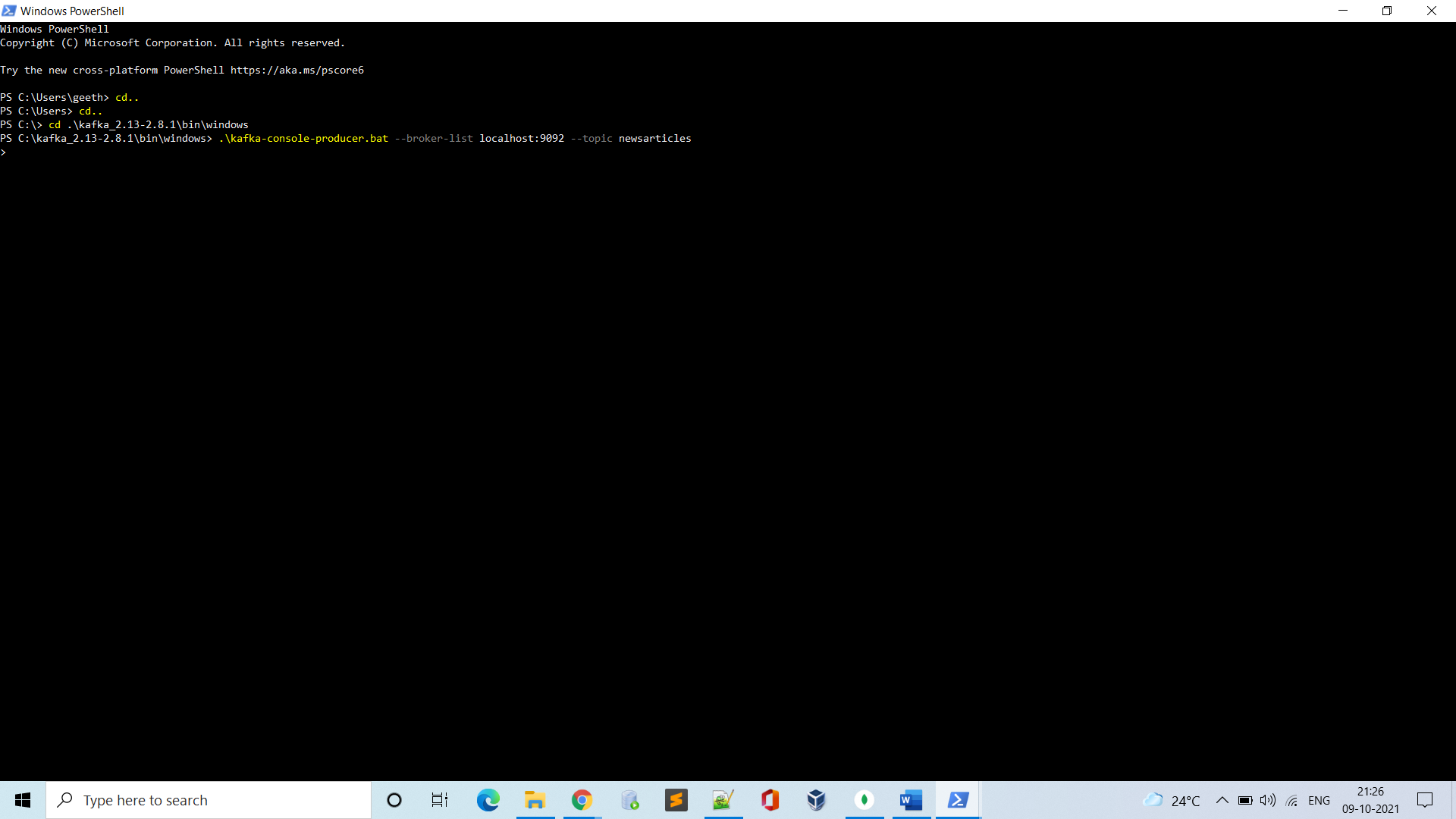
Mongo sink connection started-



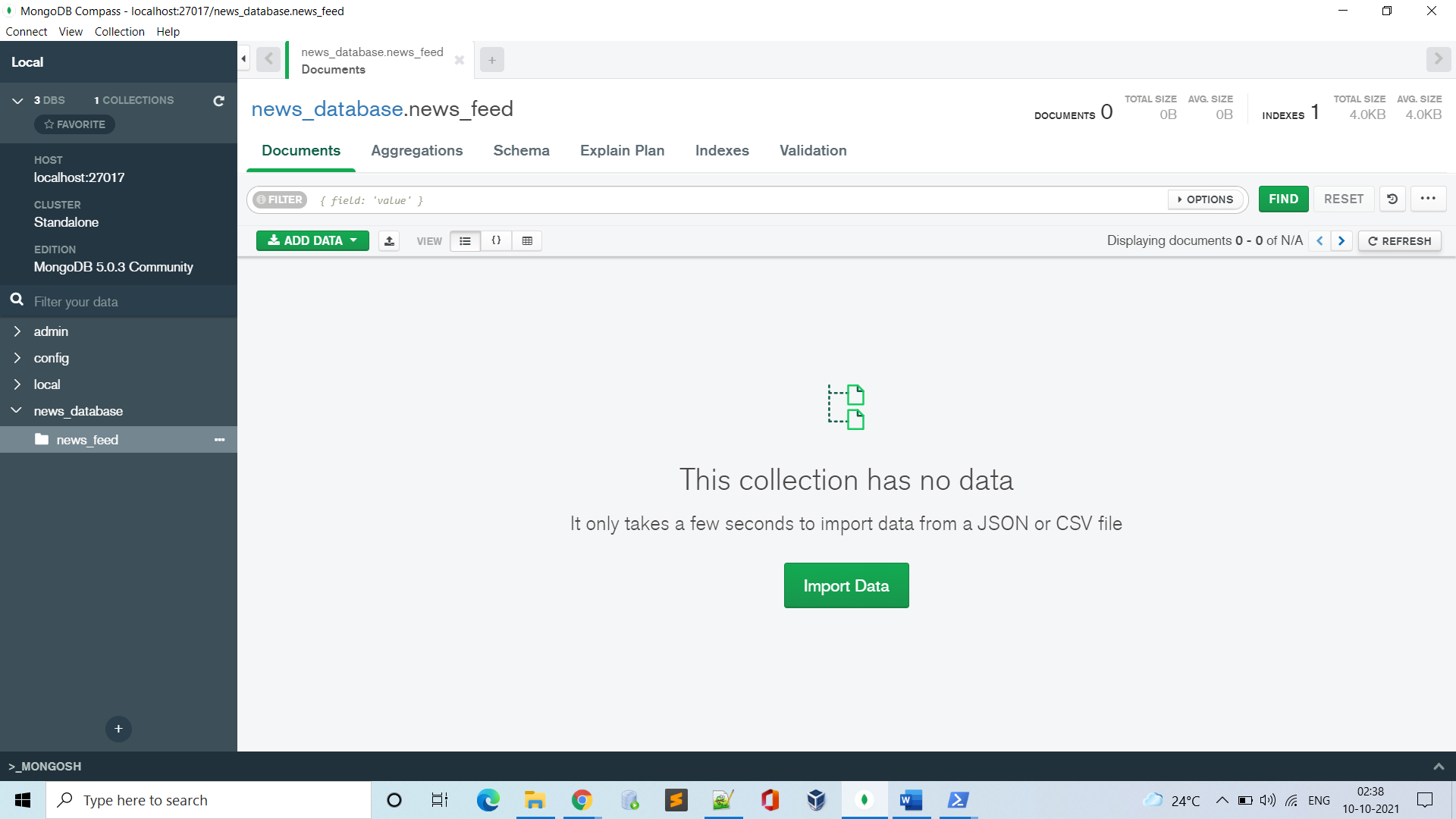
Now we open a new terminal again to start the Kafka producer and stream data to required topics (This topic is created while configuring the Kafka).

Using command (here the topic name is newsarticles)-

* .\kafka-console-producer.bat --broker-list localhost:9092 --topic newsarticles



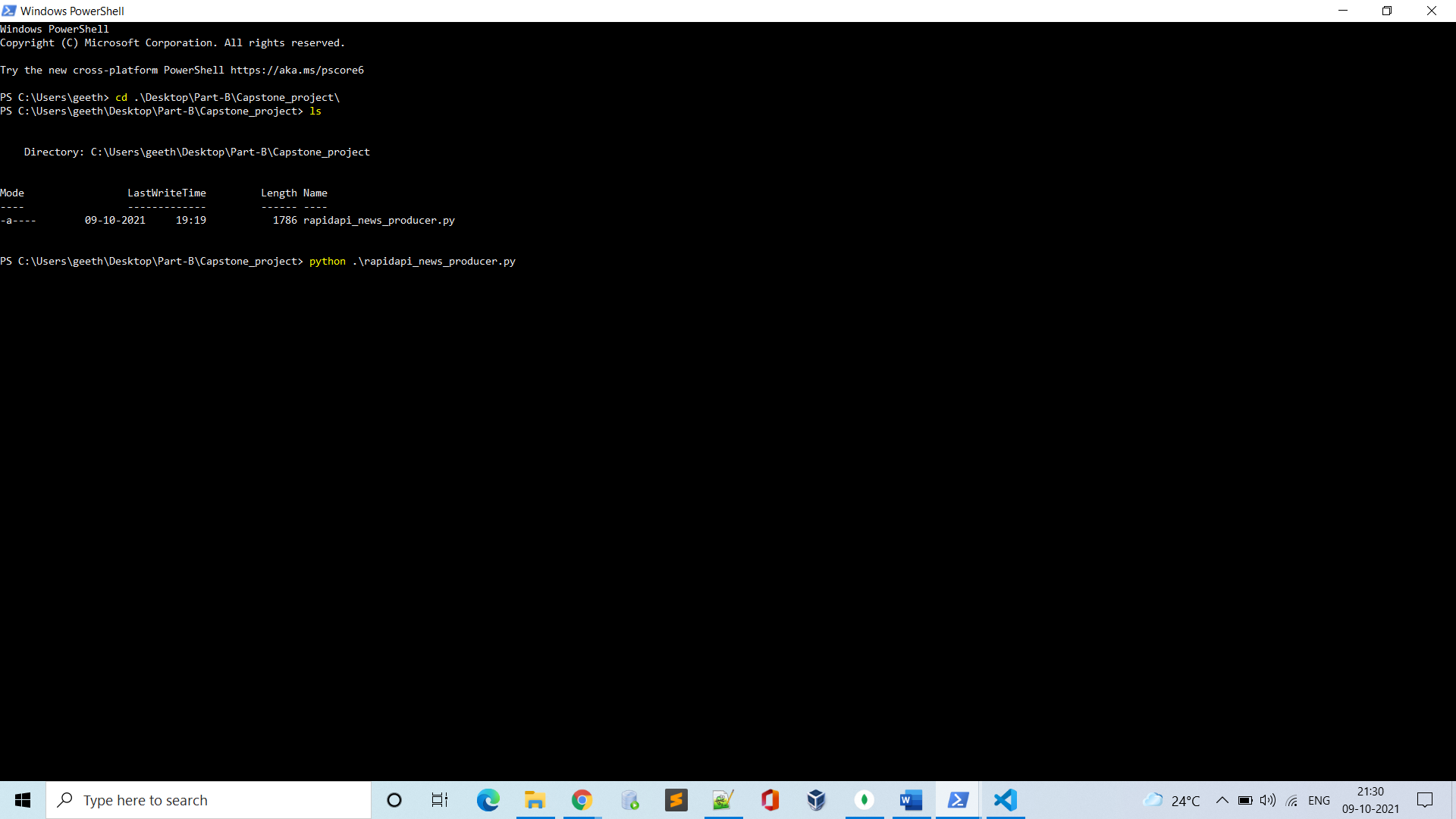
For now, the DB is empty without any records-



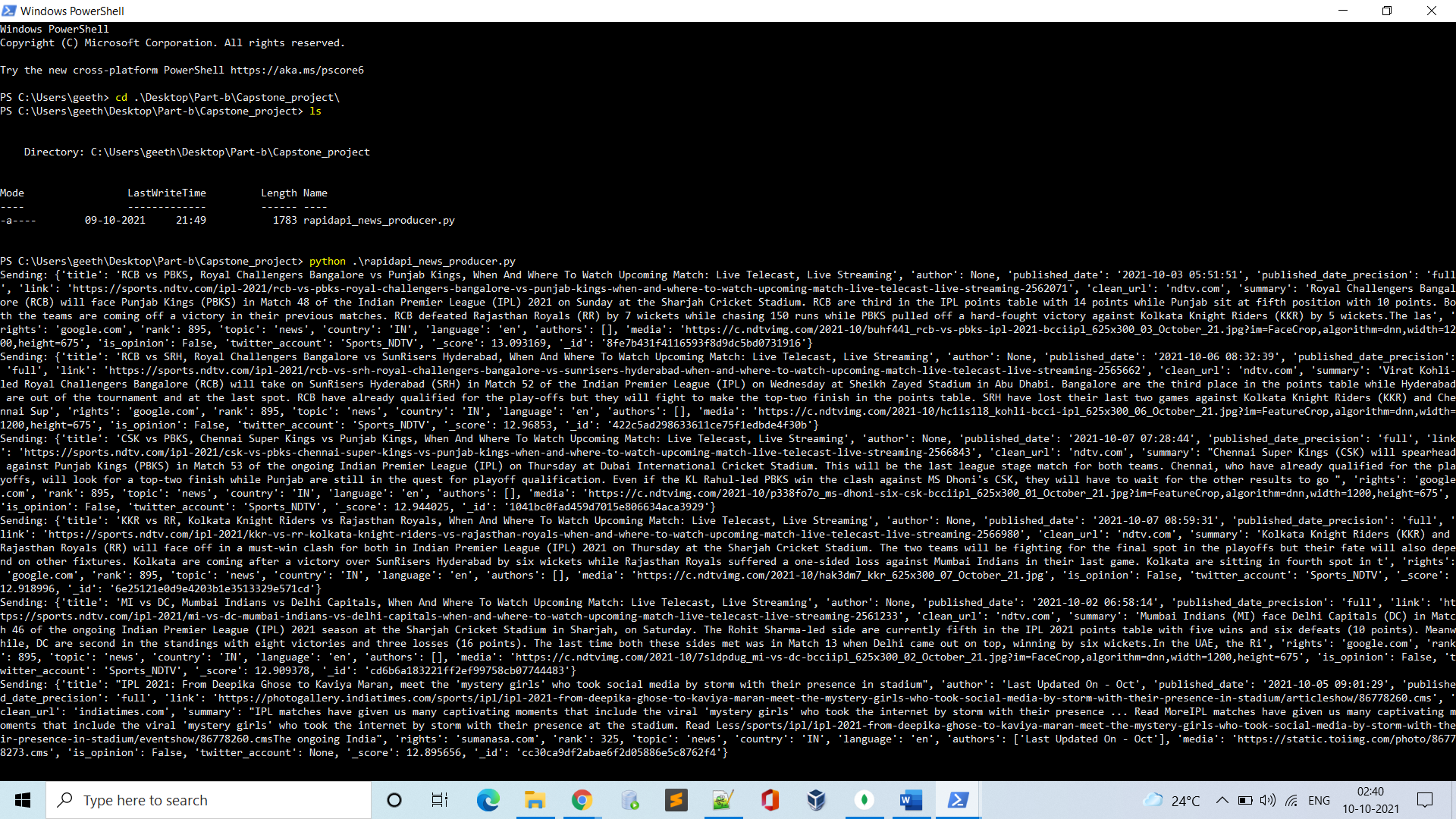
Now we are ready to stream data-

But here we are using the python file to stream the data into producer by running the python file on the same port number.

Hence we open another terminal to run the python file-

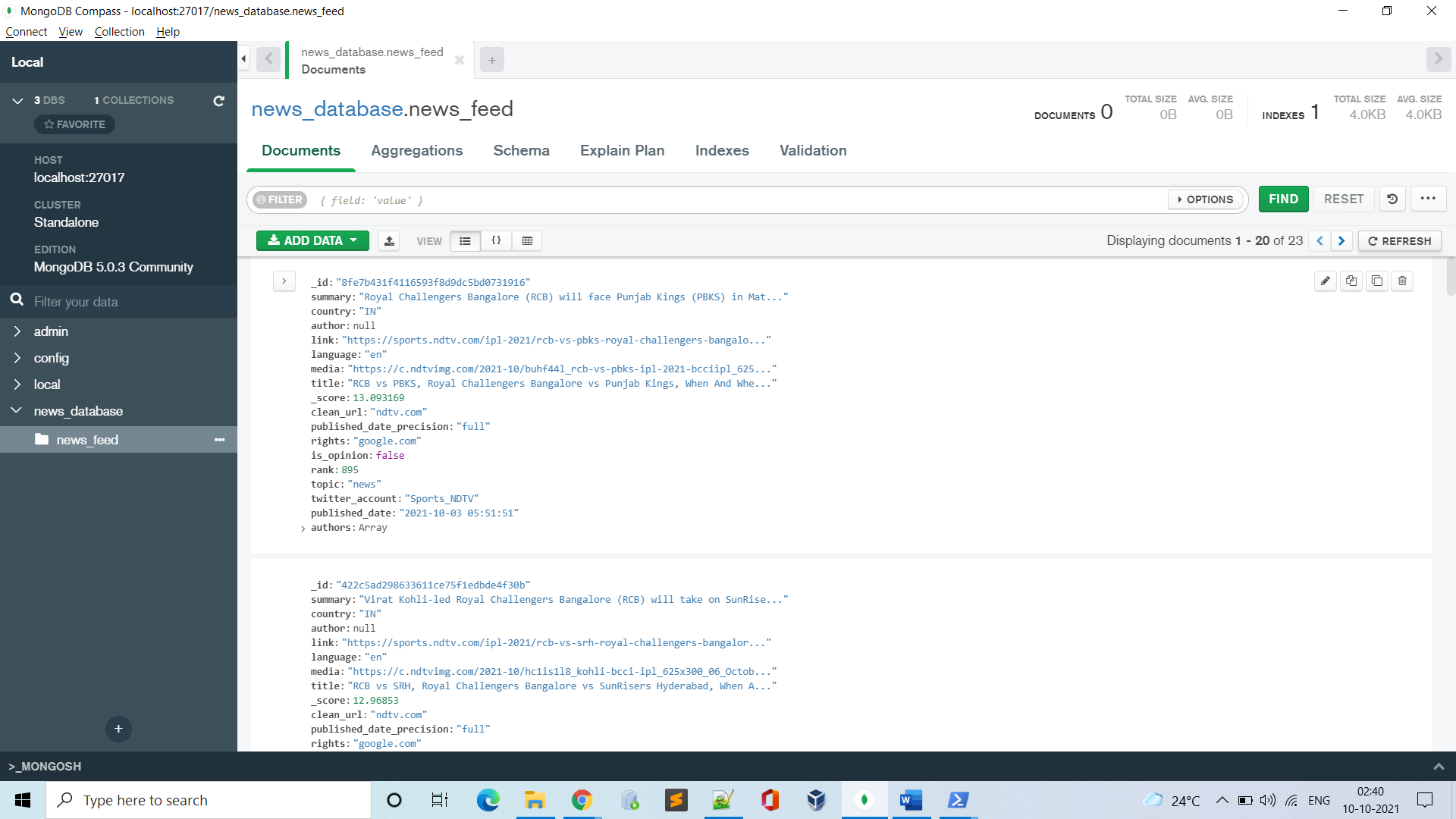


After running the above command, we see the news API responses are recorded and the same are sent to mongo dB-



# Output:

* In Mongo:



All the data is dumped into DB from the rapid API-

We can loop in many keywords as input to a rapid API and get huge amounts of news data which can be used to perform data cleansing and further used for model training.

Challenges faced:

* Environment Setup (Finding appropriate sink connections to Kafka and configuring it)
* Finding a way to send JSON format data into the topic and in python json format is considered as a string data type. Used serializers to overcome.

**Week 2:**

For Week 2: Data Preparation

* The Objective of this week is to clean and preprocess the raw data collected from week1, Raw data needs to be pre-processed to its relevant structure, by following some feature engineering rules, few of them are removing duplicates, null values, and dropping the unwanted columns.

# Goals for Week 2 Milestone

1. Load the data from Raw data Source into Spark by using a relevant connector for PySpark.
2. Perform data Cleaning and Preprocessing using the spark session.

# Environment Details required/Used:

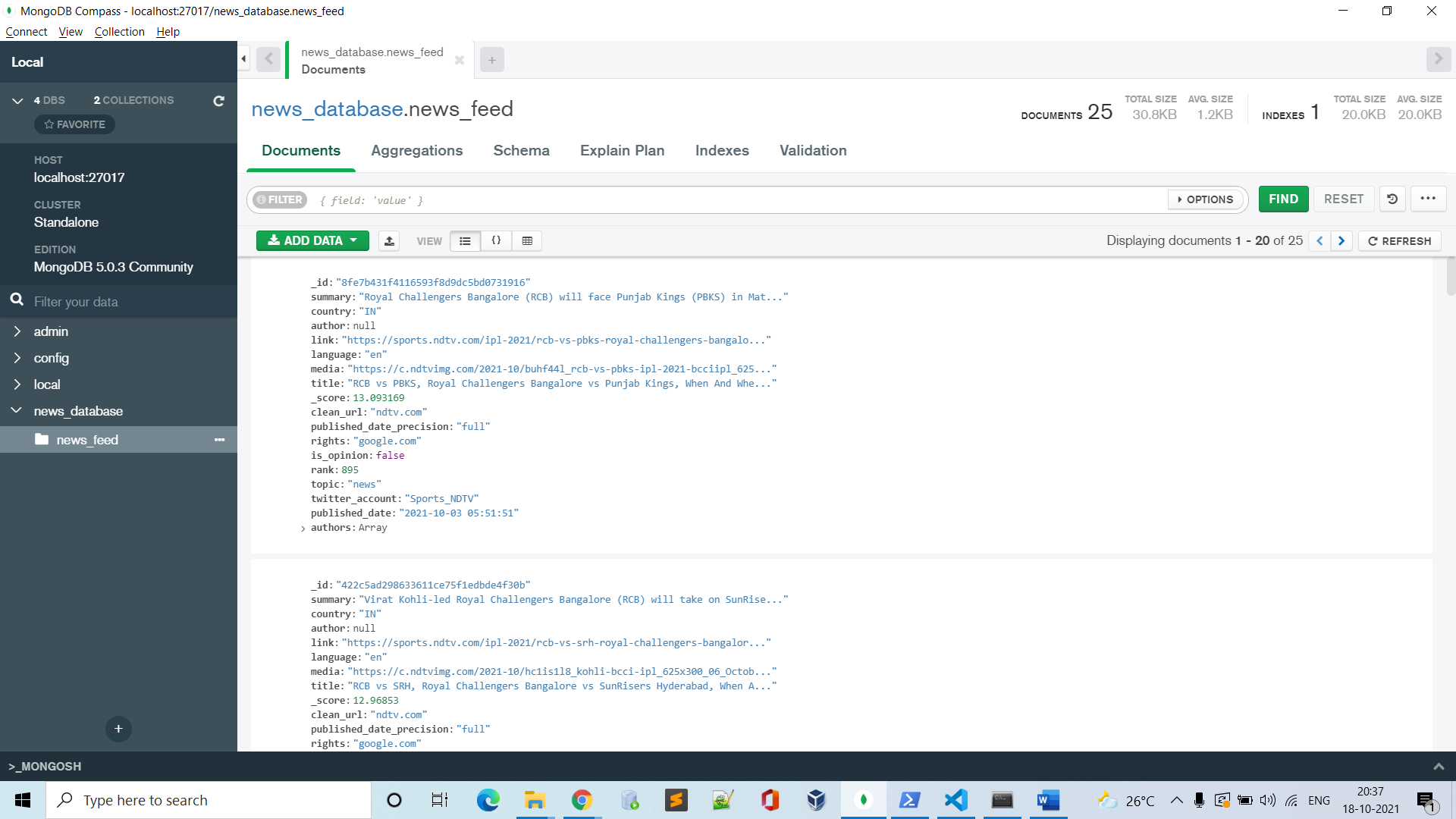
1. OS with Pyspark instance configured.
2. MongoDB to spark Connector (JAR files)
3. BSON (JAR file) to make mongo dB Json format compatible.

# Process of Input/commands and Output: Input/commands

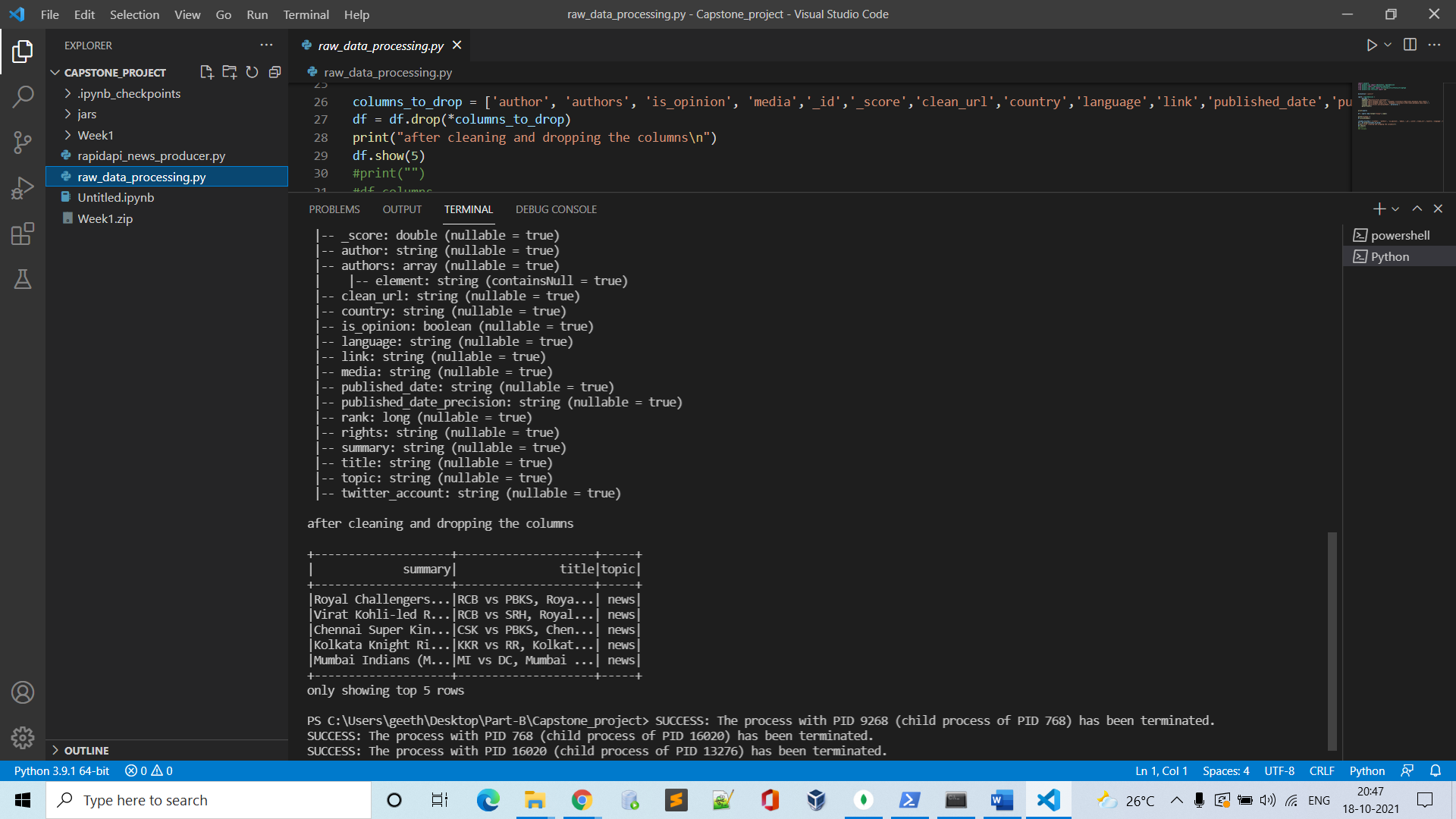
Here In This Week 2:

Raw data from mongo dB needs to be sent to a spark session where the preprocessing of data is taken care of and by using the properties of spark, the data is cleaned and only the required attributes which are to be sent into the model are stored and kept in a dataframe.

Mongo DB raw data-



After running raw\_data\_processing.py file where all the code for cleaning and storing only the required columns is taken care we get the following output-



Challenges faced:

* Environment Setup (Finding appropriate mongo dB to spark connectors and configuring it)
* Choosing the appropriate columns for the model training.

**Week 3:**

For Week 3: Model building

* The Objective of this week is to Segregate the preprocessed data into training and test clean from the data collected from week2(which is preprocessed data), Then complete the model training and deploy it in a flask application, also since training the model every time is makes the application very slow in starting, we use a pickle file of the saved model after performing model training. And when the model is reused, it just gets trained with newly added data sets instead of all the earlier ones again.

# Goals for Week 3 Milestone

1. Segregate the preprocessed data into training and test split.
2. Perform model training and saving into a pickle file.
3. Deploy the model into a flask application to use the model for prediction.

# Environment Details required/Used:

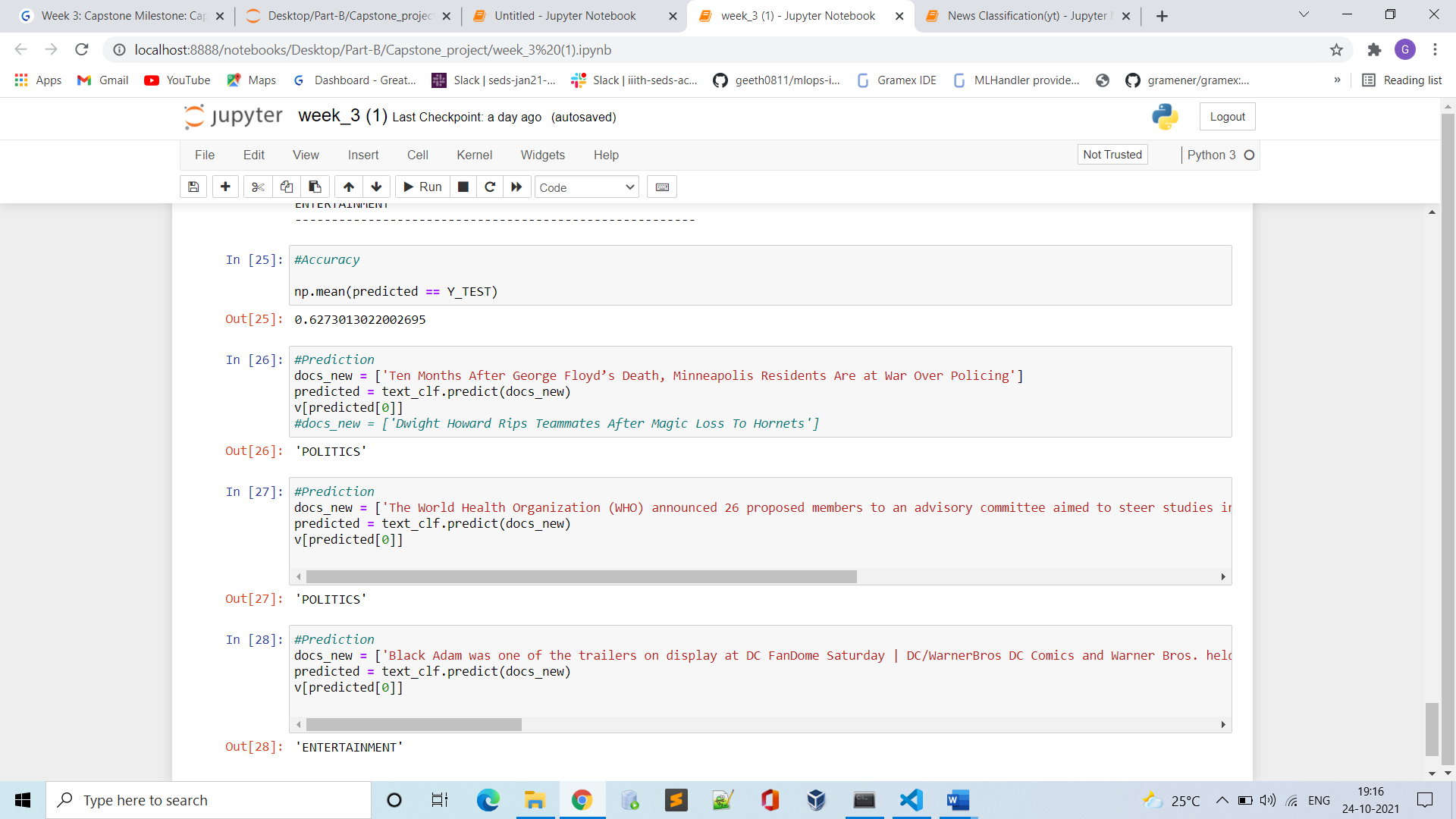
1. OS with some python IDE and Jupyter notebook configured to perform model train and split.
2. Flask API configuration files installed.
3. Pickle configuration installed.

# Process of Input/commands and Output: Input/commands

Here In This Week 3:

Preprocessed data from week2 is segregated into Test and train split so the model can be trained and evaluated, later we need to perform the model training and save it into a pickle file so that it can be reused for retraining with new data sets. Later we need to deploy the model into the flask application for it to perform prediction based on the input text provided.

For now, we have the model ready in the notebook file and would add the same to the flask app.



Challenges faced:

* System hardware configurations for training large data sets
* Selecting the appropriate ML model for prediction

**Week 4:**

For Week 4: Final Working Model

* The Objective of this week is to Finalize the working model and User Interface - A simple Stream lit app or a basic HTML page containing a form can be exposed in this project. It should take the description/ summary of the article, calls the flask API for prediction, and shows the predicted classification.

# Goals for Week 4 Milestone

1. Expose model via model-prediction-service in the form of flask API.
2. Dockerize all the projects by adding appropriate Docker file
3. Prepare a simple HTML page containing a form to take an article as input and print the predicted category.

# Environment Details required/Used:

1. Flask API configuration files installed.
2. Building Dockerization and Docker Compose for the predicted model
3. HTML Page By Using CSS and JavaScript Properties for a webpage.

# Process of Input/commands and Output: Input/commands

The following has a docker-compose.yml file which helps in building the image and starting the web application for prediction of news.

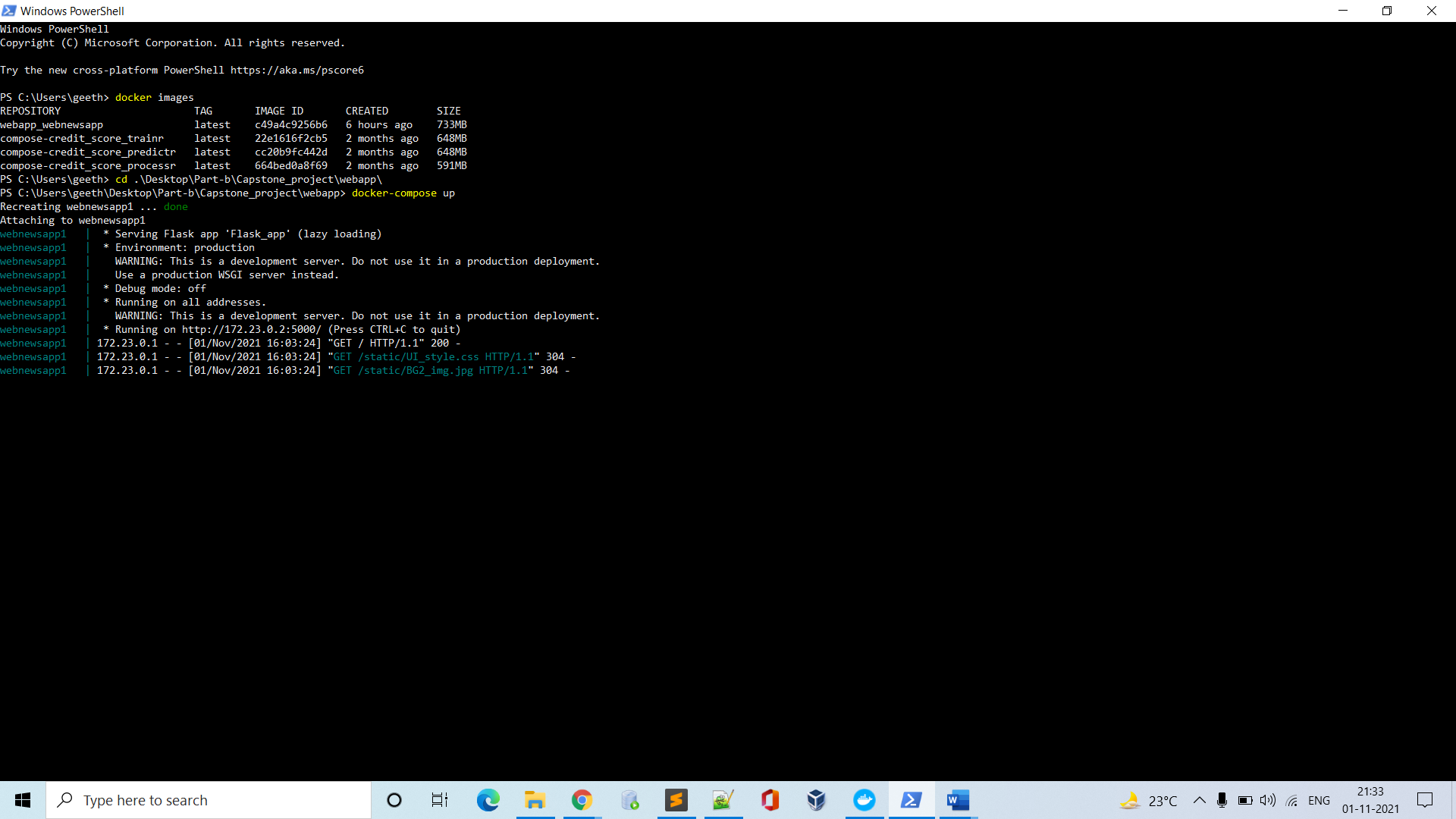
Use the following commands-

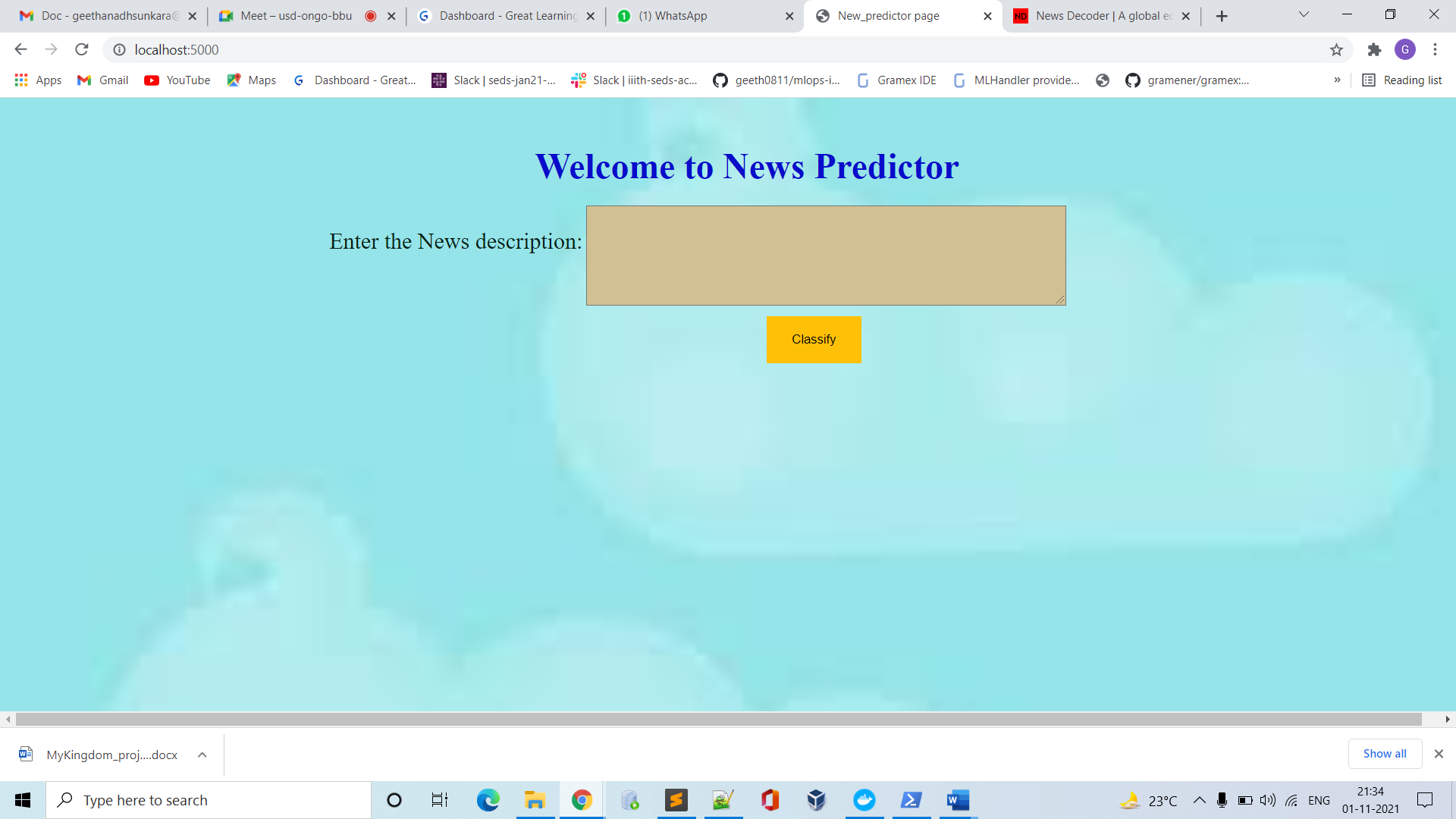
To check docker-compose installed 🡪 docker-compose –version

To build image and run the app 🡪 docker-compose up

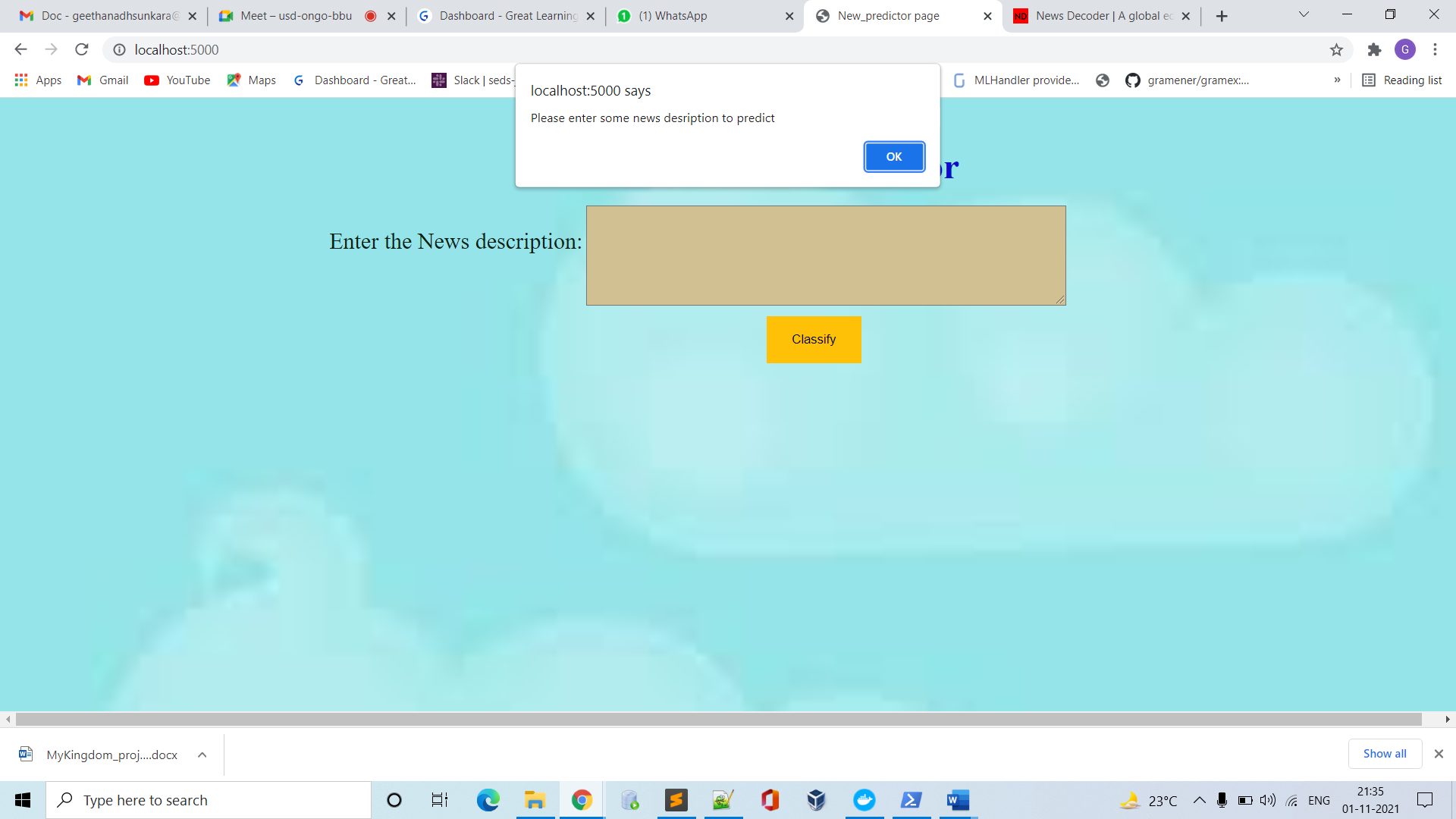
You would get a URL running which is mapped to the localhost of our local machine where the app starts.

Screenshots-

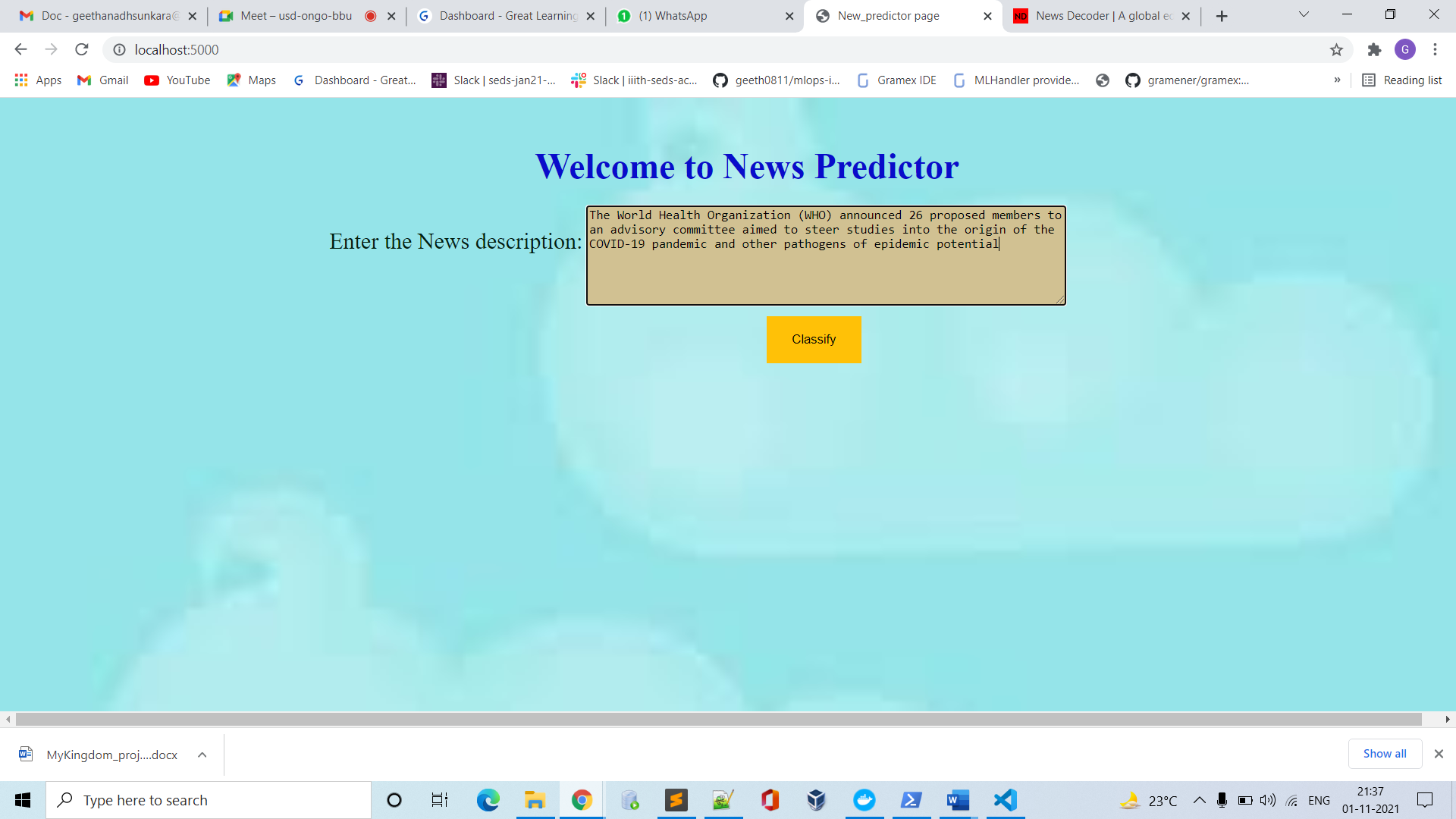




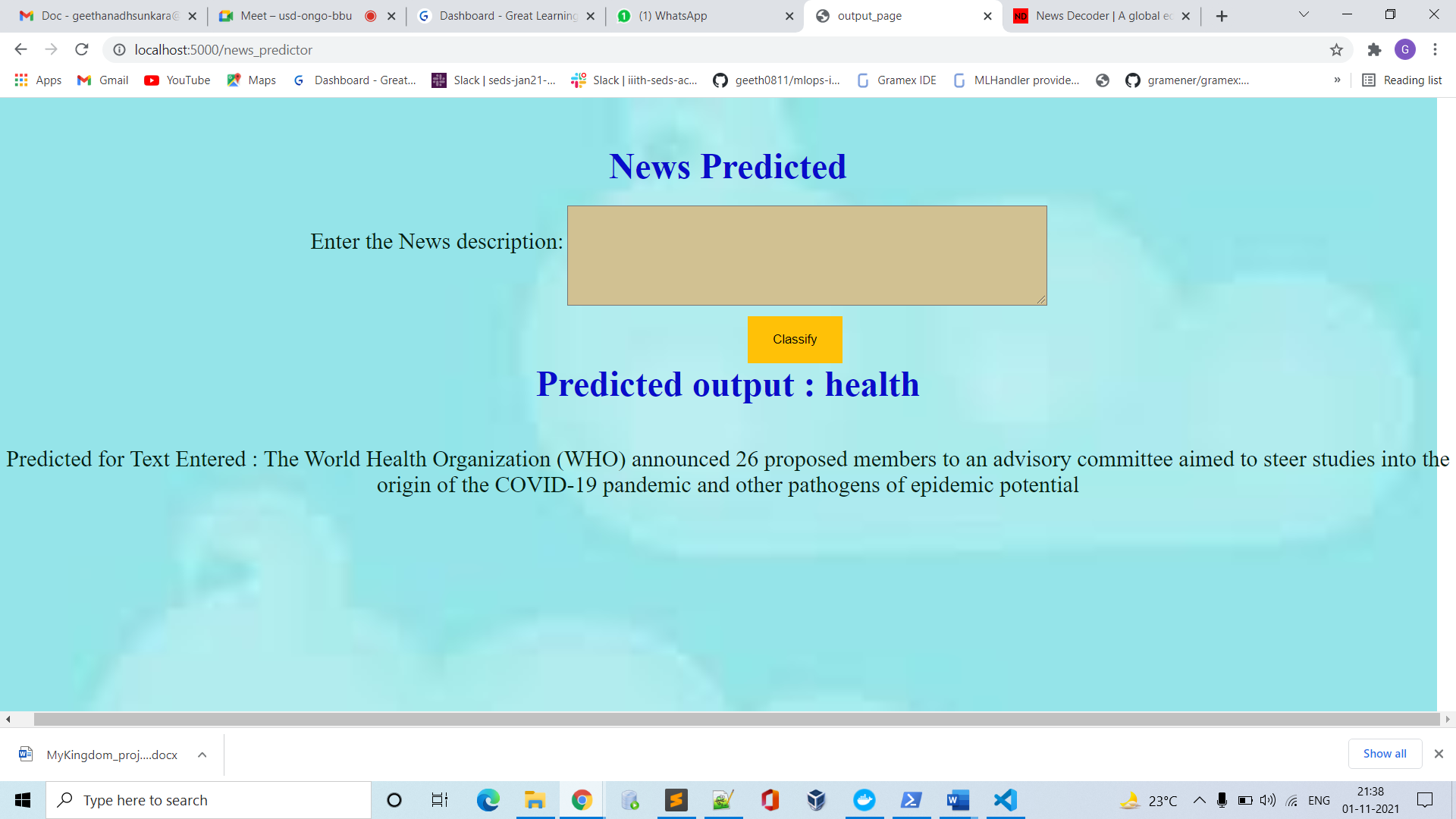
Alert if no text/description entered using JavaScript-



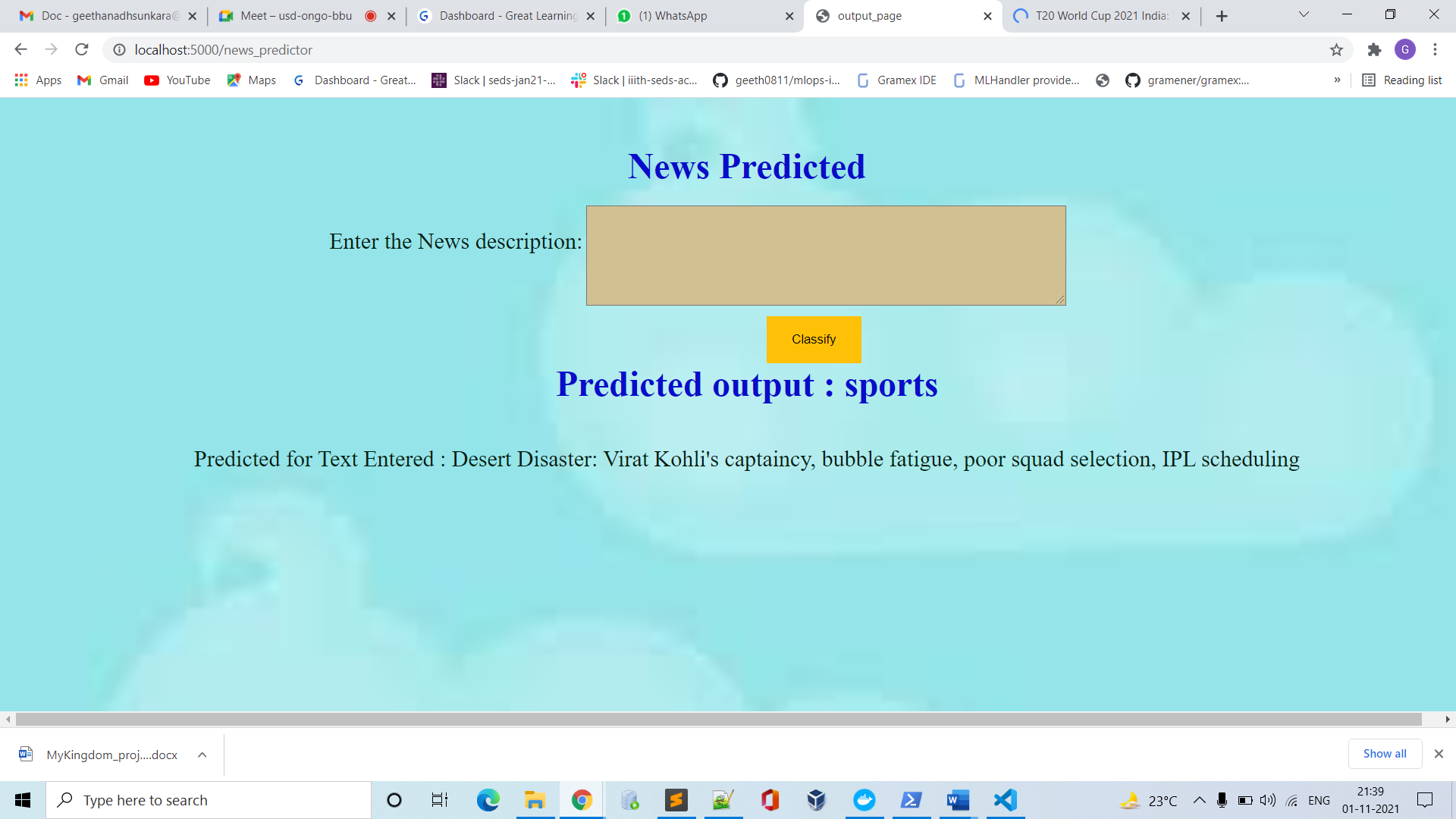
For Text entered- The World Health Organization (WHO) announced 26 proposed members to an advisory committee aimed to steer studies into the origin of the COVID-19 pandemic and other pathogens of epidemic potential



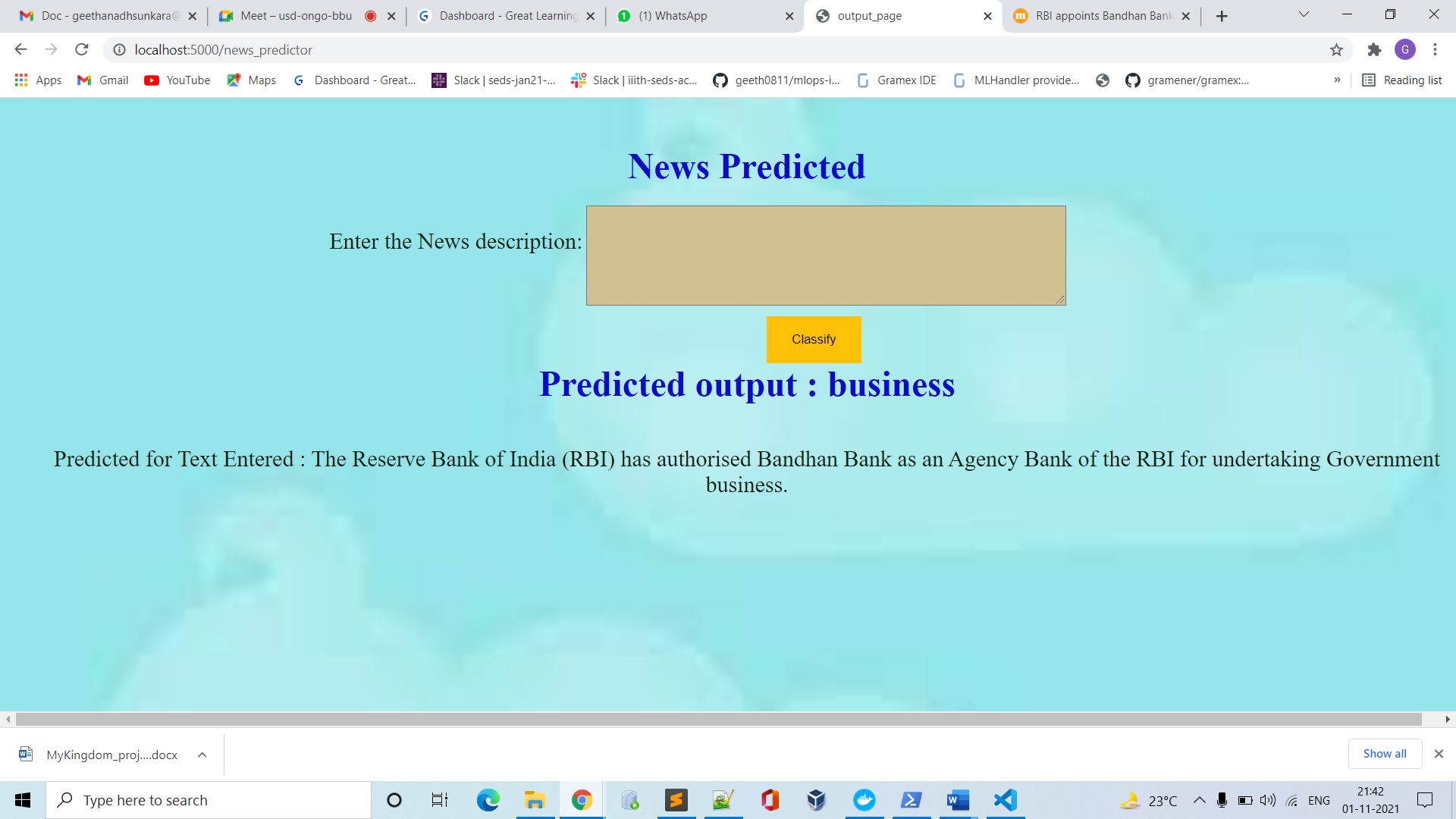
After clicking on classify button-



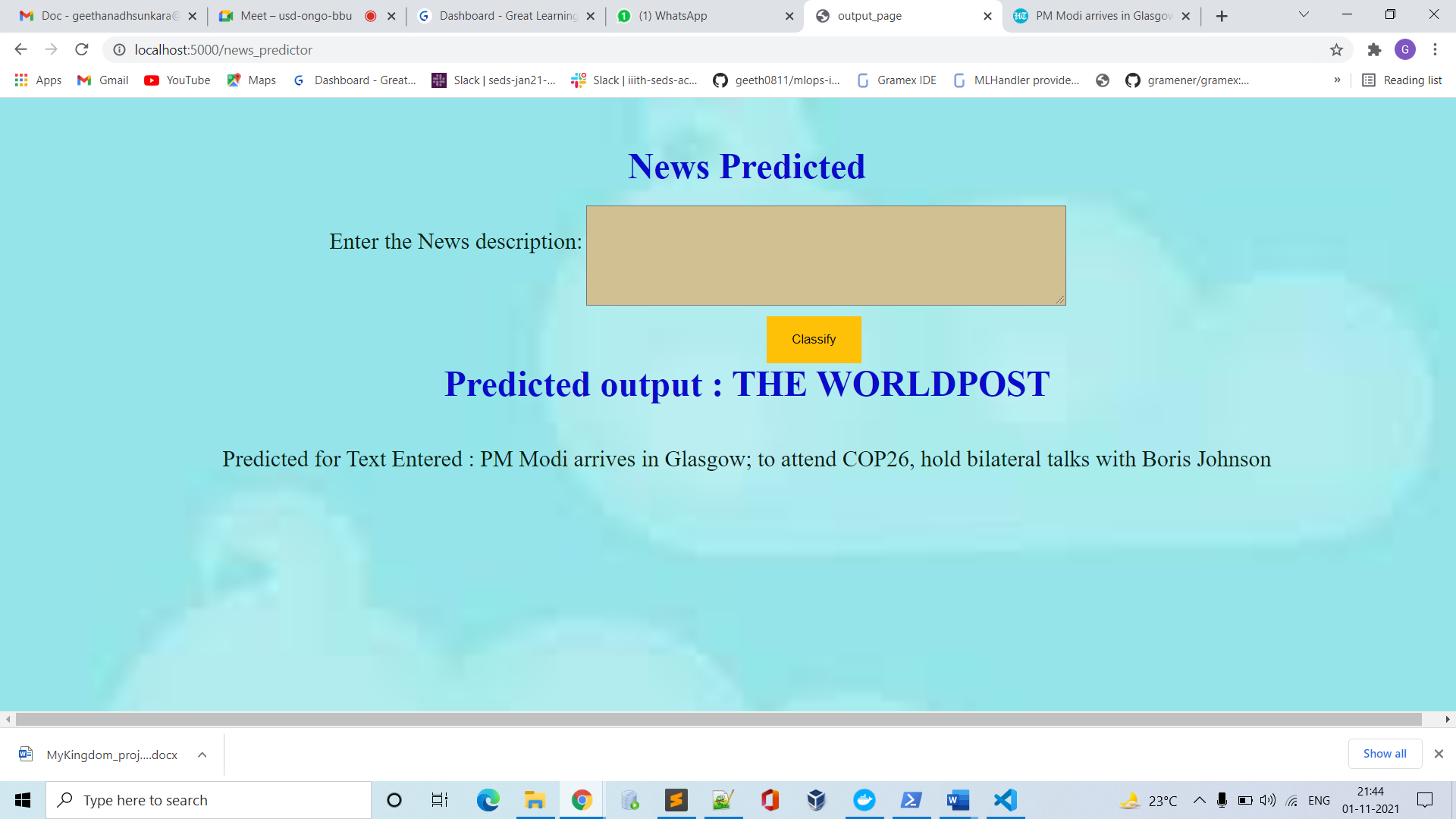
Sports news- Desert Disaster: Virat Kohli's captaincy, bubble fatigue, poor squad selection, IPL scheduling



The Reserve Bank of India (RBI) has authorized Bandhan Bank as an Agency Bank of the RBI for undertaking Government business.



PM Modi arrives in Glasgow; to attend COP26, hold bilateral talks with Boris Johnson



The above are the predicted results for few text inputs.

Thanks and regards

Team: Mykingdom