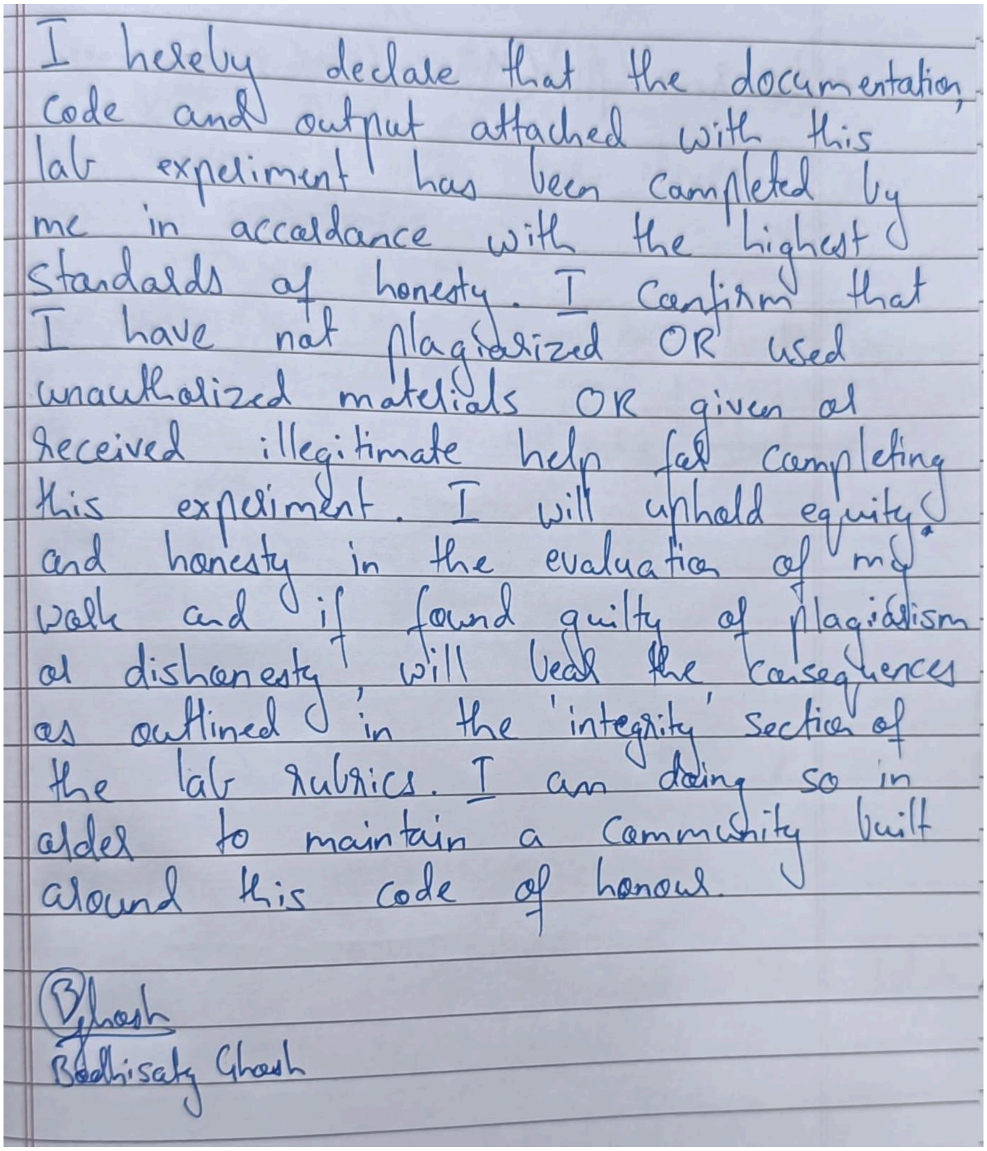
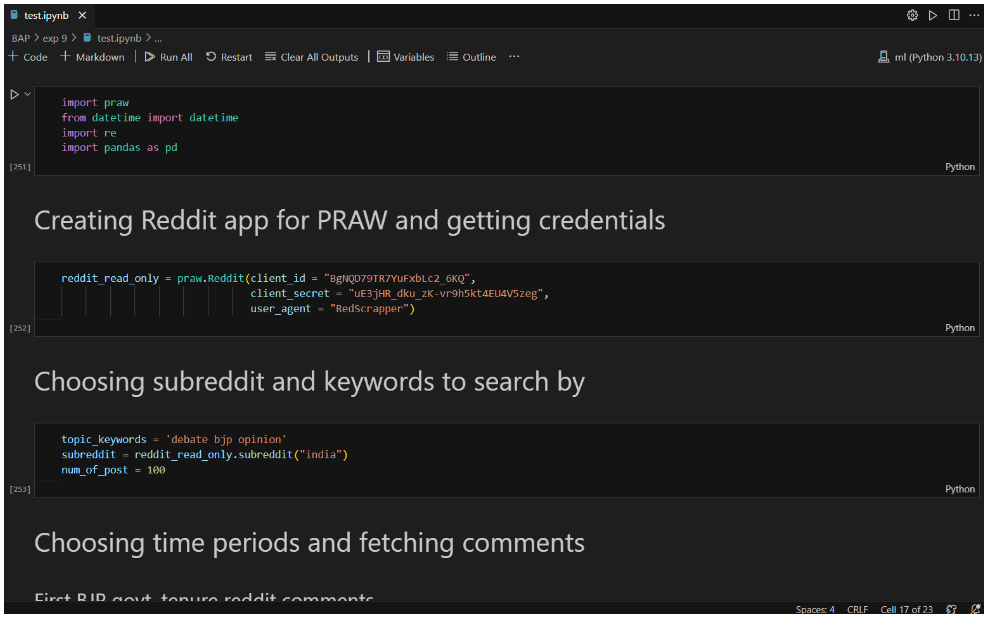
Name	**Bodhisatya Ghosh**
UID no.	**2021700026**

Experiment 9	
HONOUR PLEDGE	 I hereby declare that the documentation, code and output attached with this lab experiment has been completed by me in accordance with the highest standards of honesty. I confirm that I have not plagiarized OR used unauthorized materials OR given or received illegitimate help for completing this experiment. I will uphold equity and honesty in the evaluation of my work and if found guilty of plagiarism or dishonesty, will bear the consequences as outlined in the 'integrity' section of the lab rubrics. I am doing so in order to maintain a community built around this code of honour. B Ghosh Bodhisatya Ghosh
PROBLEM STATEMENT :	**Sentiment Analysis on Social Media Data** 1. Select one of the social media channels you would like to analyze the data of - Twitter, Instagram, Reddit, YouTube EDIT: Based on this article, I feel it might be much harder to scrape

	tweets now as compared to before 2023. So I would recommend looking at other sources beyond Twitter (X) as your data source. 2. Select value/s for relevant attributes like 'author', 'topic', 'mention', #tag', 'country' etc. to narrow down the scope of what you would like to analyze. 3. Fetch the narrowed down data across a specific time window using python libraries. This time-window would allow to compare and contrast sentiments across different times and for different events that occur during the time-window 4. Process the text and do sentiment analysis on it using relevant libraries (Twitter lexicon would require a more specialized library) 5. Plot the results and state your analysis **Bonus**: The top Result Visualization and Analysis for this experiment would be awarded an extra **+2 points**. This will be solely used to cover up for previous or future performance gaps.
THEORY:	PRAW, short for Python Reddit API Wrapper, is a powerful Python package that provides convenient access to Reddit’s API. With PRAW, developers can easily interact with Reddit, retrieve data and perform various actions. PRAW aims to be as easy to use as possible and is designed to follow all of Reddit’s API rules. Sentiment analysis is the process of analyzing digital text to determine if the emotional tone of the message is positive, negative, or neutral. Today, companies have large volumes of text data like emails, customer support chat transcripts, social media comments, and reviews. Sentiment analysis tools can scan this text to automatically determine the author’s attitude towards a topic. Companies use the insights from sentiment analysis to improve customer service and increase brand reputation. **Analysis topic**: I have chosen to compare the sentiments over time on the subreddit “r/india” regarding the ruling government party at the time. I have used keywords such as “bjp”, “debate”, “opinion”.

PROGRAM:



```python
import praw
from datetime import datetime
import re
import pandas as pd
```

Creating Reddit app for PRAW and getting credentials

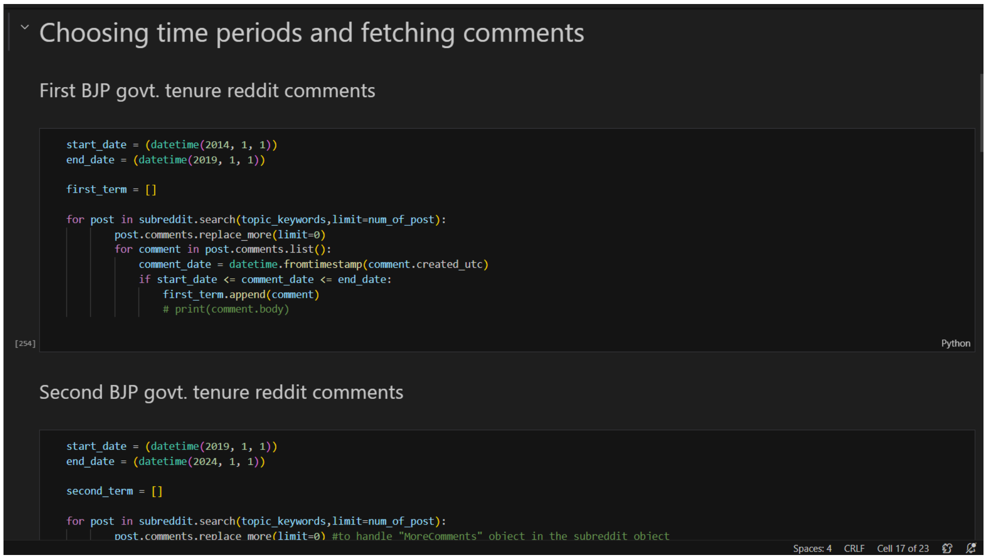
```python
reddit_read_only = praw.Reddit(client_id = "BgNQD79TR7YuFxbLc2_6KQ",
                                client_secret = "uE3jHR_dku_zK-vr9h5kt4EU4V5zeg",
                                user_agent = "RedScrapper")
```

Choosing subreddit and keywords to search by

```python
topic_keywords = 'debate bjp opinion'
subreddit = reddit_read_only.subreddit("india")
num_of_post = 100
```

Choosing time periods and fetching comments

First BJP govt. tenure reddit comments



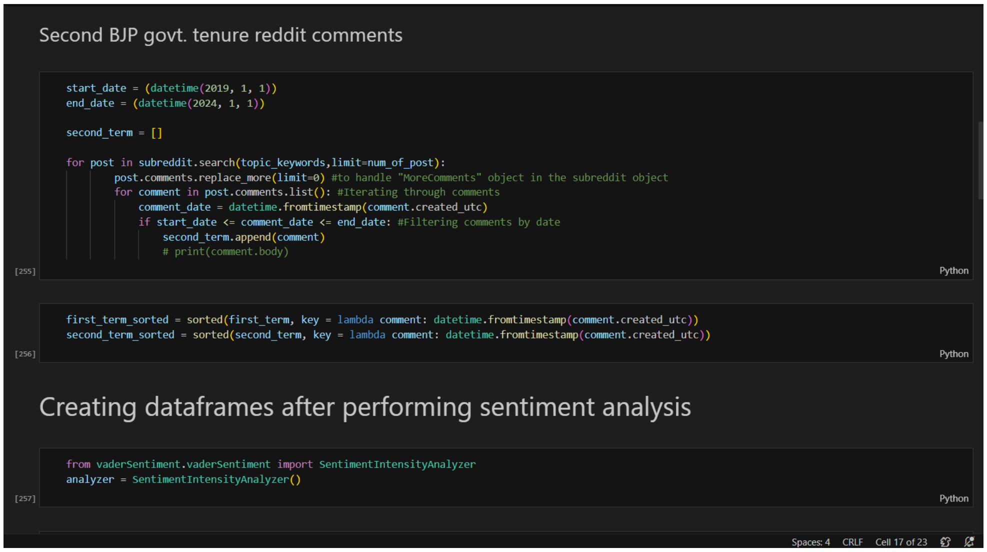
Choosing time periods and fetching comments

First BJP govt. tenure reddit comments

```python
start_date = (datetime(2014, 1, 1))
end_date = (datetime(2019, 1, 1))

first_term = []

for post in subreddit.search(topic_keywords,limit=num_of_post):
        post.comments.replace_more(limit=0)
        for comment in post.comments.list():
            comment_date = datetime.fromtimestamp(comment.created_utc)
            if start_date <= comment_date <= end_date:
                first_term.append(comment)
                # print(comment.body)
```

Second BJP govt. tenure reddit comments

```python
start_date = (datetime(2019, 1, 1))
end_date = (datetime(2024, 1, 1))

second_term = []

for post in subreddit.search(topic_keywords,limit=num_of_post):
        post.comments.replace_more(limit=0) #to handle "MoreComments" object in the subreddit object
```



Second BJP govt. tenure reddit comments

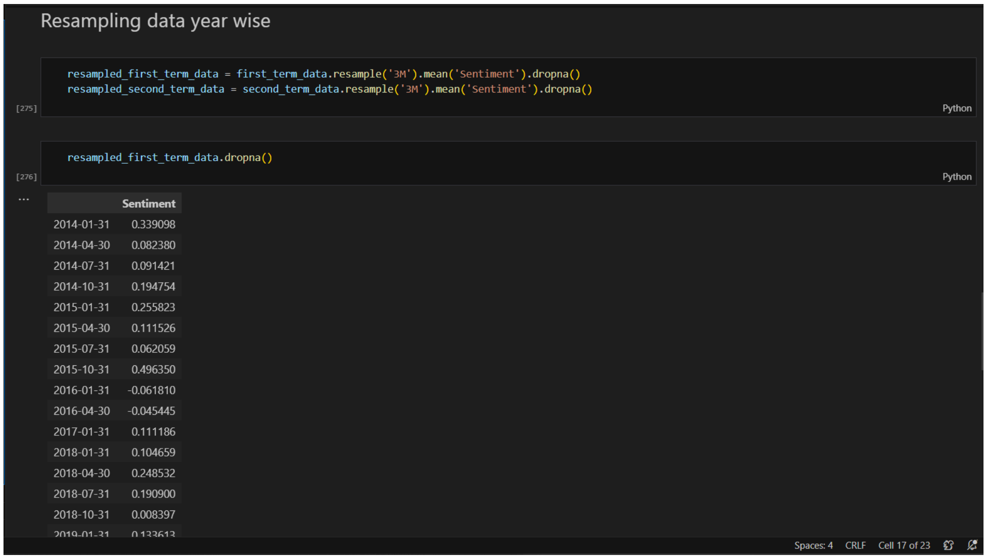
```python
start_date = (datetime(2019, 1, 1))
end_date = (datetime(2024, 1, 1))

second_term = []

for post in subreddit.search(topic_keywords,limit=num_of_post):
        post.comments.replace_more(limit=0) #to handle "MoreComments" object in the subreddit object
        for comment in post.comments.list(): #Iterating through comments
            comment_date = datetime.fromtimestamp(comment.created_utc)
            if start_date <= comment_date <= end_date: #Filtering comments by date
                second_term.append(comment)
                # print(comment.body)
```

```python
first_term_sorted = sorted(first_term, key = lambda comment: datetime.fromtimestamp(comment.created_utc))
second_term_sorted = sorted(second_term, key = lambda comment: datetime.fromtimestamp(comment.created_utc))
```

Creating dataframes after performing sentiment analysis

```python
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
analyzer = SentimentIntensityAnalyzer()
```

Resampling data on a quarterly basis:

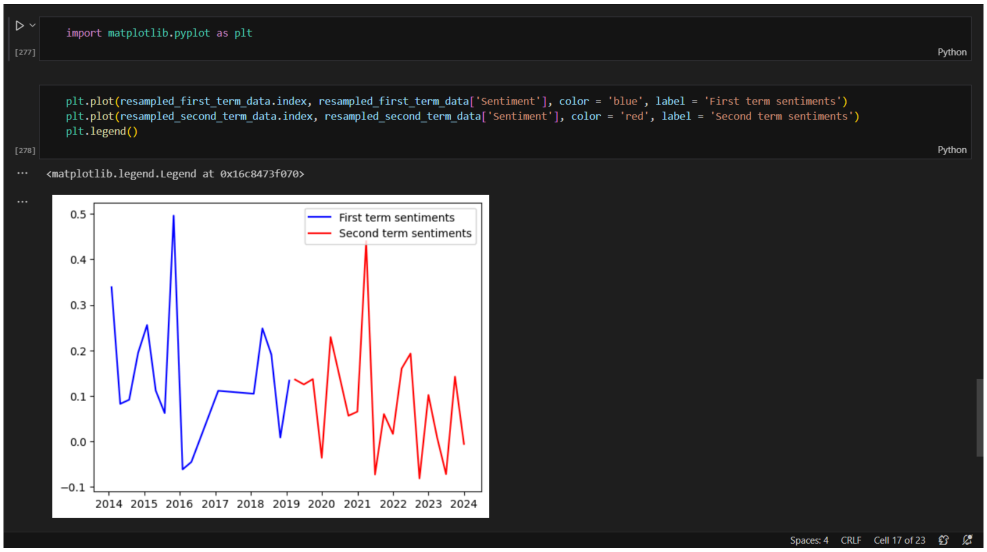


Resampling data year wise

```python
resampled_first_term_data = first_term_data.resample('3M').mean('Sentiment').dropna()
resampled_second_term_data = second_term_data.resample('3M').mean('Sentiment').dropna()
```

```python
resampled_first_term_data.dropna()
```

	Sentiment
2014-01-31	0.339098
2014-04-30	0.082380
2014-07-31	0.091421
2014-10-31	0.194754
2015-01-31	0.255823
2015-04-30	0.111526
2015-07-31	0.062059
2015-10-31	0.496350
2016-01-31	-0.061810
2016-04-30	-0.045445
2017-01-31	0.111186
2018-01-31	0.104659
2018-04-30	0.248532
2018-07-31	0.190900
2018-10-31	0.008397
2019-01-31	0.133613



```
import matplotlib.pyplot as plt
```

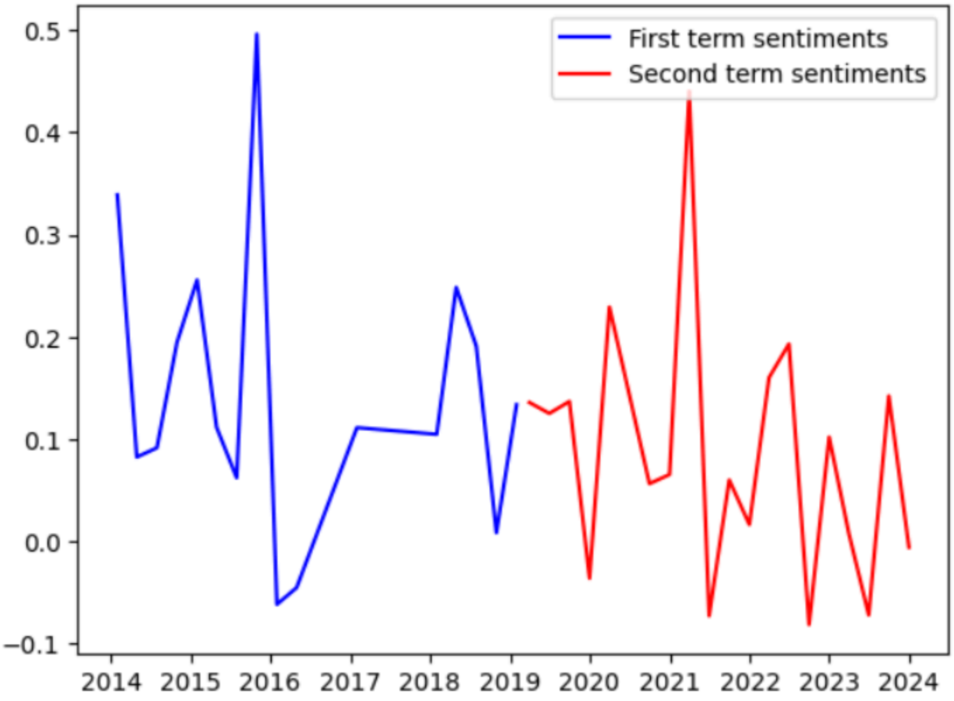
[277] Python

```
plt.plot(resampled_first_term_data.index, resampled_first_term_data['Sentiment'], color = 'blue', label = 'First term sentiments')
plt.plot(resampled_second_term_data.index, resampled_second_term_data['Sentiment'], color = 'red', label = 'Second term sentiments')
plt.legend()
```

[278] Python

```
<matplotlib.legend.Legend at 0x16c8473f070>
```

RESULT:



<table>
<tr><td><b>References:</b></td><td>What is Sentiment Analysis? - Sentiment Analysis Explained - AWS (amazon.com)

Scraping Reddit Data Using Python and PRAW : A Beginner’s Guide | by Archana Kokate | Mar, 2024 | Medium</td></tr>
<tr><td colspan="2"><b>CONCLUSION:</b>

In conclusion, the above mentioned topics and reddit scraping techniques have been used in order to collect and visualize the average sentiment of the reddit comments on “r/india” as given by the python module “VADER Sentiment”.</td></tr>
</table>