You need to select ONE of the following tasks:-

Task A

1. Task A requires you to perform some data collection from Twitter and perform some intelligent analysis using the collected data. We would like you to collect data via the Twitter REST API using hashtags relevant to both **PM Narendra Modi** and **Delhi CM Arvind Kejriwal**. You need to perform analysis on the tweets that you shall collect. We expect you to curate at least 10,000 tweets and store them in a MongoDB collection.

2. From the tweets database that you curate, we would like you to generate a Web Application with visualisation plots and graphs. **The visualization task is slightly open ended so you have freedom to display the data you collect in step-1 in whichever ways you desire. There is no set number of graphs that you should do, however at least 5-6 would be good**: Some basic visualizations we'd like to see are:

- Locations of the Tweet (Use any geo-plotting library of your choice. A few to name are Google Charts, CartoDB)
- Who is more popular **in the state of Delh**i, Narendra Modi or Arvind Kejriwal? (You can create a metric for yourself, and define what it is in the Readme)
- List of Top 10 Hashtags being used in the stream
- Distribution of Original Tweets vs Retweeted Tweets
- Distribution of favorite counts on Original Tweets
- Distribution of Type of Tweet i.e. Text, Image, Text+Image

You would be required to upload your MongoDB collection into a github repository. The WebApp should have cool visualizations of the data that you have collected. Dynamic and interactive graphs will earn brownie points. You are free to explore more visualizations than the ones we have suggested you.

3. Using the data that you have collected in step-1, train a regression model to **predict the number of likes on a tweet.** There should be a section in the web app where we can input a the URL of a tweet and your trained model predicts the number of likes on it.
   For this task you need to use different features to train the model on. The number of features is not limited, and you are free to use whatever data you think contributes to the like count. Some knowledge of how to train machine learning algorithms would be a pro.

   To **test the model** generated in this step, collect the likes count for ~100 tweets (again using the Twitter REST API), and run your model on these URLs. **The input should be the URL of the tweet**. Your trained model will try to predict the number of likes on the tweets in this test set and you will calculate the average accuracy of the model based on

the error in its judgement. Please report this average accuracy of the model when you report results.

Things you will be evaluated upon:-

1. Code Clarity
2. How do you explain your approach: We expect you to put a README.
3. The accuracy of the model
4. The design of your plots and graphs for the visualisation.

Task B

1. Task B requires you to collect data from Facebook and perform analysis using the data. We would like you to collect Facebook Event's data via the Graph API for the following keywords:

- Travel
- Food
- Music
- Art
- Education

You need to perform analysis on the events that you shall collect. We expect you to curate data related to at least 1000 events and store them in a MongoDB collection.

2. From the event database that you curate, we would like you to make a Web Application with visualisation plots and graphs:

- Venue of the Event (Use any geo-plotting library of your choice. A few to name are Google Charts, CartoDB)
- Out of the five categories - travel, food, music, art and education, which category of the events is most popular? (You can create a metric for yourself, and define what it is in the application) -- MUST
- Word Cloud of description of the events
- Distribution of positive response and negative response of the users for various categories.

You would be required to upload your MongoDB collection into a github repository. The WebApp should have cool visualizations of the data that you have collected. Dynamic and

interactive graphs will earn brownie points. You are free to explore more visualizations than the ones we have suggested you.

3. Using the data that you have collected in step-1, train a model to **predict the interested_count on an event.** There should be a section in the web app where we can input an Event ID and your trained model predicts the **interested_count** on it. For this task you need to use different features to train the model on. The number of features is not limited, and you are free to use whatever data you think contributes to the interest_count.

To **test the model** generated in this step, collect the interested_count for 50 events, and run your model on these events. **The input should be the event ID**. Your trained model will try to predict the number of users interested in the event in this test set and you will calculate the average accuracy of the model based on the error in its judgement. Please report this average accuracy of the model when you report results.

Things you will be evaluated upon:-

1. Code Clarity
2. How do you explain your approach: We expect you to put a README.
3. The accuracy of the model
4. The design of your plots and graphs for the visualisation.

Task C

Task C involves development of a Chrome Extension. **The extension should allow user to input a set of user handles and keywords for tracking.** The extension give the user a pop up browser notification when:-
1. Any of the handles that have been added for tracking, post a tweet
2. A tweet is is posted containing a keyword that has been added for tracking

You can use the Twitter Streaming API for both the cases. The extension should allow you to navigate to that specific tweet from the extension button in a new tab.

You will be required to share your output in the form of a **crx file** in a git repository.

For the evaluation we will add a user handle/keyword to the extension in real time to access your performance. Some twitter handles that we would like your extension to follow by default are @narendramodi, @ArvindKejriwal, @ponguru and @jigsaw2212.

Things you will be evaluated upon:-

1. Code Clarity
2. How do you explain your approach: We expect you to put in a README.
3. Design of the extension
4. The real time performance of the extension