

A person's hands are shown typing on a laptop keyboard. The laptop screen displays a web interface, likely a social media platform. A semi-transparent grey diagonal overlay covers the right side of the image, containing the text 'Twitter Sentiment app Part -2' in a bright orange, handwritten-style font. The background is a blurred office or home workspace.

Twitter Sentiment app Part -2

So, what do we have so far....

We built some helpful function to make the connection with the twitter API and further, we were able to get a function to plot the results of the sentiment using the textblob library.

What we were building?

Twitter Sentiment Analyser

Keep your number of tweet less than 1000

Enter the topic..

So, how are we going to build that application...

The application that you just saw is a really big application (although it looks small). There are many small pieces attached to this app which we need to first understand before we actually can start building that app. This is why, we will building this application one by one. In this video, we will be learning about the backend and in the next video, we will build the frontend part. So, let us get started.

Agenda for this video... (App Without any style)

1. Create a templates folder in the project directory
 - a. Create a file called index.html
 - b. Create a file called display.html
2. Create a new folder for static content which also holds a folder named images
3. Initialize a new app with the `static_url_path = "/static"` (This is important)
4. Set up the route ("/") with GET and POST methods
 - a. If `method == post`, then display the desired html page (display.html with an image). But before that, call the functions that we created in the last class to display the image. Save the image in the images folder so that the html page can pick it up on submit.
 - b. Otherwise display an index.html page with two inputs and submit button.

Add Style to the APP

Style for the index page:

1. Add the background color of orange
2. Change the display position of form element to flex
3. Change the flex direction property
4. Set the fixed width to the form and set margin to auto so that app comes to the center.
5. Change the font size
6. Change the color of the button.
7. Change the Header color and button background color
8. Change the button background on hover.

Add Style to the APP (Continued)

Style for the index page:

There is not much to do. Most of the things are there in the image.

1. Change the color of the text
 2. Change the width and margin to auto
 3. So, that the display screen comes to the center of the browser screen
- And we are done..



Thank you