Deployment Instructions, Requirements and Description:

- Firstly, we have to install all the below required libraries(Assumption: Windows OS)
    - Requests
        - pip install requests
    - Bs4
        - pip install bs4
    - Gmplot
        - pip install gmplot
    - Geocoder
        - pip install geocoder
    - Lxml
        - pip install lxml

- Incase, if we have a Google API key, the task is much more easier. Everything can be done accurately and efficiently with minimum latency. Processing of the behind screen activities is fast.
- Else, we have to go for other available options.
- I have divided the entire task into three sub tasks.

❖ SUB - TASK 1

➢ Web Scraping

- Url = https://wiki.openstreetmap.org/wiki/India/Boundaries/States_and_Union_territories
- From the above url, web scraping has been done.
- We have to identify the commonalities of classes,html tags, attribute names etc.. that exist for the required data to be extracted.
- The **Requests** module lets us to integrate python programs with web services.
- **BeautifulSoup** module is used to screen - scrape quickly.

❖ SUB - TASK 2

➢ Geocoding

- In Geocoding, having the address of the location, we can get the Latitude and Longitude details.
- As we don't have the Google API key, I have found a couple of ways to do the same task (Free sources)
  - **Geocoder**
    - https://geocoder.readthedocs.io/providers/OpenStreetMap.html
  - Folium
    - https://python-graph-gallery.com/312-add-markers-on-folium-map/
  - Pygmaps
    - https://www.geeksforgeeks.org/python-plotting-data-on-google-map-using-pygmaps-package/

- The better option among the above three was **Geocoder**.

❖ SUB - TASK 3

➢ Plotting the coordinates on the map.

- Python library **Gmplot** served the purpose.
- Plotting data on Google Maps.
- A matplotlib-like interface to generate the HTML and javascript to render all the data we would like on top of Google Maps.
- This service of Google can also be used without Google API key, but, “For development purposes only” text is displayed on map.
- Reference:
  - https://github.com/vgm64/gmplot

❖ Final Output

➢ After executing **task1.py**

- A new **task_result_map.html** is generated at the same level as that of task1.py
- The states marked on the Google map can be seen by opening it in the browser.

➢ The images related to the output are also attached.(please check it out)

➢ For verification of the output, delete the html file and re execute the task1.py file and task_result_map.html is created again