Subject: Documentation outlining the approach, libraries used and a brief explanation of the code

## APPROACH

1.UNDERSTAND THE TASK GIVEN

The important things to notice in the given task are:

>Creating an automated system using Python

>Using a Linkedin API

>Collecting the competitor’s user ID to their respective linkedin accounts

>Monitoring their linkedin accounts for specific things, like:  
 New connections made, About Us section, Job description, recent posts of the new connection

>The automation should generate a hyper-personalized connection request from my account

2.DATA COLLECTION

To begin the task, various forms of data need to be collected, like:  
Competitors’s ID’s

Linkedin API key

Research on the packages to use

Looking into similar projects from different sources

Researching on Linkedin’s Terms for data scraping

3.PROGRAM WRITING

Outlining the required program structure:

Packages to select

Modules that are needed to use in the package

Dividing the code to required classes

Organizing the methods used to write a clean code

Checking for errors

Testing the code

4. FINISHING THE TASK

Creating a repository for the task

Uploading the files to the repository

Rechecking with task for any changes required

## LIBRARIES USED

The solve the task given, three different libraries are used

>requests

This library is used to send HTTP requests to the linkedin API and retrieve the data from it

You can find the official documentation for the requests library at <https://docs.python-requests.org/en/latest/>

>json

This library is used to handle JSON data, it parse the JSON response received from linkedin API and extract relevant info from them

Its documentation can be found at: <https://docs.python.org/3/library/json.html>

>nltk.sentiment (Natural Language tool kit):

This module from NLTK library is used to perform the sentiment analysis on the text data,

Mostly on the abouts us section of the linkedin profiles

The official documentation of NLTK library is found at: <https://www.nltk.org/>

## EXPLANATION OF THE CODE

The python script automates the monitoring of competitors linkedin activity and generates hyper personalized connection requests

The summary of its functionality is:  
>LinkedinAutomate class is defined which handles the automation tasks

>the class constructor (\_\_init\_\_) initializes the linkedin API and sets up the header for making the API requests

>moniter\_comptitor’s\_activity method takes a list of competitors ID’s as input and perform the tasks like:  
1.retrive new connections for the competitor using LINKEDIN API

2.prints the list of new connection

3.for each new connection, generates a hyper personalized connection request message and sends the connection requests

4.prints the response of each connection request

>””get new connection”” method retrieves the new connections for a specific competitor by making a GET request to the linkedin API

>””generate connection request”” method generates a hyper personalized connection request message for a specific connection based on the sentiment analysis of their linkedin profile about section

>””analyze sentiment”” method uses the “sentiment intensity analyzer ” from the NLTK library to perform sentiment analysis on a given text

>””send connection request ”” method sends a connection request ot specific linkedin profile using the linkedin API

>””main func”” method is the entry point of the script, it defines the competitor ID to monitor and calls the “monitor competitors activity” method to start the automation process

>the linkedin access token is provided and an instance of the linkedin automate class is created with the access token

The main func method is called to initiate the automation

NOTICE-

Access key or LINKEDIN API are needed

libraires requests, json, nltk are needed

Competitor's ids are needed