# DA5402

Assignment 2


Name - Patel Shrey Rakeshkumar
Roll number - ME21B138

## Table of Contents

# 1 Module 1

- I have put all the required library in requirement.txt then with the help of Dockerfile I have build the docker Image which contains all the library mentioned in the requirement.txt file also Chromedriver to extract required site details.
- I have put all the dags python file named DA5402_AS2.py in folder name dags also in the same folder there is a folder named SCRIPTS which contains all the function scripts required for the tasks.
- 'module1.py' contains function named scrape_homepage which scrape google news' homepage as html file and return it.

# 2 Module 2

- 'module2.py' contains function named 'extract_top_stories_link' which takes the html containt and with the help of the class name it finds the link to the Top Stories.

# 3 Module 3

- 'module3.py' contains function named 'extract_thumbnails_and_headlines' which takes and top stories link and excess it through webdeiver chrome.
- It first deals with lazy-scroll by doing scrollby for certain number of times.
- It then goes through all the articles and find thumbnail-images and headline with other meta informations with the help of the class information provided and store them in the list of directories.

# 4 Module 4

- For this module I have created SQLExecuteQueryOperator named 'create_tables' which creates 2 tables as required one with the images and one with the headlines and other meta details also there is reference to the image id in the healine table to indicate the which image belongs to the headline with the connection made in the lab class named 'tutorial_pg_conn'.
- There is a function named 'process_records' in 'module4.py' which takes the image link and download the image and then convert it to base64 encoded which makes it database friendly.
- Then this data is sent to another SQLExecuteQueryOperator named 'insert_data' which excess postgres database and insert the datas in the list which are not already there also it track how many new entries has been added and returns it.
- This number (the number of new entries) is sent to function named 'write_status' in 'module4.py' which creates file named status in dags/run with this number inside it.

# 5 Module 5

- All the above tasks have been added in airflow dag with the help of various operators like pythonoperator and SQLExecuteQueryOperator.
- I have created the pipeline hierarchy for all the operations to do.
- The dag will run every hour to scrape new ¡image, headline¿ tuple.

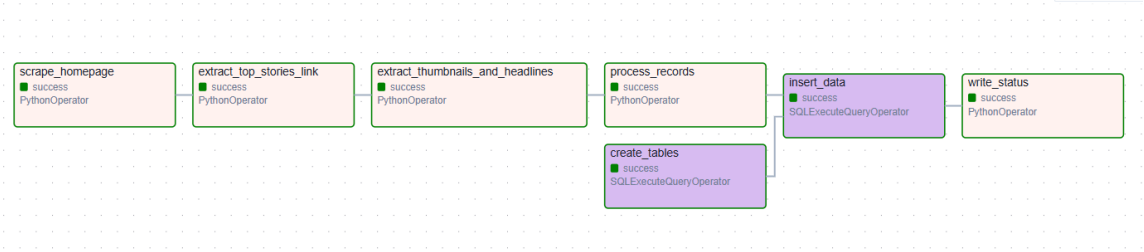


Figure 1: Graphical representation of the hierarchy of operations of DAG 1

# 6 Module 6

- In the same python file 'DA5402_AS2.py' there is another dag which take care of module 6.
- This dag use FileSensor with 'fs_default' which pokes every 30 seconds and look for file 'dags/run/status'. As soon as it finds it it returns it.
- Then the fucntion called 'send_email_notification' is called which read integer from the status file and if it is non zero then it sends the update email.
- For email SMTP is used with the help of the smtplib library.
- SMTP_HOST is set up for the gmail already if anyother mail id is to used then it needed to changed accordingly (smail works fine with gmail already.) and SMTP_PORT to 587.
- SMTP_USERNAME and MY_EMAIL is set to the mail id which needed to be mailed and accordingly set the SMTP_PASSWORD.
- After mail is send, 'delete_status_file' is called which delete the status file and set up the whole enviroment for the future runs.

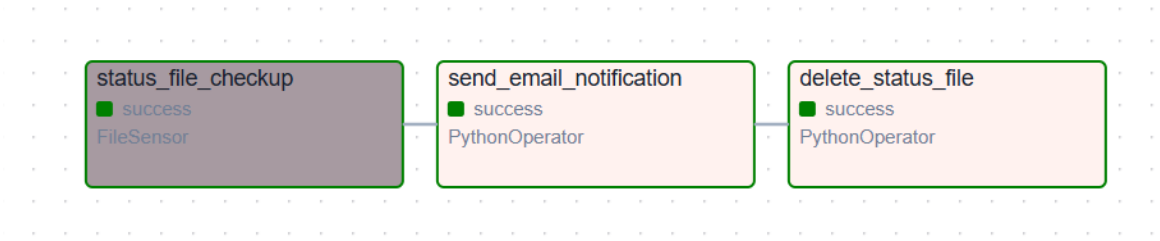


Figure 2: Graphical representation of the hierarchy of operations of DAG 2


Figure 3: Example of Email recieved