# Let’s use Operators

It’s time to add tasks to the DAG

# Waiting_file_task

- First task to execute.
- Sensor Operator
- Loops every 15 seconds to check if the file data.csv is in /home/airflow/first_dag/

# Fetching_tweet_task

- Second task to execute.
- Python Operator
- Executes the “fetching_tweet.py” script in order to produce the data_fetched.csv into /tmp/ folder

# Cleaning_tweet_task

- Third task to execute
- Python Operator
- Executes the “cleaning_tweet.py” script in order to produce the data_cleaned.csv into /tmp/ folder

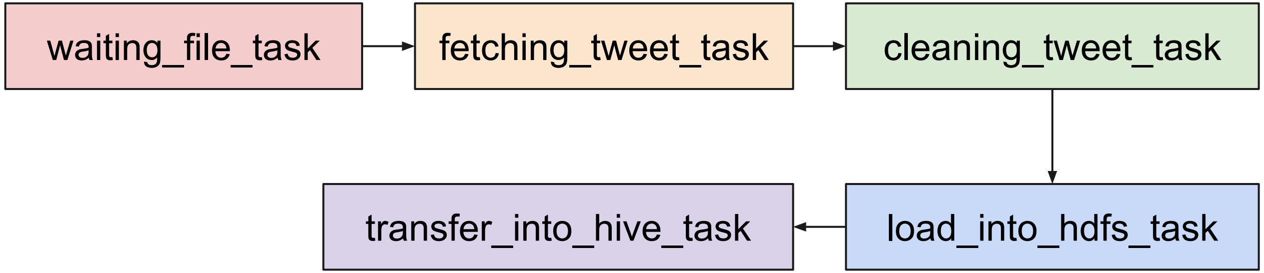
# Load_into_hdfs_task

- Fourth task to execute
- Bash Operator
- Uploads the data_cleaned.csv file into the HDFS at the following location = '/tmp'
- You can type the following command to see if your file is in the directory
  - `hadoop fs -ls /tmp`

# Load_into_hive_task

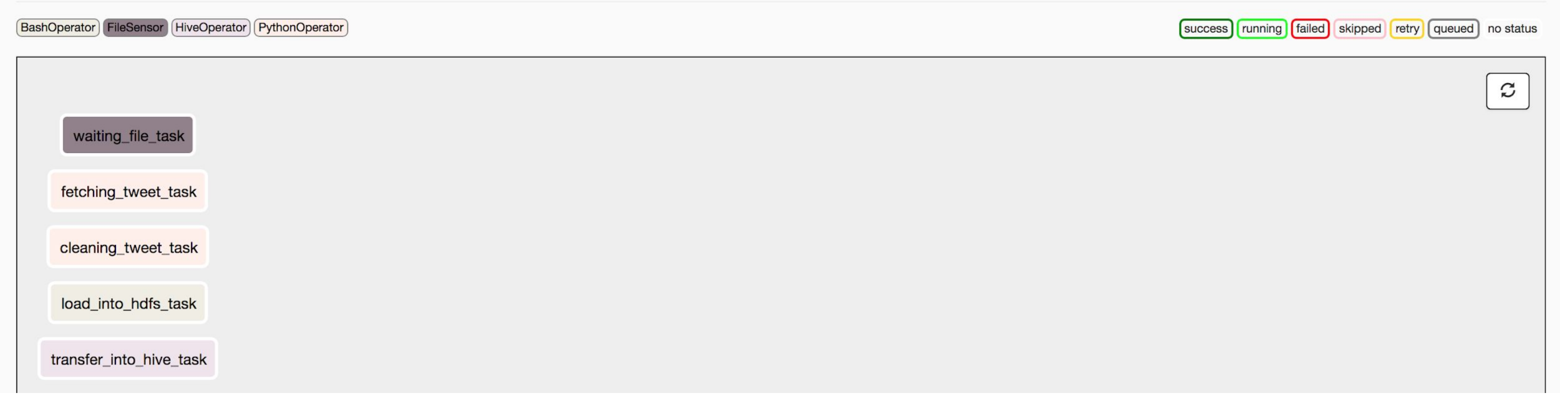
- Fifth task to execute
- Hive Operator
- Load the data from HDFS to HIVE in order to use SQL to request data.

# DAG Schema


waiting_file_task
fetching_tweet_task
cleaning_tweet_task
load_into_hdfs_task
transfer_into_hive_task

# Airflow UI

- Now if you click on your DAG and go to the GraphView from the Airflow UI you should see this:



# What’s Next?

- So, what we’ve done so far?
  - We created python scripts to fetch and clean tweets.
  - We initialized the DAG with default arguments.
  - We initialized different operators according to the tasks we want to achieve.
- As you have seen from previous slides, what we have from the DAG schema differs from the DAG showed into the Airflow UI.
- In the next section we gonna add the missing dependencies.