# Building ETL pipeline with Reddit data using Airflow

Steps to perfrom the whole process :

1. Extracting data from reddit api using python, transforming the data and loading the final results in csv format (ETL).
2. Installing ubuntu in Windows using wsl2.
3. Running airflow on ubuntu in windows using wsl2.
4. Creating DAG(Directed Acyclic Graph) using python operator provided by Airflow.
5. Running the DAG and loading the csv file in the local machine

## To extract data using reddit api follow the instructions provided by https://www.jcchouinard.com/reddit-api/

Create the ETL script using python

In [ ]:
```
import requests
import pandas as pd
import json

subreddit = 'jokes'
limit = 100
timeframe = 'month' #hour, day, week, month, year, all
listing = 'hot' # controversial, best, hot, new, random, rising, top

def run_reddit_etl():

    def get_reddit(subreddit,listing,limit,timeframe):
        try:
            base_url = f'https://www.reddit.com/r/{subreddit}/{listing}.json?limit=
            request = requests.get(base_url, headers = {'User-agent': 'yourbot'})
        except:
            print('An Error Occured')
        return request.json()

    def get_results(r):
        #Create a DataFrame Showing Title, URL, Score and Number of Comments.
        myDict = {}
        for post in r['data']['children']:
            myDict[post['data']['title']] = {'url':post['data']['url'],'score':post
        df = pd.DataFrame.from_dict(myDict, orient='index')
        df.to_csv("D:/Airflow_Project/Reddit_hot_100_month_jokes.csv")
        return df

    r = get_reddit(subreddit,listing,limit,timeframe)
    df = get_results(r)

if __name__ == '__main__':
    run_reddit_etl()
```

## To run ubuntu on WIndows using wsl2 follow this link https://learn.microsoft.com/en-us/windows/wsl/install

Run these commands once ubuntu is up and running.

1. sudo apt-get update
2. sudo apt install python3-pip
3. sudo pip install apache-airflow
4. sudo pip install pandas
5. sudo pip install requests

Next create the DAG file

In [ ]:
```
from datetime import timedelta
from airflow import DAG
from airflow.operators.python_operator import PythonOperator
from airflow.utils.dates import days_ago
from datetime import datetime
from reddit_etl import run_reddit_etl

default_args = {
    'owner': 'airflow',
    'depends_on_past': False,
    'start_date': datetime(2020, 11, 8),
    'email': ['airflow@example.com'],
    'email_on_failure': False,
    'email_on_retry': False,
    'retries': 1,
    'retry_delay': timedelta(minutes=1)
}

dag = DAG(
    'reddit_dag',
    default_args=default_args,
    description='Our first DAG with ETL process!',
    schedule_interval=timedelta(days=1),
)

run_etl = PythonOperator(
    task_id='complete_reddit_etl',
    python_callable=run_reddit_etl,
    dag=dag,
)

run_etl
```

Run these commands in the ubuntu window and follow the instructions.

1. airflow db init
2. airflow users create --username admin --firstname --lastname --role Admin --email admin@example.com
3. To start airflow just run the command : airflow standalone
4. Visit http://localhost:8080 in your web browser to access the Airflow web interface.

Airflow will start running

Next to create the DAG in airflow and running the etl pipeline we need to follow the next steps.

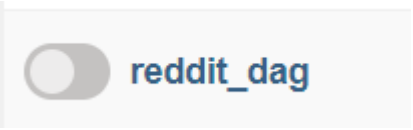
```
root@Galib_MSI:~# ls
airflow
root@Galib_MSI:~# cd airflow
root@Galib_MSI:~/airflow# ls
airflow-webserver.pid  airflow.cfg  airflow.db  logs  reddit_dag  webserver_config.py
root@Galib_MSI:~/airflow# vi airflow.cfg
```

Once inside the vi editor make sure to change the DAG name to reddit_dag

```
[core]
# The folder where your airflow pipelines live, most likely a
# subfolder in a code repository. This path must be absolute.
#
# Variable: AIRFLOW__CORE__DAGS_FOLDER
#
dags_folder = /root/airflow/reddit_dag

# Hostname by providing a path to a callable, which will resolve the hostname.
# The format is "package.function".
#
# For example, default value "airflow.utils.net.getfqdn" means that result from patched
# version of socket.getfqdn() - see https://github.com/python/cpython/issues/49254.
#
# No argument should be required in the function specified.
# If using IP address as hostname is preferred, use value ``airflow.utils.net.get_host_ip_address``
#
# Variable: AIRFLOW__CORE__HOSTNAME_CALLABLE
#
hostname_callable = airflow.utils.net.getfqdn
```

Refresh airflow and we will see a new dag created by name of reddit_dag




Trigger the DAG and once the job is finished we could find the final csv file in the appropriate location.