Setup+Project: Reddit, Airflow, AWS

1) Create a VM on Google Cloud. Edit Default network and enter 8080 in TCP port.

2) In SSH

```
sudo apt install python3 python3-pip python3-venv
source airflow-env/bin/activate
python3 -m venv ~/airflow-env
pip3 install apache-airflow
pip install apache-airflow apache-airflow-providers-amazon praw pandas
pip install apache-airflow-providers-amazon
pip install apache-airflow-providers-http
pip install boto3 praw pandas
pip install praw
pip install boto3
nohup airflow standalone > airflow.log
```

3) Open Airflow→Admin→Connections

   # *Create an AWS Connection (Optional, for modularity)*:-

   Conn Id: aws_default
   Conn Type: Amazon Web Services
   Standard Fields:
      Login: AWS _ID *(Enter your AWS_ID)*
      Password: *(Enter your AWS Password)*
   Extra Fields JSON:

```
{
"region_name": "us-east-1"
}
```

   # *Create an Reddit API (Optional, for modularity)*:-

   Conn Id: reddit_api
   Conn Type: HTTP
   Host: https://oauth.reddit.com

4) Create a new App on Reddit website.
      Name: RedditETL
      Type: script
      Redirect URI: http://localhost:8080
   (Note down the client id, secret id)

5) In AWS IAM, Create a New User (I have named it as “airflow-s3-user”).
   For the New User, Add Permissions→Attach policies directly:
      AmazonS3FullAccess

AWSGlueServiceRole (for later Glue steps)
AmazonAthenaFullAccess (for future use)
AmazonRedshiftFullAccess (for final stage)

Create Access Key for the IAM User:

Open User→Security credentials→Access keys section→Create access key→use case→Application running outside AWS→Description→”airflow pipeline”→click create access key

(Note down the Access Key ID, Secret Access Key)

6) In Airflow→Admin→Variables

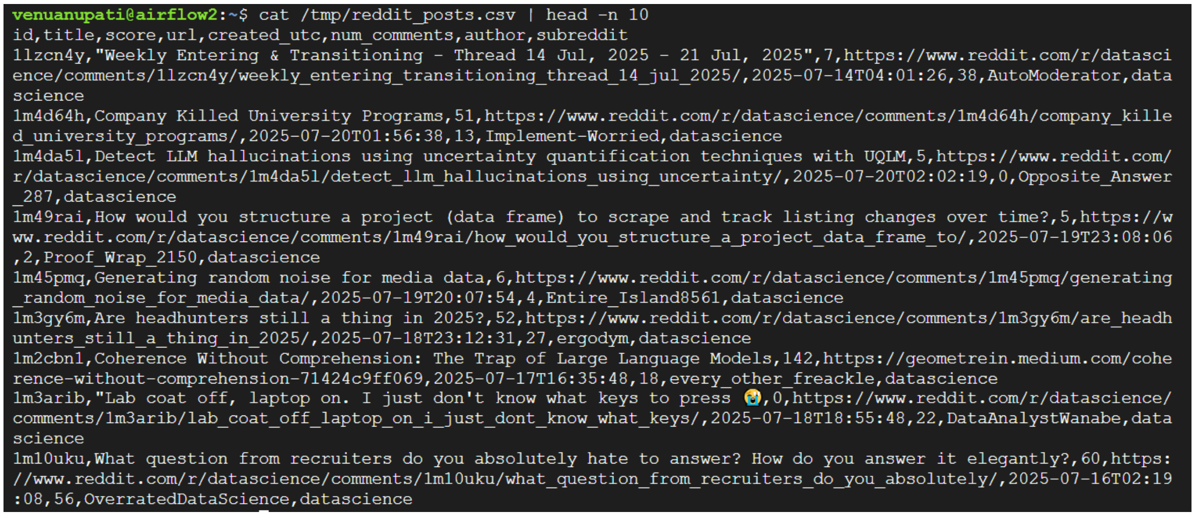
# *Create a Reddit Variable*:-
Key: reddit_config
Value:

```
{
"client_id": "YOUR_CLIENT_ID",
"client_secret": "YOUR_CLIENT_SECRET",
"user_agent": "airflow-reddit-etl-script by OkCellist3772" (Edit this as per your choice)
}
```

# *Create a AWS Variable*:-
Key: aws_credentials
Value:

```
{
"aws_access_key_id": "YOUR_ACCESS_KEY_ID",
"aws_secret_access_key": "YOUR_SECRET_ACCESS_KEY",
"region_name": "us-east-1"
}
```

7) IN SSH, Create Dags directory and create dag1.py file and paste the code from “DAG_1 (Extract Data from Reddit).py”. This should import the dag to the airflow. Trigger the DAG and it will create a CSV file in your VM. Check it by running “cat /tmp/reddit_posts.csv | head -n 10”. This will print something like in the below image.



```
venuanupati@airflow2:~$ cat /tmp/reddit_posts.csv | head -n 10
id,title,score,url,created_utc,num_comments,author,subreddit
1lzcn4y,"Weekly Entering & Transitioning - Thread 14 Jul, 2025 - 21 Jul, 2025",7,https://www.reddit.com/r/datasci
ence/comments/1lzcn4y/weekly_entering_transitioning_thread_14_jul_2025/,2025-07-14T04:01:26,38,AutoModerator,data
science
1m4d64h,Company Killed University Programs,51,https://www.reddit.com/r/datascience/comments/1m4d64h/company_kille
d_university_programs/,2025-07-20T01:56:38,13,Implement-Worried,datascience
1m4da5l,Detect LLM hallucinations using uncertainty quantification techniques with UQLM,5,https://www.reddit.com/
r/datascience/comments/1m4da5l/detect_llm_hallucinations_using_uncertainty/,2025-07-20T02:02:19,0,Opposite_Answer
_287,datascience
1m49rai,How would you structure a project (data frame) to scrape and track listing changes over time?,5,https://w
ww.reddit.com/r/datascience/comments/1m49rai/how_would_you_structure_a_project_data_frame_to/,2025-07-19T23:08:06
,2,Proof_Wrap_2150,datascience
1m45pmq,Generating random noise for media data,6,https://www.reddit.com/r/datascience/comments/1m45pmq/generating
_random_noise_for_media_data/,2025-07-19T20:07:54,4,Entire_Island8561,datascience
1m3gy6m,Are headhunters still a thing in 2025?,52,https://www.reddit.com/r/datascience/comments/1m3gy6m/are_headh
unters_still_a_thing_in_2025/,2025-07-18T23:12:31,27,ergodym,datascience
1m2cbn1,Coherence Without Comprehension: The Trap of Large Language Models,142,https://geometrein.medium.com/cohe
rence-without-comprehension-71424c9ff069,2025-07-17T16:35:48,18,every_other_freackle,datascience
1m3arib,"Lab coat off, laptop on. I just don't know what keys to press 😭,0,https://www.reddit.com/r/datascience/
comments/1m3arib/lab_coat_off_laptop_on_i_just_dont_know_what_keys/,2025-07-18T18:55:48,22,DataAnalystWanabe,data
science
1m10uku,What question from recruiters do you absolutely hate to answer? How do you answer it elegantly?,60,https:
//www.reddit.com/r/datascience/comments/1m10uku/what_question_from_recruiters_do_you_absolutely/,2025-07-16T02:19
:08,56,OverratedDataScience,datascience
```

8) Next Step is to Upload the File to AWS S3 via Airflow. Open AWS S3, Create a new bucket (I have named it as “reddit-etl-pipeline-data”).
   Create a new VS Code file with the name “DAG_2 (Store Data into S3).py”. In addition to the previous code, add a new function to pick the CSV file that we saved previously and to store it into the S3 bucket. Also mention a new DAG Task code for this process. In SSH, paste this code in a new “dag2.py” file in the dags folder and save it.
   Trigger the new DAG, the cvs file will be successfully stored in the S3 bucket.

9) Create a Glue Database:

   Open AWS Glue→Databases→Add database→Named it “reddit_etl_db”→Create

   Create a Glue Crawler:

   Click Crawlers→reddit_posts_crawler→In Data Source Choose S3 Bucket→Choose crawl subfolders→Next→Create new IAM role→Name it AWSGlueServiceRole-reddit-etl→Next→Choose previously created Glue database: reddit_etl_db→ Schedule: Choose Run on demand→Next Click Create

   Run the Crawler, This will move the data file from S3 to the **Tables** under Glue.

10) In Tables→On the data file that we moved just now, click data quality→create data quality rules→Name it and add Rules
    I have added below rules:

    Rules = [IsUnique "id", IsComplete "id", IsComplete "title"]

    Save it and test it. Good that I got “Data quality score is 100% (3/3 rules passed)”

11) Use Athena to query the Glue Table to confirm that Glue correctly cataloged the CSV. Open AWS Athena, and set the query result location (I have set it as “s3://reddit-etl-pipeline-data/athena-query-results/”) and save it.

Next, Open the Database→Tables→Under View data, click Table data. You can see the Reddit post data returned from S3 via Athena SQL.

Run a Custom SQL Query (Example):

```
SELECT title, score, num_comments
FROM reddit_posts
ORDER BY score DESC
LIMIT 10;
```

Or:

```
SELECT COUNT(*) AS total_posts, MAX(score) AS max_score
FROM reddit_posts;
```

This confirms that the glue table structure is accurate, athena can query it successfully, and we can now transform, filter, or clean data with SQL if needed.

12) Load the data into Redshift for long-term warehousing.

    Click Create Cluster→Choose Redshift Serverless (Workgroup name: reddit-etl-wg, Namespace: reddit-etl-ns)→Choose default admin user or Create (Username: awsuser, Password: ****)→Keep defaults for rest→Click Create Workgroup

    Wait 2 to 4 mins for setup.

13) Configure S3 access in Redshift.

    Go to IAM → Roles→Create a new role (Trusted entity: Redshift, Policies: Attach AmazonS3ReadOnlyAccess)

    Attach this IAM role to your Redshift workgroup (Go to Redshift→Workgroups→Click your workgroup→Click Permissions→Associate IAM Role

    This lets Redshift access the S3 bucket holding the CSV.

14) Connect to Redshift Query Editor V2.

    Go to Query Editor V2→Choose (Workgroup: reddit-etl-wg, Database: dev or whatever you chose, User: awsuser)

15) Create a table in Redshift to match CSV. Paste this SQL:

```
CREATE TABLE reddit_posts (
    id VARCHAR,
    title VARCHAR,
    score INTEGER,
    url VARCHAR,
    created_utc TIMESTAMP,
    num_comments INTEGER,
    author VARCHAR,
    subreddit VARCHAR
);
```

16) Next, copy data from S3 to Redshift. Use this command:

```
COPY reddit_posts
FROM 's3://reddit-etl-pipeline-data/raw/reddit_posts.csv'
IAM_ROLE 'arn:aws:iam::<your-account-id>:role/<your-redshift-s3-role>'
FORMAT AS CSV
IGNOREHEADER 1;
```

Run the query. You will see COPY complete with rows loaded. Next, Validate the load by running the below command:

```
SELECT COUNT(*) FROM reddit_posts;
```

17) Did some analysis by running some queries in the Query Editor v2.

Find Top Reddit Posts:

```
SELECT title, score, num_comments
FROM reddit_posts
ORDER BY score DESC
LIMIT 10;
```

Aggregate Stats:

```
SELECT COUNT(*) AS total_posts, AVG(score) AS avg_score,
MAX(num_comments) AS most_commented
FROM reddit_posts;
```

Trends by Date:

```
SELECT DATE(created_utc) AS post_date, COUNT(*) AS daily_posts
FROM reddit_posts
GROUP BY post_date
ORDER BY post_date DESC;
```

18) Create a Full DAG: Reddit → S3 → Redshift. This DAG will:

Extract posts from Reddit (praw),
Save as CSV,
Upload to S3 (boto3),
Load into Redshift via COPY command

In Airflow→Admin→Variables
<u>Create a Redshift Credentials</u>:-
Key: redshift_credentials
Value:

```
{
"host": "your-redshift-endpoint.region.redshift.amazonaws.com",
"database": "dev",
"user": "awsuser",
"password": "your_password"
}
```

19) Next, I created a new file on VS Code with the name ”DAG_3 (Reddit ETL Full DAG).py”. In addition to the previous code, included a new task to Load data into Amazon Redshift via COPY from S3. This makes the final DAG code a End-to-End ETL pipeline: Extract → S3 → Load into Warehouse.
In SSH, create a new file in the dags folder with the name “dag3.py”. Paste this code in the file and save it. Trigger this DAG in the Airflow. All the 3 tasks successfully ran: extract_reddit_data, upload_to_s3, load_to_redshift.

Note: This is an ELT pipeline. In the entire document I was mentioning the word “ETL”. But it is an ETL process. I will re-edit the document changing the mention of the word “ETL” to “ELT”.