Practical Exercise: Automating an ETL Workflow Using Apache Airflow

Step-by-Step Instructions

Step 1: Set Up Apache Airflow

- Follow the Apache Airflow Installation Guide to install Airflow.
- Once installed, start the Airflow scheduler and web server:
  - airflow scheduler airflow webserver - port 8080.
  - Access the Airflow UI at http://localhost:8080.

Step 2: Set Up MongoDB

- Install MongoDB by following the MongoDB installation guide.
- Start the MongoDB service locally or use MongoDB Atlas, a cloud-based service.

Step 3: Create the DAG File

- In the dags/ directory of your Airflow installation, create a DAG file, for example: etl_pipeline_mongodb.py.
- Define the DAG with three tasks: extract_data, transform_data, and load_data. You can use the following code to define the DAG:

```
from airflow import DAG
from airflow.operators.python_operator import PythonOperator
from datetime import datetime
import pandas as pd
from pymongo import MongoClient

# Define default arguments for the DAG
default_args = {
    'owner': 'airflow',
    'start_date': datetime(2023, 1, 1),
    'retries': 1,
}

# Define the DAG
dag = DAG(
    'etl_pipeline_mongodb',
    default_args=default_args,
    schedule_interval='0 6 * * *',  # Run every day at 6:00 AM
```

)

Step 4: Create the Sales CSV File

Create a file named sales.csv in a directory accessible by your Airflow DAG. It should contain sales data like the example below:

```
transaction_id,customer_id,product_id,quantity,price
T001,C001,P001,2,100
T002,C002,P002,1,200
T003,C003,P003,3,50
```

Step 5: Run the DAG

- Start the Airflow web server and navigate to http://localhost:8080.
- Activate your DAG (etl_pipeline_mongodb) in the Airflow UI.
- Manually trigger the DAG to test it. Verify the tasks in the DAG and check the logs to ensure successful execution.

Step 6: Schedule the DAG

By default, the DAG will be scheduled to run at 6:00 AM every day (schedule_interval='0 6 * * *'). Verify the schedule by checking the DAG's configuration in the Airflow UI.