Task Title: Automate Apache Airflow Infrastructure Setup on AWS Using Terraform and Jenkins

==========

Objective:
==========

Set up a fully automated process to deploy Apache Airflow in a Dockerized environment on AWS using Terraform. The entire deployment should be triggered and managed via a Jenkins CI/CD pipeline.

Expected Deliverables:
==========

=> *** Terraform Code ***

- To provision AWS infrastructure:

- EC2 instance for Airflow (Ubuntu + Docker)

- RDS PostgreSQL instance for Airflow metadata

- S3 bucket for DAGs and logs

=> *** Airflow Deployment ***
==========

- Use Docker Compose to deploy Airflow services: webserver, scheduler, worker, and triggerer

- Configure connection to the RDS instance

- Configure DAG sync from S3 bucket to Airflow DAGs folder

- Jenkins Pipeline Setup

- Jenkins should be set up on a separate EC2 instance

=> Jenkins pipeline stages should include:

- Terraform plan, apply, and destroy

- Docker-based Airflow deployment after successful infrastructure provisioning

Optional: Upload DAG files from GitHub to S3 or EC2

==========

How to Approach the Task:
==========

=> Phase 1: Infra Provisioning

- Write Terraform modules to provision the EC2, RDS, and S3 bucket

- Test them manually

=> Phase 2: Airflow Container Setup

- On the EC2 instance, install Docker and Docker Compose

- Deploy Airflow using docker-compose.yml

- Connect Airflow to RDS, sync DAGs from S3

Phase 3: Jenkins Automation

- Set up a new EC2 instance and install Jenkins

- Configure a Jenkins pipeline using Jenkinsfile

- Automate the above phases using Jenkins jobs