**Professional Curriculum Overview: Data Engineering Program for Trestle Academy Ghana**

* **Duration**: 2 Months (8 weeks)
* **Frequency**: Weekly, 2-hour sessions
* **Goal**: Equip students with job-ready, practical data engineering skills, focusing on deploying pipelines with real-world tools.
* **Tools**: Python, SQL, AWS (S3, Redshift), Snowflake, Apache Airflow, Docker

**Week-by-Week Curriculum**

**Week 1: Data Engineering Foundations & Python Basics**

* **Objective**: Introduce data engineering, set up Python environments, and cover Python basics for data manipulation.
* **Topics**:
  + Introduction to data engineering and pipeline basics.
  + Python basics: data types, variables, and control structures.
  + Python libraries for data engineering (Pandas, NumPy).
* **Hands-On**:
  + Students create Python scripts to clean and manipulate a sample dataset.
* **Project**: Build a data cleaning script to normalize a messy dataset, preparing it for pipeline ingestion.

**Week 2: SQL Fundamentals and Databases**

* **Objective**: Learn SQL basics and relational database concepts essential for data extraction.
* **Topics**:
  + SQL commands: SELECT, WHERE, ORDER BY, GROUP BY, JOINs.
  + Database design and normalization basics.
  + Connect Python to SQLite or PostgreSQL for data extraction and transformation.
* **Hands-On**:
  + Write SQL queries to filter, aggregate, and join data.
  + Extract data from a local database, process it in Python.
* **Project**: Create an SQL-based data exploration and reporting script that extracts insights from a sample database.

**Week 3: Cloud Storage with AWS S3**

* **Objective**: Understand cloud storage and work with AWS S3 for data storage and retrieval.
* **Topics**:
  + Introduction to AWS S3 and cloud storage concepts.
  + Uploading and retrieving data from S3 using boto3.
  + Managing permissions and storage classes.
* **Hands-On**:
  + Set up an S3 bucket, upload datasets, and retrieve data using Python.
* **Project**: Build an S3 data ingestion pipeline where cleaned data is uploaded to S3 for future analysis.

**Week 4: Data Warehousing with AWS Redshift & Snowflake**

* **Objective**: Learn about data warehousing concepts and use AWS Redshift and Snowflake for large-scale data storage.
* **Topics**:
  + Introduction to data warehouses and ETL vs. ELT.
  + Working with AWS Redshift: loading and querying data.
  + Overview of Snowflake and its architecture.
* **Hands-On**:
  + Load data from S3 into Redshift, and query it using SQL.
  + Connect to Snowflake, create tables, and import data.
* **Project**: Implement an ELT process that loads data into Redshift and performs a simple analysis on it.

**Week 5: Orchestration with Apache Airflow**

* **Objective**: Introduce data pipeline orchestration and automate workflows with Apache Airflow.
* **Topics**:
  + Introduction to Apache Airflow and DAGs (Directed Acyclic Graphs).
  + Setting up Airflow locally, writing and scheduling DAGs.
  + Integrating with S3 and Redshift for automated ETL tasks.
* **Hands-On**:
  + Create an Airflow DAG that ingests data from S3 and loads it into Redshift.
* **Project**: Build an automated ETL pipeline with Airflow to extract data from S3, transform it, and load it into Redshift on a schedule.

**Week 6: Containerization with Docker**

* **Objective**: Learn the basics of containerization to deploy reproducible data engineering environments.
* **Topics**:
  + Introduction to Docker and containerization principles.
  + Writing Dockerfiles to containerize Python and Airflow applications.
  + Deploying a containerized Airflow pipeline.
* **Hands-On**:
  + Containerize the Python and Airflow setup created in previous weeks.
* **Project**: Build a Docker container for the Airflow ETL pipeline, deploy it, and test the deployment on a local machine.

**Week 7: Deploying Data Pipelines on AWS (S3, Redshift, Lambda)**

* **Objective**: Deploy data pipelines to AWS, leveraging serverless and managed services for scalability.
* **Topics**:
  + Setting up AWS Lambda to trigger pipeline components.
  + Integrating S3, Redshift, and Lambda for automated workflows.
  + Security best practices (IAM roles, permissions).
* **Hands-On**:
  + Use Lambda to trigger data loading processes into Redshift from S3.
* **Project**: Deploy a serverless data pipeline with AWS Lambda that ingests data from S3, transforms it, and loads it into Redshift, triggered by file uploads.

**Week 8: Capstone Project & Deployment to Cloud**

* **Objective**: Implement and deploy a full-scale, end-to-end data pipeline project.
* **Project Overview**:
  + **Objective**: Design a data pipeline that extracts, transforms, and loads data from a public API to a data warehouse.
  + **Steps**:
    1. **Data Extraction**: Pull data from an API.
    2. **Data Transformation**: Clean and transform the data with Python.
    3. **Data Storage**: Upload data to S3.
    4. **Data Loading**: Load data into Redshift.
    5. **Orchestration**: Use Apache Airflow to automate the pipeline.
    6. **Containerization**: Deploy the pipeline with Docker.
* **Presentation**:
  + Students present their projects, explaining pipeline components, data flow, and deployment steps.