DATA ENGINEERING | INTERMEDIATE | 5 DAYS

Data Engineering on AWS

Build a strong foundation in building modern data platforms on AWS, from data lake design to batch and streaming pipelines. Learn how to use S3, Glue, Athena, Redshift, and orchestration patterns to deliver secure, scalable, observable pipelines aligned with real enterprise data engineering practices.

Why AWS Data Engineering?

Faster cloud adoption

Build scalable data platforms using managed AWS services.

Improved reliability

Leverage AWS-native monitoring and resilient architectures.

Cost-effective scalability

Pay-as-you-go infrastructure supports growth without upfront investment.

Accelerated time-to-insight

Deliver analytics faster with integrated services like S3, Glue, Athena, and Redshift.

Better security controls

Use IAM, encryption, and governance tools to protect data assets.

What you'll be able to do

- Understand modern data engineering architectures on AWS for batch and streaming workloads.
- Design end-to-end AWS data pipelines for ingestion, storage, transformation, orchestration, and serving.
- Ingest data using batch, event-based patterns, and CDC concepts.
- Build scalable storage layers using Amazon S3 with partitioning and columnar formats like Parquet.
- Transform and query data using AWS Glue, Amazon Athena, and Amazon Redshift.
- Orchestrate workflows using Amazon MWAA (Managed Airflow) and understand Step Functions (overview).
- Build real-time streaming pipelines using Amazon Kinesis Data Streams and Firehose.
- Implement governance and security practices using IAM, KMS encryption, and Lake Formation concepts.
- Apply data quality and observability practices for reliable pipeline operations.
- Deliver a complete end-to-end AWS data engineering capstone project.

Who this is for

- Data Engineers
- Analytics Engineers
- Cloud Engineers supporting data platforms
- Data Platform Engineers
- DevOps Engineers working with AWS data systems
- BI Engineers who want deeper engineering capability

Curriculum

01 AWS Data Engineering Architecture (Batch and Streaming)

- Modern AWS data platform reference architectures (lake, warehouse, lakehouse concepts)
- Batch vs streaming workloads and how AWS services map to each
- End-to-end pipeline stages (ingestion, storage, transform, orchestration, serving)
- Choosing services (S3, Glue, Athena, Redshift, Kinesis, MWAA)
- **Hands-on:** Activity: Design an AWS reference architecture for an analytics and streaming use case

02 Designing End-to-End AWS Data Pipelines (Ingest → Store → Transform → Serve)

- Pipeline design patterns (raw → cleansed → curated → marts)
- Data contracts, schema evolution concepts, and partition strategy planning
- Operational requirements (SLA, backfills, idempotency basics)
- Cost and performance drivers across storage and query layers
- **Hands-on:** Workshop: Create an end-to-end pipeline blueprint with data zones, partitions, and SLAs

03 Ingestion Patterns on AWS (Batch, Event, CDC Concepts)

- Batch ingestion patterns to S3 (scheduled loads, file drops)
- Event-based ingestion concepts (S3 events, pub/sub patterns overview)
- CDC concepts (insert/update/delete capture and downstream processing)
- Idempotency patterns for ingestion pipelines
- **Hands-on:** Lab: Implement a batch ingestion workflow into S3 raw zone with partitioned folder layout

04 Scalable Storage on S3 (Partitioning + Parquet)

- S3 as a data lake foundation (buckets, prefixes, lifecycle basics)
- Partitioning strategies (by date, region, tenant) and common pitfalls
- Columnar formats (Parquet) and why they improve query performance
- File sizing and small files problem (compaction concept)
- **Hands-on:** Lab: Convert raw CSV/JSON into partitioned Parquet in S3 and validate folder structure

05 Transform and Query with AWS Glue (ETL Core)

- Glue jobs overview (Spark ETL, DynamicFrames, DataFrames)
- Glue Data Catalog usage (databases, tables, crawlers, partitions)
- ETL patterns (cleanse, dedup, join, aggregate) into curated zone
- Incremental processing concepts (partition loads, watermark ideas)
- **Hands-on:** Lab: Build a Glue ETL job to transform raw data into curated Parquet/Delta-style layout (Parquet-based)

06 Analytics with Athena (Querying the Lake)

- Athena fundamentals (serverless SQL on S3)
- Partition pruning and performance best practices
- Using Glue Catalog tables with Athena
- Building views for reporting and BI consumption
- **Hands-on:** Lab: Query curated S3 datasets using Athena and build analytics views

07 Serving Layer with Amazon Redshift (Warehouse Overview + Integration)

- When to use Redshift vs Athena (workload and performance trade-offs)
- Redshift basics (schemas, distribution/sort keys concept)
- Loading curated data into Redshift (ELT patterns overview)
- Warehouse modeling basics (facts/dimensions for analytics)
- **Hands-on:** Lab: Load curated datasets into Redshift and run KPI queries (revenue, top customers)

08 Orchestration with Amazon MWAA (Managed Airflow) + Step Functions Overview

- MWAA fundamentals (DAGs, tasks, scheduling)
- Orchestrating ingestion → Glue → Athena/Redshift workflow
- Retries, alerts, and operational visibility in Airflow
- Step Functions overview (when to use vs Airflow)
- **Hands-on:** Lab: Build an MWAA DAG to run an end-to-end batch pipeline with retries and monitoring

09 Streaming Pipelines with Kinesis Data Streams and Firehose

- Streaming fundamentals (events, partitions/shards, consumer groups concept)
- Kinesis Data Streams architecture (producers, shards, consumers)
- Kinesis Firehose delivery patterns (to S3/Redshift destinations concept)
- Streaming-to-lake pattern (real-time landing in S3 + batch transforms)
- **Hands-on:** Lab: Build a simple streaming pipeline using Kinesis → Firehose → S3 and query results

10 Governance and Security (IAM, KMS, Lake Formation Concepts)

- IAM for data platforms (least privilege access to S3/Glue/Athena/Redshift)
- KMS encryption concepts (at rest and in transit awareness)
- Lake Formation concepts (centralized permissions, data access governance)
- Secure data sharing and access patterns for teams
- **Hands-on:** Lab: Apply least-privilege IAM policies and validate encrypted access paths

11 Data Quality and Observability (Reliable Operations)

- Data quality checks (schema, nulls, duplicates, ranges)
- Pipeline observability (logs, metrics, run history, SLA/freshness checks)
- Alerting and failure handling (retries, DLQ concepts for streaming)
- Operational runbooks and incident readiness basics
- **Hands-on:** Lab: Add validation checks to batch pipeline and build a simple quality report + alerts

12 Capstone Project (End-to-End AWS Data Engineering)

- Capstone goal: Build an AWS end-to-end data pipeline (batch + optional streaming)
- Ingest raw data to S3 (partitioned), transform with Glue, query with Athena
- Orchestrate with MWAA and publish curated datasets
- Apply governance/security (IAM/KMS) and add quality checks + observability
- **Hands-on:** Capstone Lab: Deliver architecture diagram, pipeline code, orchestration DAG, and KPI query outputs