[First Last Name]

[City, State] | [email@example.com] | [+1 (555) 555-5555] | [LinkedIn URL] | [GitHub/Portfolio URL]

PROFESSIONAL SUMMARY

Results-driven **Data Engineer** with [X]+ years of experience designing and implementing **scalable data pipelines**, ETL processes, and cloud-based data platforms. Proven track record of enabling **analytics, BI, and machine learning** use cases through robust data modeling, orchestration, and governance. Adept at collaborating with cross-functional stakeholders to translate business requirements into **high-performance, reliable data solutions**. Passionate about automation, data quality, and continuous optimization in modern **cloud data architectures**.

EXPERIENCE

[Senior Data Engineer] | **[Tech Company A]**
[MM YYYY] – Present
[City, State]

- Architected and implemented end-to-end **ETL/ELT pipelines** using [Apache Airflow], [Python], and [SQL] to ingest data from [X+] sources into a [cloud data warehouse] ([Snowflake]/[BigQuery]/[Redshift]), reducing manual data preparation time by [~60%].
- Designed and maintained **star-schema and normalized data models** for analytics and reporting, improving query performance by [30–40%] and enabling self-service BI in tools like [Looker]/[Tableau]/[Power BI].
- Implemented robust **data quality and observability** checks using [dbt tests]/[Great Expectations]/[custom Python validators], establishing SLAs for critical datasets and reducing data-related incident tickets by [~35%].

[Data Engineer] | **[Analytics Company B]**
[MM YYYY] – [MM YYYY]
[City, State]

- Developed batch and streaming **data ingestion workflows** using [Apache Spark]/[Kafka]/[Kinesis] to process [X GB+/day] of clickstream and transactional data into a centralized data lake on [AWS S3]/[Azure Data Lake]/[GCS].
- Optimized **SQL and Spark jobs** through partitioning, clustering, and resource tuning, reducing average pipeline runtime by [20–30%] and lowering compute costs in [EMR]/[Databricks]/[Dataflow].
- Collaborated with data scientists and analysts to productionize **ML-ready feature datasets**, standardizing data definitions, metadata, and documentation in [Confluence]/[Data Catalog] and improving model deployment cadence.

EDUCATION

[Bachelor of Science in Computer Science] | **[University Name]**
[MM YYYY] – [MM YYYY]
[City, State]

- Relevant coursework: [Database Systems], [Distributed Systems], [Data Structures & Algorithms], [Big Data Analytics], [Cloud Computing].

[Additional Certification in Data Engineering or Cloud] | **[Certification Provider]**
[MM YYYY]
[Online / City, State]

- [e.g., Google Cloud Professional Data Engineer, AWS Certified Data Analytics, Databricks Data Engineer Associate].

SKILLS

Programming & Querying: [Python], [SQL], [Scala], [Bash]

Data Engineering & ETL: [Apache Airflow], [dbt], [Apache Spark], [Kafka]/[Kinesis], [ETL/ELT design], [Data Modeling]

Cloud & Data Platforms: [AWS]/[Azure]/[GCP], [Snowflake], [BigQuery], [Redshift], [Databricks], [Data Lake/Lakehouse architectures]

Databases & Storage: [PostgreSQL], [MySQL], [NoSQL (e.g., DynamoDB, MongoDB)], [S3]/[ADLS]/[GCS]

Data Quality & Governance: [Great Expectations]/[dbt tests], [Data Catalogs], [Version control with Git], [CI/CD for data pipelines]

Analytics & BI: [Tableau]/[Power BI]/[Looker], [Dimensional modeling], [Performance tuning for analytical queries]
Soft Skills: Cross-functional collaboration, stakeholder communication, requirements gathering, documentation, problem-solving, mentoring/junior support

PROJECTS

[Real-Time Streaming Analytics Pipeline]
[MM YYYY] – [MM YYYY]

- Built a **real-time data pipeline** using [Kafka] ? [Spark Structured Streaming] ? [Cloud Data Warehouse] to process and aggregate event data with sub-minute latency for operational dashboards.
- Implemented **exactly-once processing** semantics and idempotent writes, handling late-arriving data with watermarking and windowing strategies.

[Enterprise Data Warehouse Modernization]
[MM YYYY] – [MM YYYY]

- Migrated legacy on-prem ETL workflows to a **cloud-native ELT architecture** using [dbt] and [Snowflake]/[BigQuery], consolidating [X] data marts into a unified semantic layer.
- Established **version-controlled, test-driven data transformations** with automated CI/CD, improving reliability of production releases and reducing deployment errors.