Google Cloud Data Engineer

What is a Data Engineer?

A **Data Engineer** is someone who **collects, organizes, and prepares data** so that companies can use it to make decisions, build reports, and train AI/ML models.

Think of a Data Engineer like the person who:

- Brings data from different sources (websites, apps, sensors, databases).
- Cleans and arranges the data so it makes sense.
- Builds pipelines to move data to storage systems (like BigQuery or Data Lakes).
- Makes the data available in a ready-to-use format for **analysts, data scientists, and business teams**.

In short 👉:

Data Engineers don't just store data — they make sure the right data reaches the right place, in the right format, at the right time.

Why is Data Engineering Important?

Every company today generates a **huge amount of data –** from apps, websites, sensors, payments, customer activity, and more.
But **raw data is messy**. It can't be used directly. That's where **Data Engineers** come in.

Importance of Data Engineering:

- **Organizes data** → turns raw data into clean, structured information.
- **Makes data fast & accessible** → so companies can get real-time insights.
- **Enables decision making** → managers, analysts, and AI models rely on well-prepared data.
- **Supports AI & ML** → without good data pipelines, AI systems can't learn properly.
- **Drives business growth** → companies use data to improve products, cut costs, and serve customers better

In today's world, data is like "oil," and Data Engineers are the ones who refine it into fuel.

Why Choose Google Cloud for Data Engineering?

Google is the **pioneer in handling massive data —** the same technology that powers **Google Search, YouTube, Gmail, and Maps** is available for companies through **Google Cloud Platform (GCP)**.

Why GCP for Data Engineering:

- **BigQuery** → one of the fastest and most scalable data warehouses in the world.
- **Dataflow & Pub/Sub** → powerful tools for **real-time streaming and batch data pipelines**.
- **Dataproc & Data Fusion** → flexible options for big data processing (Spark, Hadoop, ETL)
- **Spanner, Bigtable, CloudSql**→ scalable databases for every type of workload.
- **Enterprise Security & Governance** → IAM, VPC-SC, and Data Catalog for secure data handling.
- **Global infrastructure** → low latency and high availability across regions.

Simply put:

If you want to build data pipelines at scale, GCP is the best platform — built by the company that created MapReduce, Kubernetes, and BigQuery.

Career Demand & Salaries for Data Engineers

- One of the fastest-growing tech roles worldwide.
- Needed across every industry — finance, healthcare, e-commerce, entertainment.
- Average Salaries:
  - India: **₹12 – 25 LPA** (mid-level), **₹30 – 45 LPA+** (experienced).
  - US & Global: **$110K – $160K per year**.
- At i27Academy, many students in our Cloud & DevOps tracks have secured **30–60 LPA packages**. With Data Engineering, the opportunities are even bigger.

“If Cloud is the backbone of IT, Data Engineering is the brain that powers decisions”

Why i27Academy for Google Cloud Data Engineer?

- **5+ Years of Google Cloud Training** → Experts in Cloud Engineer, Architect & DevOps tracks.
- **Thousands of Students Trained** → i27Academy is the **go-to institute for Google Cloud in India**.
- **Specialized Focus** → We are **laser-focused on Google Cloud**, not generalist training.
- **Hands-on Projects** → Every topic taught with **real labs & case studies**.
- **Trusted Learning Path** → Our students work in top MNCs with **30–60 LPA packages**.
- **Proven Approach** → The same structured training method that made us a leader in Cloud & DevOps is now applied to **Data Engineering**.

👉 At i27Academy, we don’t just teach theory — we make you **job-ready to work on projects from Day 1**.

Who Can Join This Course?

This program is ideal for professionals and aspirants such as:

- **Database Engineers** → looking to move beyond traditional RDBMS into BigQuery & Spanner
- **Big Data / Hadoop Engineers** → wanting to upskill into Google Cloud's modern data stack
- **ETL / Data Warehouse Engineers** → ready to design scalable pipelines with Dataflow & Data Fusion
- **Application Programmers** → interested in working with real-time data and analytics
- **Test Engineers** → exploring career transition into high-growth Data Engineering roles
- **Data Analysts** → who want to step into advanced engineering and automation

👉 Even **fresh graduates** with interest in Cloud & Data can join, as we start from fundamentals and build step by step.

Course Content:

Module 1: Introduction to Cloud & GCP Fundamentals

- Introduction to Cloud Computing
- Roles & Responsibilities of a Cloud Data Engineer
- Overview of Cloud Platforms (AWS, Azure, GCP)
- Overview of Google Cloud Platform (GCP)
- Overview of Analytics Services on GCP (BigQuery, Pub/Sub, Dataflow, Dataproc, etc.)
- GCP Account Setup & Management
  i. Setup GCP for an Individual Account
  ii. Overview of GCP Projects, Billing & GCP Credits
- Accessing GCP Services
  i. Console
  ii. Google Cloud Shell
  iii. Google Cloud SDK (CLI)
- Creating & Managing Projects
  i. Understanding Project ID, Project Number, Project Name

Module 2: SQL for Data Engineers

- SQL Fundamentals
  i. DDL (CREATE, ALTER, DROP)
  ii. DML (INSERT, UPDATE, DELETE)
  iii. TCL (COMMIT, ROLLBACK, SAVEPOINT)
  iv. DCL (GRANT, REVOKE)
- Filtering & Expressions
  i. WHERE, ORDER BY, GROUP BY, HAVING
  ii. String, Date, Format & Cast Functions
  iii. Conditional Expressions (CASE, COALESCE, NULLIF)
- Joins
  i. INNER JOIN
  ii. LEFT JOIN / RIGHT JOIN
  iii. FULL OUTER JOIN
  iv. CROSS JOIN
  v. SELF JOIN
  vi. ANTI JOIN / SEMI JOIN
- Advanced Queries
  i. Subqueries (Scalar, Correlated, Nested)
  ii. Common Table Expressions (CTEs)

iii. Set Operators (UNION, UNION ALL, INTERSECT, EXCEPT/MINUS)

- Aggregations
  i. COUNT, SUM, AVG, MIN, MAX
  ii. GROUPING SETS, ROLLUP, CUBE
- Window Functions
  i. RANK, DENSE_RANK, ROW_NUMBER
  ii. NTILE, LEAD, LAG
  iii. CUME_DIST, PERCENT_RANK
  iv. Moving Averages (CSUM, MSUM, MDIFF, LAG/LEAD-based)
  v. PARTITION BY & ORDER BY in Window Functions
- Performance & Optimization
  i. Indexes & Partitioning (RDBMS vs BigQuery)
  ii. Query Optimization Techniques
  iii. Explain Plans & Execution Strategies

Module 3: Python for Data Engineering

- Python Basics
- File Handling (CSV, JSON, Parquet)
- Data Cleaning & Transformation (pandas)
- Logging & Error Handling
- GCP Libraries (BigQuery, GCS, Pub/Sub)
- Modular Python Scripts for Pipelines
- CI/CD for Python Projects (pytest, packaging, Git)

Module 4: Google Cloud Storage (GCS) – Data Lake Setup

- Overview of Google Cloud Storage and its role in Data Lakes
- Creating, uploading, and deleting buckets, folders, and files using:
  i. GCS Web Console
  ii. gsutil /gcloud CLI commands
  iii. Python APIs
- Setting up Google Cloud libraries in a Python virtual environment
- Handling multiple files in GCS programmatically using Python
- Processing data stored in GCS with Pandas
- Performing data conversions and writing outputs back to GCS
- Validating files in GCS using Python, gsutil(gcloud), and Pandas

Module 5: BigQuery – Data Warehouse Setup

- Introduction to Google BigQuery.
- Data Warehouse Concepts (OLTP vs OLAP, Dimensions, Facts, SCDs)
- CRUD & Merge/Upsert Operations
- Database operations in BigQuery:
  i. Using Web UI
  ii. Using SQL Commands
- Loading data into BigQuery (from Cloud Storage, local files, streaming)
- Execution Plan in BigQuery (EXPLAIN, Query Stages)
- Table Management in BigQuery:

    i. Partitioned Tables
    ii. Clustered Tables
    iii. External Tables
  - External Queries & External Connections
  - SQL in BigQuery (from basics to advanced)
  - Integration with Python:
    i. Using BigQuery Client Library
    ii. Pandas integration (read/write)
  - PostgreSQL Integration with BigQuery
  - Views & Materialized Views

Module 6: Cloud Composer (Real-Time Orchestration with Airflow / Cloud Composer)

  - Introduction to Cloud Composer (managed Apache Airflow on GCP)
  - Airflow basics: DAGs, tasks, operators, scheduler
  - Create and manage Composer environments (versions, scaling, upgrades)
  - Write and deploy DAGs (UI, gcloud, Git sync/artifacts)
  - Core operators & patterns:
    i. PythonOperator, BashOperator
    ii. BigQueryOperator, Dataproc SubmitJobOperator, Pub/Sub operators
    iii. Branching, sensors, ExternalTask/Trigger DAG runs
  - Variables, Connections, XCom, and Jinja templating
  - Secrets & credentials management (Airflow Connections, Secret Manager)
  - Orchestrating data pipelines end-to-end:
    i. GCS → Dataflow/Dataproc → BigQuery
    ii. Event-driven pipelines with Pub/Sub triggers
    iii. Validation & data quality steps inside DAGs
  - Monitoring & reliability: logs, retries, SLAs, alerts (email/Slack/Cloud Monitoring)
  - Performance & cost tips: parallelism, pools, task chunking, environment sizing
  - Best practices: idempotent tasks, modular DAGs, config via env/vars, CI/CD for DAGs

Module 7: dbt (Data Build Tool) — Data Transformation

  - Introduction to dbt (modern ELT vs traditional ETL)
  - Setting up dbt with BigQuery (local & cloud environments)
  - Understanding dbt project structure (models, seeds, snapshots, tests)
  - Building and testing models (SQL + version control)
  - Using Jinja templates & Macros for dynamic transformations
  - Data quality testing and documentation with dbt
  - Orchestrating dbt runs with Cloud Composer / other schedulers
  - Best practices for managing dbt projects in production

Module 8: Big Data Concepts for Data Engineers

  - What is Big Data? (Volume, Velocity, Variety, Veracity, Value — the 5 Vs)
  - Types of processing:
    i. Batch processing vs Real-time processing
    ii. ETL vs ELT
  - Data Lake vs Data Warehouse vs Data Mart

- Hadoop Ecosystem overview (HDFS, MapReduce, Hive, Spark)
- Apache Spark basics:
  i. Why Spark replaced MapReduce
  ii. Components (Spark SQL, Spark Streaming, MLlib)
- Dataflow vs Dataproc vs Databricks (managed big data choices on GCP)

Module 9: Cloud Dataproc — Big Data Processing on GCP

- Introduction to GCP Dataproc
- Setting up Dataproc clusters (development & production)
- Overview of HDFS commands & `gsutil` on Dataproc
- Handling files in HDFS:
  i. Local files
  ii. GCS files
- CLI connectivity in Dataproc:
  i. PySpark jobs
  ii. Spark Scala jobs
  iii. Spark SQL queries
- Submitting jobs:
  i. Spark SQL scripts
  ii. PySpark jobs via gcloud
- ETL pipeline creation & execution using Dataproc
- Running & validating **ELT data pipelines** on Dataproc
- Managing Dataproc workflows (multi-step pipelines)
- Dataproc jobs/workflows with gcloud commands
- Monitoring jobs with logs, UI, and Cloud Monitoring
- Cost optimization (autoscaling, preemptible workers, serverless Spark)

Module 10: PySpark (Data Engineering with Spark on GCP)

- Introduction to PySpark (Spark ecosystem, Spark vs MapReduce)
- RDDs vs DataFrames
- Core transformations: filter, map, joins, groupBy, aggregations
- Advanced functions: Window functions, UDFs, null handling, complex types
- Performance tuning: partitions, caching, broadcast joins, shuffle optimization
- PySpark with structured streaming:
  i. Source: Pub/Sub → process → sink (BigQuery/Cloud Storage)
  ii. Checkpointing, watermarks, fault tolerance
- Writing outputs to BigQuery & Cloud Storage
- Packaging & deploying PySpark jobs on Dataproc

Module 11: Databricks on GCP (Enterprise Data Engineering)

- Introduction to Databricks (why use over Dataproc)
- Setting up Databricks workspace on GCP
- Databricks clusters (standard vs high-concurrency, autoscaling)
- Databricks CLI & DBFS (file system) operations
- Building & running Spark SQL queries in notebooks

- Creating pipelines & jobs in Databricks workflows
- Integration with BigQuery & GCS
- Collaboration features: notebooks, repos, versioning
- Monitoring, debugging, and optimizing jobs in Databricks

Module 12: Cloud Pub/Sub — Real-Time Data Ingestion

- Introduction to Pub/Sub (publish–subscribe model, push vs pull delivery)
- Core concepts:
  i. Topics & Subscriptions
  ii. Publishers & Subscribers
  iii. Message ordering & delivery guarantees (at-least-once, exactly-once with Dataflow)
- Creating & managing Pub/Sub topics and subscriptions
- Publishing and consuming messages using console,cli and python sdk
- Message filtering & dead-letter topics (DLQ)
- Integrating Pub/Sub with other GCP services

Module 13: Cloud Dataflow — Batch & Streaming Pipelines

- What is Dataflow? (managed Apache Beam service for batch + streaming)
- Difference between **Batch vs Streaming pipelines**
- Core concepts: PCollection, Transform, Pipeline
- Dataflow vs Dataproc (when to use what)
- Deploying a batch pipeline
- Deploying a Streaming Pipeline

Module 14: Google Cloud Data Catalog — Metadata & Governance

- What is Data Catalog? (metadata management & data discovery)
- Why do we need it?
- Enabling Data Catalog in GCP
- Creating tags, entries, and entry groups
- Searching datasets & assets in Data Catalog (Console & CLI)
- Attaching metadata (business terms, owners, classifications) to services in gcp.
- Policy Tags for Column-Level Security in BigQuery

Module 15: Bigtable (NoSQL Database)

- Introduction to Bigtable (wide-column store)
- Bigtable Data Model (Tables, Rows, Row Keys, Column Families)
- Creating Instances & Clusters
- Schema & Row Key Design Best Practices
- Integrating between pyspark and bigtable.

Module 16: Cloud Spanner (Distributed SQL DB)

- Introduction to Cloud Spanner (horizontal scaling + SQL + ACID)
- Creating Instances & Databases
- Spanner Schema Design & Querying
- Integration with Dataflow & BigQuery

Module 17: Logging , Monitoring & Operations

- Cloud Logging & Cloud Monitoring for Data Pipelines
- Debugging Dataflow Jobs (backlog, errors, retries)
- Debugging Dataproc Jobs (YARN, Spark UI)
- Debugging Composer DAGs (logs, retries, alerts)
- Building Dashboards for Data Pipeline Health

Module 18: Cloud Data Fusion (Visual ETL on GCP)

- Introduction to Cloud Data Fusion (managed CDAP)
- When to use Data Fusion vs Dataflow/dbt
- Data Fusion architecture (Pipelines, Wrangler, Wrangler service)
- Setting up a Data Fusion instance
- Building pipelines with Wrangler (UI-based transformations)
- Batch vs Streaming pipelines in Data Fusion

Module 19: Looker (Business Intelligence & Visualization on GCP)

- Introduction to Looker (Looker vs Looker Studio / Data Studio)
- Role of Looker in the Data Engineering ecosystem
- Connecting Looker to BigQuery and other GCP data sources
- LookML Basics:
  i. Models, Views, Explores
  ii. Dimensions & Measures
  iii. Relationships & Joins
- Creating Dashboards & Visualizations
- Scheduling reports and alerts

Module 20: BigQuery ML — Basics

Module 21: CI/CD + Git for Data Engineers

- Git Essentials
- CI/CD Pipelines in GCP (Cloud Build / Jenkins / GitHub Actions)

Module 22: Google Cloud Certified Data Engineer – Exam Prep

Capstone Projects — End-to-End Implementation

Project 1: Retail E-Commerce Analytics Pipeline

- Data Source: Order transactions, product catalog, customer data (CSV/JSON)
- Ingestion: Load raw data into **Google Cloud Storage (GCS)**
- Processing: Clean & enrich using Dataproc (PySpark)
- Orchestration: Automate daily ETL pipelines with **Cloud Composer**
- Analytics: Store curated datasets in **BigQuery**
- Visualization: Build dashboards in **Looker** for product sales & customer behavior

Project 2: Online Banking: Fraud Detection & Customer Insights

- Data Source: Transaction logs, customer accounts, merchant info
- Ingestion: Store raw logs in **GCS**
- Processing: Detect anomalies with **Dataproc (PySpark/Spark ML)**
- Orchestration: Manage hourly fraud checks + daily summaries in **Cloud Composer**
- Analytics: Push results into **BigQuery** for compliance & reporting
- Output: **Looker dashboards** for fraud alerts, risk scores, and transaction heatmaps
- Governance: Secure sensitive data using **Data Catalog + Policy Tags**

Project 3: Healthcare: Patient & Hospital Analytics

- Data Source: Patient records (de-identified), admissions, equipment usage, lab results
- Ingestion: Upload raw CSV/JSON into **GCS**
- Processing: Cleanse & join datasets with **Dataproc (PySpark)**
- Advanced Analytics: Predict patient wait times using **Spark ML**
- Orchestration: Automate weekly/monthly reporting pipelines with **Cloud Composer**
- Analytics: Store KPIs in **BigQuery** (admissions, occupancy, treatment outcomes)
- Visualization: **Looker dashboards** for bed utilization & patient inflow trends
- Governance: Use **Data Catalog** for metadata & compliance

Career Readiness & Next Steps

Resume & Portfolio Building

- How to highlight GCP Data Engineering skills in your resume
- Adding capstone projects (Retail, Banking, Healthcare) as real-world case studies
- GitHub portfolio setup with SQL, dbt, Dataflow, and PySpark code samples

Job Profiles to Target

- Google Cloud Data Engineer
- Big Data Engineer (GCP focus)
- Cloud Data Platform Engineer
- Analytics Engineer (BigQuery + dbt)
- DataOps / Pipeline Engineer

Interview Preparation

- Google Data Engineer interview questions topic wise discussion
- Scenario-based discussions (which we face in realworld)