

Abhiraj Karpe

Data Engineer

github.com/abhh10 | candidates.intervues.club/u/abhiraj-karpe

SUMMARY

Data Engineer and former Software Engineering Intern experienced in building data ingestion pipelines, automated device profiling workflows, and real-time CDC systems. Proficient in transforming raw, unstructured data into structured formats, reducing storage overhead and downstream processing latency. Skilled in leveraging cloud platforms, distributed processing frameworks, and modern lakehouse architectures.

EXPERIENCE

Software Engineering Intern

Jio Platforms Limited

Jan 2026 – Jul 2026

- Converted legacy ADB script outputs from raw text to structured JSON across OTT profiling workflows, reducing storage overhead by 30–40% and cutting downstream parsing time by ~35%.
- Optimized memory and CPU profiling workflows for OTT applications, reducing job runtime by ~20% and improving data collection consistency across device models.
- Designed and deployed a production ingestion API using Node.js and Express.js for crash and profiling data, integrating ADB tooling, custom parsing, and database storage for daily QA cycles.
- Built automated collection scripts for device-level metrics using Python and shell scripts, emitting structured JSON for backend ingestion.
- Resolved data-loss and parsing issues across ADB, JSON/TXT handling, and backend database storage to ensure accurate logging of key pipeline metrics.

Software Engineering Intern

Jio Platforms Limited (JPL)

Jan 2026 – Jul 2026

- Converted legacy ADB script outputs from raw text to structured JSON across OTT profiling workflows, reducing storage overhead by 30–40% and cutting downstream parsing time by ~35%.
- Optimized memory and CPU profiling workflows for OTT applications, reducing job runtime by ~20% and improving data collection consistency across device models.
- Designed and deployed a production ingestion API using Node.js and Express.js for crash and profiling data, integrating ADB tooling, custom parsing, and database storage for daily QA cycles.
- Built automated collection scripts for device-level metrics using Python and shell scripts, emitting structured JSON for backend ingestion.
- Resolved data-loss and parsing issues across ADB, JSON/TXT handling, and backend database storage to ensure accurate logging of key pipeline metrics.

EDUCATION

Ajeenkya D Y Patil University · B.Tech · 2022 – 2026

JAIN INTERNATIONAL SCHOOL AURANGABAD · India · 2021 – 2022

The Stepping Stone School · India · 2020 – 2021

Tender Care Home · 2020 – 2020

SKILLS

Languages and Frameworks: CSS, Java, JavaScript, Kotlin, Python, SQL, TypeScript, Apache Spark, Express.js, LangChain, Node.js, PySpark

Databases: Amazon Redshift, BigQuery, MongoDB, MySQL, PostgreSQL, Snowflake, Vector Databases (ChromaDB)

Tools: ADB, Airflow, Apache Iceberg, Athena, Auto Loader, AWS CLI, Boto3, CDC, CI/CD, CloudWatch, Data Modeling, Databricks Workflows, dbt, Debezium, Delta Lake, Docker, ETL/ELT, Git, GitHub, Glue, Hive Metastore, IAM, Incremental Processing, JSON, Kafka, Lakebase, Lakeflow Declarative Pipelines, Lambda, Linux, Medallion Architecture, MinIO, OpenAI API, Postman, RAG, REST APIs, S3, Snowpipe, Unity Catalog

PROJECTS

CDC Lakehouse Pipeline

PostgreSQL, Debezium, Apache Kafka, Apache Iceberg, MinIO, Hive Metastore, Trino Built an end-to-end CDC pipeline streaming PostgreSQL inserts, updates, and deletes through Debezium and Kafka into Apache Iceberg tables with 1.03s P95 latency.
PostgreSQL, Debezium, Apache Kafka, Apache Iceberg, MinIO, Hive Metastore, Trino
github.com/abhh10/cdc-lakehouse

Databricks Taxi Analytics Pipeline

Databricks, PySpark, Delta Lake, Unity Catalog, AWS S3 Built an incremental Bronze–Silver–Gold pipeline processing 355M+ taxi trip records with schema normalization, data quality filtering, deduplication, and partition-aware design.
Databricks, PySpark, Delta Lake, Unity Catalog, AWS S3
github.com/abhh10/databricks-nyc-trip-pipeline

Real-Time Crypto Data Engineering Pipeline

Apache Kafka, AWS S3, Snowpipe, Snowflake, dbt, Airflow Built a real-time pipeline streaming CoinGecko market events through Kafka into Amazon S3 and Snowflake via Snowpipe, using incremental dbt transformations with MERGE updates.
Apache Kafka, AWS S3, Snowpipe, Snowflake, dbt, Airflow
github.com/abhh10/crypto-data-engineering-pipeline

AWARDS

AWS re/Start Graduate, Introduction to Data Engineering, LeetCode and GeeksforGeeks Coding Achievement