

# Technical Phone Screen — Data Engineer

Phone Screen

Data

Screen for data engineering: SQL proficiency, pipeline design, and data modeling.

## Evaluation Criteria

### SQL Proficiency & Query Design

Evaluates depth of SQL knowledge including window functions, CTEs, and query optimization.

Rating:  1  2  3  4  5

#### Sample Questions:

- Explain the difference between a window function and a GROUP BY — when do you use each?
- How would you find the top 3 products by revenue per category in a single query?
- What strategies do you use to optimize slow-running queries on large datasets?

#### ✓ STRONG SIGNAL

Candidate writes or describes complex queries fluently, uses window functions and CTEs naturally, and explains query execution plans. They discuss partitioning and indexing strategies for analytical workloads.

#### ✗ WEAK SIGNAL

Candidate struggles with basic JOINS, cannot explain when to use window functions, or has only worked with simple SELECT statements.

### Data Pipeline Design & ETL/ELT

Assesses experience designing and maintaining reliable data pipelines at scale.

Rating:  1  2  3  4  5

#### Sample Questions:

- Walk me through how you would design a pipeline to ingest data from an external API into a data warehouse.
- What's the difference between ETL and ELT, and when would you choose each?
- How do you handle schema changes in upstream data sources?

#### ✓ STRONG SIGNAL

Candidate describes orchestration tools (Airflow, Dagster, Prefect), discusses idempotency, retry logic, and backfill strategies. They consider data freshness requirements and SLAs.

#### ✗ WEAK SIGNAL

Candidate has only written standalone scripts with no orchestration, doesn't consider failure modes, or cannot explain how they ensure pipeline reliability.

### Data Modeling & Warehouse Design

Evaluates understanding of dimensional modeling, star/snowflake schemas, and warehouse architecture.

Rating:  1  2  3  4  5

#### Sample Questions:

- How do you decide between a star schema and a snowflake schema?
- Explain slowly changing dimensions — which types have you used and why?

- How do you model a many-to-many relationship in a dimensional model?

✓ **STRONG SIGNAL**

Candidate explains fact/dimension tables clearly, discusses trade-offs of different SCD types, and considers query patterns when designing models. They mention tools like dbt for transformation layers.

✗ **WEAK SIGNAL**

Candidate has no exposure to dimensional modeling, conflates OLTP and OLAP design, or cannot explain why warehouse design differs from application database design.

## Data Quality & Testing

Assesses the candidate's approach to ensuring data reliability and catching issues proactively.

Rating:  1  2  3  4  5

### Sample Questions:

- How do you ensure the quality of data flowing through your pipelines?
- What data testing frameworks or approaches have you used?
- Describe a time you caught a data quality issue before it reached stakeholders.

✓ **STRONG SIGNAL**

Candidate describes data contracts, schema validation, row count checks, freshness monitoring, and anomaly detection. They mention tools like Great Expectations, dbt tests, or Monte Carlo.

✗ **WEAK SIGNAL**

Candidate does no data validation, waits for downstream users to report issues, or assumes source data is always correct.

## Cloud & Big Data Technologies

Evaluates hands-on experience with cloud data platforms and big data processing frameworks.

Rating:  1  2  3  4  5

### Sample Questions:

- What cloud data platforms have you worked with, and how did you choose between them?
- When would you use Spark vs. a simpler approach like SQL on a warehouse?
- How do you manage costs in a cloud data environment?

✓ **STRONG SIGNAL**

Candidate has hands-on experience with platforms like Snowflake, BigQuery, Redshift, or Databricks. They can reason about when distributed processing is needed vs. overkill and discuss cost optimization.

✗ **WEAK SIGNAL**

Candidate lists tools they've only read about, cannot explain when big data tools are actually necessary, or has no awareness of cost implications.

## Red Flags

- Cannot write a basic SQL JOIN or explain how GROUP BY works
- Has never considered data quality or pipeline reliability
- Claims Spark experience but cannot explain what a partition is or when distributed processing is needed

## Notes & Overall Recommendation

Strong Hire  Hire  No Hire  Strong No Hire

Notes: \_\_\_\_\_

