

Data Engineer interview questions

careertips.com.au: common questions and what they're really testing

What to expect

Data Engineer interviews in Australia usually combine technical depth on pipelines and databases with practical scenario questions about handling failures or messy data, plus a check on how well you communicate with analysts and other non-engineering stakeholders. Interviewers often probe how quickly you can get productive on existing infrastructure rather than testing abstract computer science theory.

- **Technical:** Questions on SQL, schema design, Spark and cloud platforms to confirm hands-on capability with the actual tools listed in the job ad.
- **Process/walkthrough:** Questions asking you to describe how you'd design or debug a pipeline end to end, testing your practical approach rather than a single fact.
- **Behavioural:** Past-experience questions about data quality incidents, deadline pressure or cross-team requirements gathering.
- **Scenario/judgement:** Hypothetical situations, such as a pipeline failure at 2am or conflicting requirements from analysts, to see how you'd prioritise and communicate.
- **Client-facing/collaboration:** Questions on working with data analysts and data scientists who aren't engineers, checking whether you can translate requirements without over-engineering.

Expect an initial screen on background and tool experience, followed by a technical round covering SQL and pipeline design (sometimes a take-home or whiteboard exercise), then a behavioural/scenario round with a hiring manager or team lead, and possibly a final conversation focused on stakeholder collaboration and team fit.

1. Walk me through how you'd design an ETL pipeline to pull data from three different source systems into a single warehouse table.

Why they ask

Tests your practical pipeline design skills and whether you think about source reliability, transformation logic and load strategy rather than just tools.

How to structure your answer

Walk through it as a sequence: describe how you'd handle extraction from each source, what transformation and validation steps you'd add, how you'd structure the load (full vs incremental), and how you'd monitor it once live.

Example answer

"I'd start by profiling each source to understand format, update frequency and known quality issues. For extraction, I'd use scheduled jobs appropriate to each source, batch pulls for a database and an API poll for a third-party system. In the transform stage I'd standardise field types and add validation checks for nulls, duplicates and referential integrity before loading. For the load itself I'd favour an incremental approach keyed on a timestamp or change flag to avoid reprocessing the full dataset each run, and I'd orchestrate the whole thing in Airflow so failures are visible and retryable. Once live, I'd add row-count and freshness alerts so any silent failure gets flagged before an analyst notices bad numbers in a report."

2. Tell me about a time you found a data quality issue that others had missed.

Why they ask

Attention to data quality is core to this role, and interviewers want evidence you catch problems proactively rather than reactively.

How to structure your answer

Use STAR: situation, task, action, result.

Example answer

"A dashboard was showing a sudden spike in customer sign-ups that didn't match other business signals. I traced it back to a source system that had started sending duplicate records after a schema change upstream. I added a deduplication step and a row-count anomaly check to the pipeline so future spikes would trigger an alert instead of flowing straight into reporting. The immediate numbers corrected once I reprocessed the affected load, and the monitoring check has caught two similar issues since without anyone needing to notice manually."

3. How would you optimise a query that's taking minutes to run against a large table?

Why they ask

Query and schema optimisation is a named task in this role and a common technical screening question.

How to structure your answer

Give a structured technical answer: diagnose first, then list the levers you'd pull in order of likely impact.

Example answer

"First I'd look at the execution plan to see where the time is actually going, whether it's a full table scan, a poor join order or missing statistics. Common fixes include adding an index on the filter or join columns, partitioning the table by a commonly filtered date field, and rewriting subqueries as joins where the optimiser handles them better. If the table is large and queried repeatedly for the same aggregations, I'd consider a pre-aggregated summary table or materialised view rather than optimising the raw query further."

4. It's 2am and a critical pipeline has failed before the morning reporting run. What do you do?

Why they ask

Scenario question testing judgement under pressure and prioritisation, relevant given pipelines often run on schedules outside business hours.

How to structure your answer

Structure as judgement-under-pressure: immediate triage, decision on fix vs workaround, and follow-up communication.

Example answer

"I'd check the logs first to identify whether it's a source data issue, an infrastructure problem or a code bug. If it's something I can fix quickly, like a transient connection failure, I'd rerun the job and confirm the output before the reporting deadline. If it needs a real fix, I'd decide whether stale data is safe to serve for one day with a clear note to stakeholders, rather than risk pushing a rushed fix into production. Either way I'd message the relevant analysts or team lead early so they're not blindsided by late or missing numbers, and I'd log the root cause for a proper fix once business hours start."

5. How do you approach a requirements conversation with a data analyst who isn't technical?

Why they ask

Collaborating with analysts and data scientists is a listed task, and this checks communication skill, not just engineering ability.

How to structure your answer

Behavioural, STAR format, focused on communication and translation of requirements.

Example answer

"An analyst once asked for 'all the customer data' for a new dashboard, which would have meant building an unnecessarily wide and slow table. I asked what specific questions the dashboard needed to answer, which narrowed it down to about a dozen fields tied to purchase behaviour. I then confirmed the refresh frequency they actually needed, daily rather than real-time, which let me design a much simpler and cheaper pipeline. The result was a table that loaded faster and was easier for me to maintain, and the analyst got exactly what the dashboard needed without extra back-and-forth later."

6. What experience do you have with cloud data warehouse platforms like AWS Redshift, Azure Synapse or Google BigQuery?

Why they ask

Cloud infrastructure is a specialist skill and named tool area for this role, and interviewers want specifics rather than a general familiarity claim.

How to structure your answer

Direct technical answer: name the platforms used, specific tasks performed, and one limitation or lesson learned.

Example answer

"I've worked mainly with AWS, setting up and maintaining data warehouse infrastructure and managing access controls for analyst teams. I've also used dbt on top of that for transformation logic, which made version-controlling and testing SQL models much easier than running loose scripts. One lesson I've taken from that work is to be deliberate about partitioning and clustering keys early, because retrofitting them on a large table once query patterns are established is far more disruptive than planning for them upfront."