

Business Intelligence Developer interview questions

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

What to expect

Interviews for Business Intelligence Developer roles mix hands-on technical checks with questions about how you handle stakeholders who don't speak SQL. Expect a genuine technical component, whether that's a SQL exercise, a portfolio walkthrough of a dashboard you built, or a whiteboard discussion of how you'd design a pipeline.

- **Technical:** Direct questions or exercises testing SQL, ETL design, or your reasoning for choosing one BI tool or data structure over another.
- **Process:** Walkthroughs of how you'd approach building a pipeline or dashboard from a raw request through to a finished product.
- **Behavioural:** Past-experience questions about working with stakeholders, handling conflicting requirements, or catching a data problem before it caused harm.
- **Scenario:** Hypothetical situations testing judgement, such as what you'd do if a dashboard's numbers didn't match the source system just before a board meeting.

Most interviews open with a short walkthrough of your background and a dashboard or project you've built, move into technical questions or a live SQL exercise, then shift to behavioural and scenario questions about stakeholder work and data quality, before closing with your questions about the team's tech stack and current data maturity.

1. Walk me through how you'd design an ETL pipeline to bring data from three separate source systems into a single reporting warehouse.

Why they ask

This checks whether you understand the practical sequence of extraction, transformation and loading, and whether you think about data quality and reliability rather than just the happy path.

How to structure your answer

Answer as a step-by-step process walkthrough: describe extraction, transformation rules, load strategy, and how you'd monitor and handle failures, in that order.

Example answer

"I'd start by profiling each source system to understand schema, refresh frequency and known data quality issues. For extraction, I'd set up scheduled pulls appropriate to each source rather than assuming they all update at the same rate. In the transformation layer, I'd standardise formats, resolve duplicate or conflicting records, and apply business rules agreed with stakeholders. I'd load into a staging area first, run validation checks, then push to the production warehouse. Finally I'd set up monitoring and alerts so a failed load or an unexpected drop in row count gets caught before anyone opens a dashboard built on bad data."

2. Tell me about a time you had to translate a vague or conflicting reporting request from a business stakeholder into something you could actually build.

Why they ask

BI developers spend a lot of time interpreting what stakeholders say they want versus what they actually need, and interviewers want evidence you can manage that gap without endless rework.

How to structure your answer

Use STAR: describe the situation and the ambiguity in the request, your task, the action you took to clarify and scope it, and the result.

Example answer

"A finance stakeholder asked for a dashboard showing regional performance, but different team members had different ideas about which metrics and time periods mattered. I set up a short workshop with the key stakeholders, sketched out a mock layout on paper, and got explicit agreement on the metrics, filters and refresh frequency before writing any SQL. That upfront clarification meant the first version I built was close to final, and it avoided the back-and-forth revisions that had slowed down previous reporting projects."

3. A dashboard you built shows numbers that don't match the finance team's own figures, and you're being asked about it an hour before a leadership meeting. What do you do?

Why they ask

This tests judgement under pressure, and whether you can troubleshoot a discrepancy methodically without panicking or making unfounded promises about the fix.

How to structure your answer

Answer as a judgement-under-pressure structure: state your immediate priority, the quick checks you'd run, how you'd communicate uncertainty, and your next step.

Example answer

"First priority is to flag to the meeting owner that there's a discrepancy under review, rather than let incorrect numbers go into the room unqualified. Then I'd quickly check the obvious causes: whether the dashboard and finance's figures are using different date cut-offs, different filters, or different definitions of the same metric, since that's the most common cause of mismatches. If I can identify the cause quickly I'd explain it plainly. If not, I'd recommend presenting with a clear caveat that the figure is being verified, and follow up with a corrected version and an explanation of what caused the gap."

4. How do you decide between Power BI and Tableau for a given reporting need?

Why they ask

This is a genuine technical judgement question in this field, since both tools are widely used and the choice affects cost, licensing and what stakeholders can do with the output.

How to structure your answer

Answer as a comparative technical reasoning response: name the factors that drive the decision, then give a concrete example of when you'd choose each.

Example answer

"I look at the existing tech stack, licensing cost, and what the end users are comfortable with. If the organisation is already on Microsoft 365 and Azure, Power BI usually wins on integration and cost. Tableau tends to handle more complex, visually customised dashboards more gracefully and has a slight edge for advanced users who want to explore data themselves rather than just consume a fixed report. In a past role, we used Power BI for standard operational dashboards distributed org-wide, but kept Tableau for a smaller analytics team doing deeper ad hoc exploration."

5. Write a SQL query that identifies duplicate customer records based on name and email, and explain how you'd handle them.

Why they ask

Data cleaning and validation is a daily task in this role, and interviewers want to see you can write practical SQL, not just describe concepts.

How to structure your answer

Treat this as a technical exercise: talk through your query logic step by step, then explain the downstream handling decision.

Example answer

"I'd use a GROUP BY on normalised name and email fields, with a HAVING COUNT() greater than one to surface duplicates, then use a window function like ROW_NUMBER partitioned by those fields to identify which record to keep, usually the most recently updated one. Before deleting or merging anything, I'd flag the duplicates in a separate table for the business owner to review, since automatically deleting records without sign-off can destroy transaction history tied to the wrong record."*

6. Data quality issues sometimes only show up after a report has already gone out to stakeholders. How do you build monitoring to catch problems earlier?

Why they ask

Employers are hiring for people who can maintain reliable production systems, not just build one-off dashboards.

How to structure your answer

Answer as a process description: outline what you monitor, how you'd set thresholds or alerts, and how you'd respond when something trips them.

Example answer

"I'd set up automated checks at the point data lands in staging, things like row count thresholds, null checks on key fields, and referential integrity checks against known reference tables. I'd also track pipeline run times and failure logs so a silent failure doesn't just show up as stale data days later. If a check fails, I'd rather have a dashboard show a clear 'data not yet validated' state than let it publish outdated or incorrect numbers, and I'd set up an alert to the data team so it gets investigated the same day."