

Testing checklist: data pipeline

Source: <https://howtotestaproduct.com/how-to-test/data-pipeline/>

Generated: 2026-10-07. Licence: CC BY 4.0.

☐ 1. release smoke tests

Done when: After each deploy, a run on a fixed sample input produces the expected row count and totals in every target table.

Risks: data-corruption, wrong-decisions. Tool class: data validation. Usually run by: developer.

☐ 2. data integrity and migration testing

Done when: For each run, row counts and control totals reconcile between source extract, staging and target, and a mismatch stops publication.

Risks: data-corruption, wrong-decisions, regulatory-penalty. Tool class: data validation. Usually run by: QA engineer.

☐ 3. compliance checks

Done when: Every regulated report traces each figure back to its source tables, and personal fields are masked or dropped before the shared layer.

Risks: regulatory-penalty, data-breach. Tool class: test management. Usually run by: product manager.

☐ 4. integration tests

Done when: An end-to-end run against staging copies of each source and target loads a file above the largest size seen in production with no rows lost, and when the source stream stops sending mid-run, the run is reported as stalled instead of complete.

Risks: data-corruption, dependency-failure. Tool class: data validation. Usually run by: developer.

☐ 5. regression suite

Done when: Reconciliation and edge-file cases rerun on every change, and any change in sample output totals blocks the merge.

Risks: data-corruption, wrong-decisions. Tool class: test management. Usually run by: QA engineer.

☐ 6. resilience and failure testing

Done when: With the source stream or warehouse paused mid-run, the job retries, resumes from its last checkpoint and loads each record once.

Risks: dependency-failure, data-corruption. Tool class: chaos tool. Usually run by: developer.

☐ 7. unit tests

Done when: Each transform returns the documented output for empty files, oversized files, null keys, duplicate keys and timezone boundaries.

Risks: data-corruption. Tool class: unit framework. Usually run by: developer.

☐ 8. security scanning

Done when: The scan of buckets, staging schemas and service accounts shows no public access and no personal fields outside approved tables.

Risks: data-breach. Tool class: DAST scanner. Usually run by: security specialist.

☐ 9. analytics and event tracking tests

Done when: Each job emits start, end, rows read and rows written, and a run with no end event or a write count of zero raises an alert.

Risks: wrong-decisions. Tool class: data validation. Usually run by: developer.