

FREE COMPANION TO SQL RECIPES FOR DBAs

The DBA Health Check

*The quarterly checklist — plus five 30-second scripts
for PostgreSQL, SQL Server, MySQL/MariaDB, SQLite, and Oracle*

Tiye Baldwin-Anderson · Baldwin Apps

Every engine part ended with a health-check recipe — a script for PostgreSQL, SQL Server, and MySQL; a routine compiled into the application for SQLite; a posture-and-recency battery for Oracle. This closing chapter is the layer above all of them: the checklist that is true regardless of engine, phrased as questions because the asking is the skill.

Run it quarterly on systems you own, and in the first week on systems you inherit. Every "no" or "don't know" is a work item; the DBA numbers point to the recipes that turn it into a "yes."

Access

- ☐ Are administrative accounts limited — and is the who-holds-the-keys list (superuser, sysadmin, SUPER, DBA role) one you have actually read lately? *(DBA-013)*
- ☐ Are application accounts separated from human accounts, and from migration/deployment accounts? *(DBA-009-011)*
- ☐ Are read-only users truly read-only — audited, not assumed? *(DBA-014)*
- ☐ Are unused accounts removed or disabled, with every host variant and definer swept? *(DBA-012)*
- ☐ Are permissions documented — could you explain why each grant exists?

Backup and Recovery

- ☐ Are backups running — verified by recency queries, not by the absence of alerts? *(DBA-029)*
- ☐ Are backups stored safely — off the host whose failure they exist to survive? *(DBA-039)*
- ☐ Has a restore been tested — actually performed, timed, with a freshness check? *(DBA-033)*
- ☐ Is the recovery process documented and rehearsed — including point-in-time replay? *(DBA-030, DBA-034)*
- ☐ Is the recovery time acceptable — is the measured RTO a number leadership has seen?

- ☐ Is the possible data loss acceptable — does the change-stream contract (log backups, WAL, binlogs, ARCHIVELOG) match the RPO the business believes it has? (*DBA-034*)

Activity and Performance

- ☐ Are long-running queries reviewed — and long-open *transactions*, which hide from activity views? (*DBA-016*)
- ☐ Are blocked sessions investigated to the head of the chain, not the first visible waiter? (*DBA-018*)
- ☐ Are slow queries logged or tracked in whatever the engine's durable instrument is? (*DBA-026*)
- ☐ Are query plans reviewed for critical workloads — estimates against actuals? (*DBA-021, DBA-028*)
- ☐ Are indexes maintained by evidence — usage audits and deliberate surgery, not blanket rebuilds? (*DBA-022, DBA-025*)
- ☐ Are statistics current — especially after every bulk load? (*DBA-024*)

Storage

- ☐ Is database growth monitored against a baseline, not a feeling? (*DBA-007*)
- ☐ Are large tables identified, with archiving candidates named? (*DBA-008*)
- ☐ Are logs growing unexpectedly — WAL, transaction logs, binary logs, archived redo — and is the cause named, not just the symptom shrunk? (*DBA-035*)
- ☐ Is disk space safe — with action thresholds at 80%, not 95%?
- ☐ Are old exports, dumps, and temporary files cleaned up on a schedule? (*DBA-039*)

Maintenance

- ☐ Are routine maintenance tasks scheduled — the engine's real list, not another engine's rituals? (*DBA-037*)
- ☐ Are maintenance tasks monitored — with heartbeats that catch silent death, not just failure alerts? (*DBA-039*)
- ☐ Are failures visible — does anything that stops running tell someone?
- ☐ Are jobs documented — owner, purpose, and what breaks if they stop?

- ☐ Are logs reviewed — the server's own health record, on a cadence? (*DBA-036*)

Documentation

- ☐ Is the backup process documented?
- ☐ Is the restore process documented — as a checklist someone other than its author could follow at 2 a.m.?
- ☐ Are privileged accounts documented?
- ☐ Are scheduled jobs documented?
- ☐ Are emergency procedures documented — including how to get in when nothing else connects?

The best DBA health check is not a single script. It is a habit of asking whether the database is secure, recoverable, observable, and maintainable.

Four adjectives, forty recipes, five engines. The scripts in the engine parts are the thirty-second version of this chapter; this chapter is the quarterly version of those scripts; and both are the same habit at different zoom levels — the habit this book was written to build.

The five scripts

This pack includes one ready-to-run health-check script per engine: `healthcheck-postgresql.sql`, `healthcheck-sqlserver.sql`, `healthcheck-mysql.sql`, `healthcheck-sqlite.sql`, and `healthcheck-oracle.sql`. Each compresses that engine's warning signs into about thirty seconds of looking. Baseline first: the first run tells you what *is*, not what is *normal* — keep outputs and compare.

Every warning sign these scripts surface maps to a numbered recipe in **SQL Recipes for DBAs** — the same 40 tasks in all five engines, with the traps and safety notes for each. Available on Amazon.