

No-Shortcuts AI Prompt Pack

Reusable instructions for exhaustive comparisons, research, code review, troubleshooting, and long projects.

Edition: July 2026

Use: Personal and business reference

Use the universal prompts as standing constraints, then add the task-specific prompt that matches the work. The goal is not longer output. The goal is a measurable process and an honest boundary between completed and partial work.

Universal instructions

1

Master no-shortcuts instruction

Complete the requested process exactly as described. Do not replace a complete review with sampling or spot-checking. Before starting, restate the task in measurable terms. Maintain a concise process ledger listing what was inspected, what remains, assumptions, failed attempts, and validation. If the work is partial, label it partial and identify what remains.

2

Completion audit

Before saying the task is complete, compare the requested scope with the work actually performed. Report expected item count, processed item count, result count, inaccessible items, and validation performed.

3

Assumption inventory

List every assumption that could change the answer. Separate assumptions I approved from assumptions you introduced. Ask before proceeding when an unapproved assumption could materially change the result.

Finite comparisons and data work

4

Exhaustive list comparison

Compare every item in List A against the complete List B. Preserve order and duplicates. Do not sample. Report the number of input rows, rows processed, matches, non-matches, and output rows. Reconcile the counts before returning the result.

5

Data extraction audit

Extract every requested field from every record. Keep a row for records with missing values rather than silently dropping them. Report rejected or unreadable records separately and reconcile source count to output count.

6

Deduplication review

Identify duplicates using the exact rules I provide. Do not merge similar-looking records automatically. Return duplicate groups, the rule that matched each group, ambiguous cases, and the original total reconciled to retained plus removed rows.

Files, code, and websites

7

Every-file review

Open every supplied file that can materially affect the task, including configuration and build files. Maintain a list of filenames inspected. State which files were not opened and why. Do not describe the project review as complete if any relevant file was unavailable.

8

Code execution boundary

Separate what you learned by reading the code from what you verified by executing it. Do not claim the program works unless it was run through the stated test cases. List untested paths.

9

Website link audit

Check every local link and asset reference in the supplied site. Resolve paths from each page location, not just from the project root. Return broken references with source file and line or element context, then reconcile total references checked.

Research and product comparisons

10

Source coverage plan

Before researching, define the source categories required, such as official documentation, regulatory records, product manuals, independent testing, and user reports. State which categories were searched and which yielded no usable source.

11

Product comparison without filler

Compare the named products against every criterion I supplied. Use current official specifications for factual features and clearly attributed independent sources for testing or reliability claims. Do not fill missing information with an estimate unless I approve it.

12

Research completion audit

List the sources opened, not merely search results viewed. For each major conclusion, identify the supporting source. State what could not be verified and what search paths were tried.

Long conversations and revisions

13

Standing-instruction refresh

At the start of this task, restate the standing constraints that still apply. Before finalizing, audit the output against those constraints and identify any conflict or omission.

14

Version and access ledger

Record each supplied file, link, screenshot, transcript, or generated output with its purpose and version. State whether each item is currently accessible. Do not rely on a prior version when a newer version exists.

15

Revision integrity check

Apply only the requested revisions. Before returning the result, compare the new version against the previous version and list substantive changes, unchanged areas, and any unintended differences you corrected.

A process ledger is a user-facing activity record, not private chain of thought. Ask for observable actions, counts, sources, files, tests, limitations, and remaining work.