

The GA4 dashboard prompt pack

Four prompts that take you from an empty folder to a dashboard and a written weekly summary: pull the data, build the page, make it admit what it does not know, then write the update.

HOW TO RUN THESE

- ☐ Run them in order. Each one assumes the output of the previous.
- ☐ Give the assistant your real property ID and the service-account file path. It cannot guess either
- ☐ Keep the pull and the render in separate files, so a broken chart never costs you an API call
- ☐ Run prompt three before anyone else sees the page. It is the one that catches confident nonsense

PROMPT 1 • PULL THE DATA

- ☐ "You are writing a data-fetch script, not a dashboard. Using the Google Analytics 4 Data API v1beta and a service-account credential at CREDENTIALS_PATH, write a single Node script that queries property PROPERTY_ID and writes one JSON file to data.json."
- ☐ "Fetch exactly these reports: (1) sessions, engaged sessions, key events, and total revenue for the last 7 days and the 7 days before that; (2) sessions by default channel group for the same two windows; (3) sessions by landing page for the same two windows, limited to 25 rows."
- ☐ "For every report, keep the full ResponseMetaData block in the output alongside the rows. Do not strip it. I specifically need subjectToThresholding, dataLossFromOtherRow, samplingMetadatas, emptyReason, currencyCode, and timeZone."
- ☐ "Record the number of requests made and a UTC timestamp in the JSON. Handle a 429 by backing off and retrying twice, then failing loudly. Never invent a number if a request fails, and never fall back to a default."
- ☐ "Print nothing to stdout except the path written. Errors go to stderr with a non-zero exit."

PROMPT 2 • BUILD THE PAGE

- ☐ "Read data.json and write one self-contained HTML file. No external requests of any kind: no CDN scripts, no web fonts, no remote images, no fetch calls. Inline all CSS. Charts are hand-written inline SVG, not a charting library."
- ☐ "Layout, top to bottom: a header with the property, the date range, and the build timestamp; a row of four KPI tiles each showing the value, the percentage change, and a small sparkline; a wide trend chart; a channel table with a share bar per row; and a one-line status strip at the bottom."

- ☐ "Every number must come from data.json. If a field is missing, render an em-free placeholder that says 'no data' rather than a zero. A zero and a missing value are different things and must never look the same."
- ☐ "Format currency with the currencyCode from the metadata and times with its timeZone. Do not assume dollars and do not assume UTC."
- ☐ "Use a system font stack, tabular numerals for every figure, and one accent color for the whole page. It gets read on a laptop at 1,400 pixels wide and printed to PDF. Both must look deliberate."

PROMPT 3 · MAKE IT ADMIT WHAT IT DOES NOT KNOW

- ☐ "Review the dashboard you just built against its own data.json, as a skeptic. For each rendered figure, tell me whether the metadata says it was sampled, thresholded, or affected by an (other) row, and whether the page currently discloses that."
- ☐ "Add a status strip that reports, in plain language: the sampling rate computed as samplesReadCount divided by samplingSpaceSize, whether thresholding was applied, whether an (other) row is present, and how many API requests built the page."
- ☐ "Any table where dataLossFromOtherRow is true must be labeled as a top-N view, never as a total, and must show the (other) row rather than silently dropping it."
- ☐ "List every place the page currently implies more precision than the data supports: a total that is really a partial sum, a ranking drawn from a truncated table, or a change percentage computed across two windows with different row sets. Propose the exact wording fix for each. Do not rewrite the whole page."

PROMPT 4 · WRITE THE WEEKLY SUMMARY

- ☐ "You are writing the analytics section of a Monday update. You are given a JSON object with computed findings. Every number in it is already correct. Do not recalculate anything, and do not introduce a figure that is not in the input."
- ☐ "Write at most five sentences. Lead with the single largest mover in absolute terms, not percentage terms. Group anything smaller into one closing sentence."
- ☐ "If quality.thresholded or quality.truncated is true, open with one short sentence saying which parts of the picture are incomplete."
- ☐ "Do not explain why anything changed. You do not have deploy logs, campaign calendars, or seasonality. If a cause seems obvious, say what you would need in order to check it."

THE RULE BEHIND ALL FOUR

- ☐ A dashboard's job is to be trusted, which means it has to be honest about its own gaps
- ☐ Anything that has to be complete belongs in the BigQuery export, not in a Data API report
- ☐ If you cannot explain where a number came from, it does not go on the page