

# The Power Up — Prompts & Steps

Intro to Claude Cowork | May 06, 2026

These prompts were shared during The Power Up — Intro to Claude Cowork. You're going to build a recurring-analysis workflow from scratch — mock weekly Google Analytics data, run a client-facing analysis, package it as a reusable Skill, then ship the result to Slack. Use the prompts below as starting points and rephrase as needed.

## A few notes before you start:

- Prompts below are starting points — rephrase as needed. Steps that are UI actions don't have a prompt.
- Each prompt is on a single line so it copy-pastes cleanly into the chat box.
- Select **Haiku** as your model when demoing — it's faster and keeps the pace up. When you're creating documents for real, switch to the most intelligent model available (e.g. Claude Opus) for higher-quality output.

## Setup Set your account instructions

Either paste a profile someone else wrote about you into your Claude account instructions, or have Claude build it with you:

```
Interview me to build a short profile about who I am and how I work. Keep it tight – a few sentences. When you're done, format it so I can paste it into my account instructions.
```

## 1 Start a project & drop in project context

**UI action** — create a new project, then start a new task inside it. Add the prompt below to **project instructions** to define the operating frame for every chat in this project.

```
All conversations in this project pertain to Google Analytics reporting for Nike.
```

## 2 Mock the raw weekly data

Select **Haiku** first. Then run this prompt to generate two weeks of fake-but-realistic Google Analytics data to use as your inputs.

```
Make me 2 weeks of mock raw weekly Google Analytics data – two CSVs, different weeks, realistic trends. Include the standard GA columns (sessions, users, bounce rate, channel breakdown, etc.). Put these in a new "Raw Reports" folder.
```

### 3 Set up the Outputs folder + folder rule

Create the destination for your client deliverables, and teach Claude how the two folders relate.

```
Create an "Outputs" folder in our project folder, alongside the Raw Reports folder. Then update our project folder instructions so it's clear: files in the Raw Reports folder are the source data we use to produce analysis – if I ask you to produce a new report, those are the inputs you'll likely want to reach for. Customer-facing outputs always save to the Outputs folder, and default to PDF for final deliverables.
```

### 4 Flag a memory about your stakeholder

Memory is for facts; instructions are for behaviour. Drop in a fact Claude should carry into every chat without you having to repeat it.

```
Remember that our key Nike stakeholder is Sarah Chen, Head of Marketing. She cares most about organic search performance and watches that channel above all others.
```

### 5 Fresh task — run the analysis

Start a new task inside the project. Before you prompt, show the audience the Raw Reports and Outputs folders in the "On your computer" view — this is the context Claude is about to inherit. Then run:

```
Run an analysis on Week 1 of the Raw Reports data and produce a single-page client-facing PDF – trends, anomalies, what's working, what's not, recommendations. Include at least two visuals and commentary that stands on its own.
```

### 6 Iterate (optional)

One-line nudges are enough — design tweaks, callouts, reordering — until it's what you'd actually send the client. No formal prompt: just talk to it.

### 7 Package the workflow as a Skill

Turn the workflow you just ran into a reusable Skill so the next report follows the same shape with no re-prompting.

```
Help me very quickly make a skill that can produce reports exactly like this for Nike in the future. No back and forth please, just build the Skill.
```

## 8 Run the skill on Week 2 in a fresh chat

Start a **new chat** in the project, then trigger the skill on the Week 2 data.

```
Run the Nike weekly report skill on Week 2 of the Raw Reports data.
```

## 9 Post a performance overview to Slack

Ship the deliverable. The whole layered stack — account instructions, project context, folder rule, memory, skill, connector — fires from one short prompt.

```
Send an overview of performance to date across Weeks 1 and 2 to the #nikereporting channel in Slack.
```