

Claude Cowork and Agent Utilization Manual

Table of Contents, Preface, 8 Chapters, Closing Note

Claude Cowork 및 에이전트 활용 매뉴얼

Kim Kyung-jin

English PDF edition

Claude Cowork and Agent Utilization Manual

Kim Kyung-jin

Claude Cowork and Agent Utilization Manual is an online AI Library book by Kim Kyung-jin. It covers Claude Code, AI agents, coding automation, work automation, and practical agent-based collaboration.

Table of Contents

Preface

Chapter 1: File and Desktop Environment Management

Chapter 2: Finance, Accounting, and Data Analysis

Chapter 3: Marketing, Content, and Sales Support

Chapter 4: Research and Business Intelligence

Chapter 5: Daily Operations and Communication Management

Chapter 6: HR, Legal, and Customer Support

Chapter 7: IT Projects and Software Development

Chapter 8: Advanced Systems and Tool Integration Automation

Epilogue

Preface

See every chapter

Next Chapter →

Monday morning, 9:17 a.m. Your computer's downloads folder contains 34 receipt photos from last week's business trip, 5 PDF quotations from vendors, and a half-finished draft report all mixed together. You need to sort these files, organize the receipts in Excel, and complete the report. Your team leader wants to see the results by 2 p.m. You check the clock. Four hours left.

In this situation, what you do with Claude is not ask a question in a chat box. You open a folder and tell it what needs to be done.

"Sort the receipt photos in my downloads folder by date, extract the vendor names and amounts, and create an Excel file. Rename the 5 quotation PDFs to the format 'VendorName_Quotation_Date' and move them to a separate folder." Claude makes a plan, reads the files, sorts them, renames them, and creates the Excel file. You meanwhile refine the report draft. By 1:30 p.m., both tasks are complete.

This is Claude Cowork. A desktop AI agent released by Anthropic in January 2026, an automated program that receives instructions from humans and makes decisions to execute tasks on its own. It differs from the chat interface approach of getting one answer at a time. Cowork accesses your computer folders, reads and edits files and creates new ones, connects to external services, and handles multi-step tasks independently.

The core philosophy of this book, compressed into one sentence, is this: "Delegation, not conversation." When you define specifically what needs to be done, Claude decides how to do it and executes it on its own.

That said, this book is not for programmers or IT specialists. The primary readers are office workers, business owners, freelancers, government employees, and students with only basic computer skills, writing documents and sending emails. It is fine if you have never used an AI tool before. We start with installation.

This book covers eight areas of work. It begins with fundamental tasks like file organization and progresses through Excel data analysis, marketing content creation, web research, email and calendar management, HR and legal document review, software development assistance, and ends with combining multiple tools to create automated workflows. Each chapter starts with following a single simple task, then varies the conditions, and finally combines multiple features in real-world business scenarios.

Every practice example includes prompts you can copy and paste directly. Rather than vague guidance like "try doing it this way," I designed them so that entering the completed sentence produces results. I also compare effective and ineffective prompts side by side. When you see the

difference firsthand, the sense of how to write good instructions takes hold quickly.

One thing I should mention upfront. As of 2026, Cowork is evolving rapidly. Starting as a macOS research preview in January, plugins and connectors have been added, Windows support has arrived, and features enabling direct operation within Excel and PowerPoint have emerged. The screens and menu locations in this book may differ from what you actually see. When that happens, please check the official Anthropic documentation at support.claude.com. This book is not a collection of screenshots from a particular version, but rather a guide to the mindset of using Cowork and the practice of delegation. Even as versions change, those principles remain constant.

Now we begin. In Chapter 1, from installing Cowork to opening a folder and issuing your first task takes around 10 minutes. After those 10 minutes, you will experience for the first time handing work to an AI agent. That experience will change your Tuesday afternoon.

Author: Kim Kyung-jin

See every chapter

Next Chapter →

Attorney · Former Member of the National Assembly · AI Policy Researcher

kimkj.com

© 2026 Kim Kyung-jin. All rights reserved.

#KimKyungjin #AIBLibrary #ClaudeCoworkandAgentManual

Chapter 1: File and Desktop Environment Management

← Previous Chapter

See every chapter

Next Chapter →

A law office in Seocho-dong. Three in the afternoon. A lawyer sits at her monitor with her forehead in her hand. She is searching for attachments to a brief that must be submitted to the court by tomorrow morning, but the desktop is buried under 387 icons. Files named "contract_final.pdf", "contract_final_really.pdf", and "contract_final_revised_0307.pdf" sit side by side, and she has to open all three to know which is the last version. The Downloads folder is worse. Three years of email attachments, photos received on Kakao Talk, court documents downloaded from the electronic litigation system, and scanned expense receipts from business trips all mixed together. To organize them, she would have to open and check each file, a task that would consume half a day on its own. So organizing gets pushed to tomorrow, which becomes next week, which becomes next quarter.

This scene is not unique to law offices. An accountant's month-end ledger folder, a shop owner's product image folder, a graduate student's research materials folder, a self-employed person's tax documents folder, each suffers from the same chaos in its own way. Digital files, unlike paper, take up no physical space, so you feel no immediate inconvenience from leaving them disorganized. When paper documents stack to knee height, you instinctively straighten them up. But hard drives and cloud storage enable human laziness endlessly. The price is chronic time wasted searching for critical documents when you need them most.

In January 2026, Anthropic released Claude Cowork, a tool that addresses this problem directly. Cowork is a desktop agent built into the Claude Desktop app, an AI that takes instructions from a person, makes judgments independently, and executes automated tasks. It differs clearly from traditional AI chat. Chat is a conversation where the user asks questions and the AI answers. Cowork is delegation, where the user defines the desired outcome and the AI executes it. You specify a folder, describe what you want, step away, and when you return, the work is done.

This chapter covers four file management functions of Cowork. Automatic folder organization, batch file renaming, information extraction and merging from multiple documents, and automatic data generation through image analysis. Each section proceeds in the order: basic usage with step-by-step instructions, applied cases, and real-world applications. After finishing this chapter, your desktop will be clear, and your Downloads folder will contain only logically organized subfolders.

1 Automatic Organization of Complex Folders and Desktop Files

A Why You Need It

Your computer's desktop and Downloads folder resemble a package storage area in front of an apartment. At first it is meant for temporary storage, but after a few days it becomes a permanent

holding yard. When screenshots, contract drafts, meeting transcripts, image mockups, compressed files, and files received through messaging apps all mix together, the problem is not appearance. You save the same document three times over, cannot find the latest version, delete files you should not have deleted, and download files again that already exist in the cloud. File organization is not a hobby in tidying up. It is work infrastructure that cuts down on search costs and error costs.

B What Makes Cowork's File Organization Different

Methods for organizing files existed before. In Windows File Explorer you can sort by file extension, and in macOS Finder you can create smart folders that automatically gather and display files matching certain criteria. Yet these only organize the view; they do not actually move files. You see them nicely organized in the smart folder and think you are done organizing, but the files still pile up in your Downloads folder, and backup and synchronization get tangled.

Cowork is different. It does not just look at file extensions. It reads content. It analyzes what is written inside PDFs and what is photographed in images. So even though both are PDFs, a contract goes to a "legal documents" folder, a receipt goes to a "finances" folder, and a thesis goes to a "research materials" folder. Files are actually moved to subfolders, names are changed, and duplicate files are detected.

Do not worry about security. Cowork operates inside an isolated virtual machine environment on your computer. Files are not sent to external servers. It works only in folders where you explicitly grant permission. Cowork cannot see folders you do not specify. For operations like file deletion that are hard to undo, it always asks for user permission first.

What is a sandbox? To understand how Cowork works, you need to know the concept of a sandbox. A sandbox is like a fenced-in patch of sand where children play. Within the fence they can play freely, but they cannot go outside. Cowork can read, create, and move files only within the folder that you specify. The rest of the system is invisible to it.

but cannot go outside. Cowork can read, create, and move files only within the folder that you specify. The rest of the system is invisible to it.

C Basic Usage: Organizing Your Downloads Folder Step by Step

Start with the simplest task. Classify the files that have accumulated in your Downloads folder by type.

1. Run the Claude Desktop app. At the top you will see three tabs: Chat, Cowork, and Code. Click the Cowork tab.
2. At the bottom next to the text input field is a checkbox labeled "Work in a Folder". Click this checkbox.
3. A folder selection window opens. Select your Downloads folder. A permissions dialog appears, asking whether Claude can read, edit, and delete files in this folder. Click "Allow". If you plan to use

this folder often, you may select "Always Allow".

4. Copy and paste the following prompt directly into the input field.

"Please organize the files in this folder by type. Create subfolders for documents (PDF, Word), images (PNG, JPEG), spreadsheets (Excel, CSV), videos (MP4), and other, and move each file to the appropriate folder. Do not delete duplicate files; move them to a separate "review needed" folder. When done, summarize how many files you moved and where."

5. When you press Enter, Cowork does not start work immediately. First it scans the folder to confirm the file list, then creates an execution plan. A progress bar and to-do list appear in the right sidebar.

6. Review the plan and then click "Proceed". Cowork begins execution. You can watch files move and folders being created in real time. When Claude finds duplicate files, a dialog appears asking what to do. Choose "move to review needed folder".

7. When the task is complete, Cowork displays a summary. You might see something like "Classified 47 documents, 89 images, 12 spreadsheets, 8 videos, and 23 other files. Moved 15 duplicate files to the 'review needed' folder."

Real-world tests show that it organized 186 mixed files in between one and a half to two minutes. One user entrusted a Downloads folder containing 2,200 files to Cowork, which in about twenty minutes organized them into eleven major folders with subfolders, renamed over 67 files to match their content, and separated 2 gigabytes of old installer files as deletion candidates.

D Applied Case: Classification Combining Date and Content

Classification by type alone is sometimes not enough. An accountant may want to organize files by month, a lawyer by case, a researcher by project. Add conditions to your prompt.

"Please read the PDF files in this folder and classify them by content. Move contracts to the 'legal documents' folder, receipts and tax invoices to the 'finances' folder, and theses or reports to the 'research materials' folder. Create subfolders by year within each folder. If a date appears in the file, use that as the basis; if not, use the file creation date."

Cowork opens the PDFs and analyzes their contents. It recognizes the contract date in contracts, the payment date stamped on receipts, and the publication date noted in theses, then places them in appropriate locations. If a person did this, it would take 30 seconds to a minute per file. With 200 files, you would spend at least two hours.

E Real-World Application: Folder-Specific Instructions and Scheduled Tasks

Cowork has a feature called folder-specific instructions. You save rules that always apply when you select a particular folder. Set this in the Settings menu under the Cowork section.

For example, save this instruction in your "client reports" folder. "When new files arrive in this folder, rename them in the format 'clientname_reporttype_date'. If the report has no summary at the top, create a three-line summary based on the first page content as a separate text file." This instruction automatically applies each time you work in that folder.

However, as of March 2026, this feature is not offered uniformly to all users. Some users may not see a Cowork option in the Settings menu. Anthropic is rolling it out in phases.

Do not worry. There is a way to achieve the same effect. Create a text file in your work folder named CLAUDE.md. When Cowork opens a folder, it reads this file first and uses the content as instructions for the task.

For the client reports folder example, create a file named CLAUDE.md inside it with this content.

"When new files arrive, rename them in the format 'clientname_reporttype_date'. If the report has no summary at the top, create a three-line summary based on the first page content as a separate text file."

The results are identical to setting this in the Settings interface. The file-based method is actually more flexible. You can put different instructions in each folder. Because it is a text file, editing and backup are straightforward. When you move a folder to another computer, the instructions travel with it. Users who engage deeply with Cowork prefer this approach over the Settings UI.

Use the scheduled tasks function (/schedule) to run automatic organization at set intervals. Enter /schedule in the Cowork input field to open the scheduling window.

"Every Friday at 5 p.m., check my Downloads folder. Classify newly arrived files by type, and move installer files (.exe, .dmg) older than one week to a 'deletion candidates' folder."

For scheduled tasks to run, your computer must be on and the Claude Desktop app must be open. If the computer goes to sleep or you close the app, scheduled tasks will not run.

F If You Run Into These Problems

(1) If the Cowork tab is not visible, check that your Claude Desktop app is the latest version. Download and install the newest version from claude.com/download. Cowork is not available on the free plan. You need a paid subscription of Pro (\$20 per month) or higher.

(2) If a folder access permission error occurs on macOS, check that the Claude app has been granted permission in System Settings > Privacy and Security > Files and Folders. System-protected folders like Downloads may require additional permissions. You can also create a separate work folder within the Desktop or Documents folder and operate from there.

(3) If the app closes during work, closing the Claude Desktop app or putting the computer into sleep mode will interrupt the task in progress. When you reopen the app and return to the Cowork tab, the record of the previous session remains, but

the work must restart from the beginning. When organizing large quantities of files, it is safer to change the computer's sleep settings to "Never."

2 Batch File Renaming

A Why It Is Needed

A filename is an address in the digital world. Names like "IMG_2847.jpg", "Document(1).pdf", and "KakaoTalk_20260303_093200123.jpg" well reflect the moment they were saved. Urgency, carelessness, and reluctance are all packed into one line. Yet as time passes, these names say nothing. There is no way to know what is inside a file until you open it.

When hundreds rather than dozens of files are in this state, the problem becomes more than just inconvenience; it becomes a work bottleneck. When an accounting team processes hundreds of invoices quarterly, when a tax professional manages client tax documents, when a law firm consolidates years of accumulated case records, opening each file one by one and changing its name takes an entire day.

Existing batch renaming tools did exist. Windows PowerRename can change the names of multiple files at once using regular expressions, and macOS Finder offers the ability to add or replace text in the names of selected files. However, these tools only look at the "appearance" of files. They cannot read what is written inside a file. You cannot tell from just the file extension and name that the contents of "Scan_001.pdf" are a confidentiality agreement with Samsung Electronics.

B What Is Different About Cowork's Renaming

Cowork reads the content of a file and assigns its name. It analyzes text within a PDF, uses OCR (Optical Character Recognition) to read text within images, and even extracts frames from video files to understand what content the video contains. This is why "Scan_001.pdf" becomes "20260115_Samsung_Electronics_Confidentiality_Agreement.pdf" and "IMG_9921.jpg" becomes "20260307_Starbucks_Receipt.jpg".

Many people start by wondering "what pattern should I use?" when establishing a naming convention. In practice, the order is reversed. First, you must decide "what situations will I need to find this file in?" The structure of the name changes depending on whether you search by date, vendor name, or document type. A filename should be a compressed version of your search scenario.

A good filename typically contains five pieces of information: date, source (vendor or sender), content category (contract, receipt, meeting notes, etc.), unique identifier, and version or status. Like "2026-03-07_ABCCorp_MSA_Draft_v02.pdf", separate elements with underscores or hyphens, and use a sortable date format (year-month-day).

[Note] Characters That Cannot Be Used in Filenames. Windows, macOS, and cloud storage have characters that cannot be used in filenames. These include forward slash (/), backslash (\), colon (:), asterisk (*), question mark (?), quotation mark ("), angle brackets (<>), and pipe (|). While Cowork

avoids these characters when automatically renaming, it is good to remember them when establishing your own rules.

C Basic Usage: Batch Renaming Receipt Files,A Walkthrough

Assume you have dozens of receipt scans saved with random names during tax season.

- ☐ In the Cowork tab, select the folder containing the receipts. Click the "Work in a Folder" checkbox and designate the folder.
- ☐ Paste the following prompt into the input field: "Read the contents of all receipt files in this folder and rename them in the following format: 'year-month-day_vendor_amount'. For example, the format should be '2026-03-07_Starbucks_4500won.jpg'. Also create a 'changelog.txt' file that records the original names and new names."
- ☐ Cowork opens each file and analyzes its contents. It extracts the date, business name, and amount from the receipt image using OCR. You can track progress in the right sidebar.
- ☐ When the task completes, all filenames in the folder are changed at once. At the same time, a changelog.txt file is created that records the change history, such as "Scan_001.jpg → 2026-03-07_Starbucks_4500won.jpg".

Instructing it to create a changelog.txt file is an important practice in real work. If Cowork pauses partway through processing many files, or if you need to continue the work later, having this record file lets you immediately see how much has been completed.

D Applied Case: Content-Based Renaming of Video Files

Cowork's multimodal analysis capabilities,its ability to process multiple forms of data including text, images, and video together,also work on video files. If screen recording files are listed from "screen_recording_001.mp4" to "screen_recording_015.mp4", Cowork extracts frames from the video and analyzes what websites or programs appear on the screen.

"Analyze the screen recording files in this folder and rename them based on the video contents. Reflect the website or program name visible in the video and the main activity in the filename. Use the format 'date_program_activity.mp4'."

Cowork reads the screen in the video and renames files like
"2026-03-05_Figma_Logo_Design_Review.mp4" or
"2026-03-06_Excel_Monthly_Sales_Analysis.mp4".

E Practical Application: Organizing 10 Years of Case Records at a Law Firm

In actual work, there are times when you must process hundreds or thousands of files at once. Assume you are organizing case records that have been inconsistently managed over 10 years at a law firm.

"Read the contents of all PDF and Word files in this folder. From each file, extract the case number, client name, document type (complaint, answer, judgment, opinion, etc.), and creation date. Rename the files in the format 'case_number_document_type_creation_date', create subfolders for each case number and move files to the appropriate folder. Put files where the case number is difficult to identify into an 'Unclassified' folder. Create a 'Summary_Report.txt' documenting all work performed."

When there are thousands of files, Cowork internally runs multiple sub-agents simultaneously and processes them in parallel. File batches are divided so multiple agents analyze them at the same time, with the main agent aggregating results. Progress is displayed in real time, and you can adjust direction or give additional instructions mid-process.

One caution: if ink on a receipt is smudged so the date is illegible, or if a contract with a foreign company uses a date format different from Korea's, the machine will be confused. Right after Cowork renames hundreds of files, you must review the entire list

and scan for errors. Technology has rescued us from repetitive typing, but the responsibility for ensuring the accuracy of the final record remains with the person in front of the screen. Having a changelog.txt file makes this review work much easier.

F If These Problems Occur

(1) If duplicate filenames cause conflicts, having two receipts issued by the same vendor on the same day can create identical names. Cowork usually automatically appends numbers to distinguish them, but it is safer to include a condition in the prompt beforehand saying "If duplicate names arise, append a sequence number at the end."

(2) If OCR recognition is poor, a poorly scanned receipt or a photo of creased paper may have inaccurate character recognition. For files whose contents Cowork cannot understand, you can instruct the prompt not to rename them and keep the original name, or add a "Needs_Review" tag. Add: "For files whose contents cannot be read, do not rename them; only add a 'Needs_Review_' prefix."

(3) If you want to revert files that were renamed incorrectly, the original and new names are recorded in changelog.txt, so you can pass this file to Cowork and instruct it to "Refer to changelog.txt and revert all filenames to their originals." Before large operations, making a copy of the entire folder is the most reliable safety measure.

3 Extracting and Refining Large Volumes of Document Information

A Why It Is Needed

In real work, information is not contained in one neat file. The reality is that materials related to a single project are scattered across five PDF reports, three text files in notepad format, two meeting notes written in markdown, and one CSV data file. To combine these into a single report or spreadsheet, you must open each file, verify its contents, remove duplicates, standardize formatting,

and rearrange in logical order. For a person, this takes half a day.

Assume you are writing a guidebook that compares and analyzes artificial intelligence policies from sixteen countries. The original text of the EU AI Act, U.S. executive orders, parliamentary meeting records from various countries, and industry statements are stacked in folders, each in different formats.

With the old approach, you would have had to open documents one by one, display them on half the screen, open a blank Word document on the other half, and repeatedly copy and paste key paragraphs. Because each document defines terms differently and regulations vary in intensity, creating a unified reference table requires enormous patience and a high degree of focus simultaneously.

B How Cowork Handles This Task

Cowork's document merging takes place in three stages: extraction, refinement, and assembly.

In the extraction stage, Cowork pulls structured data from files in various formats. Even if a table is mixed with text in a PDF, it separates the rows and columns of the table. For PDFs without a text layer, like scans, it reads characters through OCR. It recognizes the hierarchical structure of titles and subtitles in markdown files, and it identifies column names and data types in CSV files.

In the refinement phase, we standardize the format of extracted data. Date notation may differ from file to file. One document may show "March 15, 2025," another "03/15/2025," and yet another "15-Mar-2025." Cowork detects these inconsistencies and brings them into a single standard format. The same applies to units of measurement, currency notation, and spelling differences in names.

In the assembly phase, Cowork transforms the refined data into a single document. Here, a decisive difference between Cowork and conventional chatbots becomes apparent. Cowork does not display text on a screen; instead, it generates editable Word (.docx) or Excel (.xlsx) files directly on the user's computer. The completed report, with tables and charts included, is saved in a folder.

When the volume of files is large, Cowork internally runs multiple sub-agents in parallel. It divides documents into five batches; five agents each analyze their section, extract key information, then the main agent consolidates the results, checks for contradictions, and produces the final document.

What is a token? A token is the unit by which AI processes text. In Korean, one character roughly equals one to two tokens. Cowork has a maximum context window of one million tokens, the total volume of text it can read at once. This means it can analyze approximately hundreds of pages of documents simultaneously, measured by A4 paper.

C. Basic Use: Combining Scattered Meeting Minutes into a Single Report

Assume the team held five project meetings, with meeting minutes saved separately as text files and Word documents.

1. Select the folder containing the meeting minutes in the Cowork tab.

2. Paste the following prompt into the input field.

"Read all the meeting minutes files in this folder and compile them into a single Word document with this structure. On the first page, include a project overview and executive summary. From there, organize each meeting by date, listing the key decisions, follow-up actions by owner (Action Items), and unresolved issues. At the end, include a table grouping all follow-up actions from all meetings by owner."

3. Cowork reads each file and displays an execution plan. The sidebar shows the order in which files will be analyzed, the report's skeletal structure, and the estimated time to completion.

4. Click "Proceed," and Cowork begins the task. It internally runs tools like Python's python-docx library to generate an actual Word file.

5. Once complete, a finished .docx file is created in the folder. Open it in Word to find an editable report with a table of contents, tables, and numbered headings.

D. Application Case: Integrating Multi-Format Data into a Single Excel File

Combining files of various formats into a single Excel file is a common task in practice. Assume you have three competitor analysis PDFs, two market research CSVs, and four internal memo text files.

"Read all files in this folder and create a competitor comparison Excel file. On the first sheet, include a table comparing key metrics by competitor (revenue, employee count, main products, market share). On the second sheet, place the raw data extracted from each document. On the third sheet, record the data source and reliability. If currency units differ, standardize to Korean won."

An important design principle applies here. You must define the output structure in the prompt before Cowork begins. Asking for "everything on one sheet" produces a massive table that is hard to navigate. Dividing sheets is essential. If you separate the raw extraction sheet, the refinement sheet, the final comparison sheet, and the source verification sheet, fixing errors later won't require touching other sheets.

When asking Cowork to create an Excel file, you can instruct it to include formulas. With specific requests like "add category totals using a SUMIF formula" or "highlight items exceeding targets in green with conditional formatting," it produces an actual working Excel file with formulas, not a simple CSV.

E. Practical Use: Automating Resume Screening

Assume dozens of job applicant resumes have been submitted as PDFs, and your company has a separate hiring evaluation criteria document.

"This folder contains 'hiring-evaluation-criteria.docx' and applicant resume PDF files. Review each resume according to the evaluation criteria items (years of experience, job-relevant skills, education, certifications, project experience). Analyze each applicant's strengths and weaknesses by item and assign scores of one to five per item. Export the results to an Excel file sorted in descending order by total score. Add a one-line comment in the 'Recommendation' column."

Cowork first reads the evaluation criteria to understand the framework, then applies that framework to each resume and assigns scores. For twenty applicants, a task taking a person half a day or more, Cowork completes in minutes.

Resumes contain personal information. Names, contact details, schools, and employment history are sensitive data. Although Cowork works locally on the user's computer and does not transmit files externally, you must still verify with your organization's data protection policy whether passing such data to an AI tool is acceptable. Cowork itself is in Research Preview stage, and Anthropic explicitly states it should not be used for regulated tasks.

Using AI-assigned scores directly for hiring decisions is also risky. Cowork reads text from resumes and applies evaluation criteria but cannot judge an applicant's actual capabilities or cultural fit. Use AI evaluation results as a reference for initial screening only; humans must make the final decision.

F. A Warning About Hallucinations

When summarizing and merging large volumes of documents, Cowork sometimes generates content that does not actually exist. This is called hallucination, the phenomenon in which an AI produces false information as if it were factual.

For example, when creating a report comparing AI regulation across countries, Cowork might misread context and invent legal clauses that do not exist, or incorrectly insert a country A's enforcement rule into country B's section. Hundreds

of pages of source material cannot be fully retained in human memory, so plausibly packaged errors can escape notice in a quick review.

There are two countermeasures. First, add to your prompt: "Always include the source file name and page number next to each data item." When sources are documented, verification against the original becomes much simpler. Second, for items in Cowork's output where errors are critical, numbers, dates, proper names, legal clause numbers, have a person cross-check against the original. As machines handle the physical labor of drafting and gathering data, humans take on the role of verifier, separating fact from fiction.

G. If These Problems Arise

(1) When tables break and fail to import properly into Excel: Tables within PDFs appear clean on the surface but internally mix text and images. When Cowork cannot recognize a table correctly, accuracy improves if you directly describe the table structure in your prompt, such as: "The table has

columns named 'Date, Item, Amount, Note,' and each row represents one transaction."

(2) When there are too many files and the task stops midway: Cowork has a finite context window. Processing dozens of files at once can be difficult. In such cases, dividing work by folders is more stable. When you narrow the scope, like "Process only January to March files first," Cowork handles each batch reliably.

(3) When the original file is a scan and text extraction fails: A scanned PDF is an image, not text. While Cowork uses OCR to read characters, recognition rates drop with low scan quality. Errors occur if the original is skewed, ink has smudged, or resolution is poor. Adding to your prompt "Mark items with uncertain OCR recognition with 'verification needed' in the Notes column" helps you identify points that need checking beforehand.

4. Image File Analysis and Data Transformation

A. Why It Matters

A studio in Dongdaemun. The camera flashes that crackled through the night have stopped, and hundreds of spring-season garments, shooting complete, hang on racks. On the photographer's monitor sits a folder holding thousands of high-resolution images. The shoot is done, but the work is just beginning. For these photos to appear on an online shopping mall, someone must view each image one by one and input the product name, category, color, material, and tags into Excel.

Visually determining whether a dress's neckline is V-neck or round neck, whether its color is pale cobalt blue or deep navy, whether its material is linen blend or 100 percent cotton, then typing this information demands endless patience. With ten items, it is bearable; with one hundred or five hundred, the process consumes the entire team's time. In hundreds of repetitive keystrokes, mistakes creep in: typing "black" instead of "navy" or omitting essential tags. When a fashion brand launches hundreds of new products each season, delays in product registration mean lost sales.

This is not unique to clothing e-commerce. Extracting dates and amounts from stacked business trip receipts to compile an expense report, identifying square footage and layout from real estate photos to organize a property list, documenting construction site progress from photos, all follow the same pattern. The bottleneck is converting visual information in photos into textual data.

B. What Cowork Reads from Photos

Cowork uses the visual analysis capability of multimodal AI. When it receives a photo, multiple layers of analysis proceed simultaneously.

Object detection identifies the outline of the product in the photo. It determines whether it is a shirt, pants, or bag. Color analysis extracts the dominant color and converts it to a specific shade name like navy, beige, or burgundy. Analysis of texture and gloss patterns estimates the material, distinguishing rough denim from smooth silk or fuzzy knit. OCR reads text in the photo, brand labels, care instructions,

price tags, and other text. Caption generation summarizes the entire photo in a single sentence, producing a description that combines product features: "Women's navy linen V-neck cardigan, oversized fit, casual style," for example.

All of these processes complete in seconds for a single photo.

Basic Use: Creating a Product Inventory Excel File from Clothing Photos

Suppose you have a folder containing 28 clothing photos.

1. Select the folder containing photos in the Cowork tab.
2. Paste the following prompt in the input window.

"Analyze the product photos in this folder and create an Excel file for registering products in the shopping mall. Look at each photo and extract the following items and organize them in columns: filename, product name (auto-generated by AI), main category (tops/bottoms/outerwear/accessories), subcategory (t-shirt/blouse/jeans/coat, etc.), main color, estimated material, style tags (casual/formal/sporty, etc.), recommended search keywords. Organize by color or category to improve readability."

3. Cowork starts analyzing the photos one by one. The sidebar shows progress like "Photo 1/28 analyzing... Photo 2/28 analyzing..."

4. When the work is complete, an Excel (.xlsx) file is generated in the folder. When you open the file, 28 rows are filled with data for each photo. The first column shows the filename, the second column has auto-assigned product names like "Women's Spring Linen V-neck Cardigan Sky Blue," and subsequent columns contain category, color, material, and tags.

It would take a person 1 to 2 hours to fill in this table while looking at 28 photos. Cowork completes the same task in minutes.

Practical Example: Generating an Expense Report from Business Travel Receipts

It is not just product photos. Business travel receipt photos can be processed the same way.

"Read the receipt photos in this folder and create an expense report Excel file. Extract the date, vendor name, category (meals/transportation/lodging/other), and amount. Insert a SUMIF formula in the Excel to calculate totals by category. If the photo is dark or the text is blurry and you cannot verify the amount, write 'Verification needed' in the remarks column."

Cowork extracts information via OCR even from crumpled receipts. By applying formulas in Excel, a report is created with meal and transportation totals automatically calculated. If you compare only items marked "Verification needed" against the original receipts, the time required is significantly less than doing everything manually.

Real-World Application: Building a Large-Scale Product Catalog

Suppose you must register 500 new product photos all at once at the change of season. The photos include front shots, back shots, and detail shots mixed together, along with color variations of the same product.

"Analyze all product photos in this folder. Group front, back, and detail shots of the same product as one product. Handle color variations as options for the same product. Create the following columns in Excel: product code (auto-generated), main product name, main category, subcategory, base color, color variation list, estimated material, style tags, season tags (spring/summer/fall/winter), recommended search keywords, main image filename, additional image filenames. Mark items with uncertain materials in the 'Review needed' column."

The key in this prompt is the last sentence: "Mark items with uncertain materials in the 'Review needed' column." The difference between having and not having this one line is significant. Having a person review all AI-generated data from start to finish is inefficient. When AI marks items with low confidence on its own, a person only needs to verify those specific items.

What Machines Cannot Read

Cowork's image analysis excels at extracting objective facts and numbers. It accurately captures color, shape, and text. However, it is difficult to expect it to understand the subjective sensibilities embedded in the fashion genre.

A brand planner might want to express a result that a machine reads as "faded gray cotton t-shirt" using the sensory language of "vintage wash grunge look." A machine might mistake deliberately bold design in ripped jeans as "damaged defective goods," or interpret asymmetrical design as "cutting error." This does not stem from a lack of machine logic, but from the absence of sensory perception and cultural context.

Color identification also requires care. If the photo background is cream-colored, the background color might be detected before the product color, and glossy materials appear brighter than they are due to light reflection. In practice, the color candidates extracted by AI are organized to match the shopping mall's color dictionary. For instance, ivory, cream, and off-white are all grouped as "ivory," while charcoal and dark gray are combined as "charcoal."

Material identification requires even more care. While it is possible to estimate cotton versus wool or linen versus polyester to some degree from images alone, getting the exact blend ratio is difficult. Since material labeling is information directly linked to consumer disputes, it is safer to treat automation that determines material from photos only as a "candidate" level. If you include photos of wash labels or brand tags and supplement them with OCR, accuracy improves.

The machine has organized thousands of photos into Excel. The final adjustment task of breathing the brand's unique philosophy and sensibility into that standardized table is the human's responsibility. Cowork builds the framework, and the human adds the substance. Freed from tedious keyboard typing, the worker can focus on extracting language that will capture the customer's heart from the data the machine produces.

If You Encounter These Issues

1. If the product in the photo is cut off or overlapping, making recognition inaccurate: When photographing products, use a single-color background and position the product in the center of the frame to significantly improve recognition rates. If multiple products appear in one photo, Cowork may become confused, so the rule is one product per photo.
2. If color identification differs from reality: The lighting in the shooting environment has a major impact on color recognition. Photos taken in natural light and photos taken under fluorescent lights can show the same garment in different colors. Adding the condition to the prompt "Classify colors broadly by family (blue family, red family) rather than exact single color names" reduces errors. A person makes the final confirmation of accurate colors.
3. If category classification is ambiguous: Items with blurry boundaries, like t-shirts versus sweatshirts or shirts versus blouses or loafers versus flat shoes, confuse machines too. Accuracy improves when you provide the company's category criteria directly in the prompt. "Our shopping mall's categories are as follows. Tops: t-shirt, blouse, shirt, knit, crewneck. Bottoms: jeans, slacks, skirt, shorts. Outerwear: jacket, coat, jumper, cardigan." By limiting choices this way, the machine makes decisions within that range.

The four tasks covered in this chapter share a common structure. You specify a folder, describe the desired form of output, review the plan that Cowork creates, and approve execution. When the work is complete, a person verifies the results. Once you master this flow, you can apply the same approach to financial analysis, marketing content creation, and research automation in chapters to come.

If your desktop is empty and only logical subfolders remain in your Downloads folder, you have buttoned the first button correctly. The next chapter will explore what you can do with these organized files, entering the world of financial and data analysis.

← Previous Chapter

See every chapter

Next Chapter →

Attorney · Former Member of the National Assembly · AI Policy Researcher

kimkj.com

© 2026 Kim Kyung-jin. All rights reserved.

#KimKyungjin #AIBLibrary #ClaudeCoworkandAgentManual

Chapter 2: Finance, Accounting, and Data Analysis

← Previous Chapter

See every chapter

Next Chapter →

Every month-end, a similar scene unfolds in every office. An accounting clerk spreads out a pile of crumpled receipts and alternates between staring at an Excel screen. The finance team manager downloads a CSV file from the credit card company website and grapples with pivot tables. The business planning office stays up all night creating revenue trend charts, and a new employee dispatched from an accounting firm strains their eyes trying to find why two ledgers differ by a single won.

The four tasks covered in this chapter are the repetitive manual labor of a company's financial operations. The work of transferring receipts and invoices to ledgers, classifying card expenses and finding waste, extracting management indicators from payment data, and matching numbers across multiple ledgers while building future financial models.

Cowork significantly reduces the "entering and checking time" in these four tasks. Instead of a person typing numbers, a machine reads files and creates results that include Excel formulas. However, there is one principle that runs through this chapter. Numbers created by Cowork must always be validated by a person. In accounting, "approximately correct numbers" are the same as "wrong numbers." The machine handles typing, and people make judgments. As long as you do not forget this role division, Cowork becomes the most diligent assistant to your finance team.

[Note to Remember] Frequently appearing terms in this chapter

CSV (Comma-Separated Values) is a text file that stores data separated by commas. It can be opened in Excel, and is the format used when downloading transaction histories from credit card companies and payment platforms.

A Pivot Table is an analysis tool in Excel that summarizes large amounts of data by category. It is used when converting hundreds of transaction records into summary tables organized by month and category.

1 Receipt and Invoice Settlement Automation

A Why This Function Is Needed

Expense settlement is the top-ranked task every office worker dislikes. It's the work of pulling crumpled taxi receipts from your pocket after a business trip, downloading scattered hotel invoices from email, and typing each one into Excel by hand. According to a 2023 study by the Global Business Travel Association (GBTA), the average cost for a company to process a single expense report is \$58. If errors are included, it exceeds \$100. It costs money to organize records of spending

money.

The problem is that receipts come in different formats.

Some are printed on thermal paper and the text fades after three months. Some come as PDFs, but the vendor's placement of the amount differs from one to the next. The act of a person reading these irregular fragments of information and transferring them into standardized ledgers is repetitive labor that tests patience, and it carries the risk that typing a single number wrong will throw off the entire ledger balance.

B What Cowork Does for You

Cowork analyzes an entire folder containing receipt photos and PDF files. It extracts the date, vendor name, amount, and tax from each file, and organizes this data in an Excel file. Inside the Excel are not just text but SUMIFS formulas that calculate subtotals by department, and conditional formatting that highlights budget-exceeded items in red. What Cowork creates at once is a completed expense settlement report.

The technical foundation of this process is multimodal recognition. The language model embedded in Cowork reads not just the text in an image but also infers what that text means from context. It determines from surrounding text whether a number like "2026-03-07" is a transaction date or part of a business registration number.

C Basic Usage Tutorial: Your First Settlement with 10 Receipts

What You Need: Create an "Expense Settlement-Practice" folder on your desktop and place 5 to 10 receipt photos taken with a smartphone inside. It helps if various business types are mixed together, such as cafes, taxis, and stationery stores.

- ☐ Open the Claude Desktop app and click the "Cowork" tab in the mode selection at the top.
- ☐ Press the folder icon at the bottom left to select the "Expense Settlement-Practice" folder. Grant Claude permission to read and write files in this folder.
- ☐ Enter the following prompt in the input box:

"Analyze all receipt photos in this folder for me. Extract the payment date, vendor name, and amount from each file, and organize it in an Excel file. Add a total formula to the bottom of the Excel."

- ☐ Claude displays a work plan on the screen. Steps like "Analyze 10 receipt images → Extract data → Create Excel file" are listed. After confirming the content, approve execution.
- ☐ Claude reads each receipt in order and fills the extracted data into Excel. After about 2 to 3 minutes, an "expense-settlement.xlsx" file is created in the folder. When you open it, you see the date, vendor name, and amount organized by row, and the total is calculated at the bottom using a SUM formula.

D Application Example: Adding Category Classification and Conditional Formatting

Now that the basic settlement is done, let's move to the next level. This time we'll divide expense items by category and highlight items above a certain amount in a noticeable way.

- Add more receipts to the same folder. Mix in various types such as meals, transportation, accommodation, office supplies, and others. If you have PDF invoices, include them together.

- Enter the following prompt in the input box:

"Analyze all receipt and invoice files in this folder for me. Extract the date, vendor name, amount, and payment method from each file and organize in Excel. Classify expenses into categories: meals, transportation, accommodation, office supplies, and other. Add category subtotals using SUMIFS formulas and apply yellow background conditional formatting to expenses over 50,000 won. For items where the amount is missing or difficult to read, mark "Manual Verification Needed" in the remarks column."

- Claude begins the task. Since category determination is added, it takes a bit longer. Claude classifies "Starbucks" as meals and "Kakao Taxi" as transportation. When the category is ambiguous based on vendor name alone, it refers to the amount scale and transaction time to make inferences.

- Open and review the Excel created by Claude. Category subtotals are calculated as formulas, and items exceeding 50,000 won have a yellow background. If there are rows with "Manual Verification Needed" in the remarks column, directly check the original receipt yourself and make corrections.

There is one thing you must check here. Confirm that the categories Claude assigned match your company's chart of accounts standards. AI classifies by general common sense, but accounting rules differ by company. One company puts "taxi fare" in "transportation" and another puts it in "business travel expenses." A person must correct the standards.

E Real-World Use: Building a Monthly Settlement System

If you repeat the same task every month, you can turn it into a system. Use Cowork's Scheduled Tasks feature, which automatically runs at a designated time as a recurring task.

- First, organize the folder structure. Create a "Monthly Expenses" folder on your desktop, and place a "Pending" folder and a "Completed" folder inside. For one month, you simply drop receipt photos into the "Pending" folder.

- Type /schedule in the Cowork input box.

- Claude asks you a few questions. You select the task name, execution interval, and execution time. You select "the 1st of every month at 8 a.m."

- Set the task content as follows:

"Analyze all newly added receipt and invoice files in the Monthly Expenses/Pending folder. Extract date, vendor name, category, and amount to create a '2026 Month X Expense Settlement.xlsx' file. Include category subtotal formulas and total formulas, and highlight items exceeding 100,000 won with conditional formatting. Move the processed files to the 'Completed' folder."

□ Claude creates a scheduled task. You can view registered tasks in the "Scheduled" menu in the left sidebar.

There is something to be careful about. Scheduled tasks only run when your computer is awake and the Claude Desktop app is open.

If your computer is off at 8 a.m. on the 1st of each month, the task will not run. The task will run once you open the app after arriving at work, so developing a habit of opening the app right after arriving at work will do the job.

[Note to Remember] Accuracy of Receipt Recognition

Professional document parsers like Azure's Invoice Model, AWS's Textract, and Google's Document AI show accuracy rates above 95 percent on standard-format documents.

Cowork's multimodal recognition performs at a similar level with clean photographs. However, accuracy drops with wrinkled thermal paper, blurred handwritten ink, and photos taken in dim lighting.

Google Cloud's official documentation recommends a minimum of 200dpi, preferably 300dpi or higher resolution. When photographing receipts with a smartphone camera, laying the receipt flat and shooting in bright lighting improves recognition accuracy.

If You Encounter These Issues

(1) Receipt photo is too dark and amount cannot be extracted. Claude does not arbitrarily estimate items that are difficult to read; instead it marks "Unable to verify" in the remarks column. Retake the receipt in bright lighting, or manually enter the amount from the original receipt into Excel.

(2) The same merchant appears with different names ("Starbucks Gangnam", "STARBUCKS", "Starbucks"). This is a 'normalization' issue. You can give Claude the additional instruction: "Standardize merchant names. Combine all variations of Starbucks into a single entry."

If you have a set list of frequent merchants, it is effective to create a standard merchant name list in a separate text file, place it in the same folder, and instruct Claude: "Standardize merchant names based on this list."

(3) Excel formulas appear broken. If you open an Excel file generated by Claude in LibreOffice, some formulas may break. They display correctly in Microsoft Excel. If using LibreOffice, instruct Claude: "Create formulas that work in LibreOffice as well."

2 Credit Card Expense Analysis

Why You Need This Feature

When you receive a monthly credit card statement, most people only check the total amount and make payment. Hundreds of transactions are listed line by line, but few people take the time to categorize them by type, compare them with the previous month, and identify waste. Time is the limiting factor.

The same applies to companies. Combining monthly statements from three corporate cards produces thousands of lines. The merchant field is filled with cryptic codes like "NHN KCP regular payment", "AMZN MKTP US", "SQ *SQC CAFE" from payment processors. Figuring out from memory whether this is server hosting, advertising, or employee meal expenses requires significant mental effort. In the meantime, a \$19 monthly foreign software subscription you tested six months ago continues to be charged without anyone noticing.

What Cowork Does

Cowork analyzes CSV files downloaded from your card issuer and produces three outputs.

First, a summary sheet that categorizes all transactions into categories such as meals, transportation, software subscriptions, and advertising.

Second, a summary chart showing spending by category and month-over-month changes.

Third, a cost reduction proposal that identifies cancellable subscriptions, duplicate charges, unusual transaction times, and similar issues.

Getting Started: Analyzing One Month of Card Transactions

Preparation: Download this month's transaction history as a CSV file from your card issuer's website or app. Most Korean card companies provide an "Excel download" or "CSV download" button on their transaction inquiry page. (1) Create a "Card Analysis" folder on your desktop and place the downloaded CSV file there.

(2) Connect this folder in Cowork mode.

(3) Enter the following prompt in the input box.

"Analyze the credit card transaction CSV file in this folder. Categorize all transactions into meals, transportation, shopping, software subscriptions, living expenses, and other. Calculate category totals and percentages of the whole, and create an Excel file with a pie chart visualization."

(4) Claude reads the CSV and begins classification.

It infers the merchant category from the merchant name. "Baemin" is classified as meals, "Kakao T" as transportation, and "Netflix" as subscriptions. After about two to three minutes, an Excel file with the analysis results is generated in the folder.

(5) Open the Excel file and review the classification results. If any items were classified as "other" due to ambiguous merchant names, manually correct them to the appropriate category.

Advanced Example: Combining Multiple Cards and Monthly Comparison

Let's combine transaction data from two corporate cards and one personal card, then compare them with the previous month.

(1) Place three CSV files from this month and three from last month in the "Card Analysis" folder. It is helpful to include the card name and month in the filename, like "Samsung_Card_2026_02.csv" and "Samsung_Card_2026_03.csv".

(2) Enter the following prompt:

"Analyze all six credit card CSV files in this folder. Date formats may differ by card, so standardize them. Combine all transactions into a single table and categorize them as meals, transportation, software subscriptions, advertising, entertainment expenses, office supplies, and other. Create the following sheets in Excel."

First sheet: complete transaction list. Second sheet: monthly summary by category (comparing February and March). Third sheet: a chart showing category increases and decreases. Fourth sheet: a list of recurring subscription services charged each month. Include service name, monthly charge, and initial transaction date in the subscription list."

(3) Claude begins by aligning the column structures across multiple CSVs. Card issuers use different column names. Some files use "date of use", others use "transaction date". Claude recognizes these differences and integrates them into a single standard structure.

(4) Open the resulting Excel file. The monthly summary in the second sheet shows at a glance which spending categories increased from February to March. Looking at the subscription service list in the fourth sheet, you may find services that are still being charged but no longer used.

There is one aspect often overlooked in credit card expense analysis: refunds and cancellations. In the CSV, negative amounts represent refunds or cancellations; if you simply sum them, your actual spending appears lower than it is. You can prevent this by instructing Claude: "Mark refunds separately and distinguish between net spending and total spending."

Practical Application: Cost Reduction Report

Organizing numbers is different from reducing costs. Let's create a cost reduction report to present to executives.

(1) In the folder with the analysis results you created earlier, enter the following prompt:

"Create a cost reduction report as a Word document based on the card spending data you just analyzed. Include the following items: first, unused subscription services that can be canceled

immediately and estimated savings. Second, items where the same type of service is charged multiple times. Third, estimated savings if monthly payments are converted to annual payments."

Fourth, categories with significant increases compared to the previous month and hypothesized causes. Include a three-line executive summary on the first page of the report."

(2) Claude reviews the analysis data and generates a Word document. It automatically identifies subscription services with no usage history for three months or longer and calculates the price difference between monthly and annual payment options.

Remember this: a machine can only classify a 500,000 won restaurant expense as "entertainment spending". It cannot determine whether that meal was strategic spending to persuade an investor or uncontrolled waste. Machines are fast at organizing data and spotting signs of waste, but deciding whether to keep or reduce that spending belongs to the manager.

If You Encounter These Issues

(1) Korean characters appear corrupted when opening the CSV file. Korean card companies often use EUC-KR encoding in their CSV files. You can instruct Claude: "Auto-detect the CSV file's encoding, convert it to UTF-8, and then analyze it."

(2) Merchant names are cryptic, making classification inaccurate. Create a text file with classification standards for merchants you trade with regularly and keep it in a folder. Examples: "AWS = IT Infrastructure, Baemin = Food Expenses (Employee Benefits), Kakao T = Transportation." Instruct Claude: "Please read this classification standard first and apply it."

(3) Foreign payment amounts are mixed between Korean won and US dollars. Instruct Claude: "For foreign payments, analyze based on the amount converted to Korean won and display the original currency and applied exchange rate in separate columns."

3 Real-Time Revenue and Transaction Analysis

A Why This Feature Is Needed

For online shopping mall operators, SaaS (Software as a Service, a business model where software is provided via subscription through the internet) product sellers, or subscription service providers, payment data floods in daily. Records of customer purchases, refunds, and subscription cancellations through payment platforms like Stripe, PayPal, and Toss Payments accumulate by the second.

The issue is that it takes time to convert this data into numbers usable for management.

To determine total revenue, which products sell well, customer churn rate, and whether refunds are surging, you must download data, create pivot tables, and draw charts. If you repeat this work weekly, it consumes significant time.

B What Cowork Does for You

Cowork analyzes payment data CSVs to calculate core business metrics, identify anomalies, and even create summary documents for management reporting.

Through MCP connectors (Model Context Protocol, a communications standard for AI to exchange data with external services), you can connect directly to payment platforms like Stripe, or analyze CSV files by placing them in a folder.

In February 2026, Anthropic released five finance-focused plugins.

Financial analysis, investment banking, equity research, private equity, and asset management plugins. MCP connectors from FactSet and MSCI were also launched simultaneously, allowing institutional-grade market data to be queried directly within Cowork. However, these plugins and connectors are optimized for enterprise subscriptions (Team, Enterprise), so for individual business operators, CSV-based analysis is the practical starting point.

C Basic Usage Example: Weekly Revenue Analysis

What you need: Download the past seven days of transaction records as a CSV from your payment platform. From Stripe, you can download them in the Dashboard's Payments menu; from Toss Payments, from the admin page.

- ☐ Create a "Revenue Analysis" folder and place the CSV file in it.
- ☐ Connect the folder in Cowork and enter the following prompt.

"Analyze the payment records CSV in this folder. Calculate total revenue, average daily revenue, average payment amount per transaction, number of refunds and refund rate for the past seven days. Organize the results in Excel and include a line chart showing daily revenue trends."

- ☐ Claude reads and analyzes the data. The results Excel includes a sheet with key metrics summarized and a daily revenue trend chart.

D Application Example: Category-Specific Revenue and Anomaly Detection

- ☐ Request deeper analysis using the same CSV file.

"Perform additional analysis on the data I just analyzed. Calculate revenue share by product category and show payment success and failure rates. If there are days where revenue fluctuated 20 percent or more compared to the same period last week, mark them. Also mark any days where refunds exceeded five per day. Add the results to an 'Anomalies' sheet in Excel."

- ☐ Claude identifies abnormal fluctuations using statistical standards. If refunds concentrated on a specific date, it becomes a clue for investigating the cause. If a specific product's revenue dropped sharply, it signals the need to review pricing policy or inventory issues.

According to Stripe's official documentation, revenue recognition graphs may experience up to a 72-hour delay before appearing on the dashboard. Operating metrics like payment approval rates or

failure rates differ from accounting-recognized revenue.

One is heartbeat; the other is a health exam report. When designing an executive dashboard, it is better not to mix these two in one cell.

E Real-World Application: Automating Executive Weekly Briefings

Let's run this analysis automatically each week and create a summary document for management reporting.

- ☐ Enter /schedule in Cowork.

- ☐ Configure as follows.

Task name: Weekly Revenue Briefing Execution Cycle: Every Monday at 9 AM Task Details:

"Analyze the most recent payment records CSV in the Revenue Analysis folder. Calculate last week's total revenue, revenue share by product, refund rate, and payment success rate, and show changes compared to the previous week. Alert if anomalies exist. Write the analysis results as a one-page Word summary and save it in the same folder. Use the filename format 'Weekly_Revenue_Briefing_2026_MM_DD.docx'."

- ☐ Every Monday morning, when you turn on your computer and open Claude Desktop, the briefing document is ready in the folder.

The shorter the executive summary, the better.

In the first paragraph, include three key changes this week; in the second paragraph, hypotheses about causes; in the third paragraph, immediate actions to take and items to observe next week.

No matter how excellent the dashboard, executives do not spend much time staring at the screen. Without a one-page memo, data exists but decision-making lags.

F If This Problem Arises

(1) The date format in the CSV file is being recognized incorrectly. When Excel auto-converts dates, US format (month/day/year) and Korean format (year-month-day) get mixed up. Simply instruct Claude: "Standardize the date format to YYYY-MM-DD."

(2) The payment data contains personal information (customer names, emails). Personal identification information is not needed for analysis.

Before placing the CSV in a folder, delete columns with personal information, or instruct Claude: "Ignore the customer name and email columns and perform the analysis." Cowork operates on a local computer, so data does not go external, but caution is needed even in granting folder access to folders containing sensitive data.

(3) I want to connect directly with overseas payment platforms like Stripe. You can connect to Stripe through Cowork's MCP connectors.

As of February 2026, financial data connectors like FactSet and MSCI have been officially released, and the Stripe connector is also available in settings. However, API connection setup requires technical knowledge, so it is recommended to start with the CSV download method initially and attempt connector integration once you become familiar with it.

4 Financial Modeling and Account Reconciliation

A Why This Feature Is Needed

There are two most grueling tasks in the finance department.

One is account reconciliation, which is the process of verifying that numbers in different ledgers match.

This is checking whether transaction records downloaded from the bank match the numbers in the company's internal ledger. Among tens of thousands of transactions, you must identify a few where the amounts don't match. When the accounting period arrives, accountants deploy VLOOKUP functions and pivot tables to compare the two ledgers late into the night.

The other is financial modeling.

This involves building a 3-Statement model in Excel where the income statement, balance sheet, and cash flow statement are linked together. It is a system that predicts how cash flow changes if revenue rises 10 percent, and when cash will run short if capital investments are made.

To build this model, you must apply cascading formulas to hundreds of cells, and if even one is missing, the entire structure breaks.

B. What Does Cowork Do?

Cowork compares multiple spreadsheets and automatically identifies discrepancies. It distinguishes between differences in amount, items that exist only on one side, and dates that are slightly off, and organizes them into a separate sheet.

In financial modeling, when assumptions are entered, it generates an Excel model with three linked financial statements and visualizes results as scenario variables are adjusted.

However, accuracy in this area remains limited. In a February 2026 comparative test conducted by Wall Street Prep, Claude, Shortcut, Microsoft Copilot, and ChatGPT were asked to build Apple's 3-Statement model.

Claude received the highest score among the four tools, but scored 5.5 out of 10. It did not reach the level of a junior analyst.

AI was found to inaccurately input portions of historical financial data or directly enter numbers instead of formulas (hard-coding). The financial model created by AI is a "tool that rapidly takes you to 60 percent," not "a finished product you can submit without review."

C. Basic Use: Reconciling Two Ledgers

What you need: Prepare two Excel files, one containing transaction records downloaded from the bank and one containing the ledger extracted from the company's internal accounting system.

- ☐ Create a folder called "Reconciliation Work" and place both files in it.
- ☐ Connect the folder in Cowork, then enter the following prompt.

"Compare the two Excel files in this folder for me.

One is bank transaction records and one is the internal ledger. Compare them using transaction date, amount, and counterparty, then organize the following. First, the number and amount of transactions that match on both sides. Second, a list of transactions with different amounts. Third, transactions that appear only in the bank records but not in the internal ledger. Fourth, transactions that appear only in the internal ledger but not in the bank records. Create the results as separate sheets in a new Excel file. Apply a yellow background to mismatched items."

- ☐ Claude reads both files and begins reconciliation using the key (transaction date + amount + counterparty). Even if transaction dates differ by a day or two, or counterparty names are slightly different, Claude infers the possibility that they represent the same transaction.
- ☐ The result Excel is generated. You only need to visually review the items organized in the "Mismatch" sheet.

Instead of finding a needle in a haystack of tens of thousands of transactions, the machine has narrowed it down to a few candidates.

[Note] How to Classify Reconciliation Status

Dividing reconciliation results into simply "match/mismatch" is insufficient. In practice, a more detailed classification is used.

Exact Match indicates a transaction with no issues. Amount Mismatch may be due to fees or exchange rate differences. Date Shift occurs when it is the same transaction but recorded on different dates. Missing refers to a transaction that exists on only one ledger, which could be a data entry omission or a transaction not yet processed.

D. Application Example: Creating a 3-Statement Financial Model Draft

- ☐ Create a "Financial Model" folder on your desktop. Place an Excel file containing the company's historical financial data for the past three years (revenue, expenses, assets, liabilities, etc.). If you don't have it, you can practice with hypothetical data.

☐ Enter the following prompt.

"Based on the historical financial data in this folder, create a 3-Statement financial model for the next three years in Excel. Apply the following assumptions. Annual revenue growth rate 10 percent, gross profit margin 60 percent, SG&A expense ratio 30 percent, corporate tax rate 22 percent, depreciation straight-line method 5 years. Create Assumptions sheet, income statement sheet, balance sheet sheet, and cash flow statement sheet in Excel. Connect all projected figures using formulas that reference variables from the Assumptions sheet, not hard-coded numbers."

☐ Claude creates each sheet and enters formulas. When you change the growth rate in the Assumptions sheet, the revenue in the income statement changes, and that change automatically cascades to the balance sheet and cash flow statement.

☐ Here, there is something you must verify.

Verify that Claude has not directly entered numbers in any cells instead of formulas.

In Excel, pressing Ctrl+` (backtick) displays all cell formulas. If you see cells containing numbers instead of formulas, correct them with the proper formula. This is precisely the problem flagged in the Wall Street Prep test.

Because AI knows the calculation result, it sometimes directly enters the answer without using a formula. Later, if you change an assumption, only that cell fails to update, throwing the entire model out of alignment.

E. Practical Application: Scenario Analysis and Executive Reporting

If you are preparing to raise capital or approaching a board presentation, you must show multiple scenarios at once.

☐ In the folder containing the financial model you created earlier, enter the following prompt.

"Add scenario analysis to the financial model you just created. Set the upside scenario at 15 percent revenue growth, base case at 10 percent, and downside scenario at 5 percent. Create a chart comparing net income and cash balance after three years across the three scenarios. If there is a point where cash turns negative in the downside scenario, flag it with a warning. Then summarize these analysis results in a 2-page Word document for executive presentation."

☐ Claude creates a scenario selection dropdown in the Assumptions sheet, calculates results for each scenario, and generates a comparison chart. The Word summary document organizes key numbers and decision points.

Before this output, remember one thing. A financial model created by a machine is logically smooth, and the chart curves move cleanly each time a variable changes. However, the assumptions supporting that model are merely estimates provided by humans. Variables such as shifts in consumer behavior, unexpected regulations, and the emergence of competitors do not fit into Excel

formulas.

The machine has reduced modeling work that took days to just minutes, but whether the numbers printed on that spreadsheet survive in reality depends on the real-world intuition of the person standing outside the screen.

F. If You Encounter These Problems

(1) The column names in the two files to reconcile are completely different. Tell Claude directly which columns correspond: "In the first file, 'Deposit Date' is the same as 'Transaction Date' in the second file. 'Deposit Amount' corresponds to 'Debit Amount'." Claude will reconcile based on this mapping.

(2) The balance sheet in the financial model does not balance. Total assets not equaling total liabilities plus equity means there is a missing formula connection. Instruct Claude: "The total assets and total liabilities plus equity in the balance sheet don't match. Find the cause and correct the formulas." Claude will trace formula dependencies to identify the missing item.

(3) "Can I show the auditor a model Claude created?" AI-generated models should be used as drafts, but must be verified by a human cell by cell before submission. In audits, traceability is required not only for numerical accuracy but also for where those numbers came from. If there are cells with AI hard-coded values, the audit trail breaks, so it is essential to verify during the validation process that all cells are connected through formulas.

There is a commonality among the four tasks covered in this chapter.

Cowork handles repetitive tasks like typing, classification, formula input, and pattern detection. What cowork cannot handle is judgment: "Is this number correct?" "Should this expense be reduced?" "Is this assumption realistic?" The staff member freed from receipt reconciliation no longer inputs numbers but becomes a final validator who questions and approves the reliability of machine-generated ledgers. The manager who receives card expense analysis must decide independently whether to cut costs or invest more aggressively.

Even when machines draw a perfect map, it is ultimately our own feet that must traverse the rugged terrain. In the next chapter, we examine how to create marketing content and sales materials based on this financial data.

← Previous Chapter

See every chapter

Next Chapter →

Attorney · Former Member of the National Assembly · AI Policy Researcher

kimkj.com

© 2026 Kim Kyung-jin. All rights reserved.

Chapter 3: Marketing, Content, and Sales Support

← Previous Chapter

See every chapter

Next Chapter →

On the conference room whiteboard, the distribution schedule by channel is drawn like a spider web. Blog, newsletter, landing page, email, presentation materials. Each channel requires different grammar and rhythm, and staff members explain the same product in different tones. While a copywriter squeezes out copy, a web developer translates that copy into code, and the sales team creates yet another version of presentation materials. The process that required several people to spend days bringing one new product to market takes on a different landscape in front of Cowork.

The six tasks covered in this chapter form the backbone of marketing practice. Building integrated campaigns, converting long-form content by channel, content gap analysis and calendar generation, B2B lead prospecting and scoring, SEO audits and website analysis, and creating customer onboarding packages. Each appears to be independent work, but they are tied together by one common point: people plan, Cowork executes, and people validate.

1 Building Integrated Marketing Campaigns

The afternoon landscape of the marketing department one month before a new product launch typically looks like this.

When the product manager hands over the functional specification, someone writes a blog post, someone else structures an email sequence, and another person creates presentation materials. The results that emerge from each person's corner address the same product yet have different tones. The phrase "smart work assistant" written in the blog becomes "next-generation automation platform" on the landing page.

This is because even when a brand guidelines document exists, each practitioner interprets it differently.

Cowork unites this fragmentation into a single flow. The core is to have one foundational document that readers can reference, and from that document, have blog posts, newsletters, HTML landing pages, email sequences, a 30-day launch plan, and PowerPoint presentation materials branch out.

There may be multiple channels, but the campaign should be one, and however many files there are, the message should flow as a single thread.

a Why It's Needed

Creating campaign content separately for each channel creates two problems.

One is time. If a copywriter, designer, web developer, and email manager each do their work, it usually takes more than two weeks.

The other is consistency. When blogs are educational, landing pages are sales-oriented, and emails feel urgent, customers wonder if these are messages from the same company.

It's more serious for solo entrepreneurs or small teams of three to five people. Without the resources to place professional staff in each channel, one person must handle all channels. When that one person gets tired, the quality of the entire campaign collapses.

b What It Does

When you hand a product guidelines document to Cowork, it extracts the core value proposition, target customer personas, and tone and manner from that document.

Based on this information, multiple sub-agents (auxiliary agents that perform individual tasks under instructions from the main agent) operate simultaneously.

Agent 1 writes the blog, Agent 2 codes the landing page HTML, Agent 3 structures the email sequence, and Agent 4 assembles the PowerPoint slides. They work independently while referencing the same guidelines document, so message consistency is maintained.

[To Know] What Are Sub-agents?

When you give Cowork a complex task, the main agent creates a work plan and then runs multiple auxiliary agents simultaneously. The progress of each sub-agent is displayed in real time in the right sidebar of the Cowork screen. It's like a team lead assigning work to team members who work at the same time.

c How to Do It: Basic Usage

Start with the simplest form. Let's create a blog post and an email at the same time.

Follow Along

- ☐ Create a folder called "New_Product_Campaign" on your desktop.
- ☐ Place the product specification (PDF or Word) inside it. Any document containing the product name, key features, target customers, and price will do.
- ☐ Open the Claude Desktop app and click the "Cowork" tab at the top.
- ☐ Set folder permissions in the left sidebar. Select the "New_Product_Campaign" folder to allow Claude to read and write files in this folder.
- ☐ Paste the following prompt into the input field and press Enter.

"Read the product specification in this folder and create two pieces of content for me.

A blog post explaining the core value of the product (Word file, approximately 1,500 characters) An email announcement of the new product to send to existing customers (Word file, approximately 500 characters) Save both files in this folder."

- ☐ Claude displays the work plan on the screen. Review the plan and click "Approve" or "Allow".
- ☐ After 2-3 minutes, open the folder. The files Blog_Draft.docx and Email_Draft.docx have been created.

d How to Do It: Advanced Cases

Take a step further from basic usage. Add a brand guidelines document and increase the number of channels.

- ☐ Add the brand guidelines document to the "New_Product_Campaign" folder. It's a document containing the company's tone and manner, prohibited expressions, and frequently used phrases. If you don't have one, write the following content in a Word file and save it as "Brand_Guidelines.docx".

"Our brand's tone: professional yet approachable. Prohibited expressions: 'innovative,' 'groundbreaking,' 'first in the industry.' Preferred expressions: narratives centered on specific numbers and customer cases. Target customers: CEOs and executives of small to mid-sized companies with 50-200 employees."

- ☐ Also create and add a target customer persona document. One to two pages describing the customer's age, job title, concerns, and information consumption channels are sufficient.
- ☐ Enter the following prompt in Cowork:

"Read all the product specification, brand guidelines, and customer persona documents in this folder. Based on the content of the three documents, create the following four pieces of content simultaneously.

A blog post with a subheading structure to rank well in search engines (Word file) A newsletter draft to send to subscribers (Word file), written from a different angle so it doesn't overlap with the blog A seven-email sequence for sequential sending over seven days (all 7 emails in one Word file), with day 1 addressing problem awareness, day 3 hinting at solutions, day 5 introducing the product, and day 7 driving action An internal executive summary of the launch strategy (Word file, 2 pages or less) Create a subfolder called 'Deliverables' and save all four files there."

- ☐ Claude assigns sub-agents to work in parallel, with the progress displayed in the right sidebar.
- ☐ When the work is complete, four Word files are generated in the "Deliverables" subfolder.

One important point to note: With email sequences, Cowork's draft proposes message structure and sending intervals, but when you transfer them to an actual sending system (Mailchimp, HubSpot,

etc.), you must personally verify settings that affect sender reputation and delivery rate. Email is as important in the design of sending conditions as it is in message quality.

C. How to do it: Practical implementation

In practice, you request six outputs at once, including an HTML landing page and PowerPoint presentation.

1. In the folder, place the product specification document, brand guidelines, customer persona, along with the company logo image file and sample past marketing copy.

2. Enter the following prompt.

"I'm a marketing director. Please read all the documents in this folder. Based on this information, generate the following six integrated marketing packages at once.

Three blog posts covering the product's core value from different angles (each as a separate Word file), a draft new product announcement newsletter for existing customers (Word file), a single-page HTML landing page to collect pre-orders (HTML file) with hero section, feature showcase, customer testimonial area, and CTA button, a seven-day email drip sequence (Word file) with two subject line A/B test variations for each email, a launch strategy PowerPoint for internal executive reporting (PowerPoint file, around 10 slides), and a day-by-day 30-day launch marketing plan from D-30 to D-Day (Excel file) including channel distribution schedule, internal review dates, and ad budget spending timeline. Save all deliverables to a subfolder named 'Deliverables'."

3. Cowork launches six sub-agents simultaneously. PowerPoint skills, Excel authoring skills, and HTML coding capabilities are all deployed. The work typically takes 5 to 10 minutes.

4. Open and review each completed file one by one.

When reviewing, check whether the three blog posts address different angles, whether the newsletter and blog tones align with the brand guidelines, whether the landing page CTA language matches the final call-to-action in the email sequence, and whether the PowerPoint's logical structure aligns with the 30-day plan schedule. Once Cowork completes 90 percent, you take on the editor's role for the remaining 10 percent, adding the brand's unique warmth and strategic judgment.

D. Good prompts and bad prompts

Bad prompt: "Make some marketing materials for me."

This instruction reveals nothing about what product, who the audience is, which channels to use, or what file format to request. Cowork will produce something even without context, but the result will be nearly useless.

Good prompt: "Read the product specification and brand guidelines in this folder, then create one blog post (Word, 1,500 characters) targeting executives at mid-sized companies with 100 employees

and a seven-day email sequence (Word, Day 1 problem awareness / Day 3 case study / Day 5 product introduction / Day 7 call-to-action) and save them to the Deliverables folder."

A good prompt contains four elements: reference documents, target audience, output format and length, and storage location. The more clearly you specify these four elements, the higher the quality of the output.

E. If these issues arise

(1) The blog and newsletter tones are too similar. Cause: You did not specify channel-specific tone differences in the brand guidelines document. Add this distinction to the guidelines: "The blog takes an educational, in-depth tone; the newsletter uses a short, curiosity-driven tone." Then submit the request again.

(2) The HTML landing page appears broken in the browser. Cause: The HTML Cowork generates has basic structure and text focus. CSS styling may be insufficient. Either pass it to a web developer or give Cowork the additional instruction: "Apply mobile responsive design."

(3) The company logo did not appear on the PowerPoint slides. Cause: Even if you place the logo file in the folder, Cowork may not automatically insert it unless you explicitly state in the prompt: "Place the company logo in the bottom right corner of each slide." Supplement your instructions with this specificity.

2. Multi-Channel Repurposing of Long-Form Content

It took two days to shoot and edit a 40-minute YouTube lecture. The content is dense. Yet only 300 people watched the video.

Post the same content on LinkedIn and 5,000 people could see it; send it as a newsletter and reach 2,000 subscribers. The problem is that adapting a 40-minute script to fit each platform's conventions takes as much time as producing the video itself.

A. Why it's necessary

The strategy of creating multiple channel versions from a single original piece of content is commonly called "One-Source Multi-Use."

The strategy has existed for a long time, but execution has always been the bottleneck. To convert a YouTube script into a LinkedIn post, you must extract the core idea, rewrite the opening sentence for the platform, and add hashtags.

Converting the same material into a Twitter(X) thread requires breaking your thoughts into 280-character chunks. At one to two hours per channel, five channels eat up an entire day.

B. What it does

When you hand Cowork a long script or transcript, it extracts the message suited to each channel. LinkedIn posts place industry insights upfront because professionals read them there; Twitter threads develop arguments in short bursts; blog outlines use subheading structures matched to search intent; emails deliver a single core message.

Here lies the most critical task: infusing brand voice, your brand's distinctive tone and manner of speaking. To strip away that characteristic smooth, arid smell of AI-generated text, you must include samples from your own past writing.

Cowork learns your cadence, sentence length, and recurring expression patterns from those samples.

C. How to do it: Basic use

Start by converting a single YouTube script into one LinkedIn post.

Follow along: 1. Create a folder named "Content_Repurposing."

2. Place the YouTube video's script text file (or caption file) in the folder. To download captions from YouTube, click "Show transcript" in the menu at the bottom of the video, then copy the text and save it as a text file.

3. Connect this folder in Cowork and enter the following prompt.

"Read the YouTube script in this folder and create one LinkedIn post. Place the most counterintuitive claim or data from the script at the start to grab attention in the opening sentence. Write in an expert tone, but end with a question that prompts readers to share their experience and start a conversation. Save it as a Word file."

4. Claude analyzes the script and generates a LinkedIn post draft as a Word file.

D. How to do it: Advanced application

Expand to three channels and add a brand voice file.

1. Add a file called "Brand_Voice.txt" to the folder. Write it like this: "My speaking style: I alternate between short and long sentences. I use metaphors often but never overstate. I don't use emoji. Phrases to avoid: hedging endings like 'could be said to' or 'amounts to.' Preferred closings: end with a question or a single declarative sentence."

2. Also place two or three LinkedIn posts or emails you wrote yourself in the folder as text files. These become the learning material from which Cowork picks up your voice.

3. Enter the following prompt.

"First read the 'Brand_Voice.txt' file and my past writing samples in the folder, then familiarize yourself with my voice and phrases to avoid. Next, read the YouTube script and convert it into the

following three pieces of content.

LinkedIn post (expert tone, key insights focused, hook in first line) X thread 5-7 posts (each tweet reads independently and makes sense, with original video link space in final tweet) Email newsletter (for subscribers in a 'here's this week's highlights' tone, sharing 2-3 key points only) Three contents should each cover different insights as the main focus. Save all individually as Word files.

☐ Cowork first analyzes the brand voice file, then runs three sub-agents simultaneously to generate content for each channel.

D. How to do it: practical application

In practice, expand to 5 channels and save this workflow as a Skill (a reusable set of work rules saved for future use) so the next time you can run it with a single command.

☐ With script file and brand voice file ready in a folder, enter the following prompt.

"Read the YouTube script and repurpose it into the following five content types. LinkedIn carousel post (structure three key insights as slides) X thread (5-7 threads starting with a hook) Email newsletter (storytelling format, 2-3 key points) Three YouTube Shorts scripts (60 seconds each, extract interesting parts only, include timestamps) Blog post outline with SEO keywords (include H2 and H3 tag structure) Ensure each content covers different insights as the main focus without overlap. Save all individually as Word files."

☐ If satisfied with the results, save this workflow permanently. "Save the entire content conversion process you just performed as a Skill named 'YouTube-to-Social'."

☐ Next time you put a new script in the folder and enter /YouTube-to-Social, Cowork will automatically generate content for five channels following the saved rules.

E. It is worth making conversion rule tables first. When repurposing content across multiple channels, setting channel-specific conversion rules in advance stabilizes quality. Here is an example.

LinkedIn: break into single sentences, put real-world problems up front. 3-5 hashtags. X: short and concise. One claim per tweet. First tweet is a hook, last is a CTA. Blog: establish context before argument. Headline structure matching search intent. Email: request only one action. Put the core message in the subject line.

If you keep this rule table in a text file in the folder, Cowork references it each time.

F. If this problem occurs

(1) Channel outputs look like they're simply summarizing the original script. Cause: the prompt is missing the instruction to focus on different insights for each channel, or you haven't clearly distinguished each channel's role. Specify angles like: LinkedIn from an industry trend analysis perspective, X focusing on counterintuitive facts, email emphasizing personal experience narratives.

(2) Brand voice wasn't reflected. Cause: either the past writing samples are too few, or the brand voice file instructions are vague. The more specific you write, keep sentences under 40 characters, draw metaphors from everyday objects, instead of prefers short sentences, the more effective it is.

3. Content gap analysis and calendar generation

Assume you've been running a blog for two years and written 150 posts. Each time you plan new content, you wonder: have I covered this topic before? When you search, you find three posts with similar titles, yet there's not a single post addressing questions your readers ask most frequently.

A. Why it's needed

Two problems grow as content accumulates: duplication and gaps. When you repeat the same topic with only slight variations, readers feel déjà vu and traffic plateaus. Simultaneously, if posts addressing market-driven questions are missing, competitors claim that ground first. A person would need over a day to read all 150 past posts and identify patterns, but Cowork delivers structured analysis instantly upon receiving a list.

B. What it does

When you give Cowork past content files, it classifies everything by topic and analyzes which areas are overproduced and which are blank. For instance, in an AI-focused blog, generative AI tool reviews might have 12 posts while AI ethics and regulation has only 2. Once gaps are identified, it derives new content ideas that don't overlap with existing posts and completes them as a 3-month calendar placed on a timeline.

[Know this] What is content gap?

Content gap divides into four types.

Subject gap is a question you haven't covered at all. Format gap is when you cover the same topic in blog form only and haven't adapted it to short social formats. Audience gap is when you have only beginner posts and no decision-maker materials. Funnel gap (funnel refers to the stages customers pass through from awareness to purchase) is when you have many awareness-stage posts but comparison and purchase-stage posts are absent.

C. How to do it: basic usage

You can do basic analysis with just past blog title lists.

Try it. ☐ Create a folder called Content_Analysis.

☐ Organize past blog post titles and URLs in Excel or CSV format. Four columns suffice: title, publication date, category, URL.

☐ Put this file in the folder, connect it in Cowork, then enter the following prompt.

"Read the blog list file in this folder and classify the titles by topic. Identify which topics have the most posts and which have the fewest, then organize the top 5 of each and save as a Word file."

☐ Claude generates the classification result as a Word file. This alone shows at a glance what we've talked about most and what we haven't.

D. How to do it: application examples

When you use actual blog original files instead of title lists, the depth of analysis changes.

☐ Put 20-50 past blog articles' full texts in your folder as text files or PDFs. You can also add podcast episode scripts the same way.

☐ Enter the next prompt.

"Analyze all past content files in this folder. Extract the top 5 themes and keywords I've covered most frequently. Based on past data, derive 20 new content ideas I've never covered before (avoiding duplication). Each idea should include a concrete title and a 3-line summary.

Save results as a Word file."

☐ Cowork analyzes text patterns, finding repeatedly appearing keywords and identifying blank areas yet uncovered.

There is one point to note here. The blank space Cowork finds isn't necessarily a space you should fill.

Why a certain topic wasn't covered in the past may be because it didn't align with our brand identity or it was a sensitive area we strategically chose to avoid.

Finding statistical gaps is the machine's job, but determining whether filling that gap is strategically sound is yours.

E. How to do it: practical application

Arrange the 20 ideas into a 3-month calendar.

☐ With 20 idea files generated from application cases in your folder, enter the following prompt:

"Create a three-month content calendar based on the 20 content ideas derived above. The conditions are as follows: publication twice a week; alternate search-type, case-study-type, comparison-type, and conversion-type pieces to avoid topic concentration; alternate between data-research-heavy deep analysis pieces and lighter experience-based pieces to spread out creator energy expenditure; in each row, include publication date, content title, the one-line question this piece answers, and distribution channels (blog/LinkedIn/newsletter). Save it as an Excel file."

□ Claude generates an Excel file. Once you share this calendar with team members, it becomes a collaboration tool you can use right away.

There is a reason for including "the one-line question this piece answers" in the calendar. Even if the same topic is covered, it is not a duplicate if the question differs; even if the title changes, it is a duplicate if the reader's question is the same. With this one line, you can immediately see whether similar pieces are appearing back-to-back.

If You Encounter These Problems

(1) The 20 derived ideas are too abstract. Cause: You included only a list of past content titles, not the full texts. Claude cannot discern the depth and angle of the writing from titles alone, so include the original text files when possible.

(2) The three-month calendar distribution is unrealistic. Cause: Claude cannot know the exact production difficulty of your writing. Adding realistic conditions to the prompt such as "data-analysis pieces take 5 days to produce, experience pieces take 2 days" will improve the distribution.

4 B2B Lead Generation and Scoring

Your sales team has a CSV file with prospects from 1,000 companies. How many of them are likely to actually buy your product? If a salesperson reviews all 1,000 one by one, it takes two weeks.

Why It Matters

The core of B2B sales is deciding who to meet first. You have many names but little time. The difference between sending identical emails to 1,000 companies and sending customized emails to your top 50 shows clearly in reply rates. Lead scoring, assigning scores to prospects based on fit and interest, is the work of organizing these priorities into numbers.

What It Does

Give Claude a CSV file of prospects and your Ideal Customer Profile (ICP) criteria, and it scores each company against those criteria. It identifies decision-makers at high-scoring companies and drafts customized cold emails tailored to each company's situation.

How to Use It: Basic

First, learn the principles of scoring with a small list.

Follow along: □ Create a "B2B_Sales" folder.

□ Place a CSV file with information on 30 prospects inside it. The columns are: company name, industry, employee count, annual revenue, contact name, job title, email.

□ In the same folder, create a file called "ICP.txt" and write your ideal customer criteria.

"Our ideal customer: SaaS or IT services, 50-500 employees, 5-50 billion won annual revenue, bonus points for fundraising in the past two years."

□ In Claude, enter the following prompt:

"Analyze the 30 companies in this folder's CSV file against the ICP.txt criteria. Score each company from 1 to 100 points and explain your scoring by category. Save the results to an Excel file. Format the columns as: company name, score, scoring rationale, industry match, size match."

□ Claude generates an Excel file. When you open it, you see scores and rationales organized like this: "Company A: 82 points. Industry match (+25), employee count in range (+20), revenue meets criteria (+20), fundraising history unconfirmed (+0), other (+17)."

How to Use It: Advanced

Generate customized cold email drafts for high-scoring companies.

□ With the Excel file from the basic section in your folder, enter the following prompt:

"From the lead scoring results I just generated, filter for companies with 70 points or higher. For each company, do the following: check the contact's job title from the CSV and mark the decision-maker (CEO, CTO, VP level).

Draft a cold email for each company. In the opening sentence, mention something specific about their industry or size. In the second paragraph, explain what value our service can bring. End by proposing a 15-minute call. Generate a master Excel file containing the lead score, decision-maker info, and draft email for each company."

□ Claude fills in a separate column in the Excel file with cold email drafts, each with a different opening sentence tailored to the company.

Adjusting Weights and Validation

Your sales team may disagree with the scores. If you decide that annual revenue matters more than employee count, add weights to your ICP.txt.

"Apply a 2x weight to the revenue item and 1x weight to the employee count item." Then run it again.

It is best when sales and marketing agree on the scoring criteria. Numbers look objective, but the standards that produce them are human choices. Without agreement on standards, scores will not survive long in sales meetings.

If You Encounter These Problems

(1) Scores come out similar for all companies. Cause: Your ICP criteria are too loose or have too few items. Just "industry match" and "size match" will not differentiate enough. Add specific criteria like "recent job postings include IT roles" or "company news in the past year mentions digital

transformation."

(2) The cold emails are too generic. Cause: Your CSV lacks company-specific details. Beyond company name and industry, add columns like "recent news," "core services," and "competitors." This raises the personalization level of the emails.

(3) Claude cannot search for company information on the web. Cause: The domain is not allowed in Claude's network access settings, or Claude in Chrome browser integration is not activated. Check the network allowlist in settings.

5 SEO Audit and Website Analysis

You have been running a website and asking yourself, "Why does it not appear in search?" You post content steadily and the design is clean, yet your pages are buried after page 3 of Google results. The cause usually hides in technical details invisible to the eye.

Why It Matters

An SEO audit, systematically checking your search engine optimization status, is not about manipulating keywords. It is about clearing the path for search engines and users. Empty meta tags, missing alt text on images (hidden text describing an image), slow page load speed, and broken layouts on mobile screens all drag down search rankings. It is hard for a human eye to check all these items at once, and professional SEO consulting is expensive.

What It Does

Give Claude your website URL, and it retrieves and analyzes the HTML structure of that page.

The audit items fall into four main areas: on-page SEO (title tags, meta descriptions, H1-H6 hierarchy, keyword placement), technical SEO (loading speed, render-blocking elements, cache settings), content quality (body length, keyword density, duplication), and user experience (navigation intuitiveness, CTA placement, mobile responsiveness).

When Claude in Chrome is enabled, Cowork can open an actual browser, visit the page, and evaluate the visual layout as well.

How to Use It: Basic

Let's audit one main page of your company website.

Follow Along

- ☐ Create a folder called "SEO_Audit".
- ☐ Write one URL of the webpage to be audited in a text file and put it in the folder.
- ☐ Enter the following prompt in Cowork.

"Analyze the webpage at the URL in this folder. Check the following items and save the results in a Word file.

Title tag length and keyword inclusion; existence and quality of meta descriptions; whether H1 tag appears only once and hierarchy is correct; whether images have alternative text (alt text); number of internal and external links. For each item, mark the status as 'Good', 'Needs improvement', or 'Needs immediate fix'."

- ☐ Claude analyzes the HTML of the URL and generates the audit results in a Word file.

How to Use It: Applied Example

Audit multiple pages at once and add UX analysis.

- ☐ Write the URLs of your top five traffic pages in a CSV file and put it in the folder.
- ☐ Verify that the Claude in Chrome connector is enabled.
- ☐ Enter the following prompt.

"Using the browser, visit the five webpages in this CSV directly. SEO audit: meta tags, heading structure, keyword placement, broken links, outdated information check. UX analysis: whether the value proposition is visible on the first screen, CTA button placement and visibility, mobile responsiveness. Classify each page into three levels: 'Needs immediate fix', 'Improvement recommended', 'Long-term task'. Generate a Word audit report organized by page."

- ☐ Cowork opens a browser, visits each page (the tab border displays in orange), and creates a report by analyzing both text and visual layout together.

Suggested Report Structure

A good audit report divides into four bundles: indexing and crawling (robots, noindex, sitemap, internal links), on-page structure (title, H1, image ALT, duplicate meta), page experience (HTTPS, speed, mobile, layout stability), and conversion UX (CTA placement, form count, trust elements). This division lets the development team, content team, and design team each take their own tasks immediately. A report that says "fix everything" typically accomplishes nothing. A report that says "fix only these three things this week" actually changes the site.

If You Run Into These Issues

(1) Claude cannot access the URL. Cause: The domain is not in the allowlist in Cowork's network settings. In Claude Desktop app Settings, add the domain of the site you want to analyze to the Network Allowlist.

(2) UX analysis comes out as text only. Cause: Claude in Chrome is not enabled, so visual analysis could not be performed. In connector settings, verify that the Claude in Chrome extension is installed and connected.

6 Creating a Customer Onboarding Package

The 48 hours right after signing a contract with a new client set the tone for the relationship. If there is no contact for a week after signing the contract, the client begins to wonder, "Does this company really care about our project?"

Why It Matters

Client onboarding typically requires five documents: proposal, service agreement, invoice, project timeline, and welcome email. Each looks separate, but they are actually related documents derived from a single client record. When client name, service scope, amount, or schedule changes, all five documents must update at the same time.

Doing this manually usually takes an hour or more. You find and open a template, change names and amounts, convert to PDF, and attach to email, over and over. On busy days, you end up with the invoice amount different from the contract amount.

What It Does

When you give Cowork the client information, it generates six documents simultaneously: proposal (PPT), service agreement (Word), invoice (Excel), project timeline (Excel), welcome email (text file), and internal onboarding checklist (Word). If you have existing template files in the folder, Cowork reads the format and fills in the blanks.

How to Use It: Basic

Let's start simply by creating two things: a welcome email and a project timeline.

Follow Along ☐ Create a folder called "2026_NewClients" and connect it to Cowork.

☐ Enter the following prompt.

"New client info. Company name: GlobalTech, Service: AI chatbot development, Total amount: 20 million won, Project duration: 4 months.

Create two documents based on this information.

A welcome email draft ready to send to the client (text file), including an outline of next steps, a request to schedule the first meeting, and contact information for the account manager;

A four-month weekly project timeline (Excel file) including kickoff, material submission, first review, revision, and delivery milestones. Save both files to this folder."

☐ Two files are created in the folder after two to three minutes.

How to Use It: Applied Example

Increase the number of documents and use existing templates.

- ❑ Put the invoice format (Excel) or contract template (Word) you normally use in the folder. A blank template with your company logo is fine.

- ❑ Enter the following prompt.

"Using the invoice template and contract template in this folder as reference, fill in the blanks customized for the GlobalTech client. Additionally, create a new proposal (Word) and internal onboarding checklist (Word). In the proposal, organize the agreed-upon service scope, each party's roles, and performance measurement criteria. In the onboarding checklist, organize the tasks to be completed in the first week by item."

- ❑ Cowork reads the structure of the existing templates, maintains the logo and formatting, and fills in only the customer information.

How to Do It: Practical Application

Generate six documents in parallel at once.

- ❑ Enter the following prompt.

"Create an onboarding package for the GlobalTech client.

Service: AI chatbot development, Total amount: 20 million won, Payment terms: 50 percent upfront at start / 50 percent upon completion, Project duration: 4 months. Generate six files simultaneously. Project proposal (PPT) with agreed service scope, each party's roles, and KPI specifications. Service contract (Word) with service level agreement (SLA), confidentiality clause, and termination conditions. 50 percent upfront invoice (Excel) with formula-based amount calculation. Monthly timeline by week (Excel) with key milestones and deliverables organized. Welcome email draft (text file) with project progress guidance, first meeting coordination, and contact information. Internal team onboarding checklist (Word) with tasks for the first two weeks. Save all files to the 'GlobalTech_onboarding' subfolder."

- ❑ You will see six sub-agents executing simultaneously in the right sidebar of Cowork.

- ❑ After completion, open the folder and six files will be generated. Always verify that the upfront invoice amount is calculated correctly as 10 million won, the contract duration matches the timeline duration, and the kickoff date in the welcome email aligns with week one of the timeline.

If This Problem Occurs

(1) The invoice amount does not have a formula. Cause: If you don't explicitly state "formula-based amount calculation" in the prompt, Cowork may enter numbers directly. Give a specific instruction: "Insert a formula in the total amount cell that automatically calculates 50 percent."

(2) Information is inconsistent across the six documents. Cause: Sub-agents working in parallel may have interpreted information differently. Clearly declare common variables in the first line of your prompt: "Use the following common information consistently across all documents: Company name GlobalTech, Amount 20 million won, Duration 4 months, Start date April 1, 2026."

The six tasks covered in this chapter each have a distinct purpose.

Integrated campaign building unifies fragmented channel messages into one, content repurposing expands the reach of a single original piece, gap analysis shows what was said and what was not said, lead scoring picks the 50 prospects to meet first from 1,000 targets, SEO audit uncovers invisible technical bottlenecks, and onboarding package reduces administrative time immediately after contract signing.

Cowork's role across these six tasks is consistent. It performs repetitive, time-intensive execution in your place. The role people must play is also consistent. Set the direction of the strategy, add brand-specific judgment to the output, and maintain legal and ethical boundaries. When Cowork completes 90 percent, people work in the remaining 10 percent. That 10 percent determines whether the output is useful.

[← Previous Chapter](#)

[See every chapter](#)

[Next Chapter →](#)

Attorney · Former Member of the National Assembly · AI Policy Researcher

kimkj.com

© 2026 Kim Kyung-jin. All rights reserved.

[#KimKyungjin](#) [#AIBLibrary](#) [#ClaudeCoworkandAgentManual](#)

Chapter 4: Research and Business Intelligence

← Previous Chapter

See every chapter

Next Chapter →

In spring 2023, a startup CEO in Seoul's Gangnam district was transferring prices from competitor websites into a spreadsheet until two in the morning.

With about thirty tabs open, he read App Store reviews, then moved to Blind's forum to scan job postings, then checked investment history on Crunchbase. He repeated this work every month.

In offices in other cities, similar scenes were unfolding at similar times. An investment analyst was preparing morning briefings by moving through dozens of news sites, a graduate student was filtering hundreds of papers on PubMed while drafting a literature review, and a marketing director was sifting through past campaign data to estimate the success probability of the next project.

These four tasks share a common structure.

Gathering vast amounts of information, filtering out noise, identifying patterns, and organizing the result into a form that humans can judge. They take a long time because information is scattered, and mistakes happen because human attention has limits.

Cowork handles collection and organization in these four research tasks.

It opens browsers to visit competitor websites, reads news feeds to create summaries, analyzes papers to create comparison tables, and builds predictive models from historical data. What remains for humans is to question the machine's organized results, interpret them, and make decisions.

Information gathering and intelligence are different. Information gathering is collecting "what happened," while intelligence is judging "what does this mean for us."

Cowork handles the former at overwhelming speed. The latter remains human territory. When reading a report organized by a machine, you need an eye to observe what is absent from the text as much as what is written.

1 Automated Competitive Analysis

A Why It Is Needed

Competitive analysis is not about capturing and pasting a few screenshots. It is closer to building a reconnaissance system.

You must find out whether competitors changed prices, launched new features, hired for certain roles, and where customers express complaints. This information is not in one place. It is scattered

across websites, blogs, social media, app stores, review platforms, and job postings.

Two problems arise when humans do this work. One is time. To analyze three competitors, you must check each company's pricing page, compare feature lists, read customer ratings on review sites like G2 or Capterra, and track hiring trends on LinkedIn.

Transferring this data to Excel to create a comparison table and organizing it into a report executives can read takes at least several days. By the time the report is finished and lands on the desk, the market has already moved one step further.

The other is consistency. When tired, humans skip a tab. They focus on familiar competitors and miss new entrants. An analysis written Friday afternoon differs in depth from one written Monday morning. Cowork does not skip. It investigates every item on the list with equal depth.

B What It Does

Cowork's competitive analysis combines two technologies. One is Claude in Chrome, a browser extension that allows AI to directly control Chrome. Cowork navigates to specified URLs, scrolls pages, reads price tables, and takes screenshots to analyze layouts. The other is parallel sub-agent processing. If there are three competitors, three sub-agents move simultaneously, and the main agent synthesizes the results.

There are two final outputs. An Excel matrix comparing prices, features, and target customers across competitors at a glance, and a Word-format intelligence report that includes strategic implications.

[Note] Legal Boundaries of Web Scraping

It is technically possible to collect public information from competitor websites using browser automation. However, "technical possibility" and "legal permission" are different matters. The Robots Exclusion Standard (robots.txt) has been used for over thirty years as a means for website owners to control automated access. You must first check the competitor's terms of service, robot policy, and access frequency limits. If an official API exists, prioritize it over HTML scraping. Exclude pages requiring login, paid content, and areas containing personal information from collection. Respecting the boundary between what is technically possible and what is legally permitted is the first principle of automation.

C How to Do It: Basic Use

Start by collecting public information from one competitor and creating a summary document.

Walkthrough

- ☐ Create a folder called "Competitor_Analysis" on your desktop.
- ☐ Open the Claude Desktop app and click the "Cowork" tab at the top.
- ☐ In the left sidebar, grant read/write permissions to the "Competitor_Analysis" folder.

☐ Check that Claude in Chrome is activated in Connector settings. The Chrome extension must be connected in Settings > Connectors.

☐ Enter the following prompt.

"Visit competitor A's website (www.competitor-a.com) and collect the following information. 1) Prices by plan and included features from the pricing page, 2) Core value proposition and target customer message from the product overview page, 3) Product update related content from blog posts within the last three months. Organize the collected information into a Word file and save it to this folder."

☐ Cowork creates a plan and displays it in the sidebar. Confirm the order "Visit pricing page → Visit product overview page → Explore blog" and approve execution.

☐ The browser opens automatically and Cowork traverses each page. When a tab border turns orange, it means Cowork is controlling that tab. ☐ When the work is complete, a Word document appears in the folder.

D How to Do It: Applied Examples

Expand to multiple competitors and add review data.

☐ Enter the following prompt.

"Analyze three competitors in the AI app builder market. Targets: Replit, Lovable, Bolt. Visit each company's website and collect the following.

1) Prices by plan and free trial conditions, 2) Core features and limitations, 3) Main value proposition and target customer segment, 4) Positive and negative keywords that repeat in G2 reviews over the last six months. Investigate all three simultaneously using sub-agents. Create the results as two files: an Excel comparison matrix and a Word report."

☐ In the Cowork sidebar, you see three sub-agents executing simultaneously. They are labeled as handling Replit, Lovable, and Bolt respectively.

☐ After about five to ten minutes, two files appear in the folder. When you open the Excel file, each competitor's prices, features, and review keywords are aligned in the same rows.

E How to Do It: Practical Application

Beyond the comparison matrix, create a full-fledged report that includes strategic implications and a positioning map.

☐ Enter the following prompt.

"Create a competitive intelligence report for executive briefing based on the Replit, Lovable, and Bolt data analyzed earlier. Please follow this structure.

Executive Summary with three-line key findings. Competitor Positioning Map with horizontal axis showing technical difficulty (non-developer friendly to developer-focused) and vertical axis showing price range (low-cost to high-cost). Please explain each competitor's position. Price Comparison Table comparing Free, Basic, Pro, and Enterprise tiers. SWOT Matrix showing strengths, weaknesses, opportunities, and threats for each competitor. Customer Review Sentiment Analysis with frequency of positive and negative keywords and recent trends. Implications for Our Company, including identification of positioning gaps we can target and proposed market entry strategies. Create this as a Word report and attach the original data sheet in Excel separately."

- Cowork reuses sub-agents or creates new ones to complete the task. When there are areas with insufficient data, Cowork independently determines that "additional research is needed" and explores news articles or technology blogs further. This decision-making process is displayed in the sidebar.
- When you open the completed report, the positioning map contains descriptions of each competitor's position, and the SWOT matrix contains analyses such as "Lovable has high accessibility for non-developers but is weak in scalability."

What Machines Cannot Read

The reports Cowork creates are results of analyzing text published on web pages. Information competitors intentionally hide, hidden discount conditions not revealed on pricing pages, and the context of organizational restructuring behind job postings fall outside the machine's analytical scope.

Machines cannot determine whether a competitor's sudden price cut is due to cash flow problems or is a strategic choice to monopolize the market. While review sites show rating trends, distinguishing whether those reviews are fabricated by competitors or represent genuine user voices is the work of humans.

The true value of the report begins after data collection is complete. Whether to follow the direction indicated by the numbers arrayed in the matrix, or to intuitively read the hidden sides of the market that those numbers cannot show, depends on the decision of the person holding the report.

When These Problems Arise

- (1) Unable to retrieve data from a specific competitor's website. The competitor may have applied bot protection, or it could be a dynamic page requiring JavaScript rendering. Instruct Cowork with an alternative approach: "If you cannot access that page, collect the same information from cached versions, news articles, or review sites instead."
- (2) Insufficient volume of review data. Startups or B2B products may have limited public reviews. Expand the search scope by instructing: "If there are fewer than five reviews on G2, search for mentions of that product on Reddit and Hacker News."
- (3) Item criteria in the Excel comparison table differ by competitor. When Company A lists "100 items per month" and Company B lists "unlimited," they cannot be placed in the same column. By

instructing the prompt: "For items that cannot be compared, write them exactly as they appear in the source, and add a yellow background to those cells," you make visually distinct the places where humans must later make direct judgments.

2 Latest Industry News and Trend Briefing

Why It Is Needed

When an investor or business executive opens their eyes in the morning and turns on their smartphone, they are immediately confronted with hundreds of news headlines.

New legislation from the European Union on AI regulation, earnings announcements from semiconductor firms, and geopolitical news shaking the global supply chain are all mixed together. In this massive cacophony, the task of selecting three pieces of news that truly affect my business today and summarizing their significance in three lines is a harsh ritual that must be performed every morning.

Some people use Google Alerts.

However, Google Alerts relies on keyword matching. When you set "AI regulation," it captures only articles where the text "AI regulation" appears. An article stating "The European Union announced new compliance deadlines for high-risk systems" may be missed because the keywords do not match exactly. Cowork's LLM-based filtering judges relevance at the semantic level. Even without the words "AI regulation," it detects content related to AI regulation.

What It Does

When you set up Cowork's Scheduled Task, Cowork wakes itself at the designated time, collects news, filters it, summarizes it, creates a file, and saves it to a folder. Every day, a briefing of the same quality arrives at the same time. It does not stop even when the person is busy or unwell.

The output takes two forms. One is a text briefing, where each news item is summarized in three sentences or fewer and followed by a one-sentence implication for our business. The other is an image in the form of card news. Using a Python script, key figures and one-line summaries are rendered as visual cards.

[Note] Execution Conditions for Scheduled Tasks. Cowork's scheduled tasks run locally on the user's computer, not in the cloud.

It operates only when the computer is on and the Claude Desktop app is open. If the computer is sleeping or the app is closed, it automatically runs any missed tasks when turned back on.

If a task must run at a precise time, you should adjust the computer's power-saving settings and keep the app running at all times.

How to Do It: Basic Use

Auto-generate a text briefing that summarizes five AI industry news items every morning.

Follow these steps: ☐ Create a folder on the desktop called "Daily_Briefing." ☐ Switch to the Cowork tab in Claude Desktop and grant this folder read/write permissions. ☐ Enter /schedule in the chat input box. The scheduled task creation screen opens. ☐ Enter the following prompt.

"Execute the following task every morning at 8 a.m. Through web search, select the five most important news items related to the AI industry that occurred in the past 24 hours. Summarize each news item in three sentences or fewer and include the source URL. Save the file to this folder with the filename format 'YYYY-MM-DD_AI_MorningBriefing.txt'."

☐ Cowork organizes and displays the task name, execution frequency, and task details. Click the "Schedule" button to confirm.

☐ You can view registered scheduled tasks in the "Scheduled" tab on the left sidebar.

☐ The next morning at 8 a.m., a text file appears in the folder.

How to Do It: Applied Examples

Specify news sources concretely and add relevance tags and implications. ☐ Edit an existing scheduled task or create a new one. Enter /schedule.

☐ Enter the following prompt.

"Execute the following task every morning at 7:30 a.m. Prioritize news sources from TechCrunch, The Verge, The Electimes, and the Bloomberg Tech section. I am a business development manager at a B2B SaaS company. Filter news from this perspective. Mark each news item with a relevance rating to our business as high, medium, or low, and attach a one-sentence implication to news items marked 'high relevance.' Save as a Word file."

☐ Cowork applies semantic-based filtering to screen news based on content relevance rather than keyword matching.

How to Do It: Practical Implementation

Add card news images to the text briefing and automatically send to a Slack channel.

☐ Go to Settings > Connectors and confirm that the Slack connector is connected. Specify the Slack workspace and the channel to send to (for example, #morning-briefing).

☐ Create a scheduled task with the following prompt.

"Automatically execute the following task every morning at 7 a.m. Through web search, select the five most important news items from the past 24 hours in AI, semiconductors, and cloud infrastructure. Filter out duplicate articles and prioritize fact-based articles such as earnings announcements or official product launches over rumors. I am a venture capital analyst investing in AI technology

companies. Analyze the investment implications of each news item from this perspective. Save the analysis results to a Word file named 'YYYY-MM-DD_MorningBriefing.docx' in this folder. Using a Python script, create each of the top three news items from today's briefing as a card news image (PNG). Include the news title, key figures, and a one-line implication in each card.

Send the completed three card news images to the #morning-briefing channel on Slack."

- Cowork registers a scheduled task, and on first run, uses Python's Pillow library to generate images. Card news automatically uploads to the designated channel through the Slack connector.
- On a weekly basis, you can add a scheduled task that aggregates keywords and topics appearing repeatedly in daily briefings and generates a separate 'This Week's Trend Report'.

Ba: What machines overlook

When machines filter and summarize news, algorithmic judgment enters the process. Within articles the machine discards as unimportant may lie subtle signals that could change the world.

What people easily overlook while reading news daily is slow change. Yesterday's and today's news may seem similar, but compared to a month ago, the landscape may have shifted. An agent can capture this change numerically on a daily basis, but determining whether this change represents a truly significant transition falls to humans.

The insights the machine provides are inferences based on past data, not prophecies affirming the future. As you flip through beautifully designed card news, maintain the habit of verifying source material whenever needed, rather than blindly trusting a world edited through the machine's perspective.

Sa: If these problems arise

(1) The task did not run at the scheduled time. The computer may have been asleep or the Claude Desktop app may have been closed. When you reopen the app, pending tasks run automatically. If daily scheduled execution is needed, set automatic wake-up for that time in your operating system's power settings.

(2) News summaries are superficial. If you do not specify a persona clearly in the prompt, Cowork summarizes from a general perspective. When you narrow your area of interest, as in 'I am an analyst with focused investment in the semiconductor sector. Focus on the earnings and supply chain changes of Nvidia, TSMC, and Samsung Electronics,' the depth of analysis improves.

(3) Korean text in card news images is corrupted. This occurs when Python's image generation library cannot find a Korean font. Specify in the prompt 'Use the Korean font Malgun Gothic installed on the system' or place a font file (.ttf) in advance in the folder.

3 Professional Literature and Academic Research

Ga: Why it's needed

Suppose a researcher investigates 'the effects of vitamin D supplementation on immune function.' Entering keywords into PubMed yields thousands of papers. It takes weeks to months to complete a literature review: verifying whether each paper is a meta-analysis or clinical trial, judging whether sample sizes are adequate, filtering for research from the past five years, and comparing papers with conflicting conclusions.

PubMed alone holds over 39 million biomedical publications. At this scale, searching with a single keyword misses a significant portion of relevant papers. Results from searching 'vitamin D supplementation' and 'cholecalciferol AND immune response' may overlap by less than half. Search scope broadens only when combining synonyms, subordinate concepts, and MeSH terms (Medical Subject Headings, the medical subject classification system managed by the U.S. National Library of Medicine). Designing such search strategies itself is a specialized skill.

Na: What it does

Cowork compresses much of this process. It decomposes the broad topic you provide into multiple sub-queries and submits them simultaneously to academic databases to collect papers. It classifies collected papers by research type and extracts research design, sample size, measurement metrics, and key findings from each, creating a structured comparison table. It distinguishes areas where conclusions align from those where they conflict and identifies research gaps requiring further study.

The final output is a scholarly review document with introduction, methodology, comparative analysis, and conclusion. Every claim ends with a citation to the referenced paper.

Cowork's Opus 4.6 model has a context window of 1 million tokens (approximately 750,000 words), making it technically possible to load dozens of full papers simultaneously and cross-analyze them.

[To know] What is the Evidence Pyramid?

In medical research, not all papers carry equal weight. Systematic reviews and meta-analyses occupy the top, randomized controlled trials (RCTs) come next, and observational studies and case reports occupy lower positions. This hierarchy is called the Evidence Pyramid. When instructing Cowork to conduct a literature review, guiding it to assign weights based on this pyramid raises the scholarly rigor of the output.

Da: How to do it: basic use

Start by having Cowork read 10 papers you have collected in a folder and create a comparison summary table.

Try it yourself

□ Create a folder named 'Literature_Review' on your desktop.

- ☐ Place 10 papers you downloaded earlier in this folder.
- ☐ Grant read/write permissions for this folder in the Cowork tab.
- ☐ Enter the following prompt.

"Read all 10 PDF papers in this folder and extract the following information from each to create an Excel comparison table. Column structure: Authors, Publication year, Study type (RCT/observational study/meta-analysis, etc.), Number of subjects, Main intervention, Measurement metrics, Key findings, Study limitations, Citation information. When extraction is complete, save it as 'Paper_Comparison_Table.xlsx'."

- ☐ Cowork reads PDFs sequentially and extracts data. For large volumes of papers, it divides sub-agents for parallel processing.
- ☐ An Excel file is generated in the folder. Each paper is organized into one row.

Ra: How to do it: practical examples

Instead of providing PDFs directly, you give Cowork a topic and have it find papers through web search.

- ☐ Enter the following prompt.

"Conduct a literature review on the topic of vitamin D supplementation's effects on respiratory infection prevention. Search PubMed and Google Scholar and prioritize collecting meta-analyses and randomized controlled trials published within the past five years. Secure at least 15 papers, extract research design, sample size, dosage, and key findings from each paper's abstract to create a comparison table. If there are papers with conflicting conclusions, analyze the methodological differences explaining why the results differ."

- ☐ Cowork automatically expands the search terms.

It submits variant queries simultaneously, not just 'vitamin D supplementation AND respiratory infection' but also 'cholecalciferol AND upper respiratory tract infection' and '25-hydroxyvitamin D AND pneumonia prevention'.

- ☐ It classifies collected papers according to the evidence pyramid and assigns high weight to meta-analyses.

Ma: How to do it: real-world application

Generate a complete literature review report and identify research gaps.

- ☐ Based on the data secured in the practical examples above, add the following prompt.

"Write a formal literature review report based on the collected paper data. Follow this structure. Introduction: research background and review objective Search strategy: databases used, search

terms, inclusion and exclusion criteria Evidence summary: comparison of effects between vitamin D dosage of 1000 IU daily or less versus 2000 IU or more, differences across age groups Conflict analysis: logically explain methodological differences in papers with conflicting conclusions Research gaps: identify populations (children, Asian populations, etc.) and conditions not addressed in existing research References: organize bibliographic information of all cited papers in scholarly format Generate as a Word file and attach a separate Excel comparison table."

□ Cowork separates sub-agents, with one analyzing low-dosage papers and another analyzing high-dosage papers, then the main agent synthesizes the results.

□ The completed report contains critical analysis such as 'The conclusion that daily vitamin D supplementation of 1000 IU reduced respiratory infection incidence was consistently reported in 5 RCTs, but 2 showed no significant difference. This inconsistency can be explained by differences in baseline vitamin D levels among subjects.'

F. This is not a story limited to the medical field

The same framework applies equally to law, engineering, and social sciences. When comparing international legislative examples of the AI Basic Law, if Cowork systematically collects the EU AI Act, the U.S. executive order, China's Generative AI Management Methods, and Japan's AI Guidelines and creates comparison tables, researchers can focus on analysis and interpretation. The same structure works in patent disputes when conducting prior art searches and in both patent documentation searches and technical literature reviews.

G. What machines cannot identify

Literature reviews created by Cowork are drafts. Machines are accurate in analyzing the figures and methodologies written in papers, but they cannot fully account for the complex variables of the real world in which that data was produced. It is difficult for machines to perfectly filter out research designs that are subtly skewed due to pharmaceutical company sponsorship. When an overwhelming majority supports conventional wisdom, there is a risk that machines will exclude minority hypotheses that overturn that wisdom as errors.

Evaluating the methodological validity of papers, interpreting the causes of conflicting results, and reaching final conclusions are tasks for experts. Cowork is a tool that reduces the time experts need to reach judgments, not a replacement for judgment itself. When work that used to take weeks is reduced to days, the real promise of this technology is that researchers can think more deeply in that remaining time.

H. If you encounter these problems

(1) Cowork cannot read the paper PDF. If the PDF is a scanned image (saved as a picture rather than text), OCR processing is required. Instruct it as follows: "This PDF is a scanned document. Apply OCR first and then extract the text." Low-resolution scans may have lower recognition rates.

(2) Citation format differs from requirements. Citation formats vary by academic journal. Specify the desired format in your prompt, such as APA, Vancouver, or Harvard. Instruct it specifically: "Format the references in Vancouver style with numbering."

(3) Hallucination is suspected. Cowork may cite papers that do not exist or incorrectly summarize the conclusions of actual papers. You must go through a verification process of directly checking the DOI numbers of 5 to 10 key papers listed in the results on PubMed. If you instruct in your prompt, "Include a DOI number for each citation without fail. Do not use sources that cannot be verified," the risk of hallucination decreases.

4 Building a Decision-Making Bot

A. Why It Is Needed

The marketing team leader has proposed an influencer campaign.

The proposal is to pay 30 million won to a beauty influencer with 500,000 followers for a product review. The CEO asks, "What is the expected return on investment for this campaign? What were the results when we ran similar campaigns in the past?"

The marketing team leader can answer from intuition, but it is difficult to answer with data. Past campaign results are scattered across multiple spreadsheets, and the success and failure factors have never been organized systematically.

The instincts of experienced practitioners are generally correct. However, if that instinct cannot be explained in numbers, it is difficult to have persuasive power within the organization. Converting "it feels right" into "success probability of 72% based on past similar cases," that is the role of the decision-making bot.

B. What It Does

A decision-making bot is a system that learns from past data and, when you input new project conditions, calculates the expected ROI (Return on Investment) and probability of success. Cowork implements this bot in two ways.

One is pattern analysis. Cowork reads a CSV file containing the costs, influencer profiles, and result metrics of past campaigns, and establishes statistical standards for where successes and failures diverged based on which conditions.

The other is multi-perspective scenario analysis. Rather than a single AI providing answers, within Cowork, a market analysis agent, a financial modeling agent, and a Devil's Advocate agent (whose role is to deliberately offer opposing views to find holes in decisions) run simultaneously to validate the proposal from different perspectives.

C. How to Do It: Basic Use

Begin by analyzing past campaign data to judge the probability of success for a new campaign.

Follow Along

Prepare past influencer campaign data as a CSV file. The column structure is as follows: influencer name, follower count, average views, engagement rate, content type (video/image/text), product category, payment amount, campaign duration, conversions, sales increase, ROAS (Return on Ad Spend). An analysis with 20 or more records will have higher reliability.

- ☐ Create a folder called "Decision_Making_Bot" on your desktop and place the CSV file in it.
- ☐ Grant Cowork permission to this folder and enter the following prompt.

"The CSV file in this folder is data from our company's past influencer marketing campaigns. Analyze this data and answer the following questions. 1) What was the key variable that distinguished successful campaigns (ROAS 3 times or more) from failed campaigns? 2) Was there a correlation between follower count and ROAS? 3) Were there performance differences by content type? Create the analysis results as a Word summary report and an Excel detailed analysis table."

- ☐ Cowork reads the data and performs statistical analysis. It uses Python's pandas and scikit-learn libraries to extract correlations and patterns between variables.
- ☐ The report includes analysis such as: "Engagement rate showed a stronger correlation with ROAS than follower count itself. Micro-influencers in the 100,000 to 300,000 follower range showed the highest average ROAS of 4.2 times."

D. How to Do It: Application Examples

Based on the analysis results, create a predictive model and input new campaign conditions to forecast results.

- ☐ Enter the following prompt.

"Based on the data analyzed earlier, create a predictive model that calculates expected ROAS and success probability when new campaign conditions are input. Input variables: follower count, engagement rate, content type, product category, budget. Output: expected ROAS range, success probability (probability of ROAS 3 times or higher), risk factors. Once the model is complete, test it with the following conditions: 500,000 followers, 3.2% engagement rate, video review, beauty category, 30 million won budget."

- ☐ Cowork constructs a regression analysis or decision tree model and presents test results. The format is something like: "Expected ROAS 2.8 to 4.1 times, success probability 67%, risk factor: the influencer's engagement rate has been declining over the past three months."
- ☐ You can compare multiple scenarios by changing conditions. What if the budget is reduced to 20 million won and distributed among three micro-influencers? What if the content type changes from

video to live streaming? Compare expected results side by side for each scenario.

E. How to Do It: Real-World Application

Form a team of agents and create an AI advisory committee that validates business decisions from multiple angles.

☐ Enter the following prompt.

"We're planning to launch a new AI Leadership Bootcamp for 7,500 dollars. To validate this decision, activate a three-agent team.

Agent 1, acting as a market analyst. Research from the web the price range, target customers, and differentiation points of competing bootcamps. Agent 2, acting as a financial modeler. Calculate the break-even point and 12-month revenue forecast assuming a price of 7,500 dollars, a target of 50 students, monthly marketing costs of 5 million won, and instructor costs of 20 million won. Agent 3, acting as a Devil's Advocate. Find at least 5 reasons why this project might fail. Dig deeply into realistic risks such as insufficient differentiation from existing free content and reduced demand for high-cost education during economic downturns. Consolidate the results from all three agents and create a one-page decision summary in Word recommending either Go, Conditional Go, or No-go."

☐ The three sub-agents work simultaneously. The sidebar displays the progress and intermediate results of each agent.

☐ The completed summary includes recommendations like: "Initially, the 7,500 dollar price is likely to act as a barrier to entry, so we recommend a conditional go approach, starting with a 2,000 dollar short-term course and upselling the full course to graduates." Risk items identified by the Devil's Advocate agent are organized in a separate section.

F. Not Limited to Influencer Marketing

This framework is universal.

When deciding whether to enter a new market, you can apply the same structure: analyze the success and failure factors from past market entries and feed the conditions of the new market into the model. In recruitment, you can build a model to predict turnover rates or performance for a specific position based on historical hiring data.

The same applies to litigation strategy. If you accumulate data on win rates from similar cases, judicial panel tendencies, and citation rates by issue, you create a risk analysis tool that estimates the probability of winning a new case.

What Machines Cannot Guarantee

The machine's calculated success probability and projected returns rest on the premise that past data will repeat in the same pattern in the future. Events like disease outbreaks, the arrival of new

competitors, and shifts in public taste are unknown variables not entered into the data.

If past data contains fewer than twenty instances, statistical reliability declines. When market conditions shift rapidly, past patterns cannot predict the future. The numbers Cowork presents are not fixed prophecies but conditional estimates. When you combine the bot's output with your own experience, grounded in that understanding, you arrive at better judgment.

At the moment you sign the final approval document before a seemingly perfect analysis dashboard, the risk and responsibility that follow from that choice remain with the person making the decision, not the machine.

If You Run Into These Problems

(1) Column names in CSV data mix Korean and English. Cowork may map columns incorrectly. If you specify the meaning of column names in the prompt, for example, 'the "followers" column is follower count, the "amount paid" column is payment amount', analysis accuracy improves.

(2) The model produces overly optimistic predictions. If historical data contains more success cases than failure cases (survival bias, when only records of those who survived remain), the model overestimates the probability of success. Instruct it: 'Considering that failure cases may be underrepresented in our data, apply a conservative adjustment to the success probability.'

(3) The devil's advocate agent is too harsh. Because its role is to intentionally find weaknesses, the result may skew negative. In the summary, treat the devil's advocate opinion as a reference item, but in your final judgment, consider the analyses of all three agents in balance.

The four tasks covered in this chapter are different facets of research. Competitive analysis maps the current landscape of the market, news briefing detects changes in that landscape daily, academic research organizes deep expert knowledge, and decision-making bots calculate future probabilities from accumulated data.

Across these four tasks, Cowork's role is the same: gather scattered information, filter out noise, and organize it into a form that allows human judgment. The role humans must take across these four tasks is also the same: doubt the results the machine has organized, examine what is absent from the text as closely as what is written in it, and make your decision on that foundation.

The startup founder in Seoul no longer needs to keep a tab open until two in the morning. When she arrives at the office in the morning, a report sits waiting in her folder. What has changed is what she does with the time she spends reading that report. Instead of copying data by hand, she thinks about what the data does not reveal. In that time of thinking lies the true value of competitive analysis.

[← Previous Chapter](#)

[See every chapter](#)

[Next Chapter →](#)

Attorney · Former Member of the National Assembly · AI Policy Researcher

kimkj.com

© 2026 Kim Kyung-jin. All rights reserved.

#KimKyungjin #AIBLibrary #ClaudeCoworkandAgentManual

Chapter 5: Daily Operations and Communication Management

← Previous Chapter

See every chapter

Next Chapter →

After a meeting ends, memories diverge about who was supposed to do what. Each morning, opening your inbox reveals an important message buried among advertisements. Five minutes before a meeting, you confirm the other person's name. The night before a business trip, you pack your suitcase and try to remember where your passport is. These tasks are not the work itself. They are the work needed to do the work, what we call meta-tasks.

The four tasks covered in this chapter represent the area where Cowork shows the most immediate impact.

They are: analyzing meeting transcripts, organizing email and drafting replies, preparing meetings by syncing with calendar, and planning business trips.

They share a common trait: all are repetitive, structured, and require speed more than precision. When done by a person, they take thirty minutes; delegated to Cowork, three minutes. That difference reshapes your day.

Before we begin this chapter, one point deserves attention. As of March 2026, Gmail, Google Calendar, and Google Drive connectors (the connections that link Cowork to external services) are available in Claude's desktop app chat tab, and can also be connected from the Cowork tab.

However, since Cowork is still in research preview (a trial phase before official release), the scope of connector support and how they work may change with updates. The exercises in this chapter are based on official documentation as of March 2026.

1 Recording Meetings and Analyzing Transcripts

People in the same meeting hear the same words, yet their memories of who was supposed to do what diverge. In a thirty-minute meeting, only two or three things actually get decided, while the remaining twenty-five minutes of conversation blur that decision. In the past, the junior staffer would type throughout the meeting, then afterward piece together notes from memory. Context fell away. Critical commitments were missed. Responsibility became unclear.

Tools like Fireflies, Tana, and Clova Note have solved half this problem: converting speech to text. But a one-hour meeting's transcript fills fifteen to twenty pages of A4 paper.

Reading through it all and extracting the core content becomes another task in itself. Cowork handles precisely this second half: taking raw text and building structure from it.

A. Basic Use: Summarizing a Transcript and Extracting Action Items

Start with the simplest form: hand a meeting transcript text file to Cowork and receive a summary and follow-up list.

[How to do it]

- ☐ Open Claude's desktop app and click the "Cowork" tab in the left sidebar.
- ☐ Designate a working folder. Create a folder named "Meeting Minutes" on your desktop beforehand, place the transcript text file (.txt) in it, then select this folder from the top of the Cowork screen.
- ☐ Enter the following prompt.

"Please read the meeting transcript file in this folder and organize these three things:

First, summarize the entire meeting in ten lines or fewer. Avoid vague phrasing like 'it was discussed'; instead write decision-focused language like 'Option A was adopted and Option B was deferred.' Second, organize the decisions made in the meeting as a numbered list. Third, create a table for follow-up actions (action items). Include columns for owner, task, deadline, and priority. Save the results as a Word document in this folder. Name it 'Meeting Summary_Date.'"

- ☐ Claude displays a plan. Steps appear on screen: "I will read the transcript, create a core summary, separate the decisions, build the action item table, generate the Word document." Review the plan and approve execution.
- ☐ Claude executes the work. Progress displays stage by stage. When complete, a Word document appears in the "Meeting Minutes" folder.

The key here is separating summary from action items. What was agreed in the meeting and who must do what by when to carry out that agreement are separate pieces of information. Claude distinguishes between the two.

[To know] What format should the transcript file be?

Cowork can read .txt, .md, .docx, and .pdf files. When exporting transcripts from Fireflies or Clova Note, save as a text file (.txt) or Word file (.docx). If you include the meeting date and subject in the filename, Claude can automatically match the document title, which is convenient.

B. Applied Example: Changing Output Format and Creating an Excel Tracking Table

If you received one Word document in the basic version, this time you divide outputs by use case: creating both a summary document for team sharing and an Excel tracking table for project management simultaneously.

A Word document is formatted for readers. Executives, managers, and clients look at the summary structure first. Excel, by contrast, excels at tracking. In organizations with many action items, you

need a table separate from the notes, one you can filter by owner and sort by deadline.

[How to do it]

❑ Reuse the "Meeting Minutes" folder you created before. Modify the prompt as follows.

"Analyze the transcript in this folder and create two files. First, a Word document for executive reporting. Structure it as: Meeting Overview, Attendees, Key Decisions, Risk Factors, Next Meeting Agenda. Keep it to one A4 page maximum. Second, an Excel file for tracking follow-ups. Include these columns: Meeting Date, Topic, Owner, Task, Deadline, Priority, Status, Notes. Highlight items marked 'Urgent' in the Priority column with red cell backgrounds. Save both files in this folder."

❑ Claude displays a plan. This time, the plan shows generating two files in parallel. For complex work, Claude may spawn sub-agents (auxiliary AIs handling subsidiary tasks) to work simultaneously.

❑ After completion, check the folder. Both the Word document and Excel file are saved together.

An Excel tracking table amplifies the impact of recurring weekly meetings. If you accumulate action items in the same format each week, after a month you can see which topics have cycled for weeks without resolution, which owner's items keep rolling forward. The notes become more than a record; they become a tool that measures organizational execution.

C. Real-World Application: Saving Recurring Meetings as Skills and Automating Them

If you have regular meetings, like a weekly Monday check-in or a first-week-of-month executive meeting, you need not enter the same prompt each time. Save a working workflow as a skill (reusable work instructions stored in Cowork).

[How to do it] ❑ If the transcript analysis work you just ran satisfied you, enter this in the same Cowork session: "Please save the meeting transcript analysis workflow I just ran as a skill. Name the skill 'Weekly Meeting Post-Processing.'"

❑ Claude organizes the work instructions into a Markdown file and saves it in the skills folder.

❑ When next week's transcript is ready, place it in the folder and enter just one line in Cowork: "Please apply the 'Weekly Meeting Post-Processing' skill to this week's transcript."

Go one step further and create it as a scheduled task (a recurring job that runs at a set time). Type /schedule in the Cowork input field and a scheduling screen appears. Set it to "Every Monday at 9 a.m., check the 'Meeting Minutes' folder. If there is a new transcript file, run the 'Weekly Meeting Post-Processing' skill." When you arrive Monday mornings, the organized meeting document is already in the folder.

However, for a scheduled task to run, your computer must be on and Claude's desktop app must be open. It does not work when your laptop is closed.

Organizations with frequent customer meetings can use this to accumulate meeting history by customer. Gather meeting transcripts in a subfolder labeled "Customer A." Before the next meeting, request "Extract only open items from all meeting notes in this folder" to see past discussion points and remaining tasks at a glance.

One point must be stated plainly. Claude excels at converting spoken words to text and identifying surface-level causality.

But Claude can misread subtle atmospheres in a conference room, implicit disagreement conveyed through silence, or joking sarcasm as earnest proposals. Whether Claude's action item table fully captures the truth of the conversation is the human's responsibility. The person freed from the labor of note-taking now plays the editor's role, filling in the context the machine missed.

d If You Run Into These Issues

(1) I uploaded a transcript file, but Claude says "File not found." Check that the correct folder is selected at the top of the Cowork screen. If you don't designate a folder, Claude cannot access the files. Even if you have subfolders within the folder and placed the file inside one, access may fail. Try placing the file at the top level of the designated folder.

(2) The summary is too long, or it looks like the original text was copied almost verbatim. Specify length limits in the prompt. When you include specific numbers like "within 10 lines" or "one A4 page or less," Claude adjusts the compression level. The instruction "focus on decisions" is also important. Without this one line, Claude will try to describe the discussion process as well.

(3) The owner or deadline for action items is inaccurate. When the transcript mentions the owner's name unclearly (for example, "if your side could handle it..." or "someone please take a look"), Claude may guess and assign it incorrectly. You must review the output, fix any errors, and then share it with your team. A document Claude creates is a draft, not a final version.

2 Email Inbox Organization and Reply Automation

When you open your laptop in the morning, dozens of emails have accumulated overnight. Half are newsletters, a few have your name only on the CC line, and only a few need immediate action. The problem is the time it takes to find those few messages. As you open emails one by one, judge their importance, and close them, your high-focus morning hours slip away.

When you link Claude with the Gmail connector, you can structure this process. Claude retrieves a list of unread emails, then analyzes the sender, subject, and first part of the body to categorize them by importance. For important emails, it drafts replies that match your tone of voice.

Understanding exactly what the Gmail connector does eases your mind. According to Anthropic's official documentation, the Gmail connector can search for, read, and draft emails, but it does not send mail directly on your behalf. Drafts are saved in your Gmail account's draft folder, and you must send them yourself. This safeguard prevents Claude from accidentally sending the wrong reply.

a Basic Use: Categorizing Unread Emails by Importance

We start with the process of connecting the Gmail connector for the first time.

[Follow Along] 1. Open Settings in the Claude desktop app.

2. Find the Connectors section and click "Browse connectors." Locate Gmail in the list and click "Connect."

3. The Google sign-in window appears. Sign in with the Gmail account you want to use and approve the access permissions.

4. Once the connection is complete, return to the Cowork tab. Enter the following prompt.

"Check all unread emails in my Gmail and categorize them into these three levels.

Urgent: direct customer inquiries, contracts, payments, mail from courts or government offices.

Standard: internal work requests, schedule coordination, colleague questions. Reference:

newsletters, advertisements, CC'd emails, automated notifications. Organize the classification results in a table. Include the sender, subject, one-line summary, and importance level in the table."

5. Claude accesses Gmail and scans the emails. A popup may appear asking for approval to access each email. Click "Allow."

6. Claude displays the classification results in a table on screen. A briefing is complete, such as "Three emails to check immediately today, five emails to handle today, and twelve emails to just review."

You can change the classification criteria to fit your work. For a lawyer, court mail and direct customer inquiries come first; for a shop owner, customer complaints and shipping-related emails come first. The more specific your work context in the prompt, the higher the classification accuracy.

b Application: Drafting Replies in Your Brand Tone

Once categorization is done, it's time for replies. The biggest concern with reply automation is that it sounds like a machine wrote it. Cowork offers a solution: teaching Claude your past emails to create a profile of your tone and writing habits.

[Follow Along]

1. First, teach Claude your writing tone. In Cowork, enter the following.

"Read 20 emails I wrote directly over the past 30 days in my Gmail sent folder, and create my email tone profile. Analyze greeting patterns, sentence length, frequently used phrases, signature style, and formality level. Save the result as a file named my_email_tone.md in this folder."

2. Claude analyzes your sent folder and generates a tone profile. For example, it might show an analysis like "You prefer short, clear sentences, begin greetings with 'Hello,' and close with 'Thank you, Gyeongzin Kim.'"

3. Now draft the replies. Enter the prompt as follows.

"For the emails classified as urgent and standard importance from earlier, use the my_email_tone.md file in this folder to draft replies that match my voice. Save each draft as a separate text file with the filename in the format draft_reply_[sender_name]_[date]."

4. Claude understands each email's context and drafts replies matching your tone profile. When email threads are long, Claude summarizes the full conversation context before drafting, so it won't repeat information already given.

You read the draft, adjust facts or tone, then send it directly from Gmail. Reply time drops from ten minutes per email to two or three minutes.

c Real-World Application: Automating Your Morning Email Briefing

When you turn your daily email sorting into a scheduled task, you arrive at work with a briefing document already waiting.

[Follow Along]

1. Type /schedule in the Cowork input field.

2. Configure the schedule settings as follows. Frequency: daily on weekdays (Monday to Friday). Time: 7:30 AM. Task: "Check unread emails in my Gmail and categorize them into three levels: urgent, standard, and reference. For urgent and standard emails, draft replies using my_email_tone.md. Save the categorization and drafts to the email_briefing folder with today's date in the filename."

3. Once saved, the task appears in the "Scheduled" section in the sidebar. From then on, Claude runs automatically at 7:30 AM each day.

The effect is greatest with regularly recurring email types: lecture requests, manuscript inquiries, meeting scheduling, quote requests. The details of each request differ, but the reply structure stays similar. Learning from past reply examples, Claude drafts responses to new lecture requests with available dates, fee rates, and needed information, and replies to manuscript requests with possible topics, length, and deadline details.

Remember one principle. A draft reply Claude creates must always be reviewed by a person before sending. In areas where one sentence matters, customer complaints, contract negotiations, legal correspondence, a factual error can have serious consequences. Claude takes on the labor of writing sentences, but the weight of the promises and responsibility in those sentences falls on the person who hits send.

d If You Run Into These Issues

(1) I connected the Gmail connector but got the message "Authentication failed." If two-step authentication is enabled on your Google account, you may need additional verification during the authentication process. Try reconnecting the connector after signing into your Google account in a browser first. If you're using an organizational account (Google Workspace), the administrator may need to allow connector access.

(2) The tone of the draft reply doesn't match my voice. If too few emails are analyzed when creating a tone profile, pattern recognition is inaccurate. Make sure to analyze at least twenty or more emails, and add a condition like "analyze only work emails" to avoid mixing business and personal messages. Open the generated tone profile file and correct any errors, and it will be more accurate next time.

(3) The scheduled task did not run. Scheduled tasks only execute when your computer is on and the Claude desktop app is open. If your laptop enters Sleep mode, it won't work. To have it automatically run before you

arrive at work, you need to set your computer's wake time ahead of the scheduled task execution time.

3 Calendar Integration and Meeting Preparation Automation

You have your first meeting at 9 a.m., but you do not check the calendar until 8:50 a.m. You remember the person's name, but their company affiliation is hazy, and you can barely recall what you discussed in the last meeting. In five minutes, you scramble through past emails and search for notes, but you cannot shake the feeling of walking in unprepared. On days when you have three or four meetings scheduled, the transition time between each meeting is essentially zero.

When you connect Claude to Google Calendar, this situation changes. Every morning, Claude analyzes today's and tomorrow's schedules and creates a briefing note for each meeting that includes attendee information, meeting purpose, and advance preparation items.

A Basic Use: Getting a Summary of Today's Schedule

Start by connecting the Google Calendar connector.

- ☐ Go to Settings in the Claude desktop app and navigate to Connectors.
- ☐ Find Google Calendar, click "Connect," and log in with your Google account.
- ☐ After connecting, enter the following prompt in the Cowork tab:

"Check my Google Calendar for today's and tomorrow's schedules and organize them in chronological order. For each event, create a table with the meeting name, time, attendees, and location (or video conference link)."

- ☐ Claude reads the calendar data and outputs a schedule summary.

What matters here is categorization, not just listing. Not all events carry equal weight. When you have a mix of external client meetings, internal decision-making sessions, simple check-in calls, and meetings that require travel, a summary like "10 a.m. Meeting A, 11 a.m. Meeting B" is not particularly helpful. You can add one line to the prompt: "Distinguish between external and internal meetings, and mark meetings that need advance preparation with an asterisk."

B Application Case: Auto-Generating Pre-Meeting Briefing Documents

Taking a step further from the schedule summary is automatically creating a preparation document for each meeting.

If the attendee is an external person, understand their company affiliation, title, and recent activities. If it is an internal meeting, summarize the current status of the related project.

- ☐ Create a "Meeting Prep" folder on your desktop, and if you have past meeting notes or related documents, place them in this folder.

- ☐ In Cowork, designate the "Meeting Prep" folder as your working folder and enter the following prompt:

"Check my Google Calendar for external meetings scheduled for today. For each meeting, create a briefing document in Word format with the following content.

First, provide an overview of the other party's company and recent news. Use web search to find articles from the past 30 days. Second, summarize any previous discussions with that company if notes are available in this folder. Third, draft an agenda for today's meeting. Organize it in the order of 'meeting purpose-discussion items-decisions needed-questions list.'

Name the file in the format 'Briefing_CompanyName_Date' and save it in this folder."

- ☐ Claude extracts attendee information from the calendar, collects company information through web search, and references past notes in the local folder to generate the briefing document. Since this task combines multiple data sources, you will see sub-agents working simultaneously.

- ☐ After completion, check the briefing document in the folder.

As this approach accumulates, you can also gain insights into your work patterns. By analyzing your calendar for a month, the data clearly shows what percentage of your work time is spent in meetings, which types of meetings are most common, and whether meetings tend to cluster on certain days.

C Real-World Application: Receiving Daily Morning Meeting Briefings Automatically

This task can also be turned into a scheduled workflow.

- ☐ Enter /schedule and configure it as follows. Execution frequency: run on weekdays Time: 7 a.m. Task content:

"Check all events on my Google Calendar for today and categorize them into external and internal meetings. For external meetings, create a one-page briefing document that includes the other party's company information and a draft agenda. For internal meetings, organize the current status of related projects and any carried-over items from the previous meeting. Save the complete results in the 'Meeting Prep' folder with today's date as the filename."

□ When you arrive at the office each morning, today's briefing document is already in your "Meeting Prep" folder. By reading this document in the short time it takes to walk to the conference room, you can be as prepared as if you had spent days researching.

However, data has its limits. The briefing document shows the other party's recent financial performance, but the fatigue of internal conflict may fill the face of the person sitting across the table.

A machine presents conversation scenarios by combining facts and statistics, but it cannot tell you why the other person's eyes are wavering at this very moment. The machine has taken over the labor of understanding facts through documents, but the flexibility to read the atmosphere and change the direction of conversation in the field depends on human intuition.

D If You Encounter These Issues

(1) The calendar is retrieving the events, but attendee information shows only email addresses without names. This happens when the attendee's full name is not entered in the Google Calendar event. If you record the attendee's full name and company affiliation in the notes section when registering an event, Claude can create more accurate briefings.

(2) The company information found through web search is about a different company with the same name. Errors decrease if you include the company's exact English name, location, and industry in the prompt. Specify it clearly, like "ABC, an IT services company based in Gangnam-gu, Seoul."

(3) Personal events (doctor's appointments, family plans) are mixed in with your calendar and appear in the briefing. Add a condition to the prompt: "Focus only on business-related meetings. Exclude events that contain keywords like 'personal', 'doctor', or 'family.'" If you separate personal events into a separate calendar in Google Calendar, filtering becomes much cleaner.

4 Business Trip Preparation and Schedule Planning

A business trip is an extension of work, but the preparation process drains a different kind of energy. Searching for flights and accommodations, checking local transportation, calculating travel time between meeting locations, finding dining options, and estimating expenses are each small tasks, but together they add up.

For international business trips, you must also consider time differences, visas, currency exchange, and local communication options. Few people organize all of this into a single document. Most people head to the airport carrying fragments of information scattered through their minds.

When you hand trip-related files to Cowork, it gathers the scattered information and organizes it into three documents: an itinerary, an expense report, and a preparation checklist.

A Basic Use: Creating a Business Trip Itinerary

This is the most basic form. You provide the trip destination, duration, and main meeting schedule in text, and receive a day-by-day itinerary.

- ☐ Create a folder called "Tokyo Trip" on your desktop.
- ☐ In Cowork, select this folder and enter the following prompt:

"Prepare for a business trip to Japan, Tokyo from Monday through Wednesday next week.

Depart Incheon Airport Monday at 2 p.m. (flight not yet confirmed), meeting with Company A in Shibuya Tuesday at 10 a.m., meeting with Company B in Marunouchi at 3 p.m., depart Narita Airport Wednesday at 5 p.m. on return flight.

Based on this information, create a day-by-day itinerary in a Word document. Include transportation modes between each appointment (train, taxi) and estimated travel time. Recommend a hotel location convenient to both meeting venues.

Save with the filename 'Tokyo_Business_Trip_Itinerary' in this folder.

Claude conducts a web search to identify Tokyo's transportation information and hotel areas, then organizes the itinerary by time slot.

The most common error in itineraries is not the meeting times themselves, but the travel time between meetings. Claude calculates transit time by mode between two points and presents options like "Train from Shibuya to Marunouchi: 25 minutes, taxi: 35 minutes (varies by conditions)". An itinerary prioritizes arrival times over romance.

B: Application Case: Adding an Expense Forecast Report and Preparation Checklist

Create an itinerary that includes expenses and preparation items all at once.

[Follow Along]

In the same 'Tokyo Business Trip' folder, add any booking confirmation files you already have. Format doesn't matter: screenshots of flight confirmation emails, hotel voucher PDFs, etc. Cowork can read the contents of image files.

Enter the following prompt:

"Read all the booking confirmations in this folder and combine them with the itinerary you created earlier to produce the following two additional files.

First, create an expense forecast report in Excel. Divide the items into airfare, accommodation, transportation, meals, and miscellaneous, and show estimated amounts in both Korean won and Japanese yen. Search for today's exchange rate. Include subtotal and total formulas.

Second, create a preparation checklist in a Word document. Search for Tokyo's weather that week and include clothing advice. Separate business meeting items (business cards, presentation materials, etc.) from standard international travel items (passport, charging adapter, mobile Wi-Fi, etc.). Save both files to this folder."

Claude reads flight information from the screenshot, extracts check-in times from the hotel voucher, verifies exchange rates and weather through web search, then compiles all three documents.

In organizations requiring pre-approval for business travel, the expense forecast report can be attached as-is. After the trip, it serves as a benchmark document to compare projected versus actual costs. Small recurring items like airport rail, local taxis, gratuities, and meeting room beverages often emerge during post-trip settlement; Claude anticipates these details and preemptively includes them in the "miscellaneous" category.

C: Real-world Application: Assembling a Team Business Trip Package All at Once

Now imagine five team members traveling together instead of one person. Flight booking emails, hotel vouchers, and messenger screenshots shared with colleagues are all mixed together.

[Follow Along]

Add all booking-related files to the 'Tokyo Business Trip' folder regardless of format. PDFs, screenshot images, and text files can be mixed together.

Enter the following prompt: "This folder contains all booking confirmations for our five-person team's Japan business trip. Files are a mix of PDFs, images, and text. Read all these materials and create the following travel package.

First, a daily business trip itinerary (Word) visible to all team members. Organize flight numbers, airport arrival times, hotel check-in times, meeting times, and meal times by time slot. Include the mode of transportation and duration for each leg.

Second, a team-wide expense forecast report (Excel). Separate per-person costs from total team costs.

Third, a shared preparation checklist (Word). Organize weather, business etiquette (including notes on Japanese business card exchange protocol and gift-giving culture), and essential items. Save all three files to this folder."

Claude reads the contents of all files regardless of filename or extension. It extracts flight numbers and layover times from screenshots, pulls check-in times from hotel vouchers, and reorders everything chronologically.

Once this package is complete, repeated questions in the team group chat ("What time is the flight?", "What was the hotel name again?") disappear. You only need to share the itinerary file. Even in locations with unstable internet, the team can check the schedule using a pre-saved Word document.

For frequent business travelers, this structure itself becomes an asset. Change only the destination and dates, and a new travel document is quickly completed. Save it as a skill for even faster reuse: "Save the business trip preparation workflow I just ran as a skill named 'International Business Trip Package'."

For the next trip, one line suffices: "Apply the 'International Business Trip Package' skill to this folder."

Yet even the tightest machine-made itinerary can crumble before reality's whims. Airport baggage system failures, unannounced train strikes, and sudden road accidents exist beyond algorithmic foresight. Cowork eases the burden of trip preparation, but when lost in an unfamiliar city and needing to find a new route, that work falls to the person on the ground.

D: If These Problems Arise

(1) Text recognition from screenshot images fails. Low image resolution or small text reduces accuracy. Use PDF or text files when possible; if screenshots are necessary, capture them enlarged so text appears clearly.

(2) Calculated travel times differ from reality. Travel times Claude obtains through web search are average estimates. They don't account for rush-hour congestion, local construction, or detours caused by events. It's safer to include a note in the prompt to add roughly 30 minutes of buffer time to the itinerary.

(3) Exchange rates in the expense report become outdated. The rates Claude finds reflect the search date. They may differ from actual rates on the travel date, so add to your prompt: ask for a note in the report stating "Exchange rate applied as of search date; actual rates may vary".

[← Previous Chapter](#)

[See every chapter](#)

[Next Chapter →](#)

Attorney · Former Member of the National Assembly · AI Policy Researcher

kimkj.com

© 2026 Kim Kyung-jin. All rights reserved.

[#KimKyungjin](#) [#AIBLibrary](#) [#ClaudeCoworkandAgentManual](#)

Chapter 6: HR, Legal, and Customer Support

← Previous Chapter

See every chapter

Next Chapter →

Friday at five in the afternoon, a startup on Teheran-ro in Seoul. On the HR manager's monitor sit 247 resumes, unsorted, that flooded in after the job posting closed.

PDFs, Word documents, and Hangul files are mixed together, and some applicants pasted portfolio links directly in the email body. At the desk next door, the legal officer is printing a 45-page joint venture agreement from a partner company and highlighting it with a fluorescent marker.

In the customer support room across the hall, three agents are typing without pause. Since the system update, more than 120 support tickets have piled up, a mix of emails demanding refunds for payment errors and lengthy feature requests that say "Could you build this feature?"

These three scenes have something in common.

All three involve reading text, sorting it, making judgments, and responding with text. The volume to read is enormous, the sorting criteria are fixed, and mistakes in judgment hit people directly. Miss a strong candidate and the organization loses. Overlook a toxic clause in a contract and the company is at risk. Lag on customer complaints and trust crumbles.

The four tasks covered in this chapter are pillars that support a company's back-office (the internal administrative functions not directly visible to customers). They are bulk resume evaluation and ranking, new employee onboarding and offer letter writing, inbound customer support ticket handling, and contract review and non-disclosure agreement (NDA) analysis.

Cowork's HR plugin, legal plugin, and customer support plugin were released sequentially between late January and February 2026.

Anthropic released these plugins as open source on GitHub and designed them so each company can customize them to match its own evaluation criteria, contract review standards, and customer response policies.

When the legal plugin was announced in early February, Thomson Reuters stock fell more than 16 percent in a single day, and LegalZoom dropped nearly 20 percent. A market capitalization of 285 billion dollars vanished.

The press called that day "SaaSocalypse." The magnitude of the impact a single plugin had on the market explains the potential of this tool.

HR, legal, and customer support all touch people's lives directly. Behind every resume is an applicant whose livelihood is at stake. Behind every contract is a decision on which the company's fate depends. Behind every ticket is a customer experiencing inconvenience. Cowork handles reading, sorting, and drafting. Yet deciding which applicants to reject, signing contracts, and sending final responses to customers are human judgments and human responsibilities. Machines save time, but what people do with that saved time determines the quality of the outcome.

1 Evaluating and Ranking Bulk Job Applications

When 300 people apply for one position, a skilled HR manager takes an average of seven minutes to review each resume. Three hundred resumes means 35 hours, well over four days. Over those four days, fatigue accumulates, and there is no guarantee that the first resume read Monday morning and the last resume read Friday afternoon are evaluated by the same standard. The paperwork of a high-potential applicant can be buried beneath an afternoon of weary attention. Human cognitive capacity has limits, and reading through records of hundreds of people by a consistent standard to completion is nearly physically impossible.

Cowork lifts this document review process into a completely different dimension. Using the company's pre-established hiring evaluation rubric as a core framework (the standard system that forms the skeleton of judgment), it extracts relevant items from each resume and converts them to numbers. A machine does not tire, and it reads the 300th resume with the same intensity as the first.

A. Why Is It Necessary?

The problem is that resumes come in all different formats. Some applicants put a clean table in a PDF, while others write freely in a Word file. Some list years of experience as "three years," while others list only project start and end dates.

The large language model that powers Cowork excels at extracting meaning from unstructured text like this. Regardless of format, it infers from context 'How many years of relevant experience does this person have?' and 'What technologies does this person know?'

The heart of evaluation is transparency. The basis for each score must be visible: why did the AI give this applicant 85 points and that applicant 72 points? Scores without justification are neither fair nor useful.

B. What Does It Do?

When you install Cowork's HR plugin, Claude immediately equips itself with a skill specialized for HR work (a configuration file containing domain knowledge and procedures for specific tasks). It collects applications accumulated in email using a Gmail connector, or reads all resume files gathered in a folder at once. It scores each applicant by category according to the evaluation rubric the user provides, scoring years of experience, technical alignment, project experience, and job fit. It records the total score ranking along with the justification for each score in written form. The output is an Excel file with conditional formatting applied.

C. How to Do It: Basic Usage

Let's start with the simplest form. Score 10 resumes according to the evaluation rubric.

[Follow Along]

1. Open the Claude Desktop app and click the "Cowork" tab at the top. Install the HR plugin from the "Customize" menu in the left sidebar. If you haven't installed it yet, press "Browse Plugins," find "HR" in the list, and click the "Install" button.

2. Create a folder on your desktop called "2026_Marketer_Hiring." Put three things in it: your company's hiring evaluation rubric file (PDF or Word), the job description (JD) file, and 10 resume files from applicants.

3. At the top of the Cowork screen, designate this folder as your working folder.

4. Enter the following prompt:

"Please read the hiring evaluation rubric and job description in this folder first. Based on those criteria, analyze all resume files in this folder and perform the following tasks.

First, score each category from the evaluation rubric (relevant years of experience, technical stack alignment, project experience, job fit) on a scale of zero to ten points.

Second, write the reasoning for each score in one or two sentences, explaining why that score was given. For example: 'Mentioned experience with SEO tools, but no concrete performance metrics were provided, so technical proficiency scores 6 points.'

Third, create an Excel file sorted in descending order by total score. The columns should be applicant name, contact information, scores by category, total score, rank, and summary of evaluation reasoning. Highlight the cells of the top 3 applicants in green and the bottom 3 in yellow.

Save the file as 'Marketer_Applicant_Evaluation.xlsx' in this folder."

5. Claude displays a plan: 'I will read the evaluation rubric → I will analyze the job description → I will build a scoring framework → I will analyze resumes sequentially → I will generate the Excel file.' After confirming the plan, enter 'Go ahead.'

6. After two to three minutes, an Excel file appears in the folder. When you open it, next to each applicant's name are their category scores and justifications, all sorted by total score.

D. How to Do It: Advanced Application

Now the applicant pool has grown to dozens, and you are collecting resumes directly from Gmail. 1. Connect the Gmail connector in Cowork. Open the "Connectors" option from the "Customize" menu in the left sidebar and select Gmail.

2. Enter the following prompt:

"Using the Gmail connector, find all emails in my inbox from the past two weeks that have 'Marketer Application' in the subject line. Save the attachments (resumes) from each email to this folder. Then apply the same scoring framework I created earlier to evaluate all applicants and create a new Excel file in the same format. This time

since there are many applicants, write two customized interview questions to ask each of the top 10 applicants in an additional column."

3. Claude accesses Gmail, searches for emails, downloads the attachments, and begins analysis. Progress appears in the sidebar.

E. How to Do It: Real-World Application. In large-scale hiring with more than 300 applicants, generate an evaluation report and executive summary document simultaneously.

1. Enter the following prompt:

"Evaluate all resumes in this folder according to the hiring evaluation form. There may be over 300 applicants, so please process all of them to completion. Create the following three files.

First, an overall applicant evaluation spreadsheet in Excel. Include point scores by category, total score, ranking, summary of reasoning, and conditional formatting applied (top 10% green, bottom 30% red). Second, an in-depth analysis Word document on the top 15 candidates. For each candidate, include key strengths, concerns (red flags), and three customized interview questions. Third, a one-page summary Word document for executive reporting. Include total number of applicants, average score, one-line profile summaries of the top 5, and hiring recommendations."

☐ Cowork runs a sub-agent to process resumes in parallel. It extracts information from 300 unstructured documents and scores them according to the same criteria.

☐ When you open the completed spreadsheet, it is sorted as "1st place: Jung Woojin (total score 56/60 points)," and the Word document includes interview questions.

ba Boundaries in AI-Powered Hiring Evaluation

The more effective this tool becomes, the more important it is to clarify its limits.

First, AI evaluation is a first-pass screening tool, not a final decision-making tool. Potential that does not appear in a resume, cultural fit with the organization, and communication ability shown in an interview are domains that machines cannot read. Among applicants given low scores by AI, there may be talented individuals who shine in interviews.

It is advisable to establish a safeguard in which human resources staff randomly review a certain proportion of low-scoring applicants directly.

Second, there is the issue of bias. The U.S. Equal Employment Opportunity Commission (EEOC) has warned that discrimination against applicants with disabilities may occur in hiring evaluations using software and AI.

If the evaluation criteria include bonus points for graduates of specific universities, or if rules uniformly deduct points from applicants with employment gaps, AI reflects that bias more consistently than humans. If the criteria are fair, AI becomes a fair tool; if the criteria are biased, AI becomes an amplifier of bias. Before implementing a hiring system, the fairness of the evaluation form itself must be checked first.

Third, there is the matter of explainability. While automation is useful for "sorting" candidates, if the structure allows machines to "finalize" rejections, legal risk increases. The scores produced by AI should be used as recommendations, but there must always be supporting evidence that explains why those scores were produced.

The hiring manager who receives a neatly ordered list of applicants ranked from 1st to 300th by machine has been freed from the labor of reading documents. In exchange, they take on the responsibility of looking back at the resumes pushed down to lower ranks and examining whether there might be potential that the algorithm failed to discover.

sa If This Problem Occurs

(1) The system fails to read Korean-language (.hwp) files properly. Cowork is reliable at text extraction from PDF and Word files. Korean-language files may have lower recognition rates depending on format. Guide applicants to submit PDFs, or if you convert Korean files to PDFs and place them in the folder, accuracy improves.

(2) The same applicant has sent resumes multiple times. When collecting via a Gmail connector, multiple emails from the same sender can be captured. Add the condition to the prompt: "If there are multiple emails from the same sender, use only the most recent attachment and ignore previous versions."

(3) AI gives excessively generous scores in certain categories. If the evaluation form criteria are vague, AI interprets broadly. Refine the scoring criteria precisely in the prompt: "An applicant who has written only 'SEO experience' should be capped at 5 points or lower in technical proficiency. Scores of 7 or above require both specific tool names (Google Analytics, Ahrefs, SEMrush, etc.) and performance metrics."

2 Employee Onboarding and Proposal Writing

The moment a successful candidate is confirmed, a new kind of administrative work begins. You must write an offer letter with exact figures for the agreed-upon salary and bonus, vacation days, and probation period conditions, prepare a nondisclosure agreement, and request laptop specifications to the IT department for the first day of work.

At the same time, you need to create an onboarding plan specifying what will be done from day one through the first 30 days, who will be met, and which systems the new hire will receive access to.

An offer letter is a formal document that can carry legal force, and an onboarding plan is a practical guide. Although they differ in nature, both have a repetitive structure while the specific details must differ for each candidate. These are conditions where AI can have strong impact.

ga Why It Is Needed

If mistakes occur in the offer letter, trust erodes. It happens that a salary figure from a previous candidate remains unchanged, or a performance bonus percentage inconsistent with the job level is recorded. When human resources staff handle multiple cases simultaneously, mistakes emerge in these small but critical gaps. If the onboarding plan is poor, the early retention rate of newly hired employees suffers. Multiple research findings show that employees who experience systematic onboarding have lower turnover rates within a year compared to those who do not.

na What It Does

The HR plugin in Cowork has built-in skills specialized in drafting offer letters and establishing onboarding plans. When you provide the company's standard offer letter template and compensation table, it generates a document that fills in the confirmed conditions for a specific candidate accurately. The onboarding plan takes job information, department information, and the new hire's experience level as input and creates a customized schedule and checklist.

da How to Do It: Basic Use

Let's create one offer letter.

[Walk-Through]

- Place the company's offer letter template (Word file) and successful candidate conditions memo (text file) in the "2026_Hiring_Onboarding" folder. In the conditions memo, write the name, job title, salary, start date, and special provisions.

- In Cowork with the HR plugin installed, enter the following prompt:

"Read the offer letter template and successful candidate conditions memo in this folder. Accurately reflect the conditions noted in the memo (name: Jung Woojin, job title: Content Marketing Specialist, salary: 60,000,000 won, start date: May 2, 2026, paid vacation: 20 days per year) in the template and write an offer letter Word document. Use an official tone but include a welcoming feeling. Include signature lines and acceptance deadline (7 days from send date). Save the file with the filename 'OfferLetter_JungWoojin.docx'."

- Claude generates a document that maintains the template's format and company logo while filling in the blanks with the relevant conditions.

- ❑ Be sure to open the document and verify the figures. Confirm that the salary, start date, and probation period conditions match the memo exactly.

ra How to Do It: Application Example

Create an onboarding plan in addition to the offer letter all at once.

- ❑ Add to the folder a text file that lists the company's system environment information (internal messenger is Slack, document management is Notion, email is Gmail).
- ❑ Enter the following prompt:

"Following the offer letter created earlier, additionally create an onboarding package for Jung Woojin.

First, create a time-block onboarding schedule from the first day (Day 1) through the first week (Week 1) as a Word document. Reflecting the fact that the company uses Slack, Notion, and Gmail, include specifics about when to set up accounts in which systems, when to meet team members, and when to receive the first project briefing. Second, also create an onboarding progress tracking checklist Word document. Include items such as 'Completed security training,' 'IT account created,' 'Salary account registered,' 'Team introduction meeting completed.' Third, write a welcome email draft as a text file to send to the successful candidate along with these documents. Save all three files in this folder."

- ❑ Claude generates the onboarding schedule, checklist, and email draft simultaneously.

ma How to Do It: Real-World Application

Process successful candidates from multiple positions simultaneously.

- ❑ Create separate successful candidate condition memos for a developer, a sales representative, and a designer, and place them all in the same folder.
- ❑ Enter the following prompt.

"I have notes with conditions for three successful candidates in this folder. Create offer letters, customized onboarding schedules, checklists, and welcome emails for each candidate. Since their roles differ, their onboarding schedules should differ too. For developers, code repository access and dev environment setup come first in the first week; for salespeople, CRM system and product materials training come first; for designers, Figma access and brand guideline review come first. Create subfolders for each candidate and separate the files."

- ❑ Cowork executes subagents to generate packages for all three candidates in parallel. Common items (security training, salary account registration) are structured identically; role-specific items are structured differently.

The machine created administrative documents with precision. But what a new employee truly wants is not perfectly formatted contracts or meticulously structured schedules. What eases the anxiety of

stepping into an unfamiliar environment is the warm greeting from people who welcome them. Now that the machine has lifted all administrative burden, HR managers and department leaders can focus on layering their own sincere words of welcome over those smooth documents.

If These Issues Arise

(1) The salary figure in the offer letter differs from the memo. Claude may have misread the number in the memo. If you format numbers in the memo file clearly with commas and units, like "60,000,000 won," recognition accuracy improves. The numbers in generated documents must always be verified by a person.

(2) The onboarding schedule does not match your company's actual process. The HR plugin's default skill follows a general onboarding sequence. If your company has unique procedures (for example: required executive meeting on day one, mandatory legal training in week two), customize the plugin by adding your procedures to the skill file. Press the "Customize" button to modify it in conversation with Claude.

3 Processing Inbound Customer Support Tickets

A day in the customer support department is a battle with flooding tickets. Product defect reports, refund requests, feature improvement demands, usage inquiries, and account issues come pouring in without distinction.

Support agents must read each ticket, identify the problem, prioritize it, and write a response aligned with company policy. The hardest part of this process is conveying a message of refusal to emotionally charged customers. Stripping away profanity and agitation, then identifying what technical action the customer truly needs, and composing a dry yet warm response following a set manual is enormous emotional labor.

Why It Is Needed

The bottleneck in support work often arises not from "being unable to write a response" but from "not knowing who should handle this ticket." If it is sorted from the start as to whether it is a feature request, a system failure, a usage question, or a refund complaint, response speed and quality differ greatly.

What It Does

Cowork's Customer Support Plugin automatically classifies incoming tickets, assigns priorities, and drafts responses aligned with company policy.

According to Anthropic's official documentation, this plugin has five slash commands (Slash Command, executable commands starting with the "/" symbol) built in. /triage classifies tickets and assigns priorities. /research finds answers to customer questions from multiple sources. /draft-response drafts a response suited to the situation and channel. /escalate creates an escalation brief to hand off to engineering or the product team. /kb-article converts resolved issues into

knowledge base articles.

How to Use It: Basic Usage

Let's classify 10 customer inquiry emails and create response drafts.

[Follow Along]

- ❑ Install the Customer Support Plugin in Cowork. From the Customize menu: Browse Plugins → Customer Support → Install.

- ❑ Save customer inquiry emails as text files in the "customer_support_tickets" folder. (If you have connected the Gmail connector, you can read emails directly without needing a folder.)

- ❑ Enter the following prompt.

`"/triage Analyze the 10 customer inquiry files in this folder and classify them as follows. Categories: Bug Report, Feature Request, Payment Issue, Usage Question, Account Problem. Priorities: P1 (urgent, service outage or payment error), P2 (high, core feature bug), P3 (normal, general inquiry), P4 (low, feature suggestion). Format the results in an Excel table. Columns: Ticket Number, Customer Name, Category, Priority, Key Summary (one line), Recommended Action."`

- ❑ Claude analyzes the intent and urgency of each email and generates a classification table.

- ❑ Next, request response drafts.

`"/draft-response Create an immediate response draft for the P1 tickets just classified. Start with a sentence that empathizes with the customer's inconvenience, include refund procedure guidance in the middle, and close with an invitation to reach out if further help is needed. Tone: formal but warm."`

- ❑ Claude generates customized response drafts for each P1 ticket as text files.

How to Use It: Applied Examples

Create a respectful rejection email for feature request tickets.

- ❑ Enter the following prompt.

`"/draft-response Find the feature request category tickets from the classification results. These requests are not included in our current product roadmap. Create a response draft that thanks the customer for the idea while honestly informing them that it is not in current development plans. However, the tone should make the customer feel respected. Include the sentence "Your feedback has been shared with the product team," but be careful not to make it sound like a promise. Create personalized responses reflecting each customer's name and request."`

- ❑ Claude generates customized rejection emails for each customer. Support agents review the drafts and send them.

How to Use It: Real-World Application

Process all of a day's tickets at once and generate a weekly analysis report.

□ With the Gmail connector connected, enter the following prompt.

"Use the Gmail connector to read all unreviewed tickets from today in the support email box (support@company.com). Complete the following tasks in order.

First, classify all tickets with /triage and assign priorities. Second, create immediate response drafts with /draft-response for P1 urgent tickets. Third, summarize feature request tickets and add them to the 'feature_requests_weekly_summary.xlsx' file. If the file exists, append to it; if not, create a new one. Fourth, create an engineering team escalation brief with /escalate for serious bug reports. Include bug reproduction steps, impact scope, and number of reporting customers. Fifth, analyze all tickets from this week and create a 'Weekly Customer Support Report Word document.' Include ticket counts by category, average response time, and the top 3 most frequently occurring complaint types."

□ Cowork executes multiple subagents simultaneously to process classification, draft creation, escalation, and reporting in parallel.

Once this system is in place, customer support elevates from reactive troubleshooting to an input channel for product improvement. When the data accumulated in the feature request Excel file reaches the product team, priorities can be set on evidence rather than intuition.

However, you should not send AI-generated responses as is. AI drafts are a starting point for agents, not the final product. There may be context in a customer's situation that does not appear in the text, and continuity with previous support history must also be considered. In a structure where AI completes 80 percent and agents fill in the remaining 20 percent, agents can handle more tickets per day while delivering more thoughtful service to each customer.

If These Issues Arise

(1) Claude misreads customer emotion and creates a response with an inappropriate tone. Sometimes Claude writes a serious apology when a customer made a complaint mixed with humor. Add this instruction to your prompt: "When setting the response tone, first assess the emotional intensity of the customer's message as high, medium, or low, then apply a tone that matches."

(2) Feature request rejection emails feel mechanical. The quality of a rejection email depends on how precisely you reference the customer's specific request. Give this instruction: "Always include the specific feature name the customer requested in the response, and explain in one sentence why that feature is not in the current product."

(3) The escalation brief omits information needed by the engineering team. The basic output of the /escalate command may not include technical information such as browser version, OS, and customer plan tier. Add the following condition: "Include the customer's plan tier, browser used, and error occurrence time in the escalation brief. If this information is not in the customer email, mark it as

'Information Not Provided.'"

4 Contract Review and NDA Analysis

A 45-page joint venture partnership agreement from the other party has arrived at the legal team. The review deadline is 5 business days. At the same time, three nondisclosure agreements (NDAs) with new vendors, one revised software license agreement, and one data processing contract are awaiting review. Two lawyers must review all these documents within the deadline.

Reviewing legal documents demands both deep expertise and an unrelenting, meticulous attention. Behind courteous opening language, clauses designed to shift all liability to the other party in case of dispute lie hidden within complex phrasing. Miss a single ambiguous word in Article 14, Section 2 that sets liability limits, and the company inherits billions of won in litigation risk.

Why It Is Needed

The bottleneck in legal work is reading time. Carefully reading all 45 pages from start to finish, understanding each clause, and comparing it against the company's compliance workflow (the internal review process for verifying regulatory compliance) takes more than a day. During that time, the lawyer cannot address other urgent legal issues.

What It Does

Cowork's Legal Plugin was first released on February 2, 2026. This plugin is a specialized system that automates contract review, NDA classification, and compliance workflow. According to Anthropic's official documentation, there are five core slash commands.

`/review-contract` compares the contract against the company's negotiation playbook (an internal guide that defines acceptable ranges and negotiation criteria for each clause), assigns each clause a rating of safe (GREEN), caution (YELLOW), or risk (RED), and provides revision suggestions.

`/triage-nda` quickly sorts incoming NDAs into three categories: standard approval, legal review, and full re-review. `/vendor-check` confirms vendor contract status. `/brief` generates summary briefings on legal issues. `/respond` creates template-based legal correspondence such as data subject requests or litigation hold notices.

[Note] What Is a Playbook?

The heart of the Legal Plugin is the playbook. A playbook is a document that codifies internal standards, stating "we interpret this clause this way and accept only this scope." It contains rules such as "unlimited liability clauses are RED" or "non-compete periods of two years or less are GREEN, three years or more are YELLOW." Without a playbook, the AI cannot judge what poses risk by our standards. Without one, analysis is limited to standard commercial conventions.

How to Use It: Basic Usage

Let's analyze one NDA.

[Walk-Through]

- ☐ Install the Legal Plugin in Cowork. Customize → Browse Plugins → Legal → Install.
- ☐ Place the NDA file (PDF) from the other party in the "2026_Contract_Review" folder.
- ☐ Enter the following prompt.

`"/triage-nda` Analyze the NDA file in this folder. I'm the disclosing party. Check the following. First, whether the definition of confidential information is not overly broad. Second, whether the confidentiality period is reasonable (industry practice is typically 2-3 years). Third, whether a residuals clause is included.

Fourth, whether return and destruction obligations have a backup exception. Tell me which classification applies: standard approval, legal review, or full re-review."

- ☐ Claude reads the NDA and marks the analysis result and classification for each check item.

How to Use It: Advanced Application

Apply the company's playbook to review the contract clause by clause.

- ☐ Place both the company's contract review playbook file (Word or text) and the other party's contract in the "2026_Contract_Review" folder.
- ☐ Customize the plugin. Click the Customize button and instruct Claude: "Our company's playbook file is 'ContractReviewCriteria.docx' in this folder. Apply this standard to the `/review-contract` command."
- ☐ Enter the following prompt.

`"/review-contract` Review the joint venture partnership agreement in this folder clause by clause using our playbook. Create the following outputs. First, a clause-by-clause analysis report (Word). Assign each major clause (definition of parties, capital structure, profit distribution, governance, intellectual property ownership, non-compete obligations, termination conditions, dispute resolution, and governing law) a rating of GREEN, YELLOW, or RED. For YELLOW and RED clauses, describe the nature of the risk and the recommended revision direction. Second, a one-page executive summary (Word) that executives can read in five minutes. Include the three key risks, immediate action items, and negotiation priorities. Save both files in this folder."

- ☐ Claude analyzes the 45 pages and generates a report with clause-by-clause traffic light ratings and a one-page summary.

How to Use It: Real-World Application

Handle multiple NDAs and a large contract simultaneously.

- ☐ Place all five NDAs and one partnership agreement in the folder.
- ☐ Enter the following prompt.

"Review all legal documents in this folder. Quickly classify NDA files with /triage-nda and conduct deep clause-by-clause analysis of the partnership agreement with /review-contract. First, summarize the classification results of the five NDAs in a single Excel table. Use columns for filename, counterparty company name, confidentiality period, key risk items, and classification (standard/review/re-review). Second, create a clause-by-clause analysis report of the partnership agreement based on the playbook. Third, for clauses marked RED, prepare a draft revised agreement that includes the redline language we will propose to the other party. Fourth, prepare a one-page executive summary. Save all files in this folder."

- ☐ Cowork simultaneously performs two levels of review: rapidly classifying NDAs and conducting deep analysis of the agreement.

The Absolute Boundary of AI Legal Analysis

AI analysis of legal documents does not replace a lawyer's legal judgment.

AI's role is to recognize patterns in text and surface potential issues. However, only a lawyer can determine whether a clause is actually enforceable in light of a specific jurisdiction's case law, whether the company's business strategy justifies bearing that risk, and how extensively to demand revisions in view of the relationship with the other party.

When a real problem arises from a clause that AI classified as "no risk," responsibility falls to the reviewing lawyer, not the AI. It is essential to review AI analysis results critically rather than accept them blindly.

Anthropic itself states that Cowork is in research preview and should not be used for regulated workloads. The Legal Plugin's output is "assistance," not "advice." The responsibility for final judgment and signature always remains with a person.

Cowork is an auxiliary tool that saves lawyers time from physically reading 45 pages and double-checks clauses they might miss. We have delegated the grueling work of analyzing clauses to the machine, but the weight of judging what those clauses will produce in practice and the moment of signing the contract remain the responsibility of the person at the desk.

Troubleshooting

(1) I ran /review-contract without a playbook and the analysis is superficial. Without a playbook, Claude analyzes according to general commercial standards. Since the company's specific risk tolerance is not reflected, practical value is reduced. Create a playbook file by referencing the company's contract review checklist or past opinion letters and connect it to the plugin. The playbook

should at minimum specify acceptable ranges and escalation criteria for each major clause type.

(2) Claude does not accurately recognize legal terminology in Korean contracts. For domestic legal terms such as "anticipated damages," "warranty against defects," and "loss of benefit of time," Claude may misinterpret the context. Accuracy improves when you specify in the prompt: "This contract is a Korean-language document governed by the laws of the Republic of Korea. Analyze it within the context of Korean civil law and commercial law."

(3) Too many clauses marked RED render the playbook impractical. When the playbook's standards are overly strict, most clauses get flagged RED, making it difficult to determine priorities. Revise the playbook to separate "clauses we will never accept" from "clauses open to negotiation," applying RED only to the former and YELLOW to the latter.

The four tasks discussed in this chapter share a common structure. What machines do is read vast amounts of text, classify according to criteria, and generate drafts. What machines cannot do is recognize a candidate's potential, welcome new employees with genuine warmth, understand a customer's urgency with empathy, or make judgment calls about accepting legal risk.

Freed from reading three hundred resumes, the hiring manager becomes someone who notices possibilities the machine missed. Released from document drafting, the HR specialist becomes someone who hands a cup of coffee to the employee arriving for their first day. No longer sorting tickets, the support agent becomes someone who checks whether the machine's response reached the customer's heart. Liberated from forty-five pages of text, the lawyer becomes a strategist weighing risks against opportunities.

Reports and drafts created by cwork are not finished products but starting points. What people add from that point onward determines the quality of hiring, customer trust, and contract safety.

[← Previous Chapter](#)

[See every chapter](#)

[Next Chapter →](#)

Attorney · Former Member of the National Assembly · AI Policy Researcher

kimkj.com

© 2026 Kim Kyung-jin. All rights reserved.

[#KimKyungjin](#) [#AILibrary](#) [#ClaudeCworkandAgentManual](#)

Chapter 7: IT Projects and Software Development

← Previous Chapter

See every chapter

Next Chapter →

Building software is similar to constructing a house. You draw blueprints, select materials, build the frame, finish the interior, and check for leaks.

The most time-consuming work in this process is not the act of laying bricks, but deciding where and how to lay them and verifying that they are laid correctly. Claude Code and Cowork reduce the burden on developers and planners in the areas of decision-making and verification.

This chapter differs in character from the previous chapters 1-6. Although it deals with code, we have lowered the threshold so that readers without coding experience can follow along in planning areas such as writing requirements documents (PRDs) and creating test case documents. Conversely, in areas requiring development experience such as operating agent teams or code migration, we prioritize conceptual understanding, while execution guidance is aimed at intermediate and advanced readers.

1 Parallel Work Based on Agent Teams

A One Developer Doing the Work of Four

Imagine building a single web application.

You need to build the user-facing interface (frontend), establish a server that processes data behind the scenes (backend), and design a database to store the data. Finally, you must test whether all three work together correctly.

If one person does these four tasks sequentially, the next task cannot begin until the previous one is finished. The frontend cannot retrieve data without a completed backend, and backend code cannot be written without a finalized database structure. This sequential bottleneck is the biggest cause of prolonged development time.

In actual companies, teams solve this problem. Frontend developers, backend developers, database administrators, and QA engineers work simultaneously on their respective areas and communicate and coordinate with one another as needed. Agent Teams reproduce the structure of human teams using AI.

B What Are Agent Teams?

Agent Teams are a function where multiple Claude Code instances (individual AI sessions running simultaneously) each take on independent roles to execute a single project in parallel. They were

released as a Research Preview alongside the launch of Claude Opus 4.6 in February 2026.

Previously, there was a concept called sub-agents.

Sub-agents operated by following instructions from the main agent and returning only results. The problem was that sub-agents could not communicate with each other. There was no way for the frontend sub-agent to directly tell the backend sub-agent, "The API address has changed, so fix the code." All messages had to pass through the main agent, and this bottleneck frequently caused code conflicts in complex projects.

Agent Teams changed this structure. Team member agents can exchange messages directly with each other. Through a Shared Task List, they can see who is doing what work, and when one task is complete, they take on the next one themselves. An agent in the Team Lead role coordinates the overall effort, while practical communication happens directly among team members.

[Note] What is an agent instance? It refers to a single Claude Code session. Each instance operates independently with its own context window (the range of information an AI can remember at one time). Claude Opus 4.6 supports a context window of 1 million tokens (approximately 750,000 words) per instance, allowing you to work with an entire large codebase at once.

C How to Use Agent Teams

(1) Basic Use: Enabling Agent Teams and First Run Agent Teams are an experimental feature and must be enabled separately.

- ☐ Open Claude Code's settings file (settings.json).
- ☐ Add the following line: "CLAUDE_CODE_EXPERIMENTAL_AGENT_TEAMS": "1"
- ☐ Save the settings and restart Claude Code.
- ☐ In the terminal, give the following instruction: "Analyze the README.md file and entire folder structure of this project, then create an agent team with two members, one for security review and one for code quality review, and perform a code review."
- ☐ Claude Code assumes the Team Lead role and creates two team member agents. The activity log of each agent is displayed in the terminal.
- ☐ Team member agents each analyze the code and share any issues they find with each other through messages. The Team Lead synthesizes the results and generates a final review report.

(2) Practical Example: Splitting Screen Views by Agent with tmux If you install tmux (a terminal multiplexing tool), you can divide each agent's work screen and observe them simultaneously.

- ☐ When you run an agent team with tmux installed, the screen divides into multiple panels, with each agent's terminal displayed separately.

- Press Shift plus an arrow key to move to a specific agent's panel, then give instructions to only that agent.

"Frontend agent, change the button color of the login page you just created from blue to navy."

- The agent receives the instruction, makes the change, and notifies the team members of the modification.

(3) Real-World Application: Building a Blog Platform Project A scenario for building a small to medium-sized blog platform.

- Prepare a requirements document in the project folder. (The method for writing a PRD is covered in Section 2 of this chapter.)

- Give Claude Code the following instruction:

"Read this PRD and create an agent team consisting of a backend API lead, a frontend UI lead, a database schema lead, and a test writing lead. Have the backend agent implement a REST API using Node.js and Fastify, the frontend agent implement the UI with React. Have the database agent finalize the PostgreSQL schema first and share it with the other agents. Have the test agent write unit tests for each API endpoint."

- Claude forms the team and distributes work. Once the database agent finalizes the schema first, the backend and frontend agents simultaneously write code based on that schema.

- As work progresses, the Team Lead provides a summary of status. If conflicts arise, the Team Lead judges priorities and resolves them.

D Things to Know When Using Agent Teams

Agent Teams consume a large amount of tokens. Because each team member maintains an independent context window while exchanging messages with others, costs can increase 4 to 7 times compared to having a single agent do the same work. There is no reason to use agent teams for simple tasks like script writing or file organization.

The situations where agent teams prove effective are clear: projects that must simultaneously touch multiple layers (frontend, backend, database, testing), debugging that requires validating different hypotheses at the same time, and large-scale code reviews where parallel exploration creates real value.

One more thing. While code produced by agent teams may look clean individually, it can cause unexpected conflicts within the context of the overall system. A caching structure introduced by one agent for performance may conflict with another agent's

real-time synchronization logic. Agents may write code faster, but the responsibility for judging how that code operates within the overall system still rests with humans.

E If These Problems Occur

(1) If an agent team is created but only one message is sent and the session terminates, verify that the agent team settings have been correctly saved in settings.json. Check that quotation marks and colon placement are exact, then completely close Claude Code and restart it.

(2) If team member agents simultaneously modify the same file, causing code to break, specify file ownership in the prompt. Instructing each agent to modify only files in its assigned folder and read-only access to other agents' files can reduce conflicts.

(3) If token usage runs much higher than expected, reduce team size. For tasks that five people can do with three, adding extra members only increases coordination costs. The Max pricing plan suits agent team work better than the Pro plan.

2 Software Requirements Definition and Architecture Documentation

A Why "Make an app with payment features" fails

The most common failure in asking AI to write code stems not from lack of AI capability but from vague instructions. When you say "Make a project management app with payment features," AI fills in numerous blanks on its own. Is payment by card or bank transfer? Is user authentication email or social login? Is project management organized by team or individual? When the assumptions AI supplies differ from your intent, the result is something nobody wanted.

A Product Requirements Document (PRD) is a document that fills in these blanks ahead of time. It clearly states what will be built, who it is for, what success looks like, and what will not be done this time. PRD quality determines whether a project succeeds or fails.

B What kind of PRD does Claude Code produce?

With Claude Code's Plan Mode, AI focuses entirely on analysis and design without writing a single line of code. Feed it fragmented ideas floating in your head, and AI asks systematic questions to fill the gaps, then organizes the results into a multi-page planning document.

The final deliverable includes the following. Service purpose and success criteria, core user journeys, functional and non-functional requirements (performance, security, availability), database schema (table relationships, data types, constraints), system architecture (tech stack, infrastructure layout, authentication scheme), and test scenarios.

C Let's create a PRD

(1) Basic use: extracting a question list from an idea

1. Open Claude Code terminal and switch to Plan Mode.

2. Enter the following:

"I want to build a secondhand marketplace app. Location-based search, chat, and escrow payment are the essentials. Before I write code, give me ten or more clarification questions to reduce assumptions about my project."

3. Claude presents questions like these: "What distance radius should location-based search use in kilometers?" "Will chat support image uploads?" "What commission rate for escrow payment?" "Is signup email verification or social login?" "Do you need seller authentication?"

4. Answer each question.

5. Claude generates a PRD draft based on all your answers.

(2) Advanced use: including database schema and API specification

1. Review the PRD draft from basic use, then enter additional instructions:

"Design a database schema in Prisma format based on this PRD. Include User, Product, Transaction, Chat, and Review tables, and specify each relationship and index. Then organize core API endpoints in OpenAPI format."

2. Claude generates a Prisma schema file and OpenAPI specification.

3. Open the generated documents and review the content.

(3) Real-world use: Using PRD as the agent team's execution guide. PRD's real value lies in its role as the starting point for coding.

1. Once PRD is complete, reset your Claude Code session. Mixing previous conversation into the context window creates confusion.

2. In the new session, give this instruction:

"Read the PRD.md file, schema.prisma, and openapi.yaml in this folder. Assemble an agent team according to these documents and build the app. Use Fastify for backend, React for frontend, and PostgreSQL for database."

3. The agent team begins work, treating the PRD as the source of truth for execution.

D Why you cannot use AI-generated PRD as-is

AI-generated PRD is a starting point, not a final document. AI excels at creating logically airtight structures but does not fully understand subtle business context differences. Real-world constraints like Korea's escrow payment regulations, PIPA data retention requirements, and UX preferences by customer age must be reviewed by people.

It works well to sequence review this way. The product owner confirms goals and scope. Engineers assess technical feasibility. Security staff verify authentication, encryption, and logging policy.

Operations review deployment and monitoring systems. Good design documents do not emerge complete in one pass; they crystallize decisions through review cycles.

E Territory non-developers can master

PRD writing is achievable without coding experience. Explaining your service idea and answering AI's questions belong to business sense. The same process works on Cowork's desktop interface, so readers unfamiliar with terminal should start with Cowork.

F If you run into these problems

(1) If the PRD feels too generic and misaligned with your project, it is because you skipped the clarification step and jumped straight to the PRD. Ask AI to pose questions first, answer them specifically, then have it generate the document. Quality improves.

(2) If the generated database schema is missing required tables, specify the gaps explicitly: "Add Audit Log, Notification, and Attachment tables."

3 Code Migration and Automated Testing

A The invisible debt called technical debt

Technical debt is the invisible cost accumulated by old code. Each time you add a feature, bugs surface in unexpected places. New developers need weeks just to understand the codebase. Applying security patches hits dependency snags. So companies periodically run code migrations, swapping one framework for another.

It is among the tasks development teams dread most. You are not building something new; you are changing internal structure while keeping existing features intact. Users see no visible change, yet time and cost investment is enormous.

B How Claude Code helps with migration

Claude Opus 4.6's one-million-token context window lets you analyze entire codebases of hundreds of thousands of lines at once. Older AI tools had to break code into small chunks, so inter-file dependencies were easy to miss. Now you can build transformation plans while holding the full architecture in view.

Migration typically follows four stages. First, identify the boundaries of your current code. Separate routes, middleware, authentication, error handling, database access, and external API calls. Second, freeze the current API structure in OpenAPI or JSON Schema. Third, reorganize to fit the new framework's structure. Fourth, run the old and new versions side by side and verify behavioral differences.

C Migrating from Express to Fastify

(1) Basic usage: Creating a migration analysis report □ Connect your existing Express project folder to Claude Code. □ Give the following instruction.

"Analyze this entire project. Find all routes using the Express app.get and app.post patterns, map the middleware chains, and catalog where database queries execute. Don't modify the code; instead, create a migration analysis report that assesses the risk level of converting to Fastify on a scale of 1 (low) to 5 (high)."

□ Claude scans the project and generates a report analyzing the cross-references among all files and identifying risk factors.

(2) Applied case: Executing the conversion step by step

□ Review the analysis report, then begin the conversion starting with modules that have low risk.

"Convert the routes rated 1 to 2 for risk first to the Fastify format. Replace Express's req and res objects with Fastify's request and reply objects, and add JSON Schema-based request validation to each route."

□ Claude converts those files. Rather than applying mechanical substitution, it reflects the structural differences between Express's middleware approach and Fastify's plugin approach.

(3) Real-world use: Auto-generating and validating tests

Verifying that migrated code works correctly is just as important as moving the code itself.

□ Give the following instruction. □ "Generate unit tests for each converted Fastify route. Include scenarios for normal requests, invalid parameters, authentication failures, missing resources, and server errors. Use Fastify's inject method so tests can run without opening a network port."

□ Claude generates test code for each endpoint.

□ Run the generated tests to check if they pass. "Run all the tests you just created."

□ Claude runs the tests; if any fail, it analyzes the cause and fixes the code. This process repeats until all tests pass.

If end-to-end (E2E) testing is needed, integrate Playwright (a browser automation tool). "Using Playwright, create E2E test code that tests the entire user flow of logging in, searching for products, adding items to the cart, and completing checkout."

The Limitations of AI-Generated Tests

AI-generated tests verify the code's current behavior as-is. If the current behavior itself contains a bug, a test that treats that bug as normal can be created. For this reason, human review of AI-generated test code is necessary. The process of confirming whether "the behavior this test validates matches the behavior we intended" cannot be automated.

Logic left by past developers that appears irrational may actually have been a workaround for a bug in a specific operational environment. AI, unaware of this hidden history, can refactor the code away. Determining what must be preserved during migration is a human responsibility.

When These Problems Occur

(1) If a "Cannot find module" error repeats during migration, it means a dependency package is incompatible with the new framework. Instruct Claude: "Analyze the dependency list in package.json and find packages that are incompatible with Fastify."

(2) When all tests pass but errors occur in the live service, it means there are scenarios the tests don't cover. You must reflect real data patterns from the operational environment or conditions like network latency in your tests.

4 UI/UX Web Design and Figma Synchronization

The Old Divide Between Designers and Developers

When a designer carefully crafts a design mockup in Figma (a web-based design tool) and a developer implements it in code, subtle differences emerge. Margins are off by four pixels, font weight is applied as bold instead of semibold, and shadow opacity differs. The process of handing over a design mockup to a developer is called handoff, and this phase has been one of the greatest friction points in product development.

AI is reducing friction in this handoff. It approaches this from three directions.

What Is Figma MCP Integration?

MCP (Model Context Protocol) is a communication standard that allows AI to exchange data with external tools. With Figma providing an MCP server, Claude Code can now directly read and write design information from the Figma canvas. The "Code to Canvas" feature, released on February 17, 2026, enables bidirectional synchronization.

The Figma MCP server doesn't just see the pixels in a screenshot. It reads structural information such as component hierarchy, color codes, font sizes, spacing values, and design tokens (values that define the basic units of design).

Connecting and Using Figma with Claude Code

(1) Basic usage: Setting up the Figma MCP server

- ☐ Update the Figma desktop app to the latest version.
- ☐ Enter the following command in the Claude Code terminal.

```
claude mcp add --transport http figma https://mcp.figma.com/mcp
```

- ❑ Enter the `/mcp` command in Claude Code to verify that the Figma server is connected.
- ❑ When Figma requests authentication, click "Allow Access."

(2) Applied case: Converting a Figma design to code

- ❑ Select the frame (a screen unit) in Figma that you want to convert.
- ❑ Paste the Figma link into Claude Code and give the following instruction.

"Analyze this Figma design and convert it to a React component. Use Tailwind CSS and apply responsive breakpoints. Preserve the exact color codes and font sizes defined in the design."

- ❑ Claude reads the design's structural information through the Figma MCP and generates React component code.
- ❑ Run the generated code on a local server and compare it with the design mockup.

(3) Real-world use: Sending code back to Figma in reverse

This feature is the key to achieving bidirectional synchronization. Developers can send the UI they built in code to the Figma canvas, allowing designers to receive it as an editable frame.

- ❑ With a local server running in Claude Code, issue the following instruction:

"Start my app's local server, capture the current screen, and send it to a new Figma file."

- ❑ Claude opens a browser window and displays the capture toolbar. You can capture the entire page, specific elements, or multiple screens consecutively.
- ❑ Once capture is complete, Claude provides a Figma file link. When you open this link, the captured screen appears as an editable Figma frame.
- ❑ When a designer modifies colors or spacing in Figma, Claude Code can detect those changes and apply them to the code. "Check for changes to this frame in Figma, and apply only the modified parts to the code."

D. Constraints to Keep in Mind

Some features of the Figma MCP server operate only under specific conditions. The Code to Canvas feature is currently supported only in Claude Code, OpenAI Codex, and VS Code. Figma also requires a Dev or Full plan with Organization or Enterprise tier. API calls are subject to rate limits, so converting large numbers of frames at once may slow performance.

While the visual accuracy of UI code generated by AI is improving, it does not automatically meet accessibility standards. Details such as screen reader compatibility, keyboard navigation, and color contrast ratios must be validated separately by humans. Interaction logic such as drag and drop,

infinite scroll, and complex form validation shows uneven quality in code produced by AI.

AI can quickly produce the visual appearance of a design. But the responsiveness felt at the user's fingertips, the cadence with which the screen appears when scrolling, and the tone conveyed by error messages remain areas that still require human refinement.

E. Practical Implementation in Three Stages

Trying to implement all features at once will exhaust your team. In the first stage, learn the basic workflow of converting design mockups into code. In the second stage, connect the Figma MCP to supply AI with design context (colors, spacing, component hierarchy). In the third stage, set up Code Connect (which links Figma design components with actual components in your code repository's

tool), so that instead of AI inventing new components, your team uses components it already has. Moving methodically upward is ultimately faster.

F. If You Encounter These Issues

(1) If you have connected the Figma MCP server but the `generate_figma_design` tool does not appear, a conflict may occur if a previously connected Figma MCP instance remains. Disconnect the existing connection, restart Claude Code, and reconnect. Installing through the Figma plugin is more stable.

claude plugin install figma@claude-plugins-official

(2) If the result of converting from Figma to code differs significantly from the original, working from a screenshot alone produces different results than reading the structure through MCP. Verify that MCP is properly connected using the `/mcp` command, and explicitly state in the prompt: "Fetch design context from Figma MCP and perform the conversion."

(3) If the Code to Canvas feature does not work, this feature is supported only through the remote MCP server (<https://mcp.figma.com/mcp>). If you have connected only the local MCP server (the Dev Mode MCP Server in the desktop app), the feature to send code to Figma will not work. Connect the remote server in addition.

The four areas covered in this chapter are independent features, but in actual projects they form a single workflow.

Requirements are defined in a PRD, the agent team implements the code, automated testing validates quality, and Figma sync closes the gap between design and code. AI codes faster than humans. But the decision of what to build and judgment about whether the result aligns with business context remain the responsibility of the person at the screen. Placing our organization's unique judgment and experience atop the structure the machine has built. That is the right way to use these tools.

← Previous Chapter

See every chapter

Next Chapter →

Attorney · Former Member of the National Assembly · AI Policy Researcher

kimkj.com

© 2026 Kim Kyung-jin. All rights reserved.

#KimKyungjin #AIBLibrary #ClaudeCoworkandAgentManual

Chapter 8: Advanced Systems and Tool Integration Automation

← Previous Chapter

See every chapter

Next Chapter →

This chapter serves as a capstone project that weaves together the features learned from chapters 1 through 7.

It covers four expansion features: Skills, Schedules, MCP Connectors, and Plugins. When these four work in concert, Cowork transforms from a conversation partner into an autonomous digital colleague.

If you have grown accustomed to the method of issuing a single task and receiving results in prior chapters, this chapter builds a structure that eliminates the need for instructions themselves. Once designed, you get consistent results without repeating yourself each time, tasks execute automatically at set times, and data from multiple apps flows together in one unified stream.

However, the features in this chapter are areas where Cowork is evolving rapidly while in Research Preview, a testing phase before official release. The information presented here is based on details as of March 2026, but menu locations and specific options may change with updates. I recommend making a habit of checking the official Anthropic support page (support.claude.com) for the latest information.

1 Learning Your Own Standard Operating Procedures Through Skills

A Why You Need It: The Fatigue of Repeating the Same Instructions

Assume that every time you write a blog post for your company, you must follow these rules: keep the title to 40 characters or fewer, place the customer's pain point in the first paragraph, avoid directly naming competitors in the body, and include a "Get Started Now" button text at the end. What would happen if you had to deliver these rules to Cowork as a prompt each time? Some days you would forget to mention the competitor restriction, and other days you would word the call-to-action differently. The quality of the output would fluctuate based on how completely you remembered the prompt that day.

The same issue arises in code reviews. A senior developer has certain items they always check: whether error handling is missing, whether environment variables are hardcoded into the code, whether test coverage meets standards. Putting this checklist into a prompt every time is tedious, and leaving something out degrades review quality. Skills are a mechanism that puts an end to this repetitive explanation once and for all.

B What Is a Skill?

A Skill is a text file in Markdown (.md) format.

When you organize task procedures, tone and manner (the mood and speaking style of text), forbidden phrases, preferred formats, and reference examples into a single file and save it, Claude automatically reads that file and follows the guidelines whenever receiving a related request. You no longer have to give the same instructions every time.

Technically speaking, Skill files are stored in a local folder on your computer. Cowork usually reads only the summary information at the top of a Skill file, but loads and executes the entire file only when you issue a task that triggers it. Because of this "read only when needed" structure, performance remains fast even when you have dozens of Skills saved.

[Note] What is a Markdown file? Markdown (.md) is a lightweight format for writing formatted text. You can open it with a text editor, and it uses symbols to represent formatting like headings, lists, and emphasis. You can write it without any coding knowledge. You will use this format when editing Skill files directly.

C Creating a Skill: Basic Use

Follow along:

1. Open the Claude Desktop app and click the "Cowork" tab on the left side.
2. Start a new task and give an instruction as you normally would. For example, type the following:

"Write a blog post for our company. The topic is 'Inventory Management Tips for Small Business Owners'. Keep the title to 40 characters or fewer, and put the customer's frustration in the first paragraph. Don't mention competitor names directly. Unify the call-to-action text in the last paragraph to 'Start Free Today'."

3. Claude generates the output. If you are satisfied with it, type the following in the chat:

"Based on the rules I just gave you, create a 'Blog Writing Skill' that writes blog posts applying the same rules to any topic I give you in the future."

4. Claude automatically creates a Skill file (skill.md). The skill's name, objective, execution steps, and precautions are displayed on screen.

5. Click the "Copy to your skills" button. The Skill is saved to your local folder.

From now on, if you simply type "Write this week's blog post. The topic is 'How to Reduce Delivery Packaging Costs'" in a new conversation, Claude will automatically invoke the Blog Writing Skill and produce a post with all rules applied, title length, first paragraph structure, and call-to-action text.

D Using Skills: Practical Examples

Skills are not limited to writing. They apply to any recurring work with repeating rules.

Let's take email tone as an example. Gather 20 to 30 emails you have sent in the past into a single folder. Have Cowork open that folder, then instruct it like this:

"Analyze these emails, extract my unique tone of voice, greeting style, sentence length, and frequent expressions, then create 'My Email Tone Skill' that applies this style whenever drafting emails in the future."

Claude analyzes the patterns across these emails and organizes them into a Skill file. Later, when you simply request "Draft an email to vendor rep Kim about delivery schedule changes," you get an email with your natural tone, not the stiff style typical of AI.

You can also create an expense receipt processing Skill. If you save the rules for classifying receipt photos as a Skill (date format as YYYY-MM-DD, categories as meals/transport/office supplies/other, and flag items over 100,000 won), you can simply designate the receipt folder and say "Make the expense report" to get a completed Excel file.

E Using Skills: Real-World Application

The true power of Skills lies in converting organizational knowledge into files.

When expertise that existed only in a veteran employee's mind becomes a Skill file, the team's work quality remains stable even if that employee takes leave or departs. When a newly hired employee runs a task with that Skill applied in Cowork, they can produce results that meet team standards from day one.

When work methods change, simply edit the .md file. If brand color codes change, update that line. If new regulations add a legal review item, add one line to the Skill. You do not need to memorize a new prompt or reshare it with team members. Editing one file immediately applies the change to every task using that Skill.

Remember one principle when designing Skills: Claude does not infer implicit rules that are not written in the file. Instead of vague instructions like "be kind," write specific ones: "For non-refundable cases, first acknowledge the customer's concern, state the policy basis in one sentence, and offer an alternative in the final paragraph." Abstract Skills produce abstract results. Concrete Skills guarantee consistent quality.

F Troubleshooting

(1) If a Skill is saved but does not apply in new conversations: Confirm that the Skill file was saved in the correct location. Open the Skills list under Settings > Customize and check whether that Skill is enabled.

(2) If a Skill applies but some rules are ignored: The Skill file may be too long or rules may conflict with each other. Separate "must-follow rules" from "background information for reference," and place core rules at the beginning of the Skill file, keeping them short and clear.

(3) If you modify a Skill but the previous version continues to apply: Close the Claude Desktop app completely and reopen it. Changes to Skill files take effect when the app restarts.

2 24-Hour Background Automation Based on Schedules

A Why You Need It: Daily Preparation Tasks That Repeat Each Morning

Monday morning at 9 a.m. When you arrive at the office and turn on your computer, there is something you do first. You scan through unread emails and filter out the important ones, check today's and tomorrow's schedule, search for news in your area of interest, and identify items worth sharing with your team. This takes 30 minutes to an hour. It is not time spent analyzing content and making decisions, but time spent gathering the material needed to make those decisions.

Friday afternoon follows a similar pattern. You organize files piling up in your Downloads folder, transfer this week's expense receipts to the expense report spreadsheet, and prepare next week's meeting agenda. During busy or exhausting weeks, you skip this task, and a month later you find 200 files piled up in your Downloads folder.

The Schedule (Scheduled Tasks) feature automatically executes these repetitive tasks at a set time. Whether the user is sleeping or in a meeting, Claude wakes up on its own and handles the work.

What Is a Schedule?

A schedule is a reservation command that says "execute this task, at this time, in this cycle." You can set the desired cycle, hourly, daily, weekdays only, weekly, and so on, and specify the execution time precisely. When the scheduled time arrives, Cowork opens a work session on its own, accesses the designated folder, and performs the task using pre-saved skills and connectors.

There is one prerequisite. Schedule tasks work only when the user's computer is on and the Claude Desktop app is running in the background. If you shut down the app completely or the computer enters sleep mode, scheduled tasks will not execute.

Setting Up a Schedule: Basic Usage

Follow Along

1. Open the Cowork tab in the Claude Desktop app.
2. In the left sidebar, click "Scheduled."
3. Click the "New Task" button. Alternatively, you can type `/schedule` in an existing Cowork conversation window.
4. Enter the task content specifically. For example, write like this: "Create a file called `Morning_Briefing.md` in the 'Work' folder on my desktop every morning at 8 a.m. Structure the content like this. First, today's date and day of the week. Second, the titles and one-line summaries of three major AI-related news articles from web search. Third, today's to-do checklist template (five blank

lines)."

5. Claude organizes and displays the task name, execution cycle, and task content. After confirming, click the "Schedule" button.

6. The newly created task is added to the Scheduled list in the left sidebar. If the computer is on and the app is running at 8 a.m. the next morning, the briefing file will be automatically generated in the desktop's 'Work' folder.

Using Schedules: Practical Examples

By changing the cycle and task content, various types of automation are possible.

Example of automatic file organization: "Scan the downloads folder every Friday at 5 p.m., move image files to the 'Photos' folder, PDFs to the 'Documents' folder, and the rest to the 'Other' folder. Save the list of moved files as 'Organization_Result_Date.txt' on the desktop."

Example of competitor monitoring: "Every Monday morning at 7 a.m., search the web for the past week's news about three companies, Coupang, Naver Shopping, and Kakao Commerce, and if there are any new service launches or policy changes, summarize them and save as a 'Competition_Weekly.md' file."

You can also combine skills created earlier with scheduled tasks. "When creating a summary of emails received every evening at 10 p.m., apply my Email Tone skill and attach a draft to emails that need a reply." Skills handle quality, and schedules handle timing.

Using Schedules: Real-World Application

There is something to remember about schedule automation. Auto-generated results are first drafts, not final reports.

News collected by AI in the early morning may contain errors and can miss context. In a competitor monitoring report, information stating "Company A launched a new service" might actually be from an article that is merely rumor. Automation prepares materials needed for decision-making in advance; it does not replace decision-making itself.

The most effective pattern in practice works like this. Schedule collects information in the early morning and does the first pass of organization. When the user arrives at work in the morning, they review the results in under ten minutes and adjust only the parts that need it. Preparation work that took one hour drops to ten minutes. Complete automation is not realistic; semi-automation that combines human review with AI preparation is the practical goal.

If You Encounter These Problems

(1) If a task does not run at the scheduled time: Check whether the Claude Desktop app is running and whether the computer has entered sleep mode. On Windows, adjust sleep settings at Settings >

System > Power.

(2) If a scheduled task stops mid-execution: The network connection may have been lost, or you may not have access permission to the target folder. Click the task in the Scheduled list to view the execution log.

(3) If a "Usage limit reached" message appears: Schedule tasks are included in your general Cowork usage. Pro plan users may reach the limit quickly. You can set the execution model for scheduled tasks to

Sonnet 4.6 to save usage compared to Opus 4.6. Reducing the execution frequency from daily to weekdays is also an option.

3. Controlling External Apps Through MCP Connectors

Why Is It Needed: Time Spent Switching Between Tools

Consider a project manager preparing a weekly report. They check progress from each team in Notion, search for related documents in Google Drive, search for issues discussed this week in Slack, and compile all of this into a single report. Manually combining information by switching between four tools takes over one hour. That is time spent gathering information, not writing the report.

As business tools multiply, this time spent switching windows grows longer. Each tool has its own interface and search method, so data from the same project remains disconnected. You copy from app A and paste into app B dozens of times a day.

An MCP (Model Context Protocol, the specification for AI to exchange data with external apps or services) connector is a bridge that breaks through this disconnection. You can think of it as Claude's hand reaching out to other apps.

What Is an MCP Connector?

An MCP connector is a connection device that allows Claude to access external applications directly, read data, search, and in some cases write or edit content. Once a connector is linked, users need not open a separate browser tab; they can simply say "Get this week's meeting notes from Notion" within Cowork and receive the data.

As of March 2026, Anthropic officially supports roughly ten connectors including Google Drive, Gmail, Google Calendar, Slack, Notion, GitHub, Jira, Linear, Asana, DocuSign, and FactSet, with more being added. These official connectors connect instantly after a few clicks of authentication (OAuth, a process where users grant app access).

If you use a tool not on the official list, you can connect it using Cowork's custom connector feature if the service provides an MCP server address. Connecting Zapier MCP also makes it possible to access over 8,000 external apps.

First Connector Connection Tutorial: Basic Usage

For those setting up an MCP connector for the first time, I will guide you through the entire process of connecting a Google Drive connector.

Follow Along

1. In the Claude Desktop app, click the Settings icon at the bottom left.
2. Select the "Customize" menu and click the "Connectors" tab.
3. Find and click "Google Drive" in the connectors list.
4. Click the "Connect" button to open a browser window and display the Google account login screen. Log in with your Google account and allow Claude to access Drive.

- ☐ Once authentication is complete, the connector status changes to 'Connected.'
- ☐ Return to the Cowork tab and start a new task. Enter the following: 'Find the file titled 2024 Fourth Quarter Performance Report in Google Drive and summarize the key content in five lines.'
- ☐ Claude accesses Google Drive, searches for the file, reads its content, and provides a summary.

D. Connector Use: Application Examples

When a single connector is linked, it functions as a convenient search tool. When multiple connectors are connected simultaneously, it becomes an automated workflow.

Let's revisit the weekly report preparation example mentioned earlier. With Notion, Slack, and Google Drive connectors all connected, you give this instruction:

'Collect progress updates from each team's Notion project board this week, summarize the issues discussed this week in Slack's #project-alpha channel, extract the list of related documents from Google Drive, and create a draft weekly report. Save the report as a Word file on the desktop.'

Data from three apps is collected in a single conversation, and an organized report is created. A task that took one hour now finishes with a single instruction.

With the Gmail connector linked, email tasks can also be handled within Cowork. When you request 'Find and summarize emails about contracts from Manager Kim last week,' Claude searches Gmail and organizes the essentials. This can extend to 'Draft your review comments and create a reply draft.' Reading the email, analyzing it, and writing a response connect as a single flow.

E. Connector Use: Practical Application and Security Considerations

The convenience of connectors comes with the weight of security. Since you are giving Claude access keys to company email, documents, and work channels, it is wise to establish some

principles.

First, grant permissions narrowly. When only read access is needed, there is no reason to grant write access. Start connector settings as read-only, and add write permissions only when absolutely necessary.

Second, when handling sensitive data, first review organizational policy. Cowork is currently in research preview status, and audit logs or data export functions do not capture Cowork activity. Anthropic's official documentation also recommends against using Cowork for regulated workloads.

Third, do not forget that automation can make mistakes. It is possible for Claude to send messages incorrectly to Slack or enter wrong data into Notion. For important write operations, you must include a human approval step with an approach like 'Create a draft and show it to me; I will send it after reviewing.'

F. If These Problems Occur

(1) If you see 'Authentication failed' when connecting a connector: Check whether the account already logged in to the service (Google, Slack, etc.) in your browser is the one you want. Authenticating while logged in as a different account can connect the wrong account. You can avoid this problem by attempting authentication in an incognito mode browser.

(2) If the connector is connected but 'No results found' appears: The search scope might be too broad or the keyword might not be exact. Resubmit the request with a specific filename or keyword, such as 'Fourth Quarter Report.' With Google Drive, Shared Drive files may have different search scope settings, so verify that the scope is 'My Drive.'

(3) If external app responses are slow or timeouts occur: External services like Notion and Slack have rate limits. Requesting too much data at once can cause the service to refuse a response. Narrow the request scope to something like 'Last 7 days' and try again.

4. Implementing Department-Specific Custom Plugins

A. Why It Is Needed: The Effort of Setting Up Skills and Connectors Separately

You have learned how to create and configure skills, schedules, and connectors separately. A marketing team must set up content writing skills, brand guideline skills, Notion connectors, Slack connectors, and weekly posting calendar schedules separately. Someone familiar with IT can accomplish this, but having ten team members each replicate the same setup is a different scale of problem.

A plugin bundles these pieces into a single package. Skills, connectors, slash commands (shortcut commands starting with the / symbol), and sub-agents (auxiliary AI that shares complex tasks) are packaged together, so a single installation sets up the entire AI environment needed for a specific role.

B. What Is a Plugin

Plugins fall into two main categories.

One is the official plugin provided by Anthropic by default. On January 30, 2026, Anthropic Labs released 11 open-source plugins for Sales, Legal, Finance, Marketing, Data Analysis, Productivity, Product Management, and Software Development, among others. In February, 15 additional plugins for Design, Operations, HR, and others were added. These plugins can be viewed at claude.com/plugins and installed directly from Cowork.

The other is a custom plugin created by the user. After installing a standard plugin, you can click the 'Customize' button to modify it to suit your company, or you can build a new plugin from scratch.

C. Installing Plugins: Basic Usage

Follow along: ☐ Go to the Cowork tab in the Claude Desktop app.

☐ Click the 'Plugins' menu in the lower left.

☐ A list of plugins appears. Select one from the plugins categorized by role, for example, 'Marketing.'

☐ On the plugin detail screen, verify the list of included skills, slash commands, and connectors, then click the 'Install' button.

☐ Once installation is complete, new commands appear when you type / in the Cowork conversation box. You can use commands like `/brand-review`, `/campaign-plan`, and `/draft-content`.

☐ Execute the command. For example, when you enter `/draft-content`, Claude presents a form and asks 'Which channel is this for? (Blog/SNS/Email)' and 'What is the topic?' Fill in your answers, and content is generated with the skills built into the plugin applied.

D. Plugin Use: Application Examples and Customization

Standard plugins are designed for general use and may not fit your company's situation exactly. In that case, you customize.

Follow along: ☐ Click the 'Customize' button on the detail screen of the installed plugin.

☐ Claude asks questions. 'What CRM does your company use?' 'Who is your primary customer base?' 'What is your brand tone?' Answer these questions, and the skill files within the plugin are modified to suit your company.

☐ The customized plugin can be shared with team members. For Enterprise plans, administrators can create a private plugin marketplace for organization-wide distribution.

Let me give an example of customizing a legal plugin. The standard legal plugin contains a general contract review checklist. To this, you add your company's risk tolerance criteria, frequently used

contract clause templates, and a list of toxic clauses that must be removed. Then, place partnership contracts in a folder and run the `/audit` command, and it identifies risky clauses according to your company's standards and delivers a review result with risk levels color-coded.

[Note] What is a slash command? When you type a command after the `/` symbol in the Cowork conversation box, a specific task runs immediately. It is similar to app shortcuts on a smartphone. `/schedule` creates a scheduling task, and `/brand-review` starts a brand review. When you install a plugin, the slash commands included in that plugin are automatically added.

E. Plugin Use: Practical Application and Organization-Wide Expansion

The true value of a plugin appears at the organizational level, not the individual level.

Consider a structure where ten departments each have their own plugins, but all plugins share the same company brand skills. Whether marketing uses it, legal uses it, or HR uses it, the tone and format of documents remain consistent. Individual department-specific rules are handled by each department's plugin, while company-wide consistency is guaranteed by shared skills.

Let me give an example using a new employee's first day. A new hire joining the marketing team opens Claude Cowork and activates the marketing plugin. Without any separate training, running the `/draft-content` command produces content with the team's brand guidelines, tone and manner, and CTA rules all applied. Instead of making them read dozens of pages of manuals, a single plugin does that job.

Plugin updates are also managed centrally. When the law changes and you update the legal plugin's checklist, that change is immediately reflected to all employees using the legal plugin. You don't need to notify 100 people individually that 'from now on, do it this way.'

However, you should not accept plugin-generated results as final approval. Just because a legal plugin reviewed a contract doesn't mean you can skip the lawyer's judgment, and just because an HR plugin evaluated a resume doesn't mean it can replace an interviewer's eye. A plugin's role is to handle repetitive first-pass work; final judgment and decision-making belong to people. In a corporate environment, clarity on this boundary between AI adoption and human judgment is the difference between success and failure.

In the Enterprise plan, administrators can control which plugins are deployed to which departments, which connectors are allowed, and what data can be accessed. As of the February 24, 2026 update, a private plugin marketplace, per-user automatic installation, and admin dashboard features have been added.

If these issues arise

(1) If slash commands don't appear after installing a plugin: Exit the Claude Desktop app and restart it. Check that the plugin shows as "Active" in your plugin list.

(2) If your customizations don't apply to other team members: Personal customizations only apply to you. To roll out to the entire team, the Enterprise administrator can deploy through the plugin marketplace, or you can export the customized plugin as a ZIP file and have team members each upload it themselves.

(3) If plugin skills and personally-created skills conflict: When a plugin skill and a personal skill have different rules for the same type of task, Claude can become confused. Check the priority in Settings > Customize, and disable one of the overlapping skills.

The four functions covered in this chapter, skills, schedules, MCP connectors, and plugins, are useful independently, but their combined power shows the full potential of Claude Cowork. Skills manage quality, schedules manage time, connectors link tools, and plugins tie it all together. The journey that started in Chapter 1 with organizing a Downloads folder has now arrived at designing workflows for an entire organization.

You don't need to implement all features at once. Create one skill and use it for a week, set up one schedule and check the results each morning, connect one connector and experience the convenience of search, and then naturally move to bundling with plugins. Just start with what you need right now.

[← Previous Chapter](#)

[See every chapter](#)

[Next Chapter →](#)

Attorney · Former Member of the National Assembly · AI Policy Researcher

kimkj.com

© 2026 Kim Kyung-jin. All rights reserved.

[#KimKyungjin](#) [#AIBLibrary](#) [#ClaudeCoworkandAgentManual](#)

Epilogue

← Previous Chapter

See every chapter

On the day I was finishing the manuscript for this book, I actually said this to Cowork:

"Open the Word files for the 8 chapters in this manuscript folder, count the character count for each chapter and the number of screenshot references, and organize the results into a single table."

Thirty seconds later, a table appeared on the screen. It was neatly organized with character counts by chapter, screenshot reference frequency, and the number of prompt examples. As I reviewed the table, I identified two chapters that needed more content and completed the revision work that afternoon.

Writing this book was itself the process of using Cowork. I asked it to extract key sentences from reference PDF documents and tested prompt examples directly with Cowork to see if they actually worked. When errors occurred, I captured the error screen and included it in the "If This Problem Occurs" section. Every procedure I guide readers through, I experienced first.

One thing became clear through that process. Cowork is not infallible. It sometimes enters Excel formulas incorrectly and occasionally omits file extensions when renaming files. When tasked with contract review, it sometimes skips a clause. That is why I marked in nearly every chapter of this book the points where "a person must verify the results." Delegate to AI, but the eye that validates the outcome must be human.

And another thing became equally clear. The need for verification does not diminish the value of delegation. Someone who once spent half an afternoon classifying two hundred files in a Downloads folder now writes proposals in that time. Someone who once edited emails one by one for fifty business partners now meets with clients. The time that Cowork returns is not abstract "productivity improvement." It is the margin to drink a cup of coffee at three in the afternoon on a Tuesday.

If you installed Cowork for the first time in Chapter 1 of this book and had it organize your folders, you have already completed your first delegation. If you have read through Chapter 8, you are ready to design automated workflows yourself. Now close this book, sit down at your computer, and delegate to Cowork the very first task you want to do when you arrive at work tomorrow morning.

It may work well, or it may produce an error. If an error occurs, you can open the corresponding chapter in this book again. After correcting it once or twice that way, by the third attempt you won't need the book.

That is when this book's purpose comes to an end.

Spring 2026 Kim Kyung-jin

← Previous Chapter

See every chapter

Attorney · Former Member of the National Assembly · AI Policy Researcher

kimkj.com

© 2026 Kim Kyung-jin. All rights reserved.

#KimKyungjin #AIBLibrary #ClaudeCoworkandAgentManual

Support This AI Library Book

If this book was useful, you can support future translations, public editions, and open AI knowledge projects through PayPal.



PayPal

<https://paypal.me/kimkjkj>

Scan the QR code or open the link above.