

The skill and plugin ecosystem in Claude Cowork

Learn about skills, plugins, slash commands, and sub-agents—what they are, how they differ, and which one to use in each situation.

You've used Cowork to handle your tasks. But what if Claude already knows your company's writing style? What if it understands your data formatting, report templates, and approval processes—before you even say a word?

That's what skills and plugins do. They turn your expertise into reusable packages, making Cowork significantly smarter for your specific tasks.

Anthropic's official plugin library launched with 15 plugins, encompassing over 85 skills. The community marketplace already has thousands of other plugins. And best of all? Building your own plugin only takes minutes, not hours. It's simply Markdown .

This series will guide you from using other people's plugins to building your own plugins – and publishing them for others to use.

What you will learn

1. Explain the difference between skill, plugin, slash command, and sub-agent.
2. Use the pre-built plugins from the official Anthropic library and customize them for your role.
3. Build a custom skill from scratch using the SKILL.md file and YAML frontmatter.
4. Implement MCP connectors to link Cowork with external APIs and data sources.
5. Evaluate OpenClaw extensions for security and quality before installing them.
6. Design multi-skill workflows that connect operations to automate processes from start to finish.

After this course, you will be able to

1. Build custom Cowork skills from scratch using the SKILL.md format, which codes out your team's specific workflows.
2. Connect Cowork to external APIs and data sources via MCP connectors for end-to-end automation.
3. Evaluate OpenClaw extensions for security and quality before installing them in your environment.
4. Design multi-skill workflows that connect multiple tasks into complete business automation processes.
5. Publish reusable plugins to the Cowork ecosystem, establishing your professional reputation as a skill builder.

You will be able to build:

1. **Custom collaborative skill and workflow** : A custom skill has been published with MCP integration, automating a real-world business process—including YAML frontmatter, parameterized input, error handling, and chain operations.
2. **Plugin security assessment framework** : A structured assessment template for testing third-party skills and plugins – including permission scope, data access, credential handling, and trust verification.
3. **Ability to create collaborative skills and plugins** : Demonstrate your ability to build custom skills, integrate MCP servers, evaluate extensions, and design multi-skill automation workflows.

The skill and plugin ecosystem in Claude Cowork

Learn about skills, plugins, slash commands, and sub-agents—what they are, how they differ, and which one to use in each situation.

Before building anything, you need to understand the building blocks. The Cowork ecosystem has four components, and mixing them up would be a waste of time.

After completing this lesson, you will be able to:

1. Name four components: Skill, plugin, slash command, sub-agent
2. Explain when each ingredient should be used.
3. Navigate the official Anthropic plugin library.

You should master the basics of Cowork before you begin. If you're unfamiliar with terms like "connector" or "context window," take the Claude Cowork Introduction series first.

4 components

Zinc

Skill is Claude's expertise, encoded in a file. It's a markdown file named SKILL.md, which tells Claude how to perform a specific task.

For example, the "contract review" skill could tell Claude:

1. What to look for in the contract (liability clauses, automatic renewal, termination clauses)
2. Output format of the review
3. These questions need to be marked for legal review.

When you provide Cowork with a contract for review, it reads the skill and automatically applies that expertise. A skill is the smallest unit. Each file, one capability. No coding required.

Plugin

Plugins bring everything together. Think of it as a toolkit for a specific role or area.

A "Marketing" plugin might include:

1. 6 skills (campaign analysis, content scheduling, competitor research, etc.)

2. 3 slash commands (/new-campaign, /analyze-metrics, /content-brief)
3. 2 MCP connectors (Google Analytics, HubSpot)
4. 1 sub-agent for multi-step campaign planning

Plugins are what you install and share. Skills are what make them smart.

Slash command

Slash commands are explicit triggers. You type them /tên-lệnh and they run. Unlike skills, which Claude automatically applies when needed, commands are only executed when you call them.

Suitable for: Repetitive actions that you want to trigger on demand /weekly-report./clean-data /prep-meeting

Sub-agent

Sub-agents are specialized agents within a plugin that handle complex sub-tasks. If a marketing plugin needs to analyze competitor data and generate campaign ideas, that might be two different sub-agents – each with its own skills and context.

Most users don't build their own sub-agents. They're pre-packaged in plugins.

Quick Check : What is the difference between a skill and a slash command?

Answer : The skill is applied automatically when needed; the command is activated manually by typing /name.

How they combine

Plugin ??? Skill (chuyên môn lĩnh vực ???c áp dụng t? ??ng) ??? Các lệnh slash (hành ??ng ???c kích hoạt th? công) ??? MCP connector (liên kết ??n các công c? bên ngoài) ??? Sub-agent (trình x? lý tác v? chuyên biệt)

Everything is file-based. Skills are markdown. Manifests are JSON. Commands are markdown with frontmatter. There are no build steps, no infrastructure, no deployment process.

That's the beauty of this system – a marketing manager who has never written code can build a plugin in an afternoon.

Anthropic's official plugins

Anthropic has released 15 open-source plugins on GitHub (anthropics/knowledge-work-plugins). Below are the plugins currently available:

Plugin	Field	Zinc	Key capabilities
Data Analysis	Analysis	6+	SQL querying, dashboard building, data cleaning
Design	Product design	5+	Design evaluation, user research compilation.
Engineering	Software development	8+	Code review, system architecture, testing
Finance	Financial activities	6+	Budget analysis, forecasting, and regulatory compliance.
Legal	Contract/Legal	5+	Review the contract, assess the risks.

Plugin	Field	Zinc	Key capabilities
Marketing	Marketing	7+	Campaign planning, content strategy, SEO
Fire	Business operations	5+	Process improvement, supplier management
People (HR)	Human Resources	6+	Job description, interview guide, new employee training process.
Product	Product Management	6+	Write PRD documentation, prioritizing features and user stories.
Sales	large	5+	Process analysis, proposal writing, CRM
Security	Information security	5+	Threat assessment, incident response
Strategy	Business strategy	5+	Market analysis, competitive information
Written	Content	6+	Blog posts, documents, style guides

They are all open source, all free, and all customizable.

When should you use what?

Situation	Use
You want Claude to always follow a specific format.	Zinc
You want to perform a repetitive task with just one click.	Slash Command
You need to set up a complete domain system (many skills + tools).	Plugin
You need to connect to an external API.	MPP Connector
You want to share your configuration with your teammates.	Plugins (shareable packages)

Key points to remember

1. Skill = an ability stored in the SKILL.md file, automatically detected by Claude.
2. Plugin = a collection of skills + commands + connectors + sub-agents
3. Slash command = manually activated, skill = automatically applied
4. Anthropic's 15 official plugins cover key business areas with over 85 skills.
5. It's all file-based - markdown and JSON, no coding required.

1. Question 1:

How many skills are included in Anthropic's 15 official knowledge-based job plugins in total?

1. A. 15 skills (one skill per plugin)
2. B. Approximately 30 skills
3. C. Over 500 skills
4. D. Over 85 skills for all plugins

EXPLAIN:

The 15 official plugins contain over 85 skills, over 69 slash commands, and over 40 MCP connectors. Each plugin covers a specific area (marketing, finance, engineering, etc.) with many specialized skills.

2. Question 2:

How did Claude discover and utilize these skills?

1. A. You have to load them manually each session - this seems logical at first glance, but research points in the opposite direction.
2. B. Claude automatically detects skills from SKILL.md files and applies them appropriately based on the frontmatter description.
3. C. They only work with slash commands.
4. D. They require an API key to activate.

EXPLAIN:

Skills use YAML frontmatter with a description field that tells Claude when to apply them. Claude reads this and automatically calls the skill when your mission is appropriate—no manual loading required.

3. Question 3:

What is the main difference between a skill and a plugin?

1. A. A skill is a single capability (markdown file); a plugin combines skills, connectors, commands, and sub-agents into a complete package.
2. B. Skills require payment, while plugins are free.
3. C. The plugin only works on Macs.
4. D. Skills require programming, while plugins don't – this would be great if it were true, but evidence suggests a more complex story.

EXPLAIN:

A skill is a focused ability defined in the SKILL.md file. A plugin is a complete package combining multiple skills, MCP connectors, slash commands, and sub-agents for a specific domain.

Submit your work

Training results

You have completed **0** questions.

-- / --

[Review the lesson](#)

You finished reading the article "**The skill and plugin ecosystem in Claude Cowork**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar

articles on tips and guides. Thank you for reading and for following us regularly.
