

Coding Nexus · [Follow publication](#)

★ Member-only story

The Claude Skills Cookbook: Anthropic's New Context Engine Outperforms MCP??

5 min read · 5 days ago



Code Coup

Follow



Listen



Share



More

Anthropic quietly released the **Claude Skills Cookbook** on GitHub. Initially, I assumed it was just another dull API document.

But after poking around for a few hours, I realized — this thing actually *alters* how Claude operates.

In short, **skills** are like providing Claude with a set of tools and some muscle memory.

They transform it from a chatty assistant into a practical worker that can actually *do* things — create spreadsheets, build PowerPoints, analyze data, even generate PDFs. And yeah, you can develop your **own** skills too.

Let me show you what I learned.



Image by — Anthropic

. . .

Okay, What Are “Skills” Exactly?

Think of a Skill as a small bundle of instructions that tells Claude how to behave in a particular context.

It’s part code, part playbook, part vibe.

Each Skill usually has three things:

- Some **instructions** (what Claude should do)
- A bit of **code** (so it can actually execute)
- Optional **resources** (like templates or datasets)

The idea is to give Claude *specific expertise*—similar to installing a small plugin. For example:

- “xlsx” skill → makes Excel sheets, charts, and pivot tables
- “pptx” skill → builds presentation slides
- “pdf” skill → formats and exports PDFs
- “docx” skill → writes Word docs

You don't need to load all of these at once — Skills only load when required. That's what Anthropic refers to as **progressive disclosure**, and it helps limit the model's token usage.

• • •

What's Inside the Cookbook

The Cookbook is divided into three main notebooks. Each one concentrates on a different part of the Skills workflow.

1. **Introduction to Skills** — a quick demo on how to set things up and generate your first Excel or PDF.
2. **Financial Applications** — this one's fun if you like dashboards, portfolio analysis, or anything data-heavy.
3. **Custom Skills Development** — this is where you learn to build your own Skill from scratch.

Here's how the folder structure looks once you clone it:

```
skills/
├── notebooks/
│   ├── 01_skills_introduction.ipynb
│   ├── 02_skills_financial_applications.ipynb
│   └── 03_skills_custom_development.ipynb
├── sample_data/
│   ├── financial_statements.csv
│   ├── portfolio_holdings.json
│   ├── budget_template.csv
│   └── quarterly_metrics.json
```

Each notebook guides you through a specific workflow — from creating Excel dashboards to producing polished PDF reports. And the sample datasets? They actually look pretty real.

• • •

Setting It Up (Not as Scary as It Sounds)

You'll need a few things:

- Python 3.8 or newer
- Anthropic API key
- Jupyter Notebook or JupyterLab

Then just run this in your terminal:

```
git clone https://github.com/anthropics/claude-cookbooks.git
cd claude-cookbooks/skills
python -m venv venv
source venv/bin/activate # Windows: venv\Scripts\activate
pip install -r requirements.txt
cp .env.example .env
# add your ANTHROPIC_API_KEY to .env
jupyter notebook
```

Once you're in Jupyter, open `01_skills_introduction.ipynb` and you're good to go.

. . .

A Closer Look Inside

Here's where it gets interesting.

Claude doesn't automatically "know" how to run code or download files — that's what the **beta headers** are for.

You essentially need to inform the API that it can use the Skills system and file features. Something like this:

```
from anthropic import Anthropic

client = Anthropic(
    api_key="your-api-key",
    default_headers={
        "anthropic-beta": "code-execution-2025-08-25,files-api-2025-04-14,skill"
```

```
}  
)
```

Those headers unlock:

- **Code execution** (so Claude can run Python safely)
- **Files API** (so it can create and download actual files)
- **Skills framework** (the new part)

Once those are established, Claude essentially turns into an automation engine that can operate across documents.

. . .

Example: Making an Excel File (With One Prompt)

Here's a demo I ran.

All I wanted was a basic budget spreadsheet.

```
from anthropic import Anthropic  
client = Anthropic(api_key="your-api-key")  
  
response = client.messages.create(  
    model="claude-sonnet-4-5-20250929",  
    max_tokens=4096,  
    container={  
        "skills": [  
            {"type": "anthropic", "skill_id": "xlsx", "version": "latest"}  
        ]  
    },  
    tools=[{"type": "code_execution_20250825", "name": "code_execution"}],  
    messages=[  
        {"role": "user",  
         "content": "Create an Excel file with a simple monthly budget spreadshe"}  
    ]  
)  
# extract file_id (you'll have a helper in the notebook)  
file_id = extract_file_id(response)  
# download it  
file_content = client.beta.files.download(file_id=file_id)  
with open("outputs/budget.xlsx", "wb") as f:
```

```
f.write(file_content.read())  
print("Done. Check outputs/budget.xlsx")
```

No pandas, no openpyxl, no fórmulas.

Claude simply writes out the entire Excel file and returns it. I ran it locally, and yes, it's a genuine spreadsheet with rows, totals, and even formatting.

• • •

Custom Skills: The Power Move

This is the part that caught my attention — you can *teach* Claude your own specialized workflow.

Let's say you want it to create branded financial reports every month.

You can package that logic into a Skill like this:

```
my_skill/  
├── SKILL.md  
├── scripts/  
│   └── processor.py  
└── resources/  
    └── template.xls
```

Inside `SKILL.md`, you describe what the Skill does:

- Input type (e.g., CSV, JSON)
- Output type (Excel, PDF, etc.)
- What logic to follow (e.g., apply company colours, calculate margins)

Next time you call Claude with that skill, it already knows what to do — no need to restate all your instructions.

It's like giving Claude a shortcut to your brain.

• • •

Some Handy Tips I Picked Up

- Don't load everything at once.

Use the progressive loading feature so Claude only grabs the skills it needs.

- Batch tasks.

You can have Claude process multiple datasets in a single run.

- Reuse containers.

When you create a session, reuse it instead of starting fresh every time. It saves time and tokens.

- Keep outputs clean.

Everything you generate goes into an `outputs/` folder. If you rerun a cell, it'll overwrite the file — which is fine, just something to note.

. . .

Little Gotchas

You'll probably hit one or two snags early on.

Here are the ones I ran into:

Problem	What It Means	How I F
-----	-----	-----
<code>`ValueError: ANTHROPIC_API_KEY not found`</code>	Forgot to update <code>`.env`</code>	Added <code>m</code>
<code>`Skills feature requires beta header`</code>	Missing the new header	Double-
<code>`Request exceeds token limit`</code>	Prompt too large	Broke <code>m</code>

Once those are sorted, everything runs smoothly.

. . .

Why This Actually Feels Different

If you've used GPTs, Plugins, or RAG-style workflows before, this is *not* the same.

Claude's Skills system seems more like providing the model with a **memory and a toolkit**.

It understands your goals and has the context to act on them — without constant prompting.

You could:

- Build a dashboard in Excel
 - Auto-generate a PowerPoint from it
 - Export the summary to a branded PDF
- All in one run.

That's not just a productivity trick — it's how AI gradually integrates into actual business workflows.

• • •

Final Thoughts

Honestly, this Skills system is the missing piece between **chatbots** and **real automation**.

Claude can finally *do* things with structure, not just talk about them.

The cookbook is a good starting point if you want to explore that world. It's well-documented, and the examples are ready to use—finance-heavy but easily adaptable.

If you're a developer, analyst, or someone who runs weekly reports, it's worth trying over the weekend.

You'll walk away thinking, *"Wait, did Claude just automate my job?"*

Claude

Mcp Server

AI

Ai Agent

Ai Tools



Follow

Published in Coding Nexus

8K followers · Last published 17 hours ago

Coding Nexus is a community of developers, tech enthusiasts, and aspiring coders. Whether you're exploring the depths of Python, diving into data science, mastering web development, or staying updated on the latest trends in AI, Coding Nexus has something for you.



Follow

Written by Code Coup

2.3K followers · 1 following

Code Coup: Seize the Code, Stage a Coup!


No responses yet



Bgerby

What are your thoughts?

More from Code Coup and Coding Nexus


 In Coding Nexus by Code Coup

10 n8n Tricks Beginners Wish They Knew Earlier

Everyone loves saying n8n is beginner-friendly. Honestly? Not really.

★ Sep 30 🖱️ 278 💬 6




 In Coding Nexus by Civil Learning

The Guy Who Let ChatGPT Trade for Him—and Somehow It Worked

You know how everyone says, “Don’t let AI touch your Money”? Well, someone on Reddit decided to ignore that.

✦ Oct 8 🖱 178 💬 8




 In Coding Nexus by Algo Insights

4 Open-Source Tools That Made Me Rethink My Dev Setup

I've been coding for a while now. Most of the tools we use every day... they've been the same for years. Editors, browsers, frameworks. Just...

✦ Sep 3 🖱 451 💬 9



 In Coding Nexus by Code Coup

Claude Desktop Might Be the Most Useful Free Tool You'll Install This Year

I didn't expect much when I first saw the announcement for Claude Desktop. Another AI wrapper, I thought. Maybe with a shiny UI.

★ 6d ago 🖱️ 246 💬 6



See all from Code Coup

See all from Coding Nexus

Recommended from Medium

☐ In Coding Nexus by Code Coup

Claude Desktop Might Be the Most Useful Free Tool You'll Install This Year

I didn't expect much when I first saw the announcement for Claude Desktop. Another AI wrapper, I thought. Maybe with a shiny UI.

★ 6d ago 🖱️ 246 💬 6



 Agen.cy

20+ Genius Ways Power Users Are Using Claude Code Right Now

Here are 10+ ways power users are using Claude Code 

Oct 23  29  1




 In Dare To Be Better by Max Petrusenko

Claude Skills: The \$3 Automation Secret That's Making Enterprise Teams Look Like Wizards

How a simple folder is replacing \$50K consultants and saving companies literal days of work

✦ Oct 17 🖱 398 💬 6

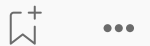


 In AI Software Engineer by Joe Njenga

This Viral DeepSeek OCR Model Is Changing How LLMs Work

This DeepSeek OCR model hit an overnight success not seen in any other release—4k+ GitHub stars in less than 24 hours and more than 100k...

✦ Oct 23 🖱 272 💬 4




 In Realworld AI Use Cases by Chris Dunlop

The complete guide to Claude Code's newest feature “skills”

Claude Code released a new feature called Skills and spent hours testing them so you don't have to. Here's why they are helpful

★ Oct 21 🖱 95 💬 4



 Micheal Lanham

Microsoft Just Solved the AI Agent Problem (And Open-Sourced the Solution)

How the new Agent Framework unifies Semantic Kernel and AutoGen to bring enterprise-grade AI agents from prototype to production

★ Oct 14 🖱 55 💬 3



See more recommendations