

Claude Code Plugins: Share Your Setup With Your Team

11 MIN · YOUTUBE · [HTTPS://WWW.YOUTUBE.COM/WATCH?V=RWAS_3GLHRU](https://www.youtube.com/watch?v=RwAS_3gLHRU)

https://www.youtube.com/watch?v=RwAS_3gLHRU

SUMMARY

The video discusses the importance of plugins in cloud code, particularly for collaborative teams. It explains how plugins allow for the packaging and sharing of skills and agents, making it easier to manage updates and maintain version control across projects.

- *Plugins facilitate collaboration by allowing teams to share skills and agents easily.*
- *They help avoid the hassle of manually copying and pasting code between different repositories.*
- *Plugins can be created from existing skills and agents, enabling better organization and version control.*
- *The video demonstrates how to create a Git repository for plugins and push updates to GitHub.*
- *Users can install plugins from their organization's GitHub or explore public plugins available in the marketplace.*
- *The importance of structuring plugins correctly is highlighted, as it affects their discoverability and functionality.*
- *Future topics will include packaging MCPS and hooks into plugins for enhanced sharing capabilities.*

All right, in this video we're going to talk about plugins. Now, plugins might not be super relevant for you if you are working in cloud code by yourself. But if your team, like my team has adopted cloud code as the default way of working, which I recommend everyone should or will get there eventually. um you start using cloud code in a more collaborative fashion, right? So, for example, um you your marketing team or your sales team starts using cloud code and you built this like amazing skill, right? And you want to be able to easily share that with um your team. Well, how do you do that, right? You could probably like let's say for example um the skill we created in um uh a couple episodes ago or videos ago I guess um let's say we created this skill. You could literally just copy this, right? Copy uh and then like paste it into a Slack message or something and send it to your friend, right? But then what if you update something because um you found a better way to do it or a more efficient way of doing it. Then you know you have to run into this issue of you run into this issue of having to um manually send it to your friend every time, right? It's not very discoverable.

Um so it's it's difficult uh to sort of share your skills, your your agents, uh maybe your MCP setup, your hooks, which we'll get into later. MCPS and hooks we haven't covered yet, but plugins are basically a way to package up the specific tool set that you've built um and be able to share that with with others, right? Um, and vice versa. You can also get plugins from other folks either on your team or just like in general um and download really good plugins uh that others have developed already and you can use it right. Uh some are public, some are private. Um today we'll focus on uh a private use case and then I'll quickly show you how you can download some of the public ones as well. Okay. So, um, let's see. I wrote this markdown file here. We're going to actually

take skills that we learned from git basics, skills, and agents. Well, we created the skill and the agent um, in previous videos, and we're going to package those up um, as plugins. So, first things first, what we're going to do is turn this um, into a Git repository, right?

So, what we'll do is I'll just ask Claude to create a Git repository for this. So, I'm going to say, "Hey, can you turn this repo into a git repository um and put it uh in uh my yeah, put it in my personal um account. So what I'm asking uh claw to do is basically turn this into a git repo and then um push it to the to GitHub so that it's um connected uh it's initialized and then connected to the remote repo so that um anytime I update things in this repo that's going to be pushed uh and I push it that's going that can be pushed into the cloud. So we'll say yes. Okay.

Yes. always create a git ignore which is always helpful. Um and then I'll add in there. Um I'm actually going to take out cloud because we might need that. So anything that you put in getit ignore means that it doesn't actually sync to git. So uh for example always holds like your API keys. So it's good practice to not sync that obviously. Um, so I'll add all of these.

Great. Okay. So, if I click on this now, I can see I can see that this has been created, right? Which is great. Now, what we're going to do is um I can see that this is green, which means that this hasn't been committed. So I'm going to say um get add and push.claude as well. Now let's turn the account research agent. So when you use the at um actually skill then look at skill research skill.m MD. When you use the at you can reference specific files and I'm going to say let's turn these into plugins in this repo. And then I'm going to say this um codecloud uh.com docs uh plugins to understand how to do this um refer to or how to package agents and skills into a plug-in.

So it's first reading those files and then now it's going to turn them into a plugin. And what I'll do is I'll actually use I'll I'll do account research and this agent. I've asked for these two things to turn into a plugin. But I'm not going to turn the skill creator into a plugin so that we can kind of see um the difference. And then once we've created these as plugins, what we can do is um use them in other projects, which is really cool. uh so other teammates or you yourself in a different repo can actually refer to these plugins and then use them. Now uh on your laptop it might be easier just to copy and paste the skill or the agent and you can do that do it that way. But then again you get two different copies right in two different repos. And so if you update one you have to manually copy and paste it into the other one. Whereas if you create a plug-in um you can e more easily update um the latest uh versions and you can version control because it's synced to git.

So anyways um okay great. So, we've pushed everything already, right? You can see here it's added the plugin, the agents and the skills, and it's committed uh and it's pushed. And so, we can install the plugin by doing this right here. So, what I'm going to do is um I'm going to open up a new repo. And what I'm going to do is I'm going to say I'm going to start Claude in here.

And I'm going to type in slash plugin and I will say marketplaces. Go to marketplaces and add marketplace. Right? So here it says owner repo GitHub and this is the one that you can look at um when you push it into your

git. So I'm going to say the workflow co and let me look at what the other name was. Okay. So that's uh Claude CC walkthrough entropic cla walkth through anthropic. So what I'm doing is um I'm telling it the owner organization name right uh slash this owner owns this repo right and I already have access to this because uh my git is synced to my organization right uh and if you don't have access to this so if you type this in as you're watching this video you'll probably get some sort of error because you don't have have access to my GitHub organization but everyone on my team does so they could do this right so the workflow So claude sec walk through anthropic. If I type enter cloning repository marketplace file not found marketplace JSON okay so this is what it says and we ran into an error and I'm going to pause this video figure it out and then I will come back.

Okay. So, uh, for the past little while, I was having issues in terms of figuring out why only one plug-in was showing, but I feel really stupid right now because the plug-in was um formatted to have an a skill and an agent. So, before it this was just called account research and then inside the plug-in actually had the agent and the skill. And so um before I kept just looking at the plugin and it said one and when I click on it you can't actually now I can see the different plugins but when there's only one um cloud code actually has a bug where it just returns you to discover.

It doesn't show you the skill um the skill or the agent. But now you can see these two, right? Because I asked Claude here um where where did I say this? Uh separate out the agent and the skill into two separate plugins in the marketplace. So the whole time I was trying to debug it had actually worked. Um it just was that it was under one plugin and it and I was just in my mental model. I just thought it would be two plugins. It was my fault for not being very specific with Claude to say, "Hey, I want the agent and the skill to be separated out into two separate plugins."

Uh, Claude Claude Code on the other hand was like, "Well, one's account research agent, another's account research skill. They sound very similar. I'm just going to package them into one account research plug-in, which makes sense." um if which um because it doesn't have the context that I'm just doing a a dummy walk through, right? So anyways now you can see WFCO tools browse plugins and there is the agent and the account research skill and I can install for me just in this repo local scope and then I have to go back here and uh browse plugins and I can um install for me local scope again I'll exit claude and I can uh if I type in like agents, you can see account research agent here, right? Um, and then if you look at skills, you can see was here account research right here, right? Yeah. Hopefully this was um hopefully this taught you about uh how to package up your agents and skills and then we'll um learn about MCPS and hooks and those can be packaged up into plug plugins as well in the marketplaces uh so that you can share them with your team. Um, last thing, uh, you saw me use the plugin, but there's you can under discover, there's a lot of good plugins that, um, you can actually use. And so, um, I definitely recommend that you look at these. These are public open- source plugins, um, that you can use, uh, like right now. Um, so yeah, uh, hopefully you found this, uh, useful, if not entertaining. Um, and, uh, as always, uh, let me know if you have any questions.