

Ten terms from Class 4. Learn them well enough to explain each one in your own words.

1. The spiky frontier

The spiky frontier is the idea that AI is brilliant at some tasks and surprisingly bad at others that seem just as easy. A model might write working software in minutes, then miscount the letters in a word or invent a fact. Its skills don't follow human logic, so being good at one thing doesn't mean it's good at the thing next to it. You learn where the spikes are in two ways: by using your agent a lot and paying attention, and by following other people's reports and examples of what it can and can't do.

2. Verification

Verification means checking that work is actually right, instead of taking it on faith. You can do it yourself, or ask your agent to do it: open the page and look at it, run the program and see what happens, compare an answer against the source it came from, or run a test. Agents do their best work in a loop: do the work, check it, fix what's wrong, and check again. Your worksheet from last week called this a verification loop. When you give your agent a task, also tell it how to check its work, and ask it to show you what it checked before it says it's done.

3. Test

A test is one kind of verification: a small program that checks whether something works, and reports pass or fail. For example, a test might open your website and fail if the signup button is missing. Unlike a model, a test gives the same verdict every time you run it, which makes it something you can trust even when you can't fully trust the model. Ask your agent to write tests for anything it builds, and to run them before it tells you the work is done.

4. Goal

A goal is a finish line you give your agent, so it keeps working on its own until it gets there. Normally an agent does one task and stops to wait for you. With a goal, it keeps going: it works, checks its progress, fixes what's wrong, and continues until the goal is met, it gets stuck, or it runs out of the budget you gave it. In Codex, you set one by typing /goal followed by what you want done. Goals work best when "done" is easy to verify, like "all the tests pass" or "every page on the site loads". A goal with a full set of tests the work must pass is sometimes called a Gauntlet Loop, and it's how you can leave an agent working for hours and still trust the result.

5. Model provider

A model provider is the company or computer that actually runs an AI model. When you send Codex a message, your computer doesn't do the thinking. The request goes over the internet to OpenAI's computers, which run the model and send the reply back. That's why you can hit usage limits, and why your agent sometimes slows down or stops working even when nothing is wrong on your end: the provider is busy or having an outage. When that happens, wait a bit or try again later.

6. Cloud Computer

A Cloud Computer is a computer you rent that lives in a data center and stays on all the time. You reach it over the internet instead of sitting in front of it. Agents and automations that run on your laptop stop when you close it. On a cloud computer, they keep running day and night, including jobs set to run on a schedule, like a summary every morning at 7. That's where always-on assistants like your Grok Bot live.

7. API

An API is a way for one program to talk to another. A website is built for people to click through. An API is the version built for programs: a program sends a structured request, like "give me today's calendar events", and gets back data it can use. Most apps you use, like Gmail, Stripe or Notion, have one. APIs are how your agent and your scripts read and change things in other apps without anyone clicking around.

8. MCP

MCP is a standard way to connect an agent to other apps and tools. It works like a universal plug: an app builds one MCP connection, and any agent that supports MCP can use it. Through it, the agent gets new tools, like "search my Notion" or "create a calendar event", usually built on top of that app's API. Some Codex plugins use MCP to connect to apps. When you want your agent to work with an app it can't reach yet, check whether that app has an MCP connection.

9. Plugin

A plugin is a package you install into Codex to give it a new ability, like working with your Gmail, your calendar, or Notion. Inside, a plugin can bundle several things: a connection to an app (through its API or an MCP connection), and skills that teach the agent how to use it well. You install it once, sign in if needed, and the agent can use it from then on. When you want your agent to work with an app, check whether there's a plugin for it before building anything yourself.

10. Secrets

Secrets are passwords, API keys and other codes that prove a program is allowed to act as you. Anyone who gets one of your secrets can use your accounts: read your email, spend your money, or run up charges on your API account. Never paste a secret into a chat, a document or a public website, and never put one directly in your code. Keep secrets in a password manager or in your computer's settings for them, and give each one only the access it needs.