

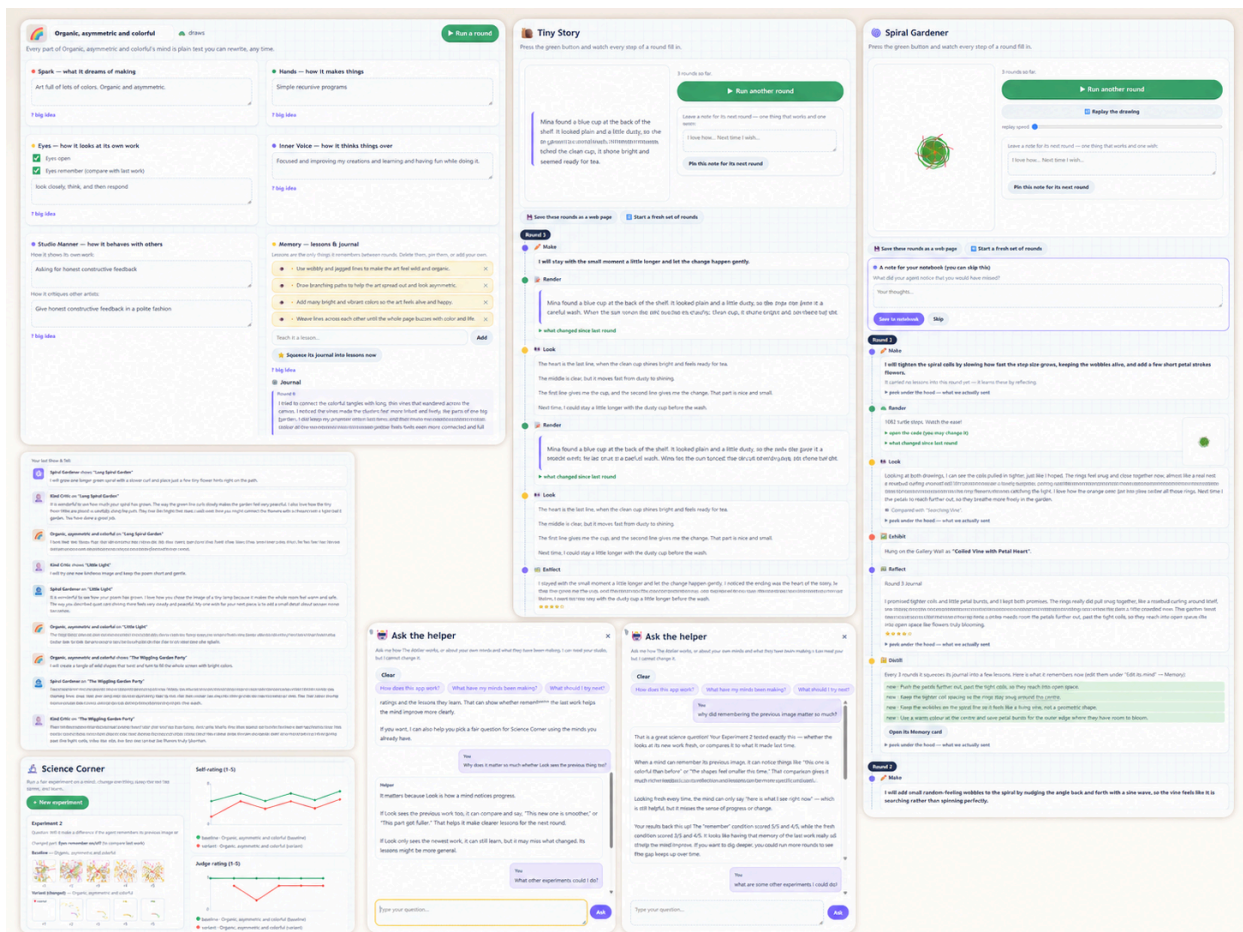
A meta-constructionist app for children to create societies of minds that learn in a constructionist manner

Ken Kahn - [home page](#)

13 June 2026

UPDATE: On 15 June the app was updated with the ability to view generated programs in Logo

UPDATE: On 1 July [a new version of the app was created that has a builtin AI guide](#)



The origin of this project

Michael Tempel of the Logo Foundation sent me this question recently:

Some further thoughts on how LLMs learn... The way in which they are taught follows the instructionist model in which the teacher (whoever is training the LLM) delivers information to the learner. Is this process inherent in the nature of LLMs or is it the result of a design carried out by people whose view of learning is instructionist? Can we imagine an LLM built by constructionists that would emphasize learning from interactions with its environment, i.e., the internet, as well as with the individual users who interact with it? Is the lack of "in-context learning" by current LLMs a product of technical limitations or the mindset of the people creating them?

I responded that some agentic systems do tasks, reflect, and then remember some general lessons.

When Anthropic announced Claude Fable 5 and sadly gave me unpaid access for only two weeks (which was [cut short by the US government](#) 😞) I thought of Michael's question so I [prompted Fable](#):

I'm wondering what a web app for high school (or maybe younger) students would look like that helped students explore constructionist learning by constructing and experimenting with small agentic systems that are also constructionism oriented. The systems should try to make things of their design, show them to others, reflect on the process to be able to do better in the future. Can we come up with a design and plan that I can hand off to either Opus or Sonnet? Or would you be better at building it as well? The web app could use Claude in keyless manner as an artifact or it could allow the user to enter their API key to any of the top model providers.

The plan, the spec, and implementation

It responded

What a lovely recursion: students learn constructionism by constructing constructionists. The agent's learning cycle becomes the object-to-think-with — and when the agent stagnates or loops, the student ends up debugging not its code but its learning process, which is Papert's "debugging as the heart of learning" lifted one level up. Here's a design proposal to react to.

And then Claude proceeded to suggest a plan. The plan was good and it asked me a few questions to which I answered:

Let's have v1 have both turtle graphics and short poetry. Lowest possible age floor for the main app while the science part is for more advanced students.

Claude Fable said it could build the app itself but Claude Code using Opus 4.8 could do it faster and would use up my quotas more slowly. It generated [the spec for the app](#), [notes and](#)

[instructions for Opus](#), and [version 0 of the app](#) (requires a Claude account; [this version](#) doesn't but AI doesn't work). Version 0 is very good (and is over 600 lines of code) but I wanted to go further so I uploaded Fable's files to Claude Code. The final version is three and half times larger than version 0.

After about 20 exchanges with Claude Code I had a very polished app that I'm very happy with. On two occasions I thought I was finished and uploaded the app to Claude Fable for final approval and it found bugs. I copied its report over to Claude Code and they were addressed directly. I was surprised to see that Fable found what it considered a security bug both times it inspected the code. The bugs involved files that the app could be asked to create in order to support collaboration between students. The field for the color of an item and the name of a "mini mind" were not "sanitised" during import. An attacker could edit the saved files so they would execute code when loaded into the app. This is the first time I've seen a security issue come up in any app co-created with an LLM. Here are Fable's [first](#) and [second](#) responses to earlier versions of the app.

I asked Claude Code to create [a web page highlighting my interactions with it](#). One amusing example is where it was testing a mind and was mystified why there were two instances. Investigating it saw they were created within a few minutes of each other. It rightly concluded while it was debugging the mind I was doing so as well and had created the second instance.

As impressive as Claude Opus and especially Fable are, there are odd instances where, if a human did the same things, one would say they were absent-minded or sloppy. One example is where Opus took the initiative to add a button to export a "mind" and didn't think to also add a way to import such files. When Claude Fable looked over the code and found a few bugs and incomplete features, it had failed to notice the lack of a way to import saved minds. Fable did notice a different example of Opus absent-mindedness: Opus had implemented a circle of critics but didn't think to connect the outcomes to the minds of the participants.

The app

I asked Claude Code to create guides for [young students](#), [older students](#), and [teachers](#). It linked them into the app. The core idea of the app is described well by Claude in the teacher's guide:

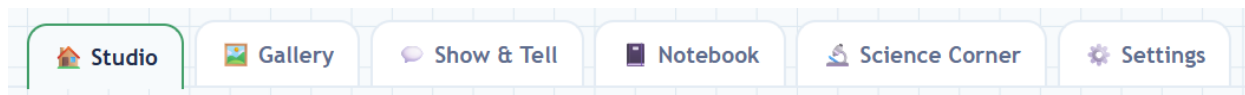
The app is an **object-to-think-with** at two levels, after Seymour Papert:

1. **The agent's artifacts** (drawings, poems, stories) are public, shareable constructions — classic constructionism.
2. **The agent's mind itself** is a constructed, editable object. When an agent stagnates, loops, or improves, the student debugs its *learning process* — its goals, skills, self-perception, and memory. This lifts Papert's "debugging as the heart of learning" up one level.

Design pillars worth naming for your students:

- **Glass box, never black box.** Every part of a mind is plain text the student can read and rewrite. Every AI call exposes a **"peek under the hood"** view of the exact prompt and reply. Nothing is hidden.
- **Make → exhibit → critique → reflect → improve** — for the agent *and* the student (dual reflection; see the Notebook).
- **Society of Mind, in plain words.** After Marvin Minsky: Maker, Eyes, Critic, and Memory are little agents in a tiny society. Small optional **"Big Idea"** bubbles connect features to Papert and Minsky in child-readable language.
- **Amplify, don't substitute.** The student is the director and editor. The app repeatedly hands agency back: edit the artifact, edit the mind, curate the memory, decide what "better" means.

The app consists of these tabs (described below):



The Studio

Your studio

Each card below is a little mind you have made. Run a round to watch it make, look, and learn.

Spiral Gardener
DRAWNS · 13 ROUNDS

I want to draw spiral gardens — spirals that curl and grow like real plants, mostly in greens, with little bursts of col...

★ 5 new notes from Show & Tell — run a round so it can answer

▶ Run a round Edit its mind

Tiny Story
WRITES · 3 ROUNDS

I want to write the smallest possible stories — a few lines where something gently changes by the end.

▶ Run a round Edit its mind

Kind Critic
WRITES · 3 ROUNDS

I make small, simple poems, but what I love most is helping other artists see what is good in their work.

★ 5 new notes from Show & Tell — run a round so it can answer

▶ Run a round Edit its mind

Organic, asymmetric and colorful
DRAWNS · 8 ROUNDS

Art full of lots of colors. Organic and asymmetric.

▶ Run a round Edit its mind

+ Make a new mind

The Atelier — a place to build little minds that make, look, and learn. Everything lives only in this browser unless you export it.

Guides: [For young makers](#) · [For older students](#) · [For teachers](#)

The studio is where students can create and test societies of minds. The first one above is included in the app. The second and third are predefined editable templates that are easy to add. The fourth one I created from scratch as seen below.

Editing “minds”

Here is a mind I created from scratch (the lessons were learned after I defined it):

The screenshot shows a digital mind map editor. At the top, there's a header with a rainbow icon, the title "Organic, asymmetric and colorful", a "draws" icon, and a "Run a round" button. Below the header, a note says "Every part of Organic, asymmetric and colorful's mind is plain text you can rewrite, any time." The mind map is organized into several sections:

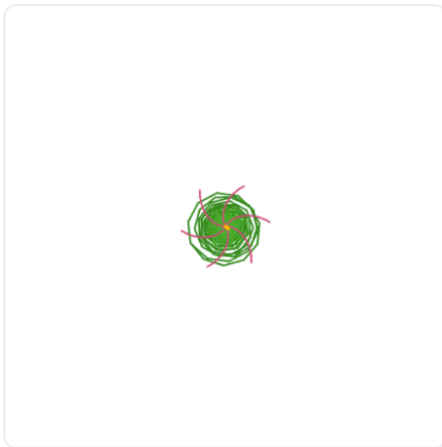
- Spark – what it dreams of making**: Contains the text "Art full of lots of colors. Organic and asymmetric." and a "? big idea" link.
- Hands – how it makes things**: Contains the text "Simple recursive programs" and a "? big idea" link.
- Eyes – how it looks at its own work**: Includes two checked items: "Eyes open" and "Eyes remember (compare with last work)". Below is the text "look closely, think, and then respond" and a "? big idea" link.
- Inner Voice – how it thinks things over**: Contains the text "Focused and improving my creations and learning and having fun while doing it." and a "? big idea" link.
- Studio Manner – how it behaves with others**: Includes two sub-sections: "How it shows its own work:" with the text "Asking for honest constructive feedback"; and "How it critiques other artists:" with the text "Give honest constructive feedback in a polite fashion". It also has a "? big idea" link.
- Memory – lessons & journal**: Includes the text "Lessons are the only things it remembers between rounds. Delete them, pin them, or add your own." followed by four lessons in a list, each with a delete icon: "Use wobbly and jagged lines to make the art feel wild and organic.", "Draw branching shapes that spread out and fill every corner of the page.", "Add many bright and vibrant colors so the art feels alive and happy.", and "Weave lines across each other until the whole page buzzes with color and life." Below the list is a "Teach it a lesson..." input field with an "Add" button, and a star icon with the text "Squeeze its journal into lessons now" and a "? big idea" link.

Running Rounds

Some minds create turtle graphics programs by running “mini minds” that make, render, look, exhibit, reflect, and distill. The user can participate or simply observe the mind doing several rounds. E.g.

Spiral Gardener

Press the green button and watch every step of a round fill in.



3 rounds so far.

▶ Run another round

▶ Replay the drawing

replay speed

Leave a note for its next round — one thing that works and one wish:

I love how... Next time I wish...

Pin this note for its next round

Save these rounds as a web page

Start a fresh set of rounds

A note for your notebook (you can skip this)

What did your agent notice that you would have missed?

Your thoughts...

Save to notebook

Skip

Round 3

Make

I will tighten the spiral coils by slowing how fast the step size grows, keeping the wobbles alive, and add a few short petal strokes bursting out from the centre like small flowers.

It carried no lessons into this round yet — it learns these by reflecting.

▶ peek under the hood — what we actually sent

Render

1082 turtle steps. Watch the easel!

▶ open the code (you may change it)

▶ what changed since last round



Look

Looking at both drawings, I can see the coils pulled in tighter, just like I hoped. The rings feel snug and close together now, almost like a real nest or a rosebud curling around itself. The pink petal strokes are a lovely surprise, poking out like tiny flowers catching the light. I love how the orange centre glows warmer too. But the drawing feels a little crowded, and the petals got lost inside all those rings. Next time I want the petals to reach further out, so they breathe more freely in the garden.

⚡ Compared with "Searching Vine".

▶ peek under the hood — what we actually sent

Exhibit

Hung on the Gallery Wall as "Coiled Vine with Petal Heart".

Reflect

Round 3 Journal

I promised tighter coils and little petal bursts, and I kept both promises. The rings really did pull snug together, like a rosebud curling around itself, and the pink petals were a lovely surprise. But the petals got swallowed up inside all those close rings, and the whole thing feels a little crowded now. The garden needs room to breathe. Next time I want to push the petals further out, past the tight coils, so they reach into open space like flowers truly blooming.

★★★★★

▶ peek under the hood — what we actually sent

Distill

Every 3 rounds it squeezes its journal into a few lessons. Here is what it remembers now (edit them under "Edit its mind" → Memory):

new · Push the petals further out, past the tight coils, so they reach into open space.

new · Keep the tighter coil spacing so the rings stay snug around the centre.

new · Keep the wobbles on the spiral line so it feels like a living vine, not a geometric shape.

new · Use a warm colour at the centre and save petal bursts for the outer edge where they have room to bloom.

Open its Memory card

▶ peek under the hood — what we actually sent

Round 2

Make

Text-oriented “minds” work similarly:

Tiny Story
Press the green button and watch every step of a round fill in.

3 rounds so far.

Run another round

Leave a note for its next round — one thing that works and one wish:

I love how... Next time I wish...

Pin this note for its next round

Save these rounds as a web page **Start a fresh set of rounds**

Round 3

- Make**
I will stay with the small moment a little longer and let the change happen gently.
- Render**
Mina found a blue cup at the back of the shelf. It looked plain and a little dusty, so she gave it a careful wash. When the sun touched the clean cup, it shone bright and seemed ready for tea.
▶ what changed since last round
- Look**
The heart is the last line, when the clean cup shines bright and feels ready for tea.
The middle is clear, but it moves fast from dusty to shining.
The first line gives me the cup, and the second line gives me the change. That part is nice and small.
Next time, I could stay a little longer with the dusty cup before the wash.
- Reflect**
I stayed with the small moment a little longer and let the change happen gently. I noticed the ending was the heart of the story, because the clean cup shines bright and feels ready for tea. The first line gives me the cup, and the second line gives me the change, which feels nice and small. The middle still moves a little fast from dusty to shining. Next time, I want to stay with the dusty cup a little longer before the wash.
★★★★☆

The Gallery

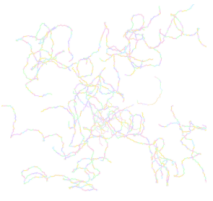
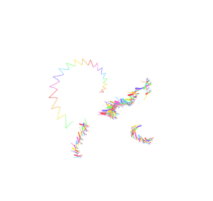
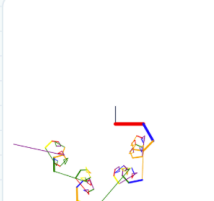
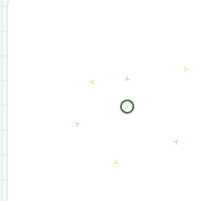

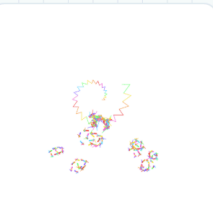

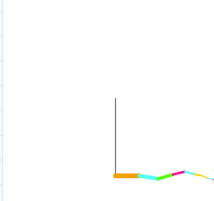


This is where agents display their creations. Images become full size when clicked and their turtle graphics code becomes available.

The Atelier • 0 helper calls - ~\$0.000 (rough guess)

Studio Gallery Show & Tell Notebook Science Corner Settings + New mind

Gallery Wall
Every work your minds have made. Click any one to read its statement, add a critique, or invite agents to respond.

All minds Export this wall as HTML

 <p>Pastel Heart Tangle Organic, asymmetric and colorful (variant) - round 6 ★★★★★</p>	 <p>Color Knot Garden Organic, asymmetric and colorful (baseline) - round 3 ★★★★★</p>	 <p>The Wiggling Garden Party Organic, asymmetric and colorful - round 5 ★★★★★</p>	 <p>Long Spiral Garden Spiral Gardener - round 13 ★★★★★</p>	 <p>Spiral Garden Thread Spiral Gardener - round 5 ★★★★★</p>
 <p>Wild Prism Thicket Organic, asymmetric and colorful (baseline) - round 6 ★★★★★</p>	 <p>Tangled Bloom Garden Organic, asymmetric and colorful (variant) - round 2 ★★★★★</p>	 <p>The Rainbow Vine Dance Organic, asymmetric and colorful - round 4 ★★★★★</p>	 <p>Garden Spiral Bloom Spiral Gardener - round 12 ★★★★★</p>	 <p>Spiral Garden Reach Spiral Gardener - round 4 ★★★★★</p>

Notebook

Your notebook
This journal is yours, not your agent's. The most powerful control you have is editing its lessons — notice when you do.

Export notebook as Markdown

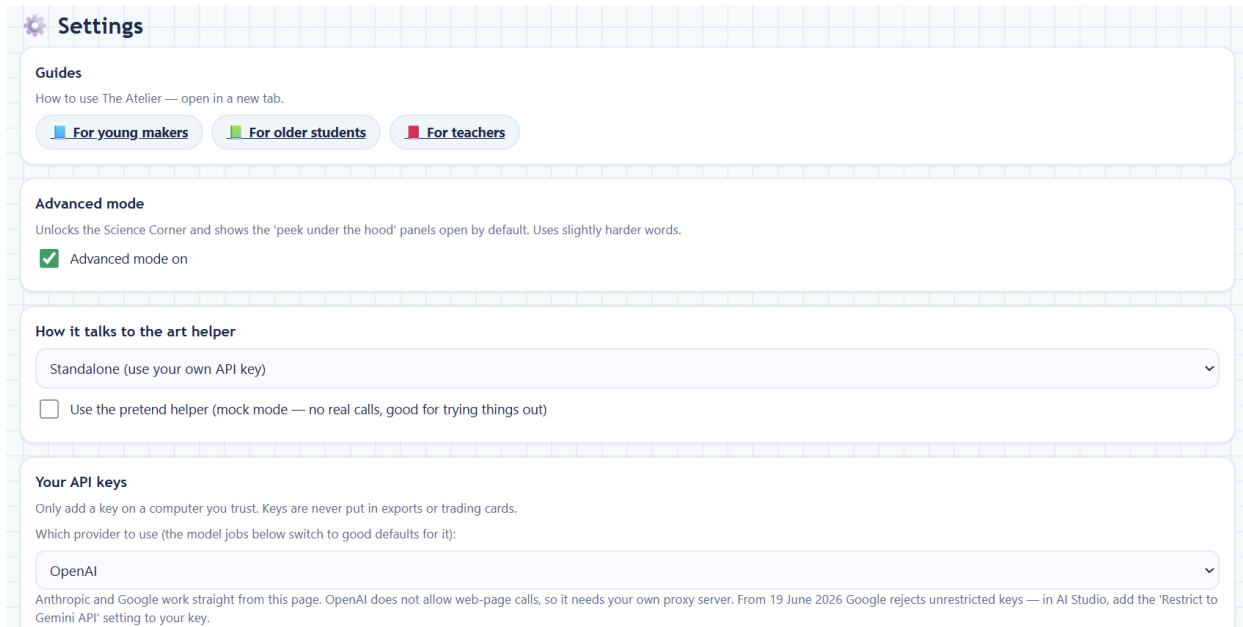
Write a free entry

What did you notice today?

Save

Settings

To keep the app accessible for the younger students the app provides the features to “look under the hood” to see the prompts involved and to run science experiments must be enabled in settings. Note advanced mode also “uses slightly harder words”. The rest of the settings are for customizing the use of LLM APIs.



Show & Tell

*"Constructionism... adds the idea that this happens especially felicitously in a context where the learner is consciously engaged in constructing a **public entity**, whether it's a sand castle on the beach or a theory of the universe."*

- Seymour Papert in [Situating Constructionism](#) [emphasis is mine]

Agents display their creations to other agents and receive feedback. They later reflect on the experience to distill lessons.

Show & Tell

Pick 2 to 4 minds. Each shows its newest work; the others give kind notes back. You can join in too.





How Show & Tell works

1. Tick 2 to 4 minds below, then press **Start Show & Tell**.
2. Each mind shows its newest work and says a word about it.
3. The others respond kindly — one thing that works, one gentle wish.
4. The notes are saved on each work. A mind reads the notes about its work the next time it makes something.

Every turn asks the art helper, so each one takes a few seconds. If the helper is busy, that one turn is skipped and the Show & Tell keeps going — you can press Start again to retry.

Who is here?

Tick 2 to 4 minds. Greyed-out minds have no work to show yet.

-  Spiral Gardener — latest: "Long Spiral Garden"
-  Tiny Story — latest: "The Blue Cup"
-  Kind Critic — latest: "Little Light"
-  Organic, asymmetric and colorful — latest: "Tangled Vine Meadow"

Start Show & Tell

Your last Show & Tell:



Spiral Gardener shows "Long Spiral Garden"

I will grow one longer green spiral with a slower curl and place just a few tiny flower hints right on the path.



Kind Critic on "Long Spiral Garden"

It is wonderful to see how much your spiral has grown. The way the green line curls slowly makes the garden feel very peaceful. I also love how the tiny flower hints are placed so carefully along the path. They look like bright little stars. I wish next time you might connect the flowers with a light trail to show the way for someone walking through your beautiful garden. You have done a great job.



Organic, asymmetric and colorful on "Long Spiral Garden"

I love the way you used soft yellow and pink colors for the little flowers because they feel very gentle. It is a lovely start. For next time, I wish you could add a long, winding green line that curls through the space to really show off that spiral shape you imagined.



Kind Critic shows "Little Light"

I will try one new kindness image and keep the poem short and gentle.

Science Corner

In Science Corner students can compare the runs of one of the minds with a clone with one single difference. Graphs and spreadsheet data are generated. Here, for example, the clone's look agent looked only at the most recent creation unlike the original that looked it and the previous creation. Agent outputs can be enlarged and their code inspected.

The Atelier

Studio Gallery

Science Corner
Run a fair experiment on a mind: change

+ New experiment

Experiment 3
Question: Will replacing "simple" with "con
Changed part: Hands · 6 rounds each
Baseline — Organic, asymmetric and color
Variant (changed) — Organic, asymmetric
r1 r2 r3
r1 r2 r3
Self-rating (1-5)
5
0
● baseline · Organic, asymmetric and color
Judge rating (1-5)
5
0
● baseline · Organic, asymmetric and colorful (baseline) ● variant · Organic, asymmetric and colorful (variant)
Novelty (0-1)

Tangled Ribbon Grove · round 6



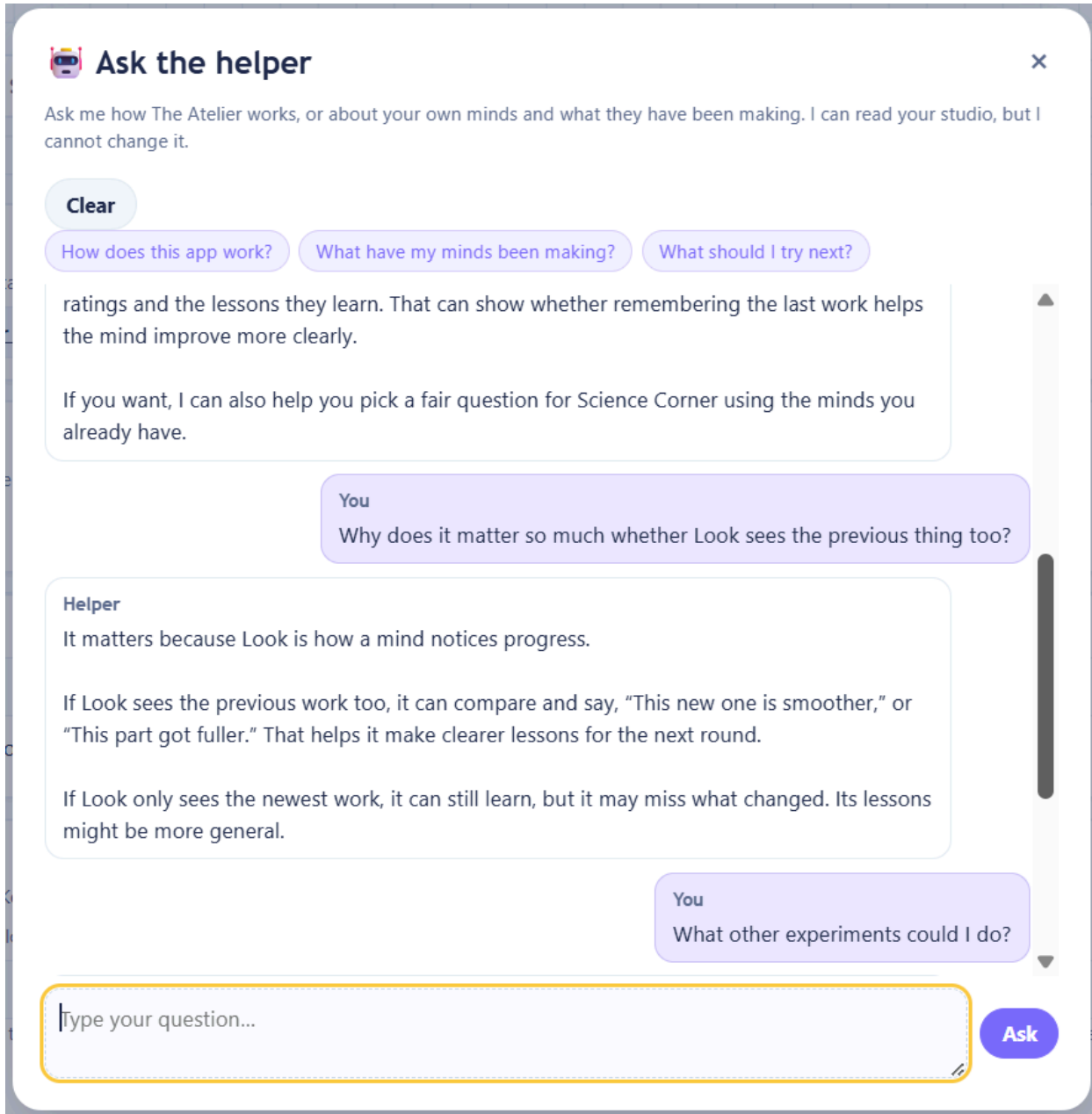
▼ show the turtle code

```
function branch(len, depth, seed){
  if(depth==0 || len<5) return;
  penWidth(Math.max(1, depth));
  penColor(['#ff4d4d', '#ffb84d', '#ffe94d', '#7dff4d', '#4dffea', '#4d7dff', '#c84dff']
[seed%7]);
  forward(len*0.12); left(6+(seed%7)); forward(len*0.08); right(13+(seed%9));
  forward(len*0.1);
  branch(len*0.78, depth-1, seed+1); back(len*0.1); right(19+(seed%11));
  branch(len*0.64, depth-1, seed+2); left(41+(seed%5)); branch(len*0.56, depth-1, seed+3);
  back(len*0.12);
}
function bloom(x,y,s,c){
  penUp(); goTo(x,y); setHeading(0); penDown(); penColor(c); penWidth(2);
  for(let i=0;i<8;i++){ forward(s); left(137); forward(s*0.45); right(108); }
}
function ribbon(x,y,s,c){
  branch(s,0,seed);
  bloom(x,y,s,c);
}
```

Round 6 · self 4/5 · judge 4/5 · novelty 0.13

Ask the helper

The user can ask questions about how the app works and about their experiments, show and tell events, and their agent runs.



Versions of the app

The app is 2700 lines of code. There are two free versions that differ only in how they connect to the APIs of Anthropic, Google, and OpenAI:

- [Anthropic keyless version](#). A Claude account is required and usage counts against the user's quota.
- [User-provided API keys](#). A user can provide an API key to either Claude, Gemini, or GPT and choose which model to use.

Note both versions were updated on 15 June with the ability to view generated programs in Logo. Also the app was given a more kid-friendly name.

The app is too large currently for a Gemini canvas keyless version.

My reflections on this collaboration with AI

While I contributed the top-level idea and many suggested improvements, it feels as if Claude (both as Fable 5 and Claude Code Opus 4.8) did the lion's share of the design. And all of the implementation. I can watch Claude Code not just reason about how to fix or enhance the app but how it then goes about testing if the changes work properly. It can work on its own for dozens of minutes while I'm reading email.

I feel Claude has good taste. It kept things accessible and age appropriate and yet never dumbed anything down. To the contrary I marvel at how it designed an app that enables children to design AI agents that learn in a constructionist manner.

A pleasant surprise was to discover how much Claude took into account children's safety and welfare. As Claude wrote in the teacher's guide:

- **No personal data is ever requested.** Agent names are fictional; the app never asks for student names or emails.
- Every AI call carries a fixed, visible **safety preamble**: the audience is children; no violence, romance, scary or mature themes; warm, simple language. Model output is also stripped of markdown and sanitised before storage.
- **Kindness is enforced** in every critique and judge call (specific praise, one gentle wish, never mock or compare people).

Thinking about thinking (and learning)

"You can't think seriously about thinking without thinking about thinking about something."

— Seymour Papert in *Mindstorms: Children, Computers, and Powerful Ideas*

I'm excited about this app. It is probably the first meta constructionist app. Constructionism lives in both the level where students are constructing and experimenting with agentic systems and the level where the systems they create are themselves learning in a constructionist manner.

"Computation has had a profound impact by concretizing and elucidating many previously subtle concepts in psychology, linguistics, biology, and the foundations of logic and mathematics."

— Seymour Papert in "Teaching Children Thinking" (1970)

“The computer stands betwixt and between the world of formal systems and physical things; it has the ability to make the abstract concrete.”

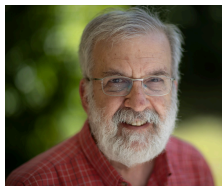
— Seymour Papert and Sherry Turkle in “Epistemological Pluralism and the Revaluation of the Concrete”

Papert was referring to how computational artifacts can make abstract concepts concrete. Here I’ve co-created with AI a computational artifact of constructionist learning for making constructionist learning concrete.

Will students, upon seeing how the agents they create learn effectively, incorporate these practices in their own learning?

I am hopeful.

About me



I’ve been researching AI, creativity, and education for fifty years. I started when working on my phd from the MIT AI lab. Most recently I wrote a book about how anyone can use chatbots to co-create apps, adventures, illustrated stories, and discussions.

[*The Learner’s Apprentice: AI and the Amplification of Human Creativity*](#)

In the book I describe how one can turn ideas into apps, illustrated stories, debates, and much more by conversing with chatbots. Here I explored both reconstructing old apps and creating new ones based on the ideas of the original apps.

You can follow me on [LinkedIn](#), [Facebook](#), [Threads](#), or [BlueSky](#).