



Configuring skills and tools

Led by Zach Muhlbauer and Meha Gupta

New Media Lab · Room 7388.01

CUNY Graduate Center

Thursday, October 15, 2026 · 2:30–4:00 p.m. Eastern

<https://cuny-ai-lab.github.io/sandbox-series/skills>



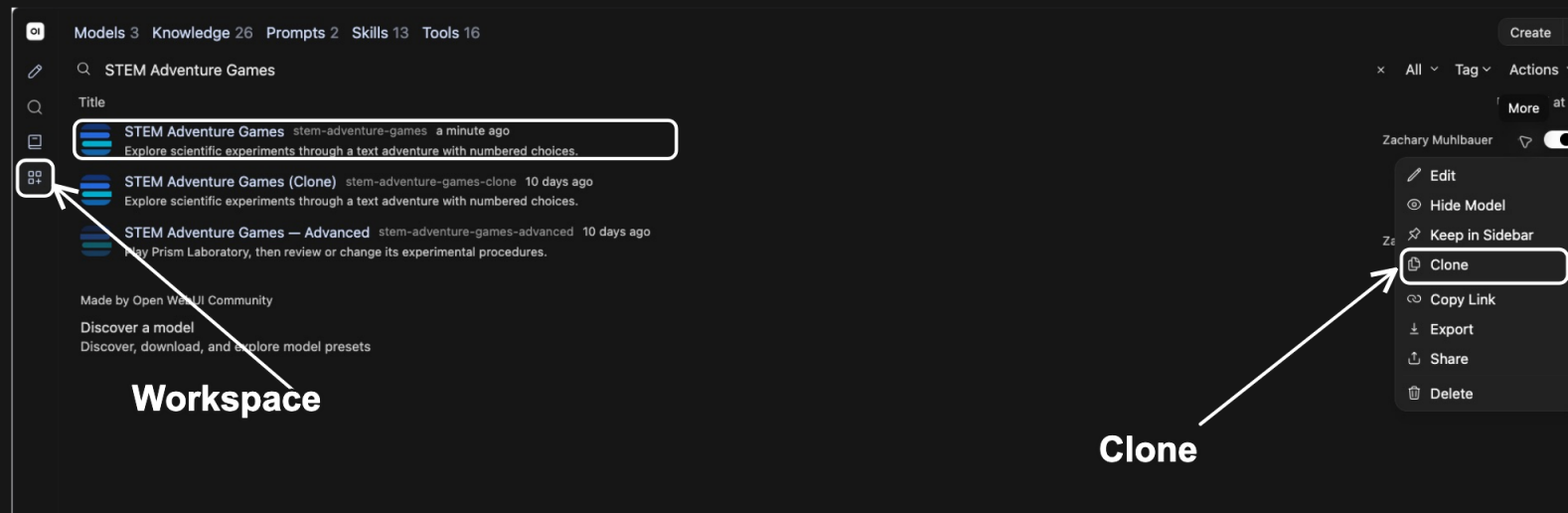
1 / 18



Outline

Choose Your Own Adventure

1. Continue your saved model, or clone **STEM Adventure Games** in **Workspace** → **Models** → ... → **Clone**
2. Open **Workspace** → **Models** → **Edit** and review your system prompt's setting, goals, and rules
3. Keep your model card open to add a skill and a tool



What Is a Skill?

Skills contain plain-text instructions for repeatable tasks

Trigger	Request or condition that starts a procedure
----------------	--

Procedure	Steps, checks, and points that require a question
------------------	---

Output	Required response content and form
---------------	------------------------------------

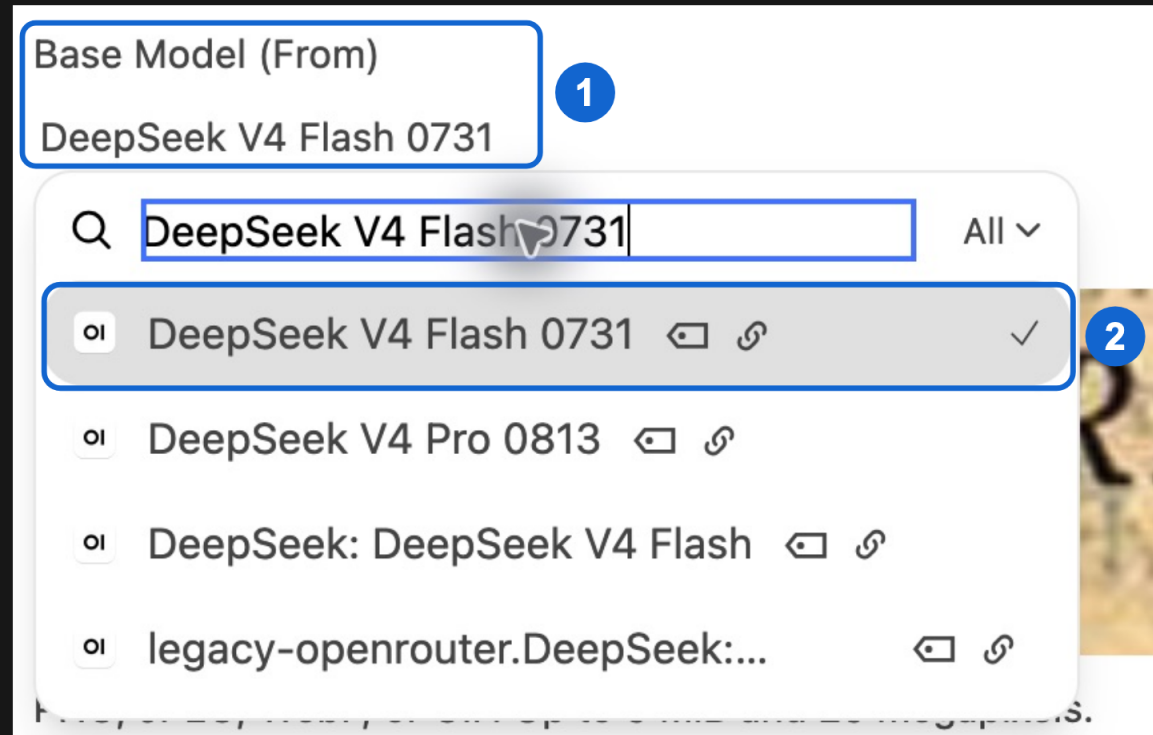
What Is a Tool?

When a model calls a **tool**, it is given the ability to execute a function and return its result, usually through a Python script

This is called **tool calling** (aka **function calling**)

Input	Information passed to a function
Execution	Code that applies specified rules
Result	Returned output for a model to inspect

Update Model for Tool Calling



Enable Filter

Enable **Model Bound Tool Enabler** on your model card, which binds assigned tools to the model's operations.

Filters

Select Filter

☒ Eco
 ☐ To select

Search filters

Enable all (3)

OpenAI GPT OSS Think and Formatter Patcher

Carbon Emission OSS120b

Model Bound Tool Enabler

Capabilities

☒ Vis
 ☐ We

☒ Terminal
 ☒ Usage
 ☐ Memory
 ☒ Status Updates

☒ File Context
 ☒ Code Interpreter
 ☒ Citations
 ☒ Builtin Tools

Add Your Skill

1. Open **Workspace** → **Models** → **Edit** → **Skills** → **Select Skill**
2. Choose **Adventure Panel Play** and read its saved instructions
3. Keep your model card open to add **Adventure Panel**

Skills Select Skill ← 1

To select

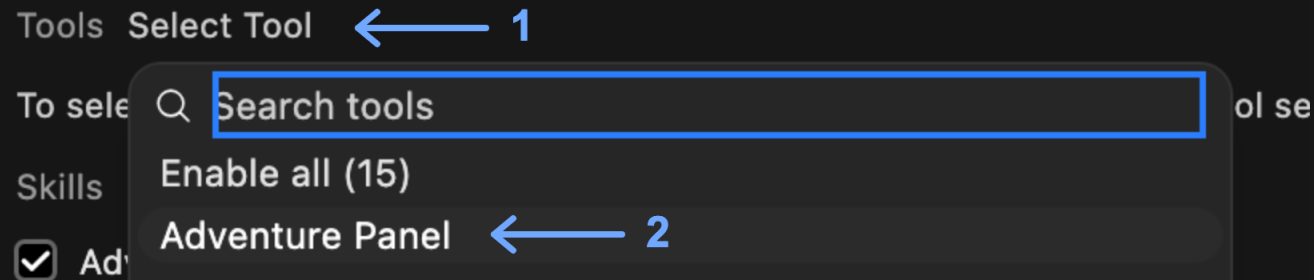
Enable all (13)

Filters Adventure Panel Play ← 2

☒ Eco Adventure Outcomes

Add the 'Adventure Panel' tool

1. Open **Tools** → **Select Tool** on the same model card
2. Choose **Adventure Panel** and keep **Adventure Panel Play** selected
3. Add the system-prompt instructions on slide 10, then select **Save & Update**



Model Card Check-In

Confirm these selections on your model card

1. **Adventure Panel** tool
2. **Adventure Panel Play** skill
3. **Model Bound Tool Enabler** filter

Keep your knowledge collection attached

Tools
Select Tool

1

☒ Adventure Panel
Disable all

To select toolkits here, add them to the "Tools" workspace or enable a tool server first.

Skills
Select Skill

2

☒ Adventure Panel Play
Disable all

To select skills here, add them to the "Skills" workspace first.

Filters
Select Filter

☒ Eco Impact Estimator
☒ Chat Metrics
3

☒ Model Bound Tool Enabler
Disable all

To select filters here, add them to the "Functions" workspace first.

Add Skill Instructions to the Bottom of Your System Prompt

[Copy prompt](#)

Load the attached Adventure Panel Play skill (extend-adventure-games) and follow it on every turn. Use `show_scene` from the Adventure Panel tool to display each scene and its choices. Follow this system prompt's setting, goals, rules, and source instructions.

Test Your Adventure

1. Run your updated adventure game
2. Verify configured skill and tool are called appropriately
3. Then play the rest of your game

Observe Game Results

1. Check the **Adventure Panel Play** skill call
2. Check the **Adventure Panel** tool call

Open call details to
confirm the skill and
show_scene function

initiate adventure

STEM Adventure Games (Clone)

4033 tokens | 95.02 sec

Explored 2 list_knowledge, view_skill, 1 dep_knowledge_files, 2 query_knowledge_files, 2 view_knowledge_file

● ADVENTURES ●

Welcome, explorer — four real experiments await, and your choices steer the science.

Explored show_scene 2

● ADVENTURES ●

Welcome, explorer. Each adventure is a documented experiment; your choices steer the method, the measurements, and the surprise. Pick a door.

1 Watch with Galileo Galilei as Jupiter's moons circle the giant planet

2 Boil broth with Louis Pasteur to test spontaneous generation

3 Weigh air with Antoine Lavoisier to hunt the gas of combustion

4 Develop Photo 51 with Rosalind Franklin and Raymond Gosling

Choose, then press Send.

The arcade door glows. Choose your experiment — the panel awaits your pick.

🌐 2 Sources

<

12 / 18

>

⌂

Outline

Explore Other Skills

1. Read **Adventure Outcomes** or **Conclude Adventure Game** after your first adventure works
2. Change one skill and save your model card
3. Run your adventure again and compare results

Skills Select Skill ← 1

To select

Filters

☒ Eco

To select

Enable all (13)

Adventure Panel Play

Adventure Outcomes

Conclude Adventure Game ← 2

History Of Ideas Abstract Critique

Choose Your Task

Revise One Item

1. Copy complete tool code or skill instructions
2. Describe an observed problem and one change

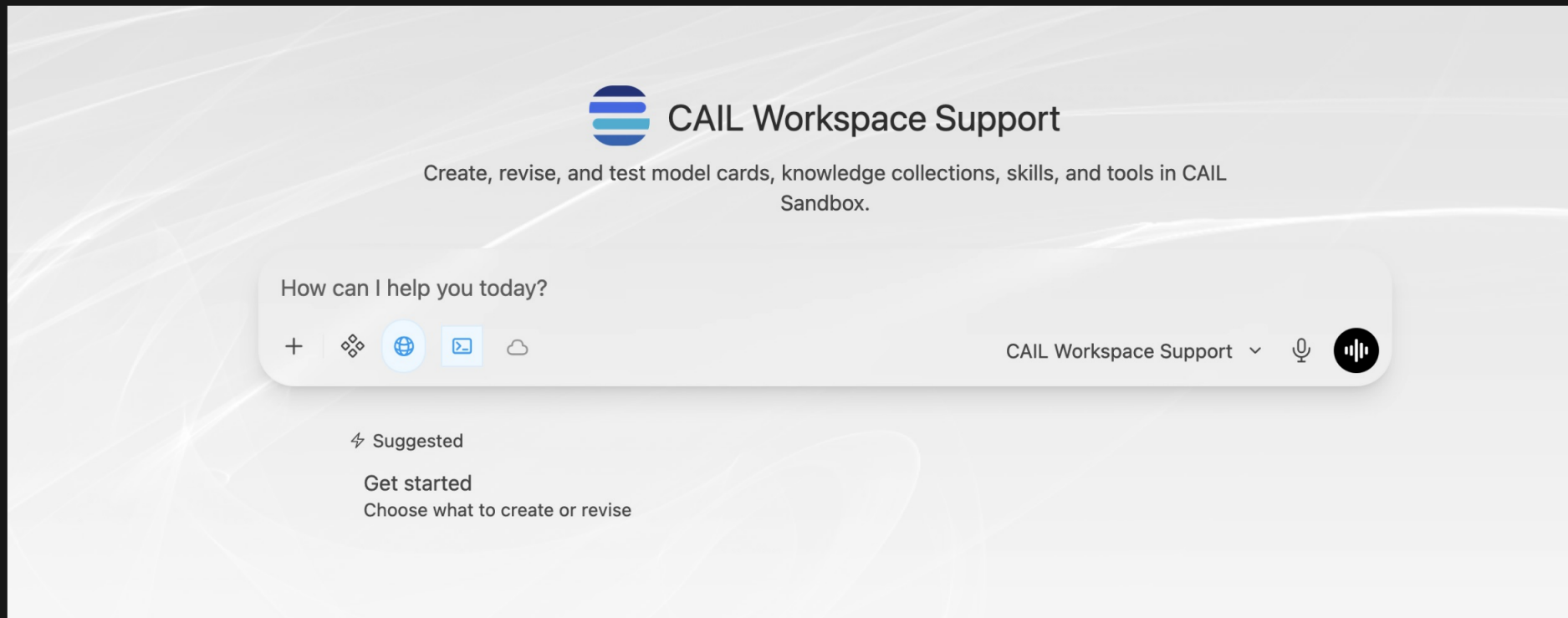
[Copy tool code](#) · [Skill instructions](#)

Create Your Own

1. Choose a task for your tool or skill
2. Specify its purpose, required input, and expected result

Work With Support

Prompt CAIL Workspace Support to generate a skill and/or tool that continues to support or further extends your adventure game, adapting an existing skill/tool or making your own.



Adapt Your Adventure

1. Give **CAIL Workspace Support** your lesson plan, learning outcome, or disciplinary focus
2. Ask it to adapt or create a skill or tool that extends your adventure toward that outcome

Add Tools or Skills

1. Copy and paste all code for tools or plain text for skills, then save your item
2. On your model card, select your new tool or skill, and signpost it in your system prompt

```
1 import os
2 import requests
3 from datetime import datetime
4 from pydantic import BaseModel, Field
5
6 class Tools:
7     def __init__(self):
8         pass
9
10    # Add your custom tools using pure Python code here, make sure to add type hints and descriptions
11
12    def get_user_name_and_email_and_id(self, __user__: dict = {}) -> str:
13        """
14        Get the user name, Email and ID from the user object.
15        """
16
17        # Do not include a description for __user__ as it should not be shown in the tool's specification
18        # The session user object will be passed as a parameter when the function is called
19
20        print(__user__)
21        result = ""
22
23        if "name" in __user__:
24            result += f"User: {__user__['name']}"
25
26        if "id" in __user__:
27            result += f" (ID: {__user__['id']})"
28
29        if "email" in __user__:
30            result += f" (Email: {__user__['email']})"
31
32        if result == "":
33            result = "User: Unknown"
```

Warning: Tools can execute arbitrary code. Only install tools from sources you trust.

Save & Create



CUNY AI Lab | Coworking Sessions ➡

Continue your Sandbox and Vibe Coding projects in coworking sessions, with time to build and troubleshoot together.

Workshop materials and guides ➡

Molnár · Quimbot / Creative Clawing