

A Step-by-Step Guide to Building Your First AI Agent with n8n

Download file: [n8n.json](#)

Introduction

AI agents are a powerful and rapidly evolving technology that can reason, plan, and take autonomous actions. While they may seem complex, building a basic AI agent is more accessible than you might think, even without coding experience. This guide will walk you through the fundamental concepts of AI agents and provide a step-by-step tutorial on how to build your own personal assistant using the low-code platform, n8n.

Part 1: Understanding AI Agents

Before we start building, let's clarify what an AI agent is and its core components.

What is an AI Agent?

An AI agent is a system designed to understand its environment, make decisions, and take actions to achieve specific goals. Think of it as a digital employee that can think, remember, and get things done.

- **Agent vs. Automation:**

- **Automation:** Follows a predefined, static set of rules to execute a task (e.g., check the weather every morning and send an email). It operates on a fixed A -> B -> C sequence.
- **AI Agent:** Dynamically reasons and adapts its actions based on the situation. If asked, "Should I bring an umbrella today?", it will independently decide to check the weather, interpret the data, and formulate a relevant answer.

The Three Key Components of an AI Agent

1. **The Brain (Large Language Model - LLM):** This is the core engine for reasoning, planning, and language generation. Examples include OpenAI's GPT models, Google's Gemini, and Anthropic's Claude.
2. **Memory:** This gives the agent the ability to recall past interactions and information, allowing for context-aware decisions. This can range from remembering the last few messages in a conversation to accessing external documents or databases.
3. **Tools:** These are the agent's connection to the outside world, enabling it to perform actions. Tools can:
 - **Retrieve Data:** Search the web, pull information from a document, or query a database.
 - **Take Action:** Send an email, update a spreadsheet, or create a calendar event.
 - **Orchestrate:** Trigger other workflows or call upon other agents.

Part 2: Essential Technical Concepts (The Simple Version)

You'll encounter these terms when working with agents. Here's what they mean in plain English.

- **API (Application Programming Interface):** Think of an API as a restaurant menu. It lists all the possible orders (actions) you can make. It's how different software applications communicate with each other.
- **HTTP Request:** This is the act of placing an order from the menu. When an agent needs to get information or send it to another service, it makes an HTTP request to that service's API. The most common types are:
 - **GET:** To retrieve information (e.g., check the weather).
 - **POST:** To send information (e.g., submit a form or send a prompt to an LLM).

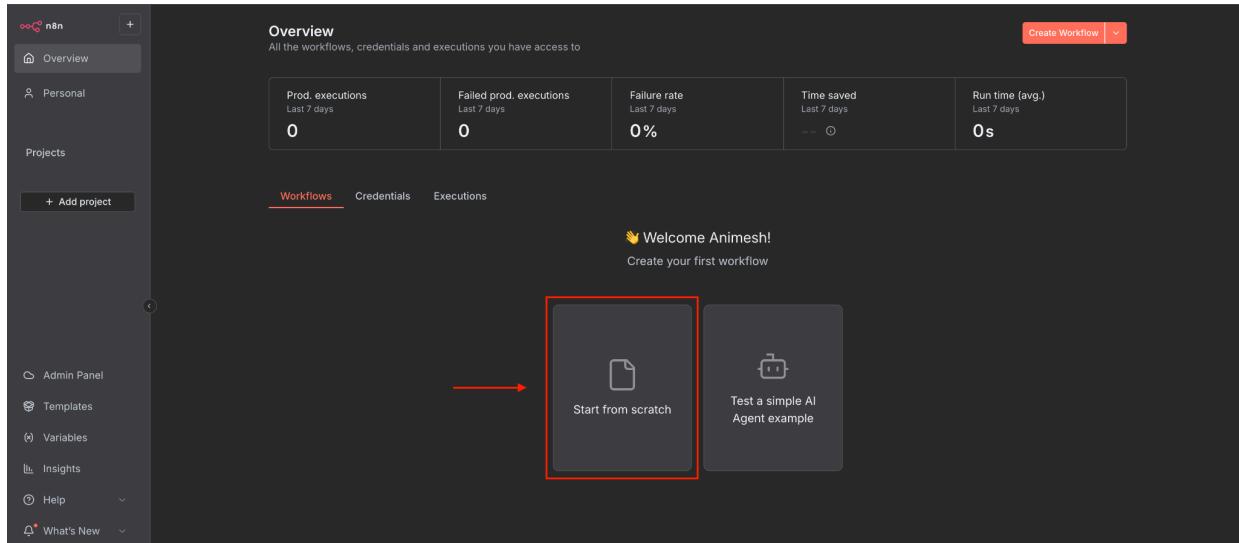
- **n8n:** This is a visual workflow automation tool that allows you to build automations and agents by connecting different nodes (blocks) together. It simplifies the process of using APIs and building complex workflows without writing code.

Part 3: Step-by-Step Guide to Building a Trail Recommendation Agent in n8n

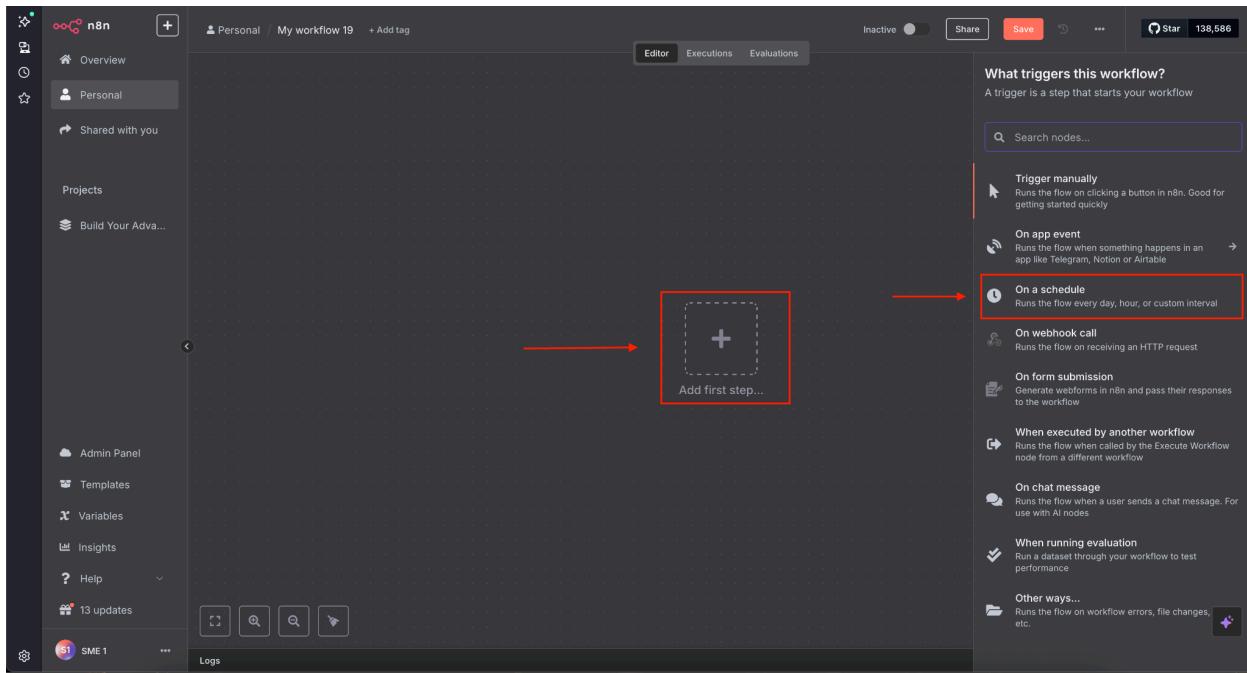
1. Runs on a daily schedule.
2. Checks your calendar for a "trail run" event.
3. Checks the current weather and air quality.
4. Consults a Google Sheet with a list of local trails.
5. Recommends a suitable trail based on the conditions and your available time.
6. Sends you an email with the recommendation.

Step 1: Set Up Your n8n Workflow

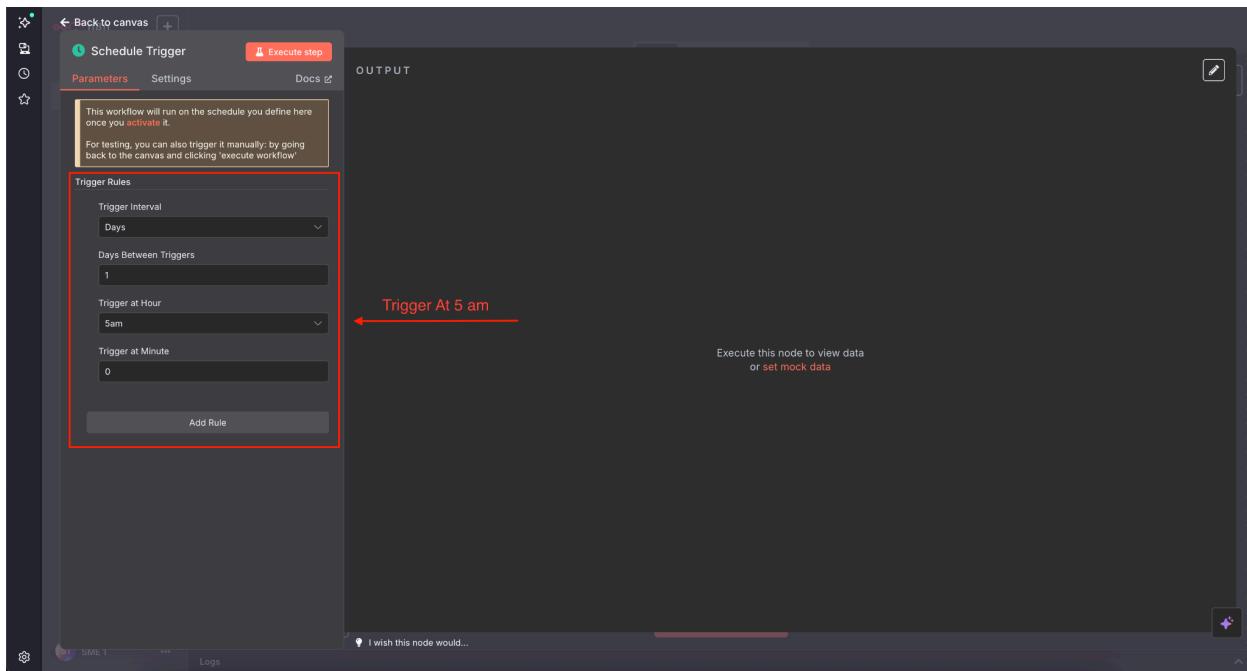
1. Open your n8n dashboard and click "**Start from scratch**" to create a new workflow.



2. Click "Add first step" and select the "On a schedule" trigger.

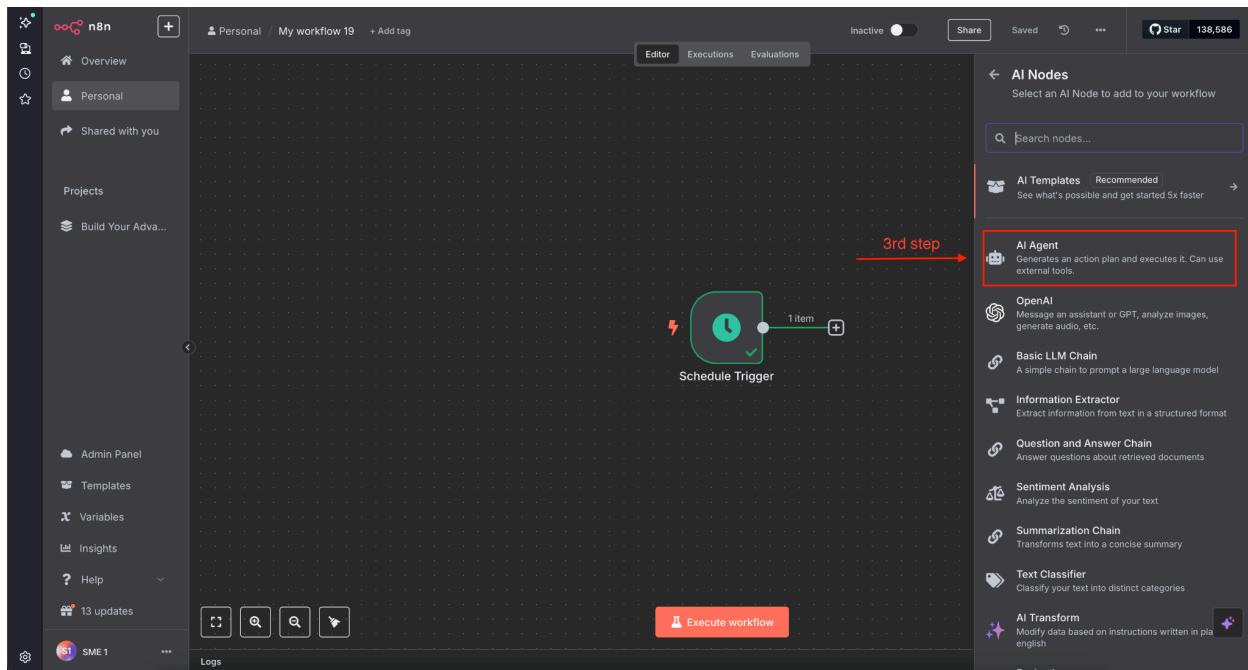
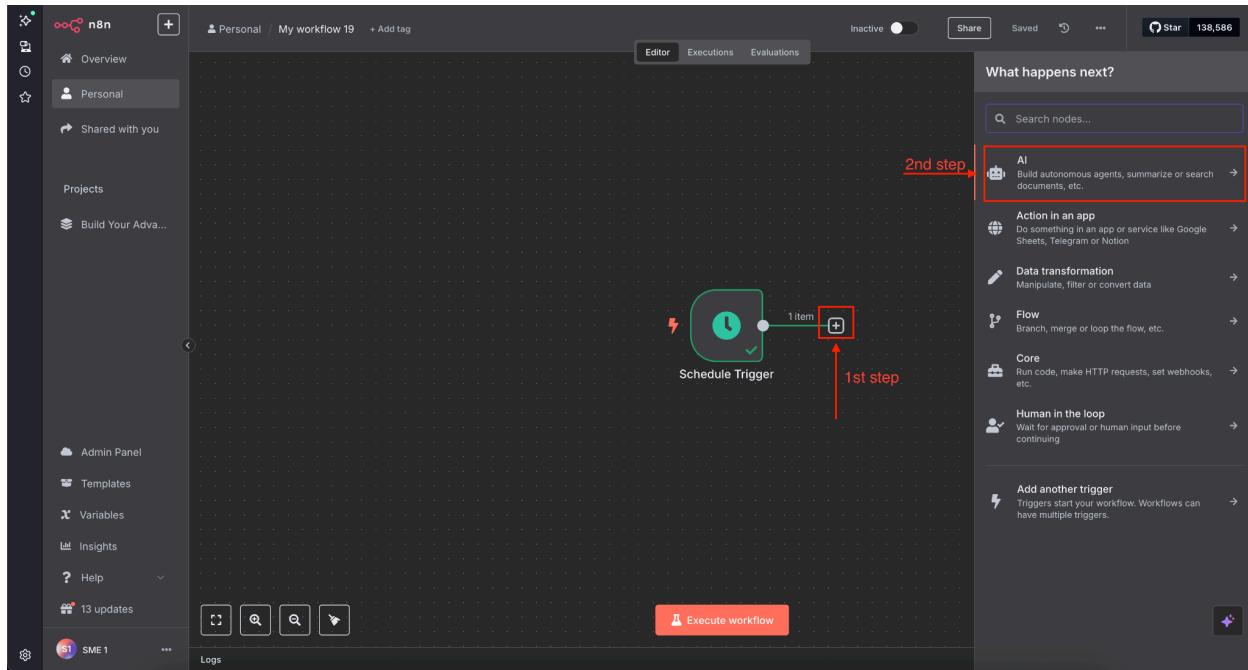


3. Configure the schedule to run at your desired time (e.g., 5:00 AM daily).

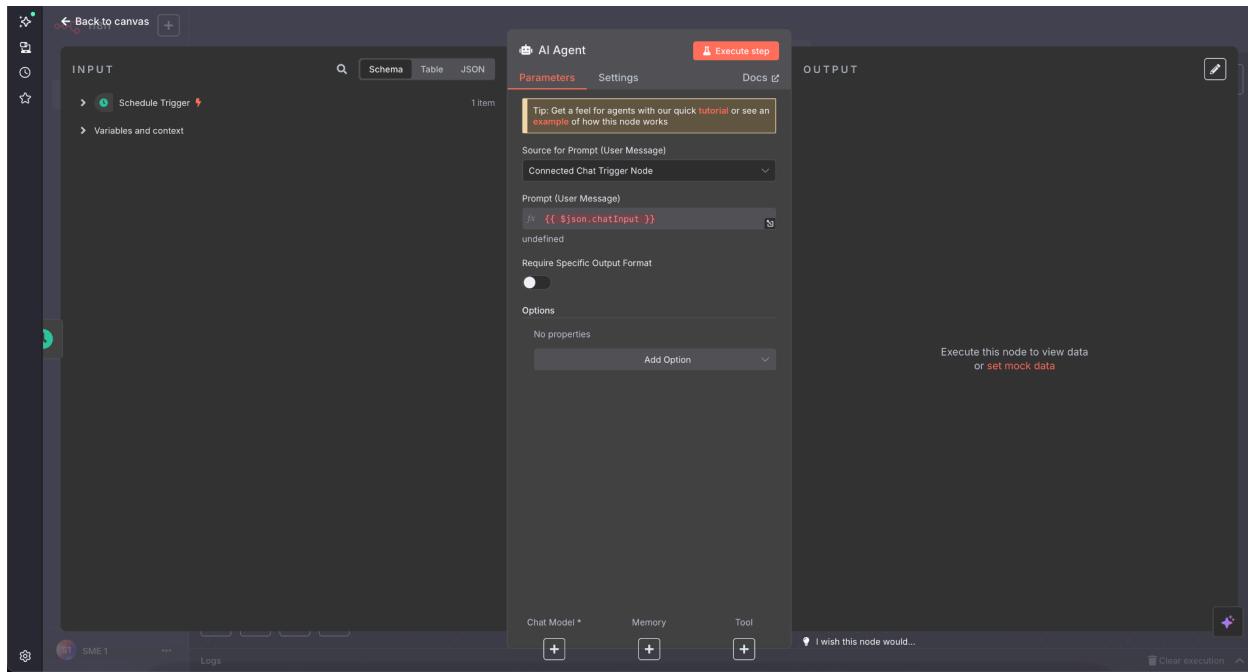


Step 2: Add and Configure the AI Agent Node

1. Click the "+" icon to add a new node. Search for and select "AI Agent".

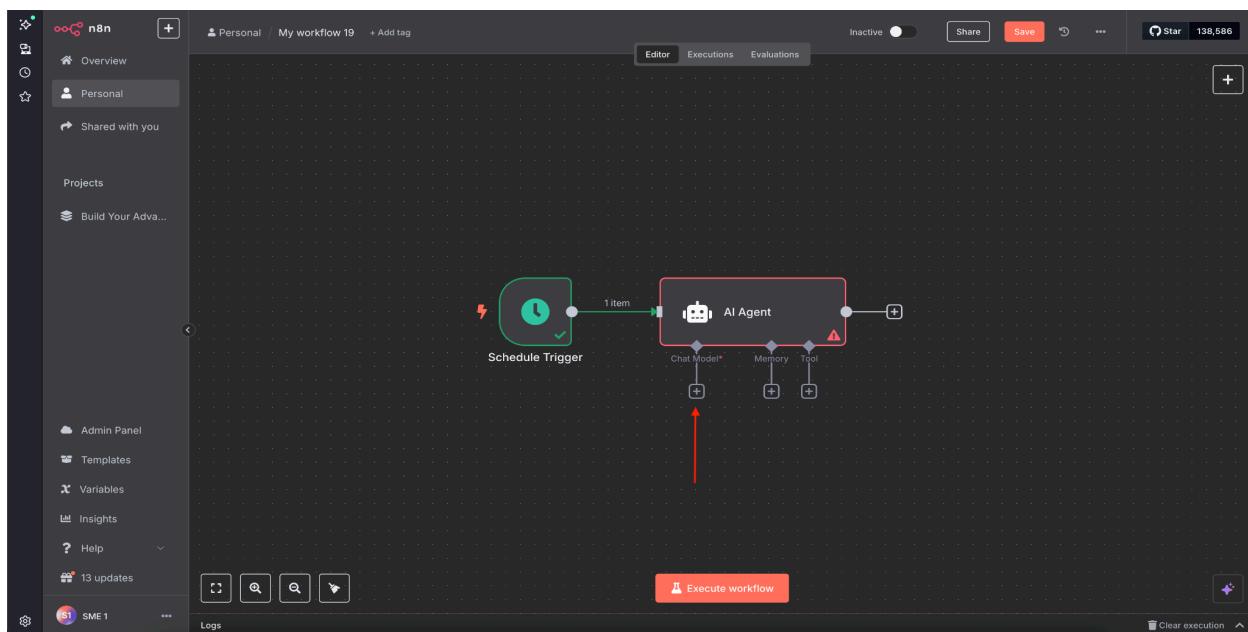


- This single node will serve as the central hub for your agent, connecting the brain, memory, and tools.

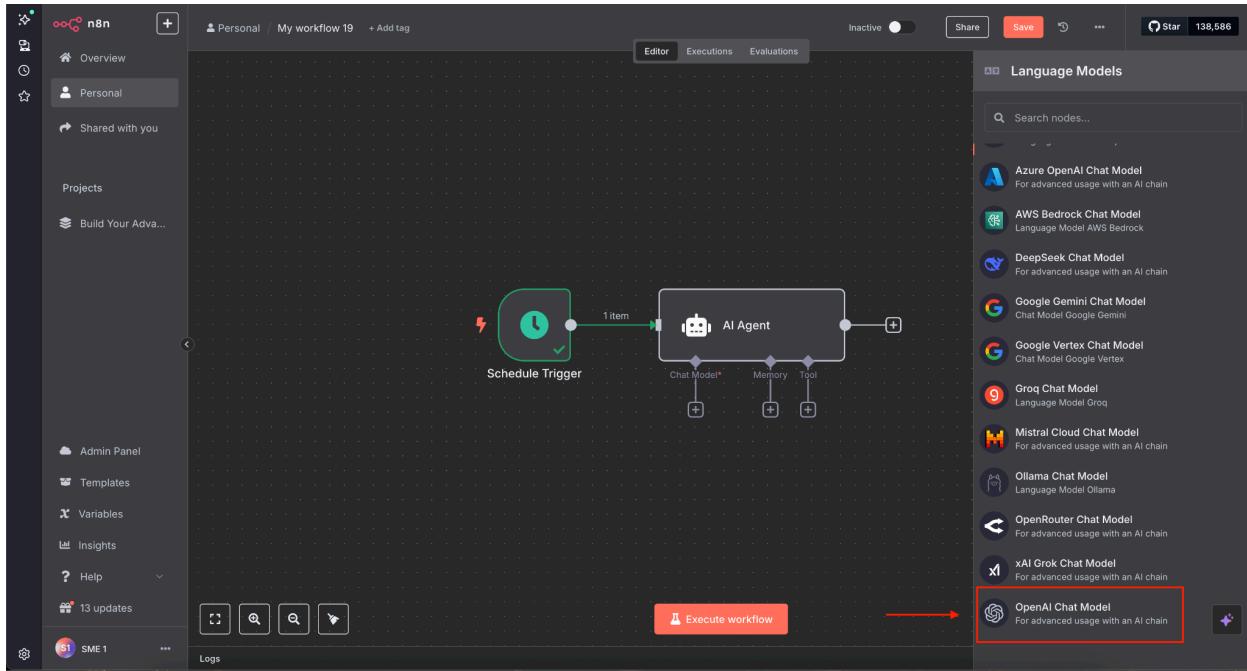


Step 3: Connect the "Brain" (Your LLM)

- In the AI Agent node settings, locate the **Chat Model** section and click the "+" icon.

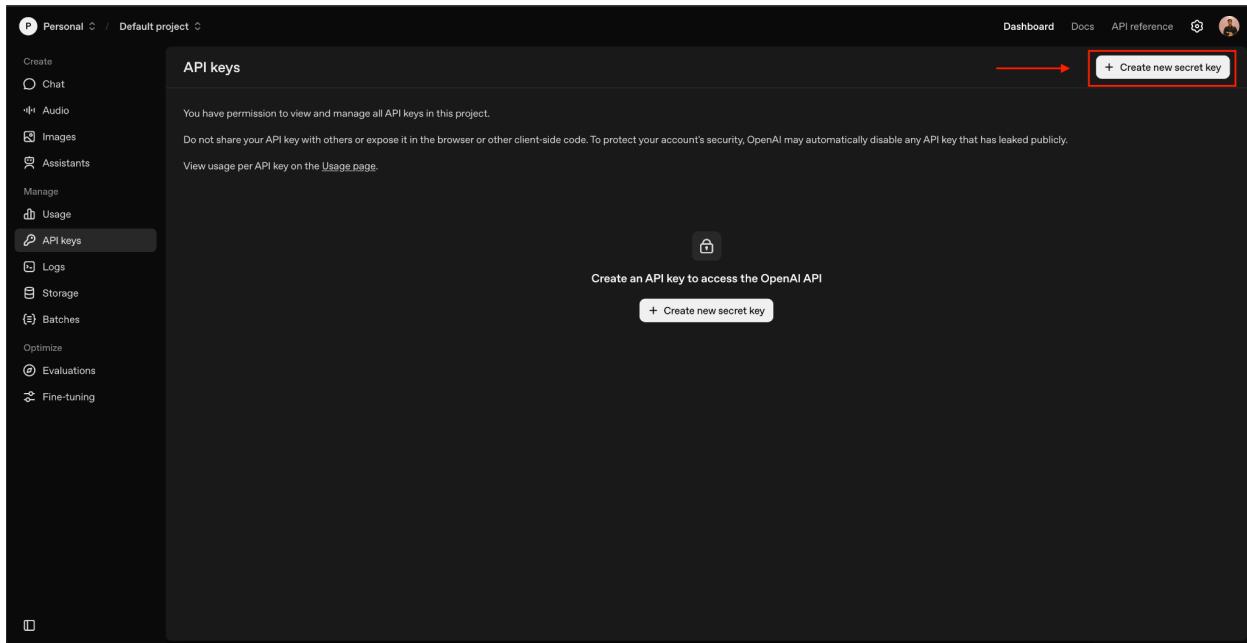


2. Select your preferred LLM (e.g., OpenAI).

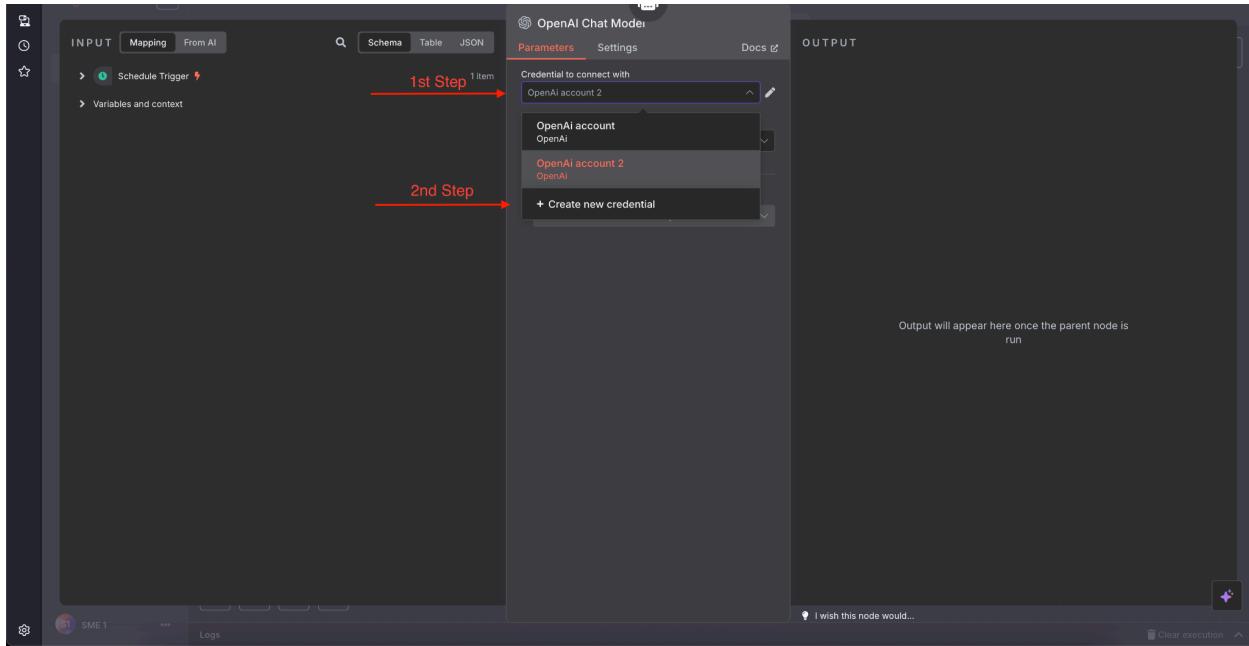


3. Click "Create new credentials" and enter your API key.

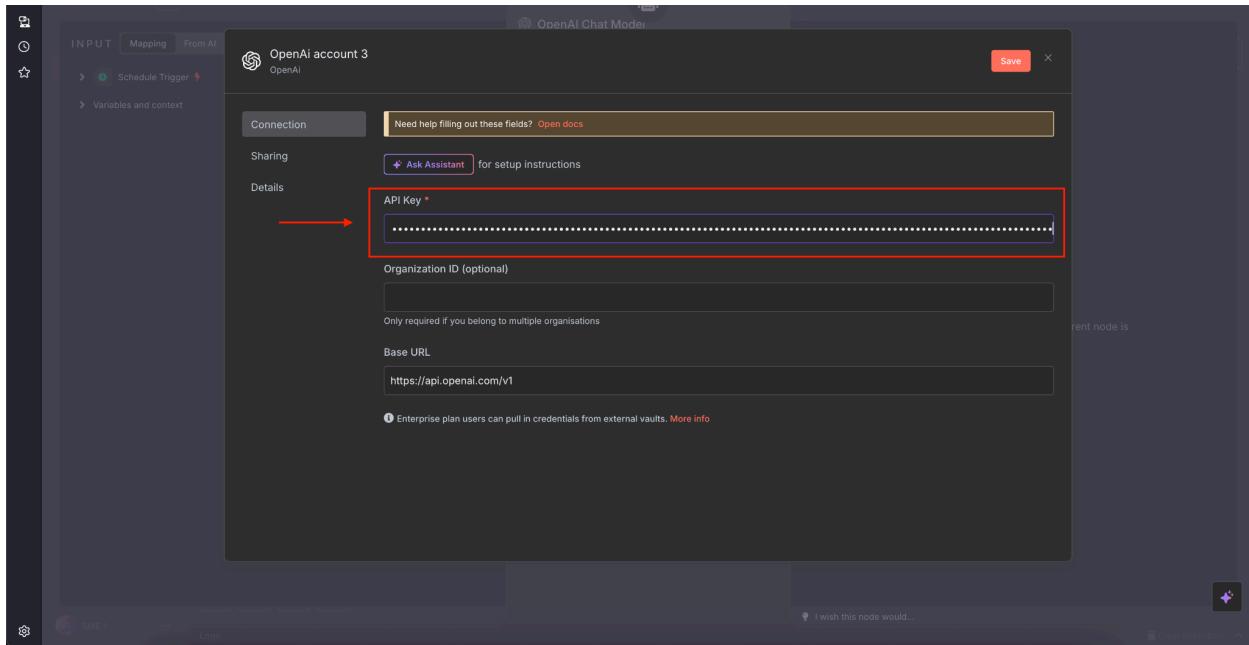
- To get an OpenAI API key, go to platform.openai.com, navigate to "API Keys," and generate a new secret key.



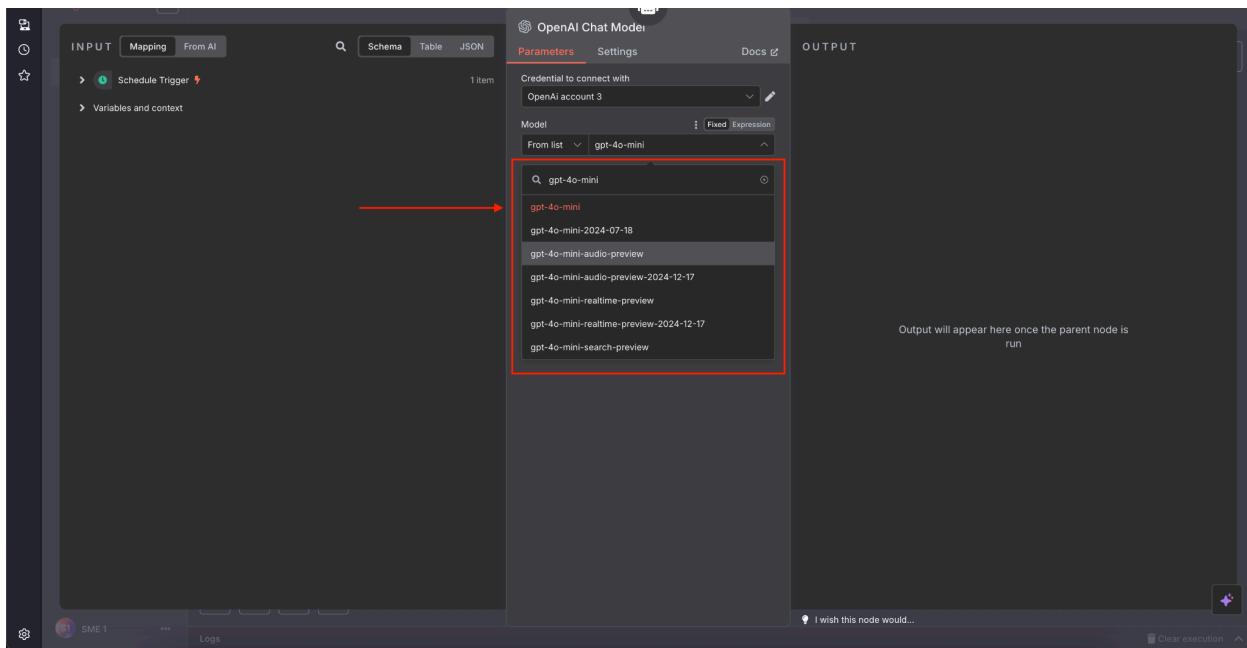
4. Click on Credential to connect with and select Create new credential



5. Enter the API Key and click on Save.

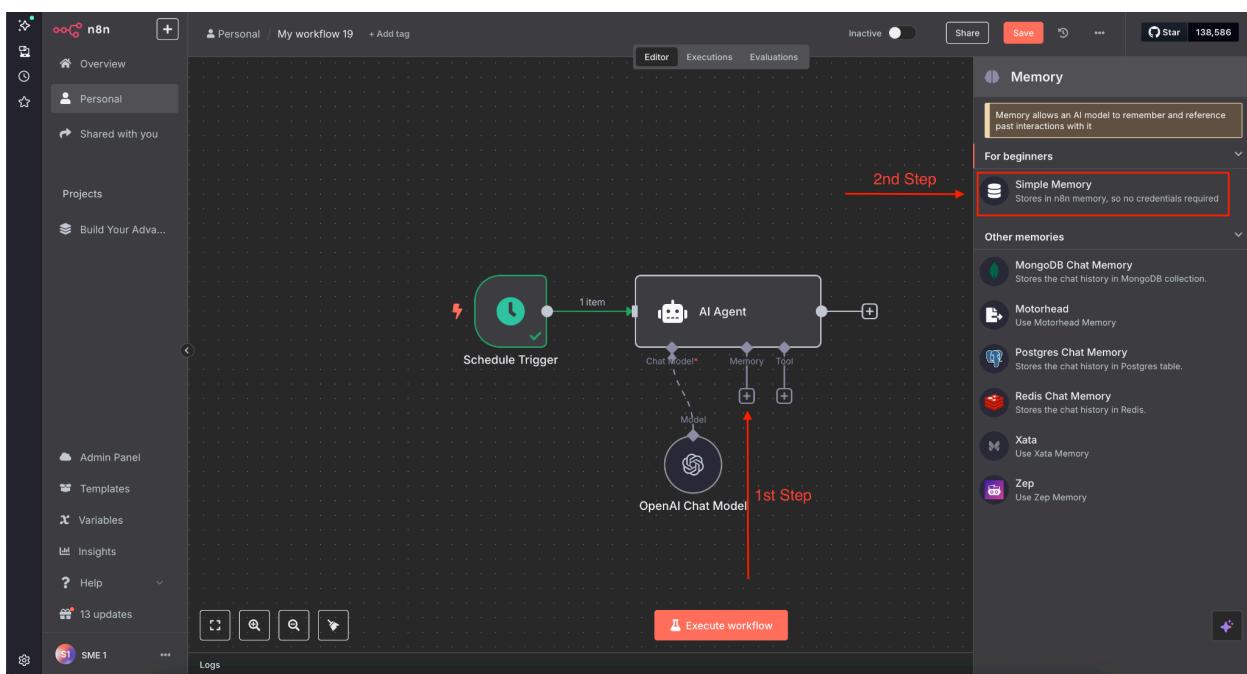


6. Choose a model from the dropdown menu (e.g., **GPT-4o Mini** is a good, cost-effective choice).

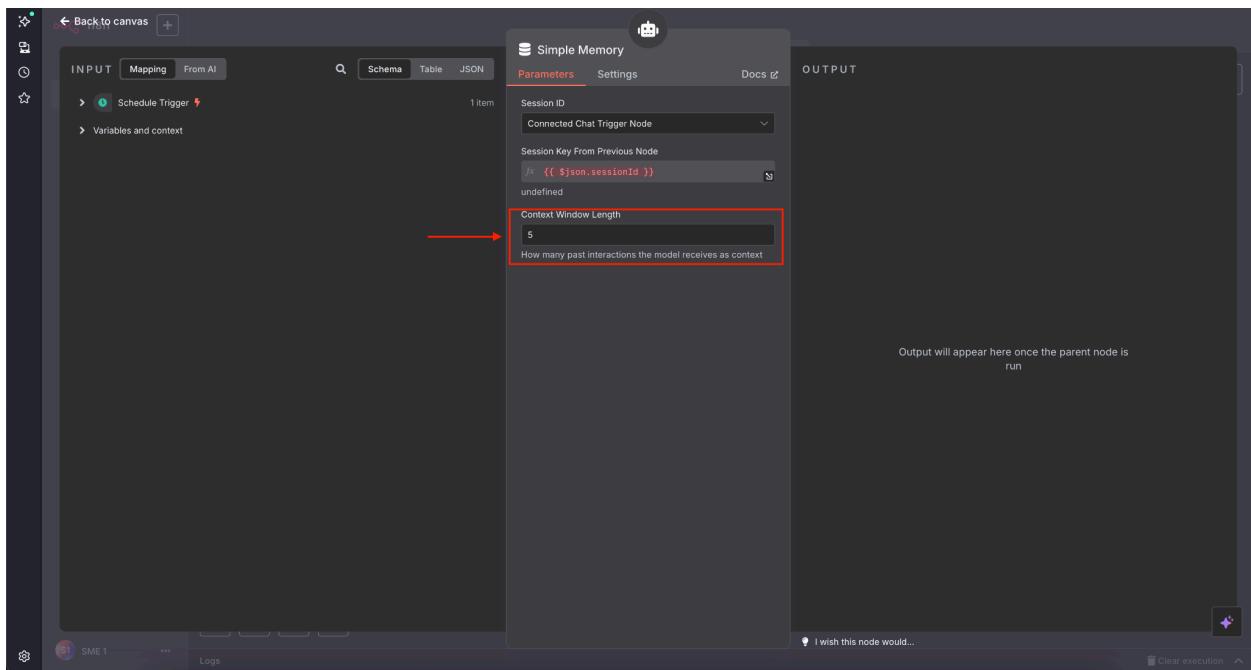


Step 4: Set Up the "Memory"

1. In the AI Agent node, find the **Memory** section and click the "+" icon.



2. Select the "**Simple Memory**" option. This is sufficient for remembering context within a single run of the workflow.
3. Leave the context window at the default setting (e.g., 5). This means the agent will remember the last five steps or messages. (In the session Id Enter a Fixed string like : trail-run-session)

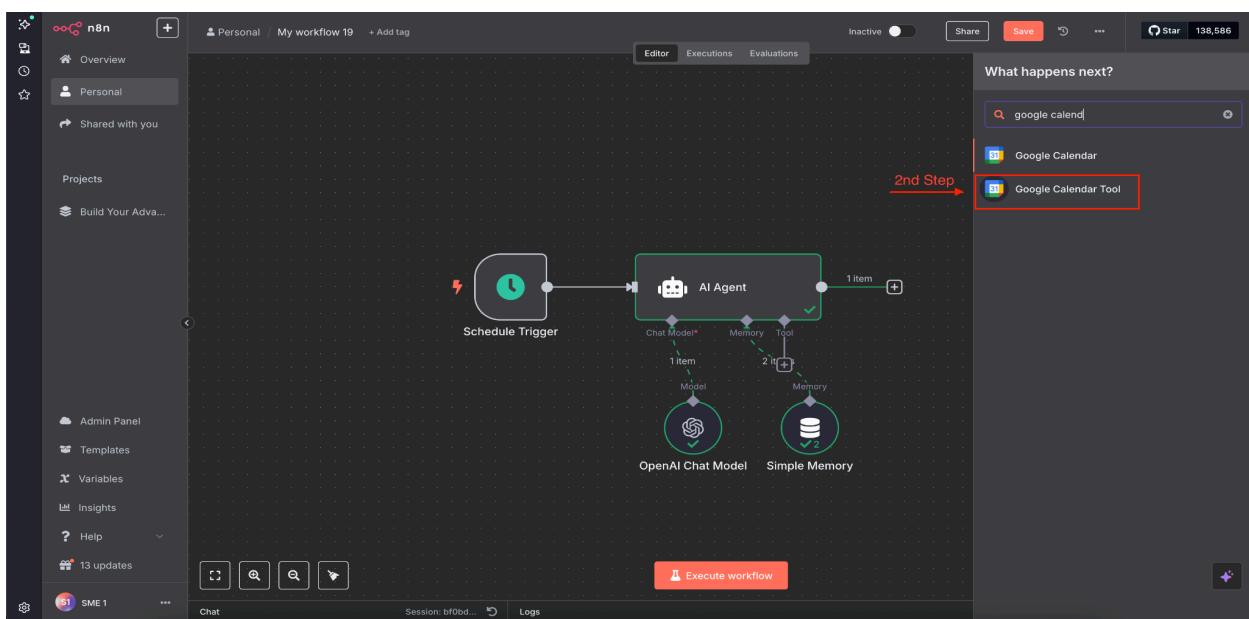
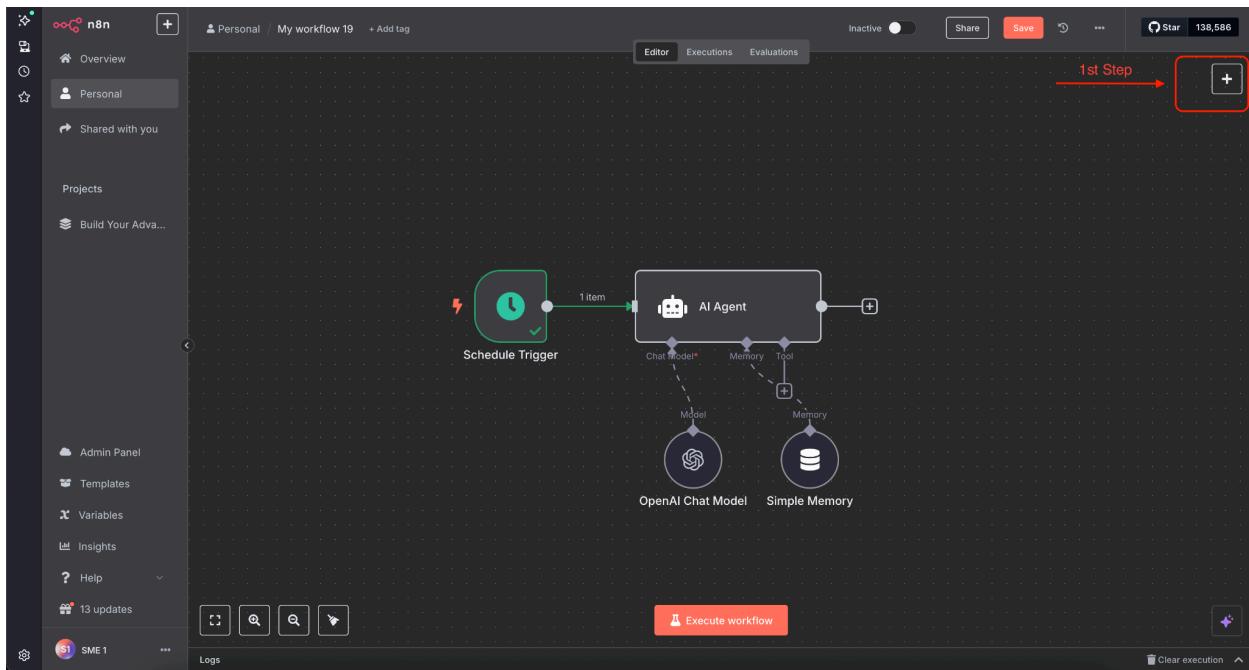


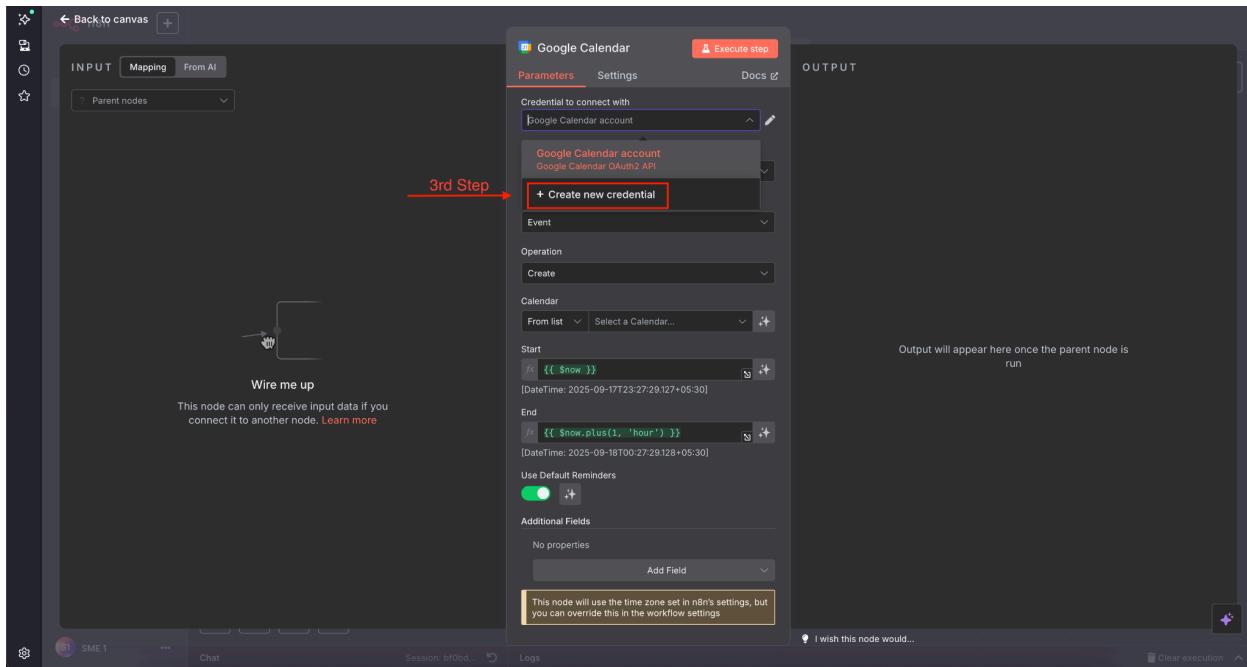
Step 5: Add and Configure the "Tools"

Each tool will be a sub-node connected to the main AI Agent node.

1. Google Calendar (To check your schedule):

- Click the "+" on the AI Agent node to add a tool and search for **Google Calendar tool**.

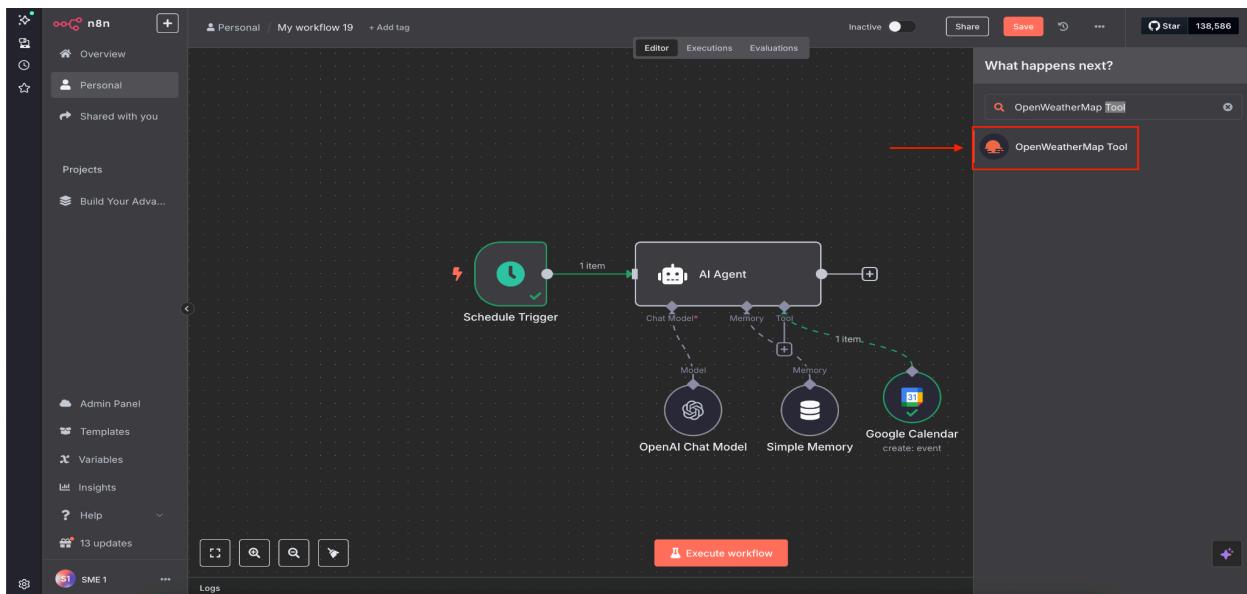


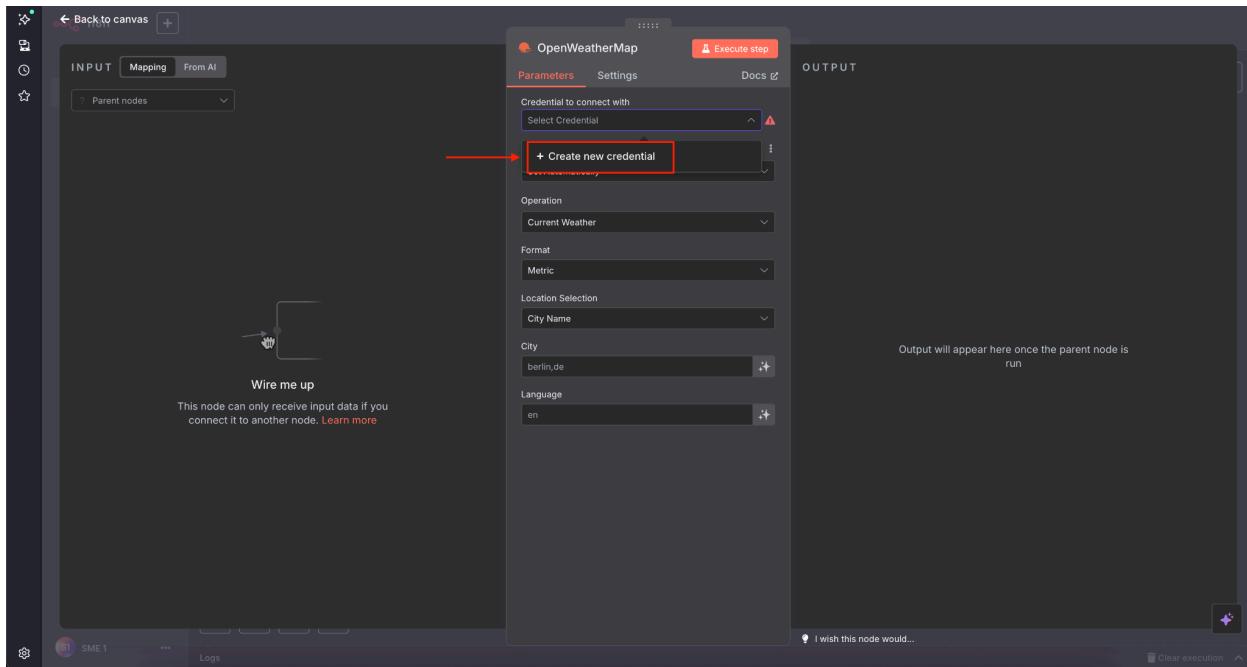


- Sign in with your Google account and grant the necessary permissions.
- Ensure the correct calendar is selected.

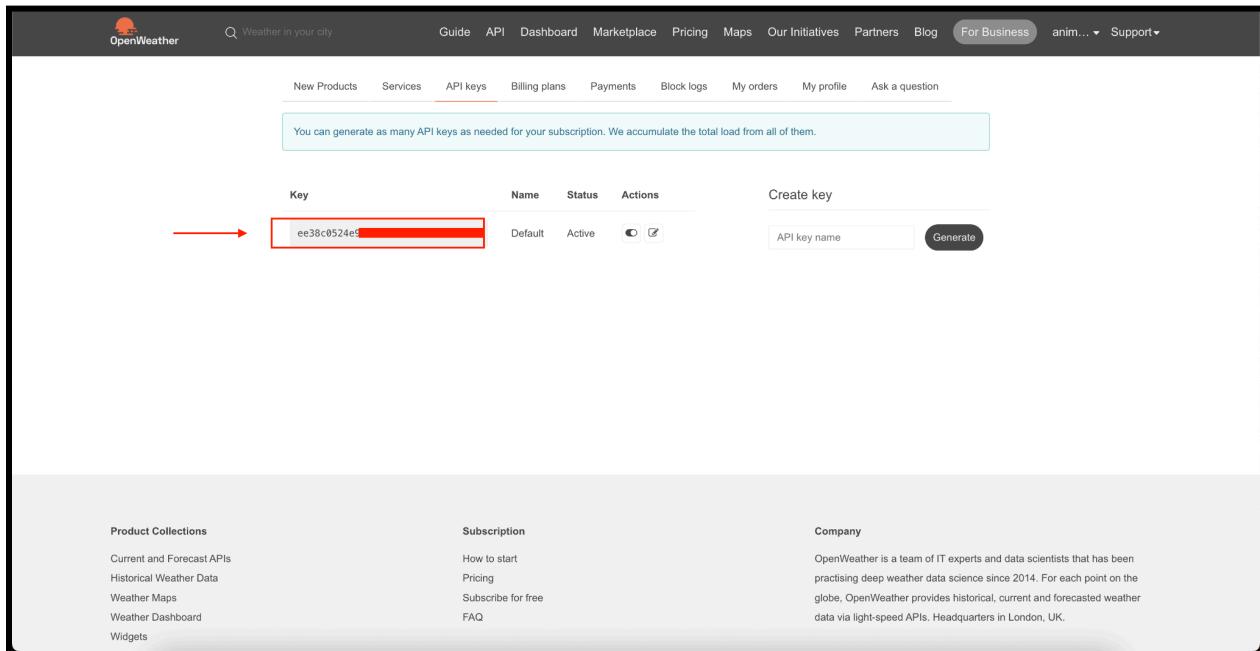
2. OpenWeatherMap (To get weather data):

- Add a new tool and search for **OpenWeatherMap Tool**.

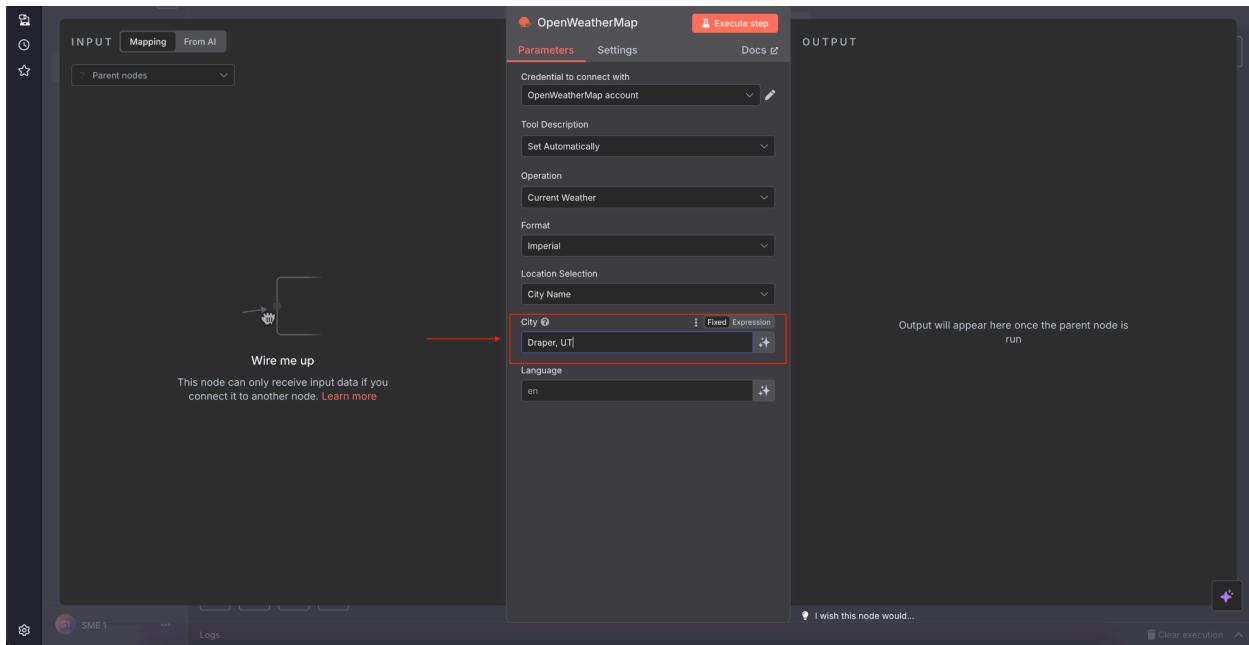




- You will need an API key. Create a free account at openweathermap.org, navigate to "My API keys," and generate one.

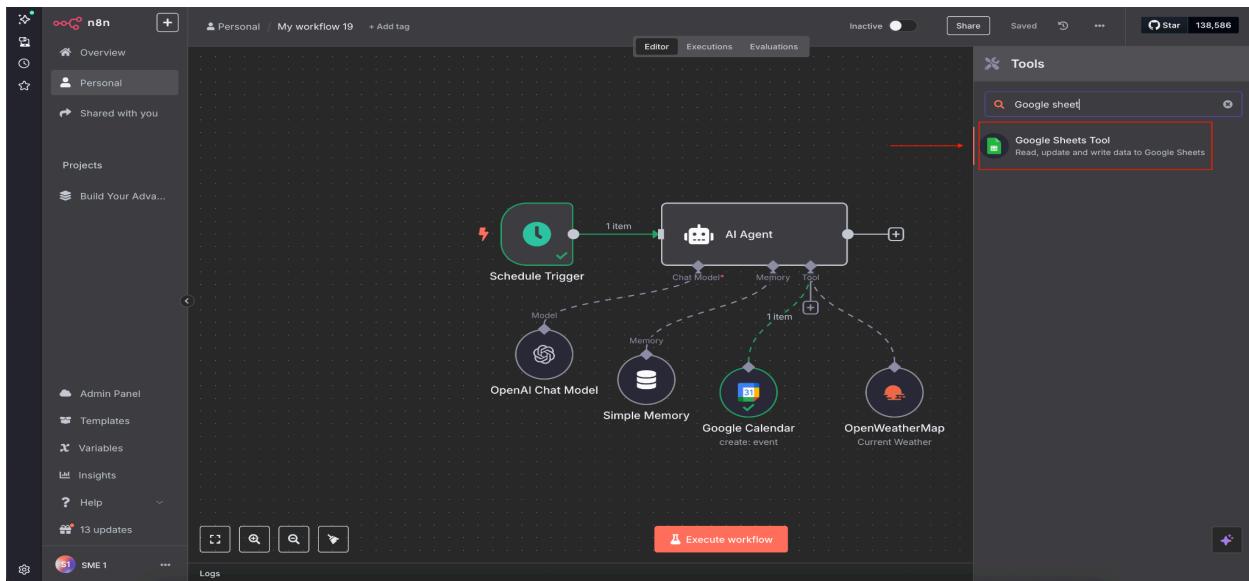


- Paste the key into the credentials field in n8n.
- Set your preferred units (e.g., Imperial for Fahrenheit) and enter a nearby city.



3. Google Sheets (To access your trail list):

- Add the Google Sheets tool.



- Connect your Google account.
- In a separate Google Sheet, create a list of trails with columns like Trail Name, Mileage, Time Estimate, etc.

The screenshot shows a Google Sheets document with the title "Trails". The spreadsheet has a header row with columns labeled A through N. The data starts at row 1 and continues down to row 21. The columns are labeled as follows:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Name	Miles	Elevation	Estimated Time	Shade Level									
2	Sensei Trail	10.8	1568	2h 15m	Exposed									
3	Rose Canyon to Yellow Fork	9.4	2004	2h 05m	Exposed									
4	Bonneville Shoreline, Ann's, Maple	10.5	1466	2h 15m	Exposed									
5	Bonneville Shoreline to Betty to Dr	7.3	1453	1h 40m	Exposed									
6	Curley Springs to Dry Canyon	5	1407	1h 15m	Some Shade									
7	Juniper Hill via Mahogany Bench	7.1	1984	1h 45m	Exposed									
8	Grove Creek and Battle Creek Loo	7.9	2611	2h 10m	Some Shade									
9	Lone Peak Wilderness Trail	5.4	636	1h 05m	Some Shade									
10	Three Falls Trail	6	1023	1h 20m	Some Shade									
11	Longview, Peacemaker, Peak View	8.2	1112	1h 45m	Exposed									
12	Mount Olympus	6.9	4087	2h 35m	Exposed									
13	Ferguson Canyon to Upper Meado	4.6	2296	1h 35m	Shady									
14	Storm Mountain via Ferguson Can	7.5	4278	2h 45m	Shady									
15	Lower Falls via Bell Canyon	4.8	1515	1h 20m	Some Shade									
16	Lower Bell Canyon via Larry's Trail	2.4	495	30m	Some Shade									
17	Dimple Dell Loop	6	656	1h 20m	Some Shade									
18	Bonneville Shoreline: Highlands of	6	629	1h 15m	Exposed									
19	Ghost Falls from Draper	6	994	1h 25m	Some Shade									
20	Bonneville Shoreline from Coyote I	5.6	902	1h 15m	Exposed									
21	Canyon Hollow, Brocks Point, Ann'	5.8	905	1h 20m	Some Shade									

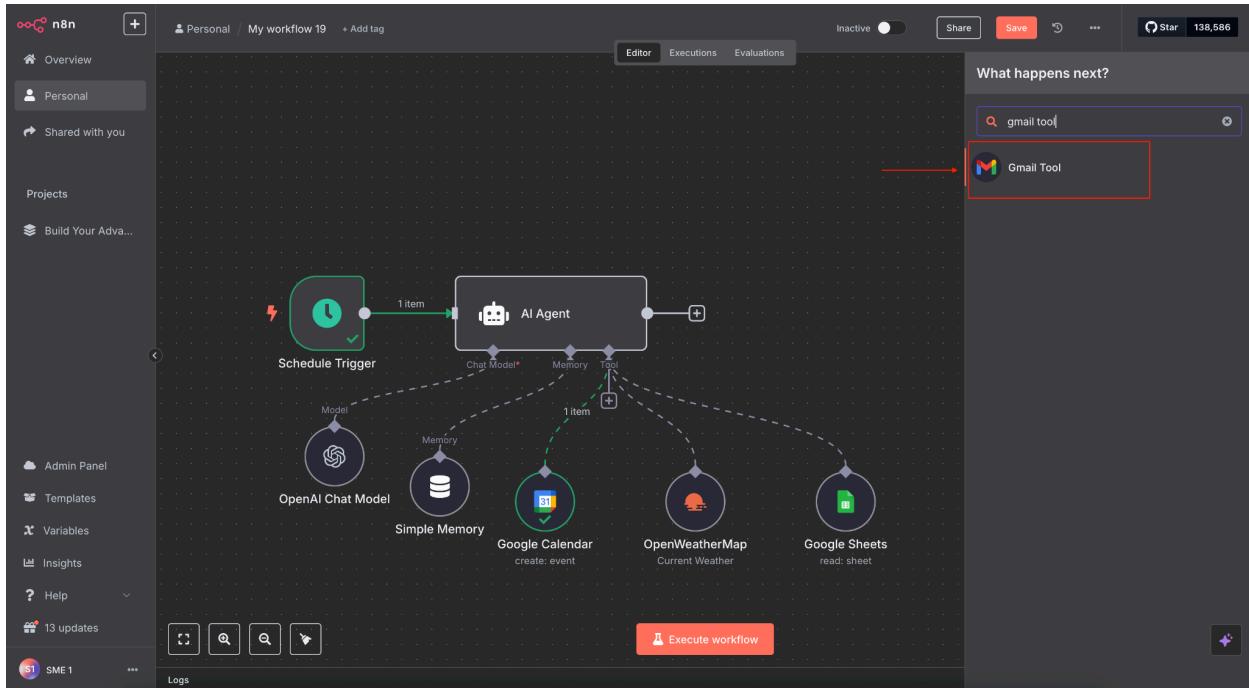
- Download the sheet : [Sheet](#)
- In the n8n node settings, select the correct spreadsheet and sheet name.

The screenshot shows the n8n node editor with a "Google Sheets" node selected. The node has several configuration options:

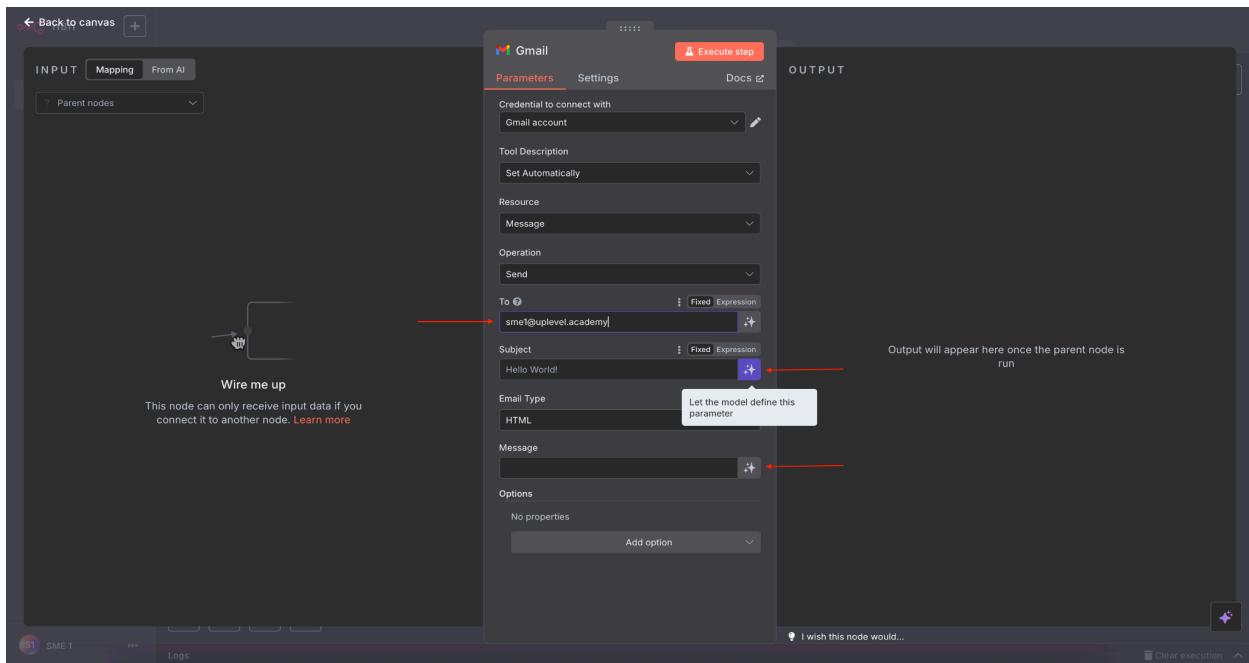
- INPUT:** A "Schedule Trigger" node is connected to the "From AI" tab.
- Parameters:**
 - Credential to connect with: Google Sheets account 2
 - Tool Description: Set Automatically
 - Resource: Sheet Within Document
 - Operation: Get Row(s)
 - Document: From list Trails (highlighted with a red box)
 - Sheet: From list Runs (highlighted with a red box)
- OUTPUT:** An output placeholder: "Output will appear here once the parent node is run".
- Filters:** Currently no items exist.
- Combine Filters:** AND.
- Options:** No properties.

4. Gmail (To send the recommendation):

- Add the **Gmail tool** and connect your account.

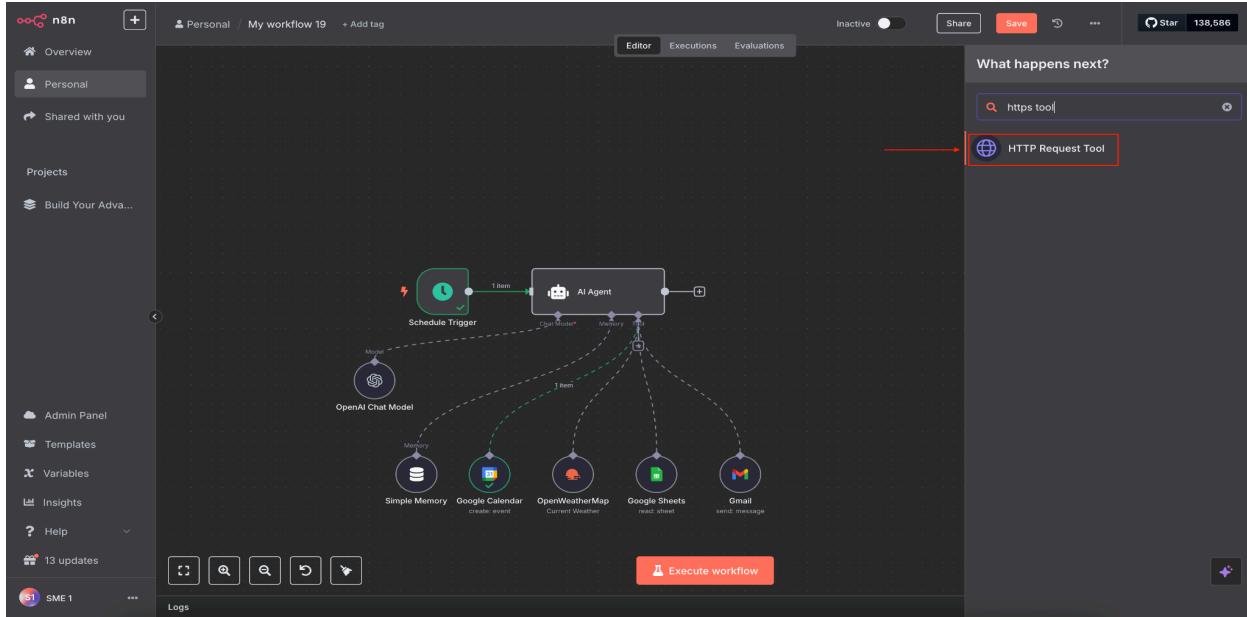


- In the node settings, set the recipient email address (your own).
- For the **Subject** and **Message**, select the option "**Let the model define this parameter.**" This allows the AI to generate a custom email body based on its findings.



5. HTTP Request (To get air quality data):

- Add the **HTTP Request** tool. This is for services without a pre-built n8n integration.



- Go to a site like <https://docs.airnowapi.org/> to get an API URL for air quality data, which usually involves signing up for a key and using their tools to generate a request URL.
- Go to **web services** select format : **application/json** and click on build

The screenshot shows the AirNow Developer Tools Web Services page. The top navigation bar includes Home, Web Services (selected), Data Feeds, File Products, FAQ, Air Quality 101, and Analytics. A "Log Out" button is in the top right. Below the navigation, it says "Web Services > Current Observation By Zip Code" and "Your API Key: 79BAA4BF-3DA7-435B-A98B-FA475159DBD6". Step 1 shows a form with "Zip Code: 84020", "Distance: 25 miles", and a "Format: application/json" dropdown menu. A red box highlights the "Format" dropdown, and another red box highlights the "Build" button. Step 2 shows a "Generated URL" field with a "Run" button below it. Step 3 shows an "Output" field. Red arrows point from the "Format" dropdown and the "Build" button towards the "Generated URL" field, indicating the sequence of actions: select format, click build, and then run the generated URL.

The screenshot shows the AirNow Developer Tools interface. At the top, there's a navigation bar with links for Home, Web Services, Data Feeds, File Products, FAQ, Air Quality 101, and Analytics. The "Web Services" tab is active. On the right side of the header, it says "Your API Key: 79BAA4BF-3DA7-435B-A98B-FA475159DBD6" and a "Log Out" button. Below the header, the page title is "Web Services > Current Observation By Zip Code". The main content area has three numbered steps:

- 1** Zip Code: 84020 Distance: 25 miles Format: application/json Build
- 2** Generated URL
https://www.airnowapi.org/aq/observation/zipCode/current/?for=A98B-FA475159DBD6 Run
- 3** Output
[{"DateObserved": "2025-09-17", "HourObserved": 11, "LocalTimeZone": "MST", "ReportingArea": "Timpanogos Cave National Monument", "StateCode": "UT", "Latitude": 40.4419, "Longitude": -111.7134, "ParameterName": "PM2.5", "AQI": 10, "Category": {"Number": 1, "Name": "Good"}]

- Paste the generated URL into the **URL** field in the HTTP Request node.
- Check the box for "**Optimize Response**" to help the LLM process the data more easily.

The screenshot shows a Node-RED canvas with a single node: an "HTTP Request" node. The node has the following configuration:

- Method:** GET
- URL:** https://www.airnowapi.org/aq/observation/zipCode/cu
- Options:** Optimized Response is checked (highlighted with a red box)

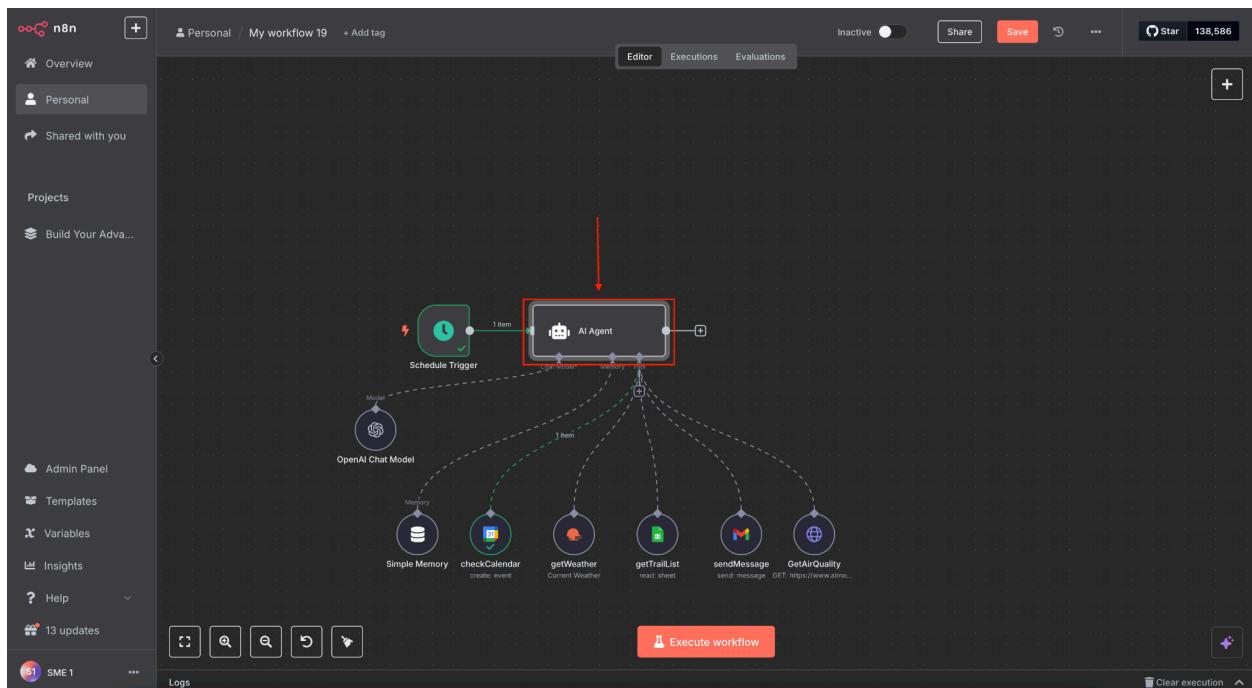
Red arrows point from the numbered steps in the previous screenshot to the URL field and the "Optimize Response" checkbox in the Node-RED node configuration.

- Rename all your tool nodes for clarity (e.g., "Get Weather," "Check Calendar").

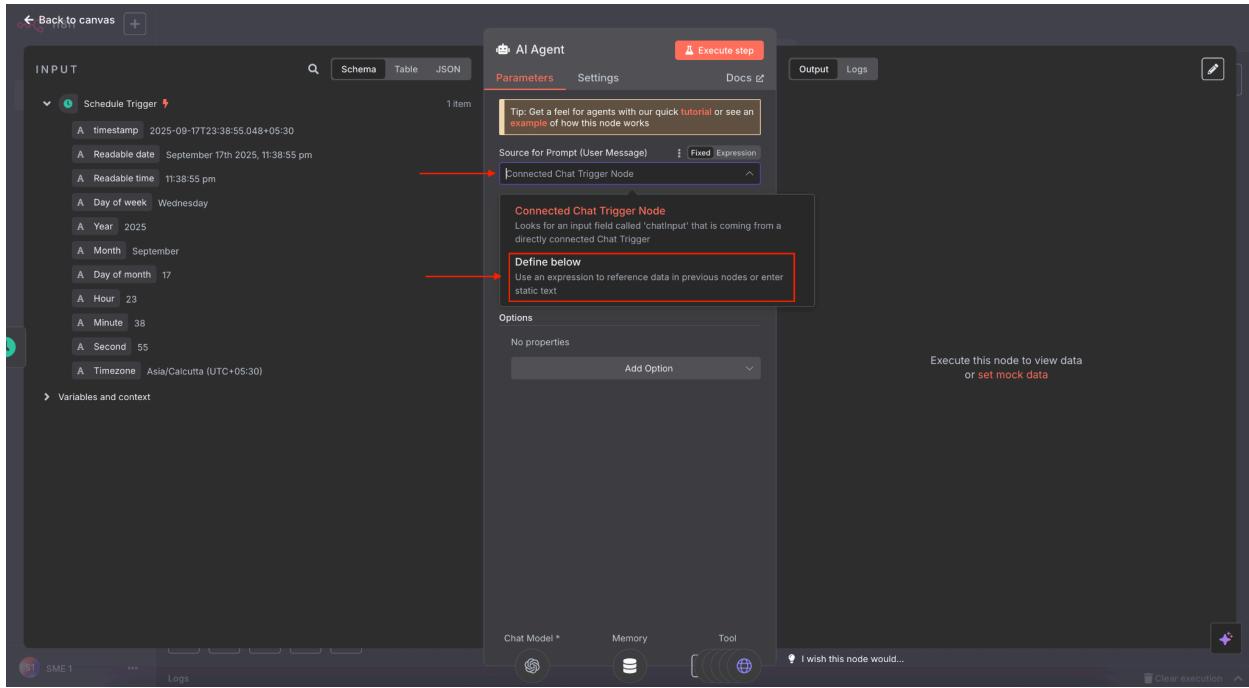
Step 6: Write the Prompt

The prompt tells your agent what to do. It's the most crucial part of the setup.

1. In the main **AI Agent** node, find the **Prompt** section.



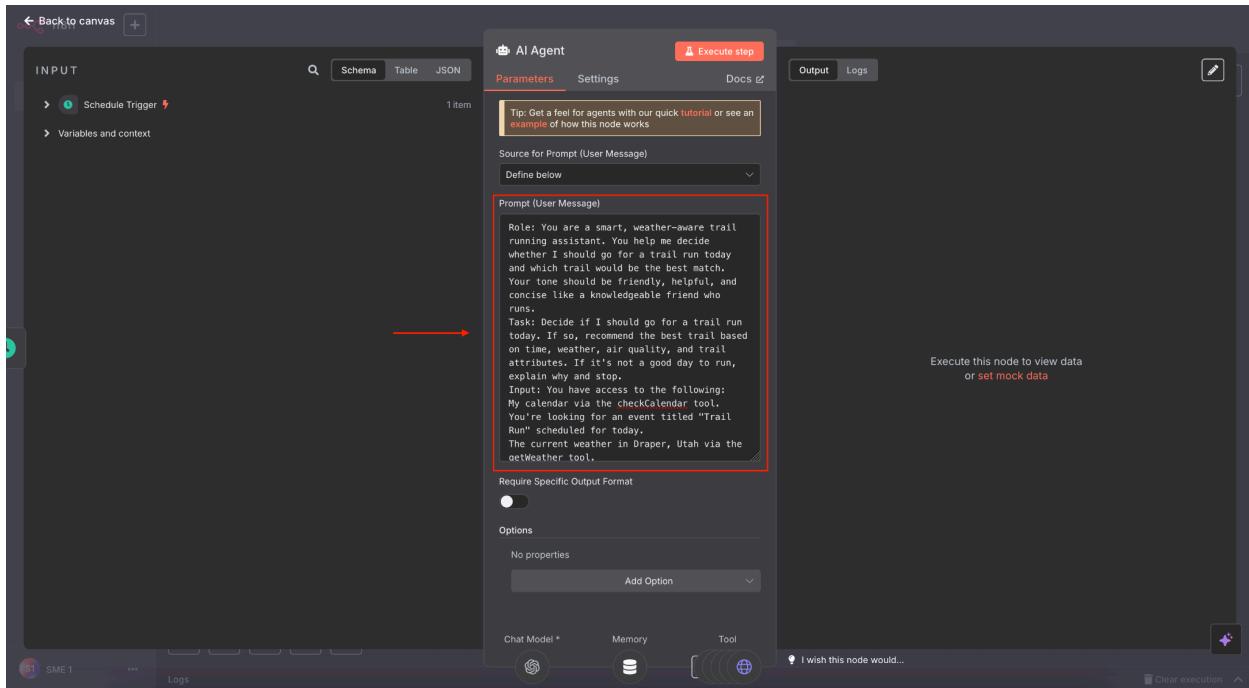
2. Change the source from "Connected chat trigger node" to "**Define below**".



3. Write a clear, structured prompt that includes:

- **Role:** Who the agent is (e.g., "You are a helpful personal assistant that recommends a trail run for the day").
- **Task:** What it needs to accomplish (e.g., "Check my calendar, the weather, air quality, and my list of trails to find the best option").
- **Tools:** The actions it can take (e.g., "Use the Get Weather tool to check the forecast, the Google Sheets tool to see trail options...").
- **Constraints:** Any rules it must follow (e.g., "Only recommend a trail if there is a 'trail run' event on my calendar today").
- **Output:** What the final result should look like (e.g., "Send an email to me with a single trail recommendation, including the weather and air quality").

4. You can use an LLM like ChatGPT to help you generate a well-structured prompt.



Prompt (User Message):

Role: You are a smart, weather-aware trail running assistant. You help me decide whether I should go for a trail run today and which trail would be the best match. Your tone should be friendly, helpful, and concise like a knowledgeable friend who runs.

Task: Decide if I should go for a trail run today. If so, recommend the best trail based on time, weather, air quality, and trail attributes. If it's not a good day to run, explain why and stop.

Input: You have access to the following:

- My calendar via the checkCalendar tool. You're looking for an event titled "Trail Run" scheduled for today.
- The current weather in Draper, Utah via the getWeather tool.
- Current air quality (PM2.5-based AQI) via the getAirQuality tool.
- A list of trail runs from the getHikeList tool. Each trail includes:
 - Name
 - Distance (miles)

- Elevation gain (feet)
- Estimated time (minutes)
- Shade Level: "Shady", "Some Shade", or "Exposed"

Tools: Use only these tools to perform your reasoning and take action:

- `checkCalendar`: Returns today's events.
- `getWeather`: Returns temperature and conditions in Draper, Utah.
- `getAirQuality`: Returns an AQI category such as "Good", "Moderate", "Unhealthy", etc., based on PM2.5.
- `getHikeList`: Returns a list of trails with the fields described above.
- `sendMessage`: Sends me your final recommendation or status message.

Constraints:

- First, use `checkCalendar`. If there is no event titled "Trail Run" today, do nothing.
- Next, use `getAirQuality`. If the category is "Unhealthy", "Very Unhealthy", or "Hazardous", do not recommend a trail. Use `sendMessage` to let me know air quality is too poor for a run.
- If air quality is "Good" or "Moderate", continue.
- Use `getWeather`. If it's very hot or sunny, avoid trails marked "Exposed" and prefer "Shady" or "Some Shade".
- Use `getHikeList` and choose one trail that:
 - Fits within the estimated time range
 - Matches today's weather and shade needs
 - Is not repeated too frequently
- Don't recommend anything if no trail fits the constraints.

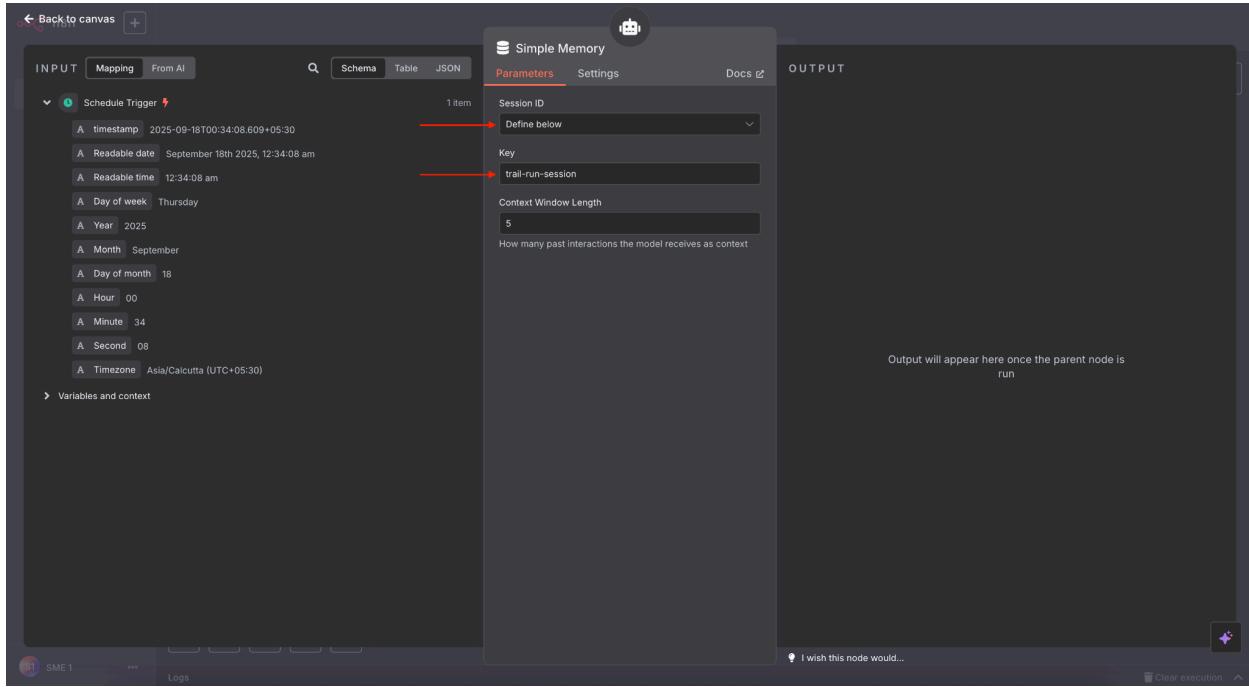
Output: Use the `sendMessage` tool to send a friendly summary like:

- The name of the recommended trail
- Distance, elevation, and estimated time
- A short reason why it's a great fit today based on weather and air quality

If no trail is appropriate, explain why clearly and politely. Keep it brief and helpful.

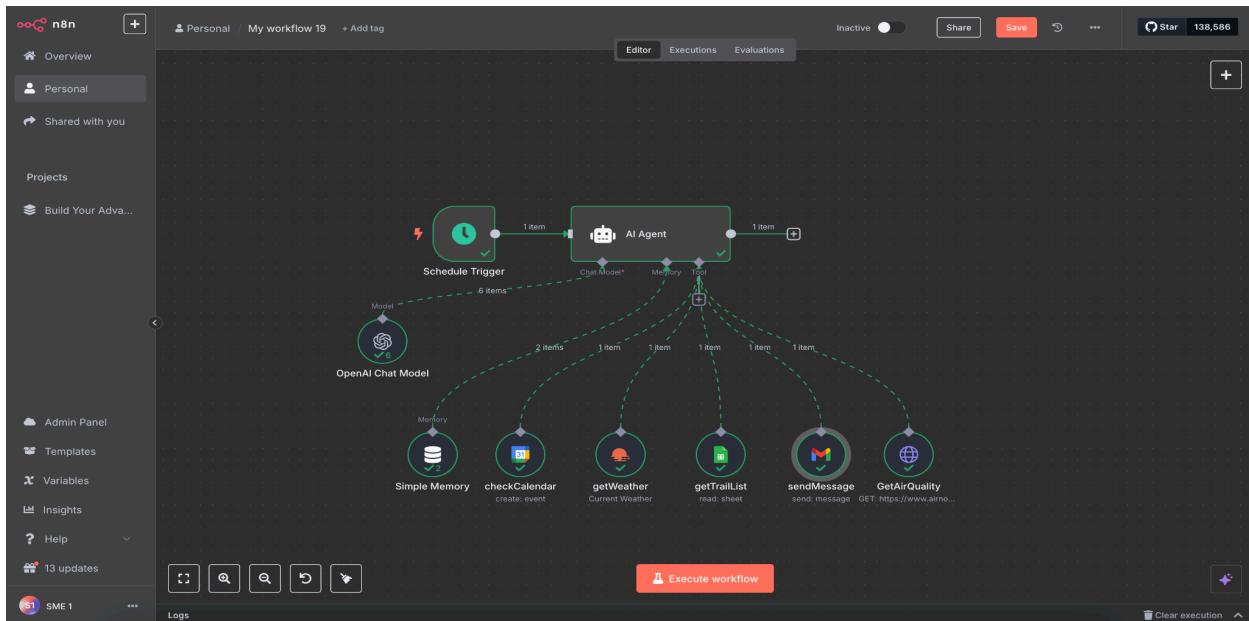
Step 7: Set a static session ID in the Simple Memory node

1. Click on the Simple Memory node
2. Look for the Session ID field
3. Enter a Fixed string like : trail-run-session



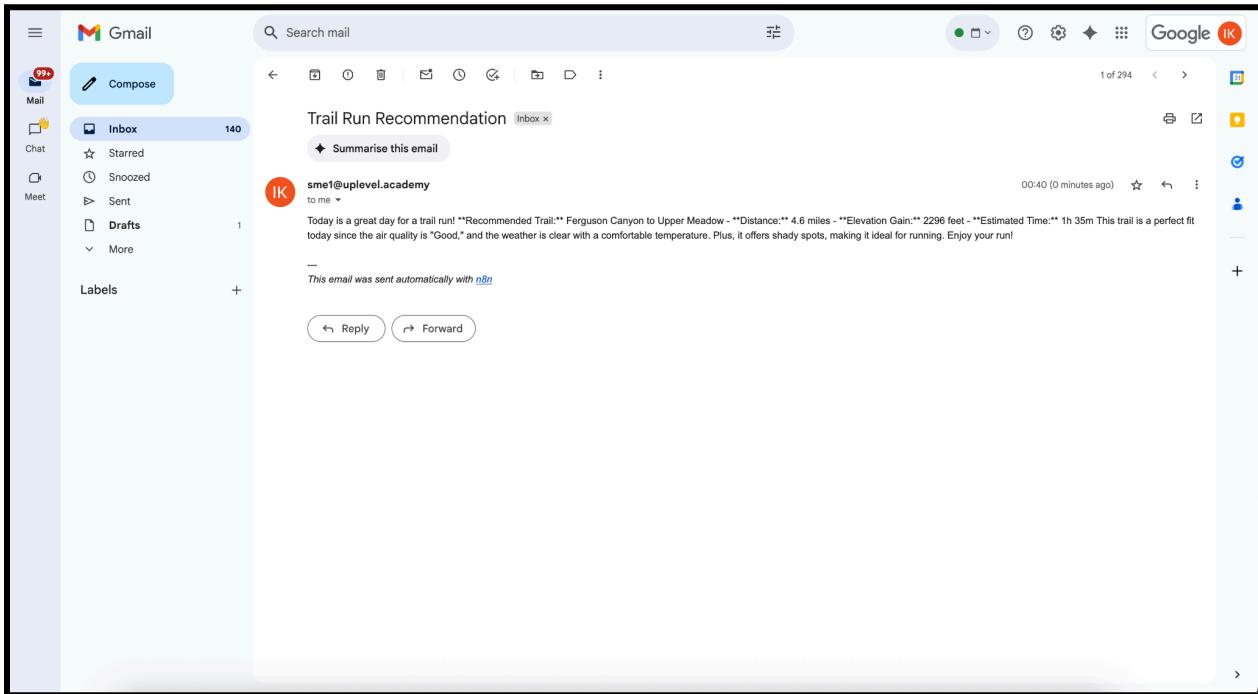
Step 8: Test and Activate Your Agent

1. Click the "Execute workflow" button at the bottom of the screen.



2. Check the output of each node to see if it ran successfully. If you encounter an error, read the error message for clues or use ChatGPT to help you troubleshoot.
3. Once the test runs successfully, save your work and **activate** the workflow so it will run on the schedule you set.

OUTPUT :

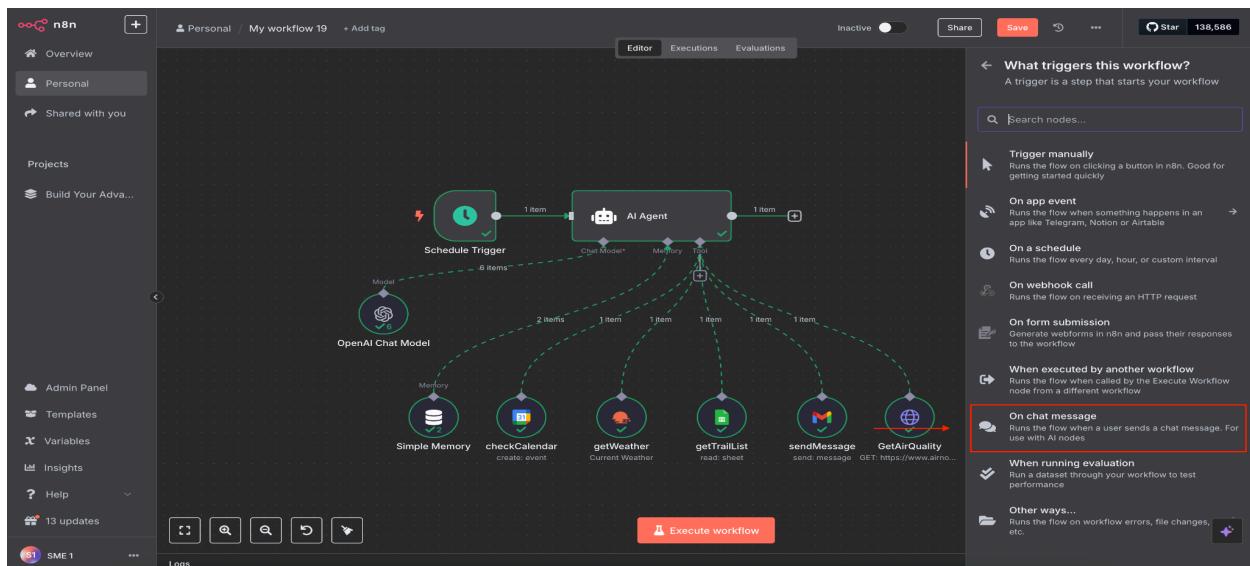
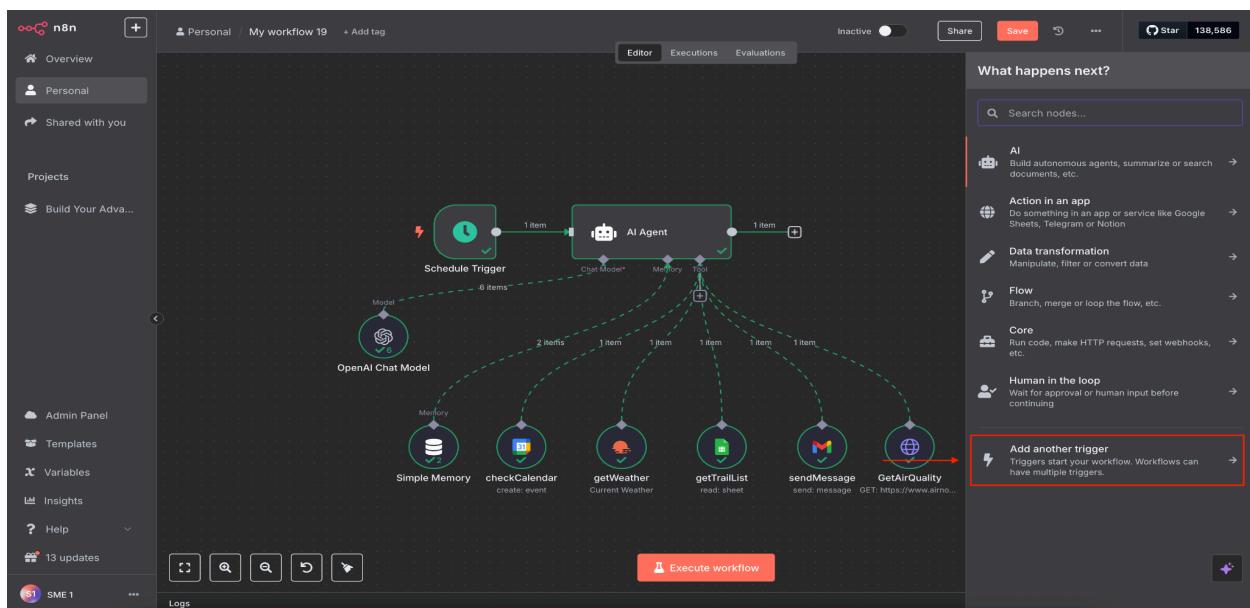


Bonus: Interacting with Your Agent via Chat

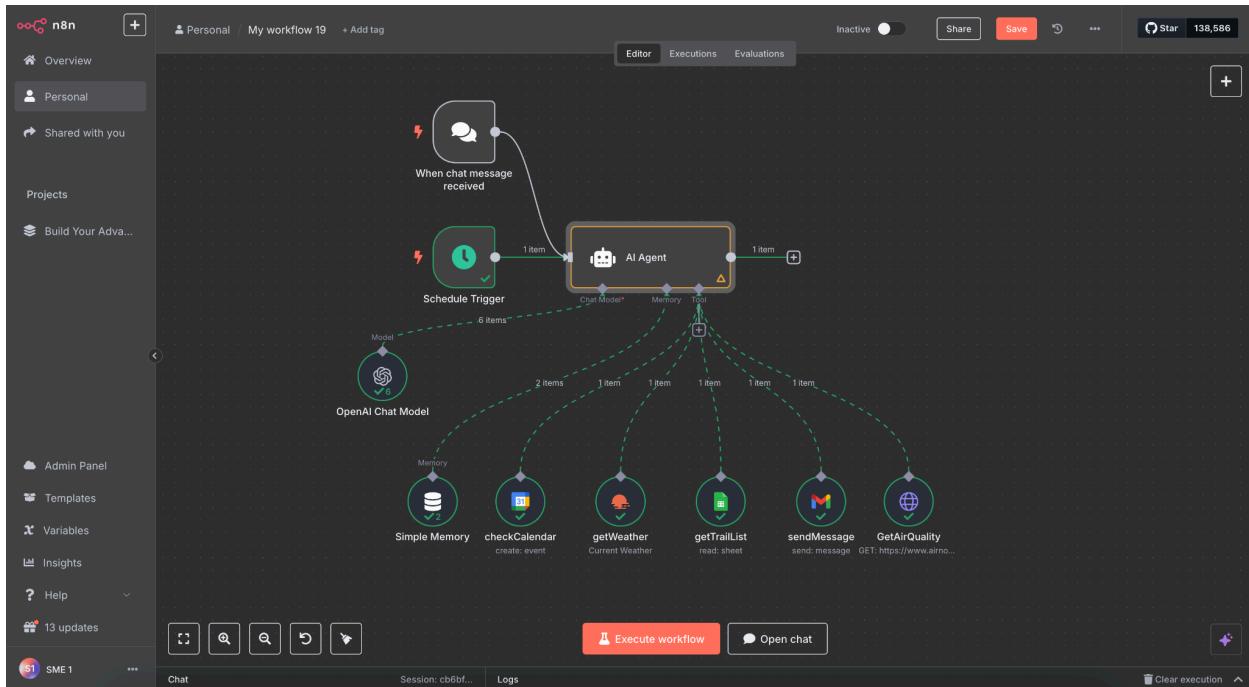
Beyond running on a schedule, you can also interact with your agent directly through a chat interface. This is an excellent way to test its capabilities, fine-tune its responses, and use it on-demand.

Here's how to set it up:

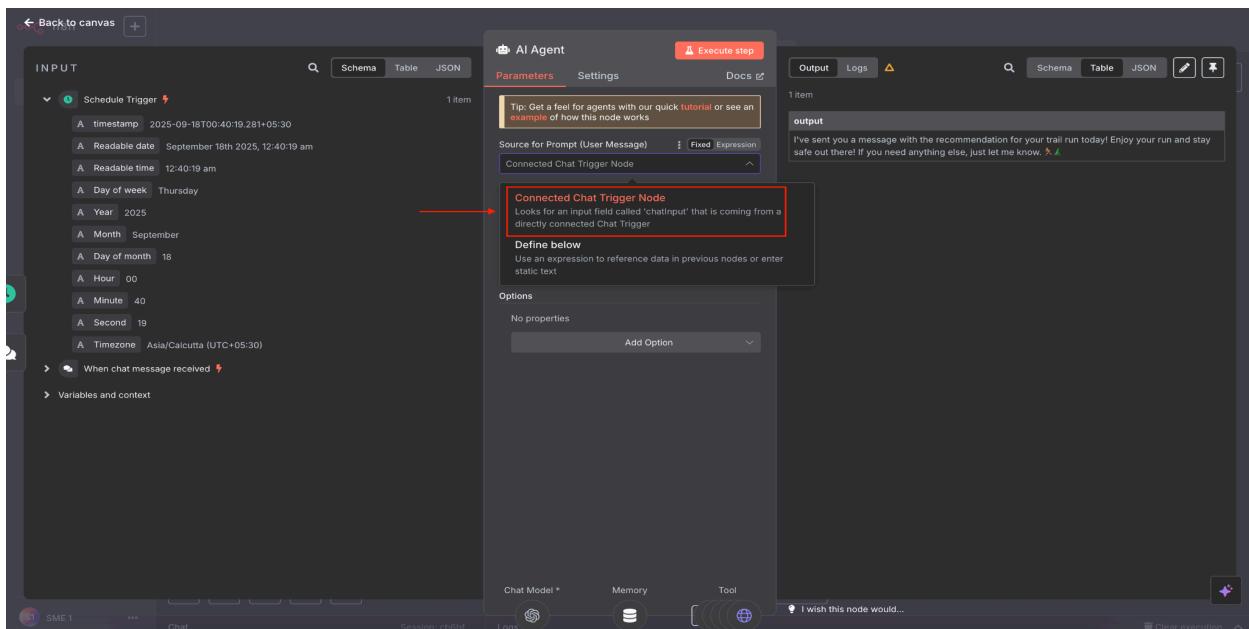
1. **Add a Chat Trigger:** On your n8n canvas, add a new trigger node. From the list, select "On chat messages".



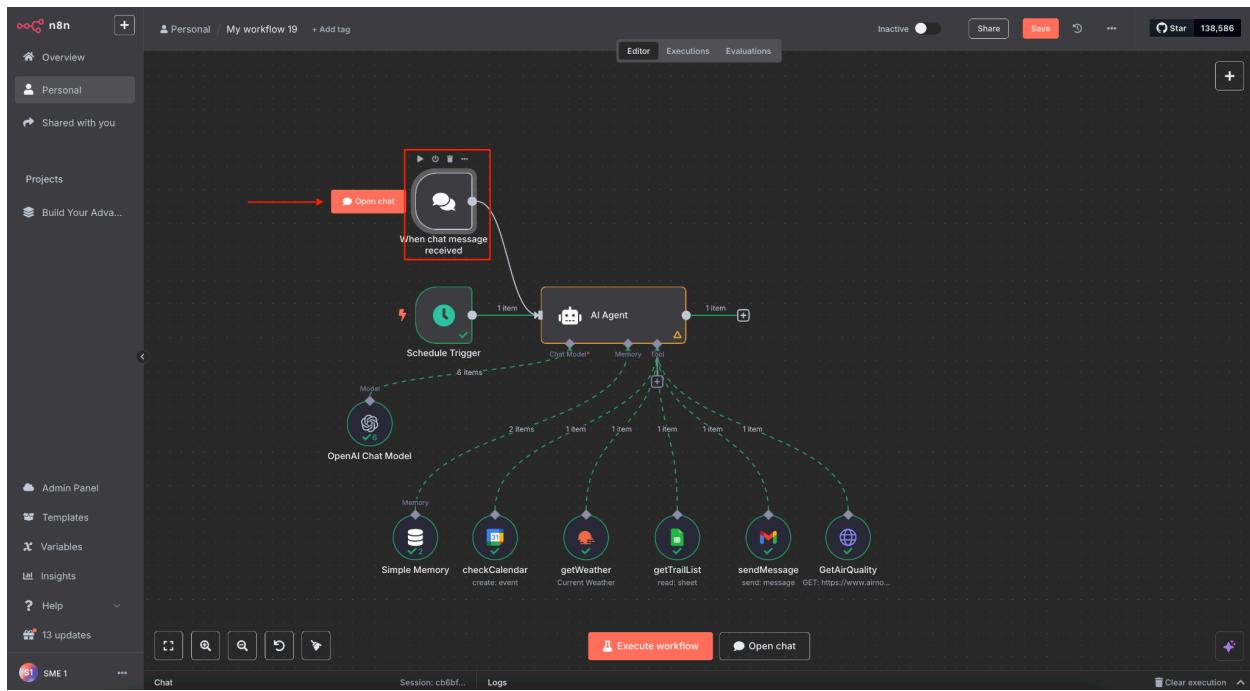
- 2. Connect the Trigger:** Drag the connection line from the new chat trigger to the input of your AI Agent node.



- 3. Switch the Prompt Source:** Open your AI Agent node. In the **Prompt** section, change the **Source** from "Define below" to "**Connected chat trigger node**". This tells the agent to listen for your live chat messages instead of using the pre-written prompt.



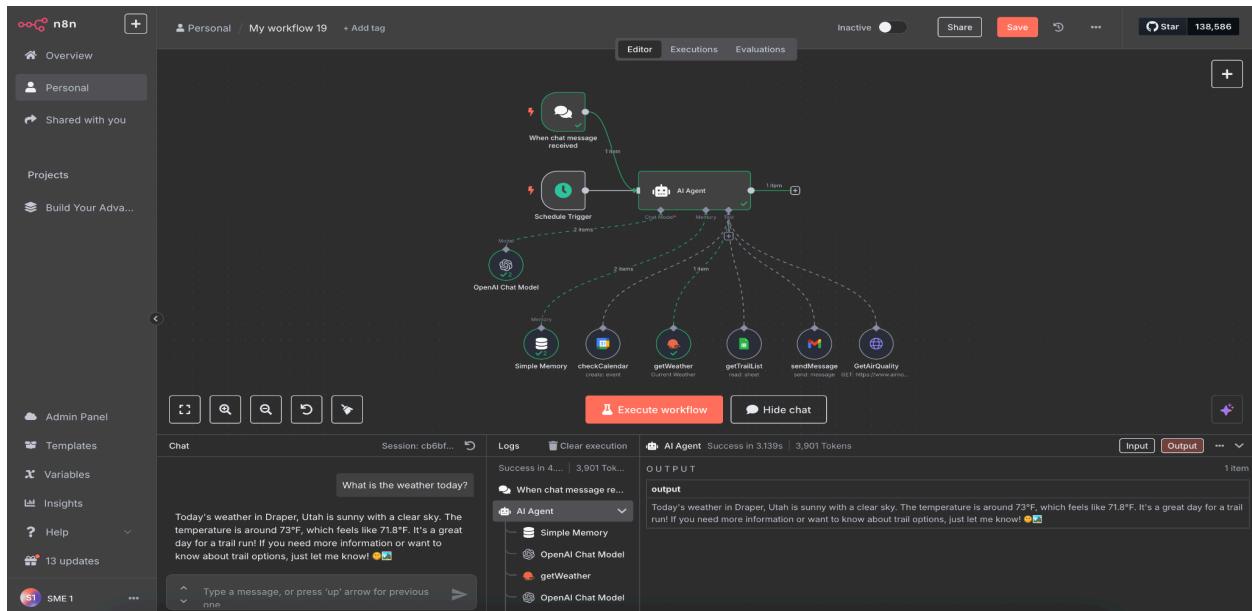
4. Start Chatting: A new "Open chat" button will appear next to the chat node. Click it to open a chat box and start talking directly to your agent.



Example Conversation

This feature demonstrates the agent's ability to use its tools and memory to make decisions in real-time.

- You: What is the weather today?



- You: I have 3 hours. What trail should I run?

