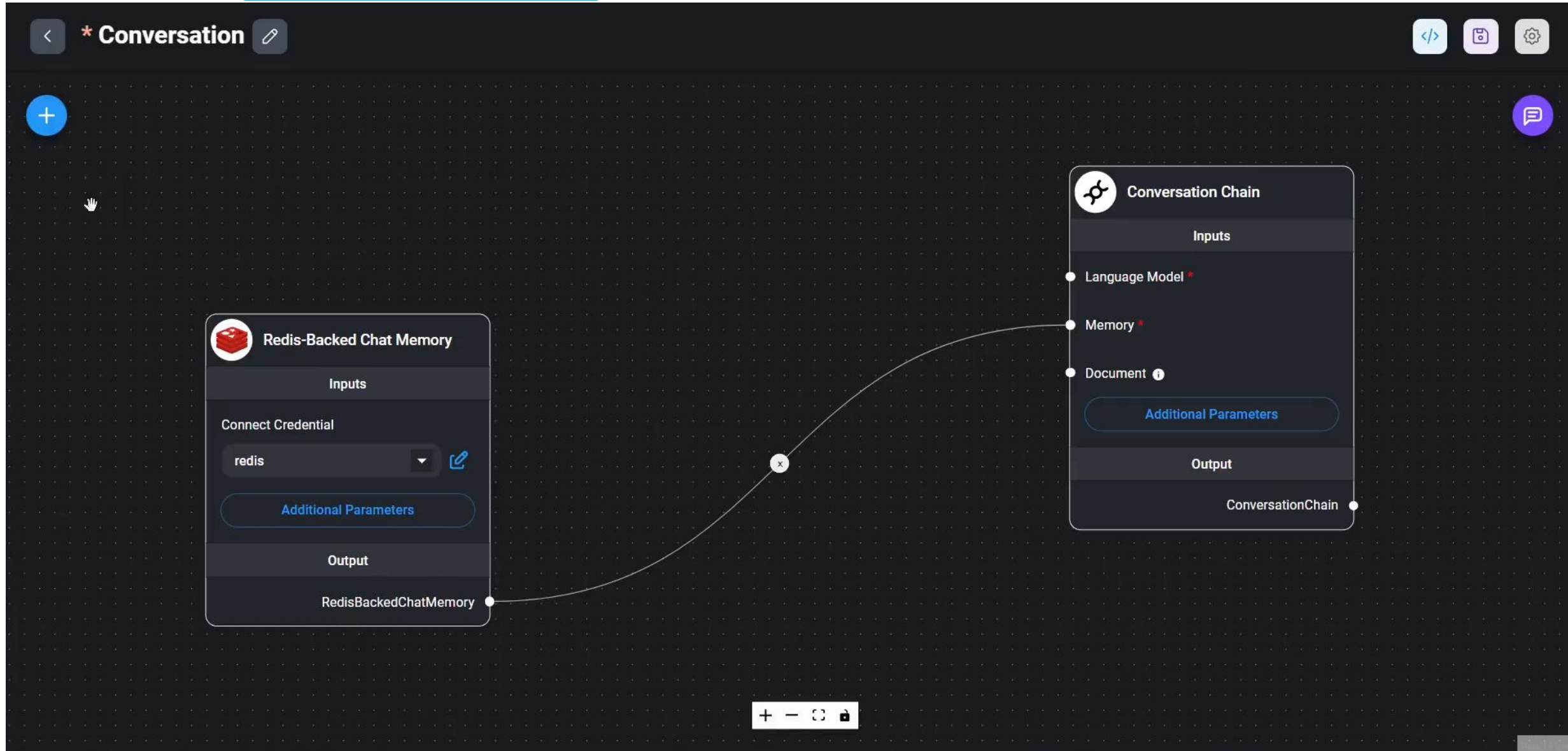


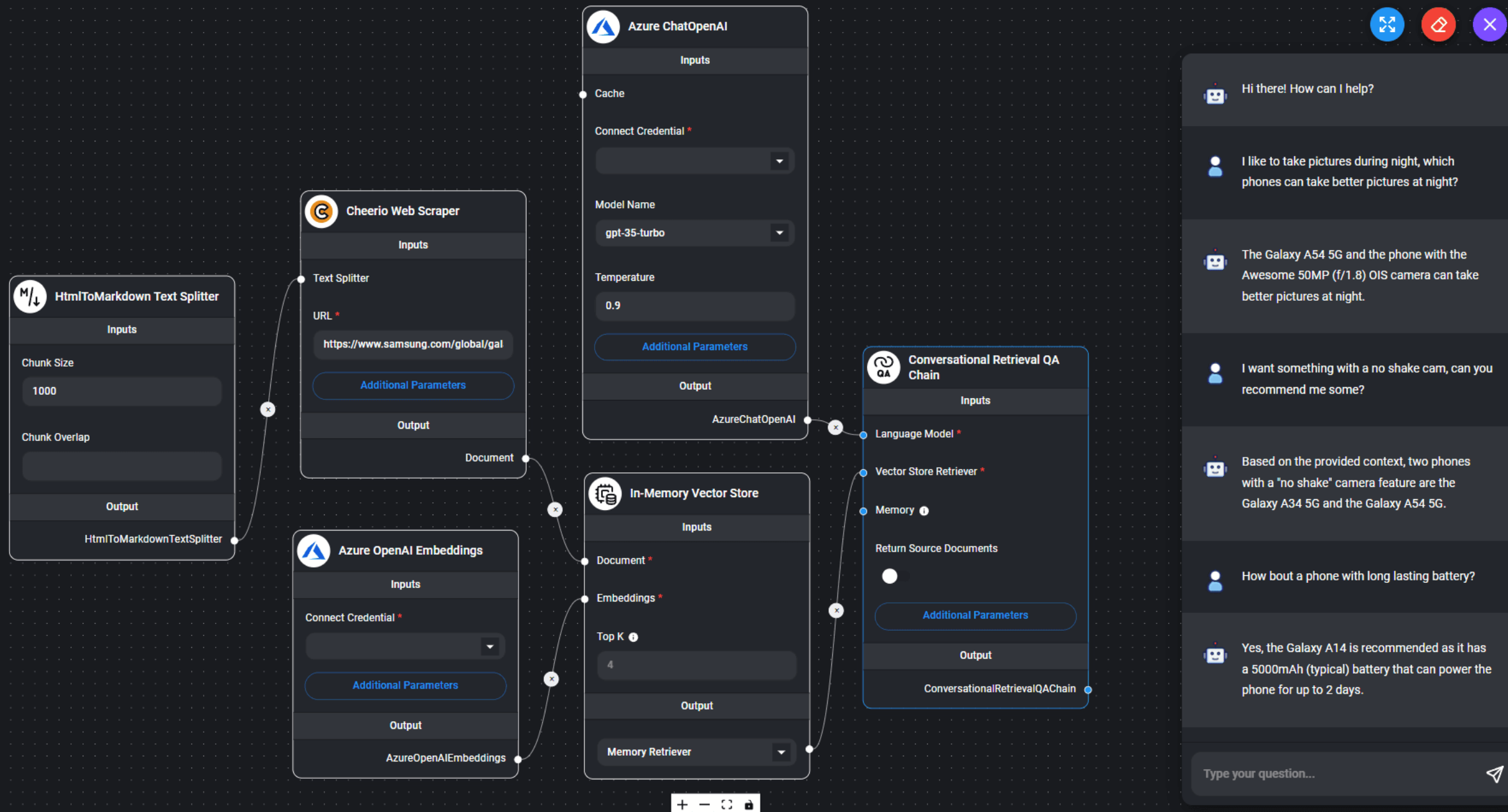
Build LLM Apps Easily

How to
Build an App
Without Coding






유스케이스- 제품과 관련된 모든 질문에 답변하는 제품 카탈로그 챗봇



유스케이스- 주어진 제품 사양에서 제품 설명 생성

 ChatOpenAI

Inputs

Cache

Connect Credential *

Model Name

gpt-3.5-turbo


Temperature

0.9

Additional Parameters

Output

ChatOpenAI

 Prompt Template

Inputs

Langchain Hub

Template *


You are a product description writer. Using the following specification, write a product description in 50 words.

Format Prompt Values

Format Prompt Values

Output

PromptTemplate

 LLM Chain

Inputs

Language Model *

Prompt *

Output Parser

Input Moderation ⓘ

Chain Name

Name Your Chain

Output

LLM Chain

Hi there! How can I help?

6.9" edge Quad HD+ Dynamic AMOLED 2X (3088x1440)
496 ppi
Infinity-O Display

Introducing our cutting-edge 6.9" edge Quad HD+ Dynamic AMOLED 2X display with an impressive 3088x1440 resolution. With a pixel density of 496 ppi, this Infinity-O display delivers stunning visuals and an immersive viewing experience. Experience the future of display technology with our top-of-the-line product.

- 12.0 MP + 12.0 MP Rear Camera Resolution (Multiple)
- F1.8 , F2.2 Rear Camera F Number (Multiple)
- Rear Camera Auto Focus
- Rear Camera OIS
- Digital Zoom up to 10x
- 10.0 MP Front Camera Resolution
- UHD 4K (3840 x 2160) @60fps Video Recording Resolution

Type your question...

유스케이스- 자연어를 사용하여 SQL 데이터베이스 쿼리하기



Hi there! How can I help?

how many rows are there

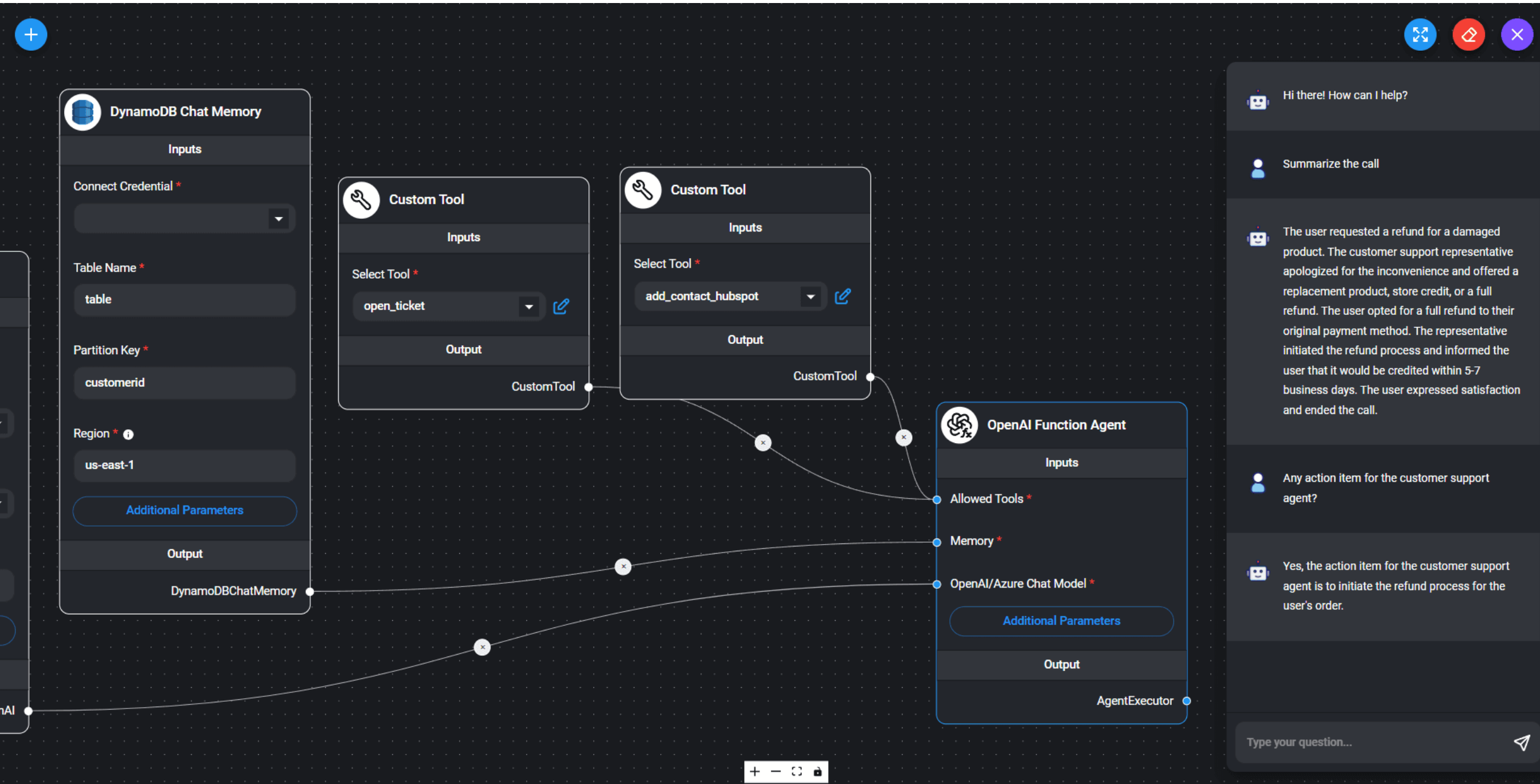
There are a total of 5 rows.

what is the address of steven

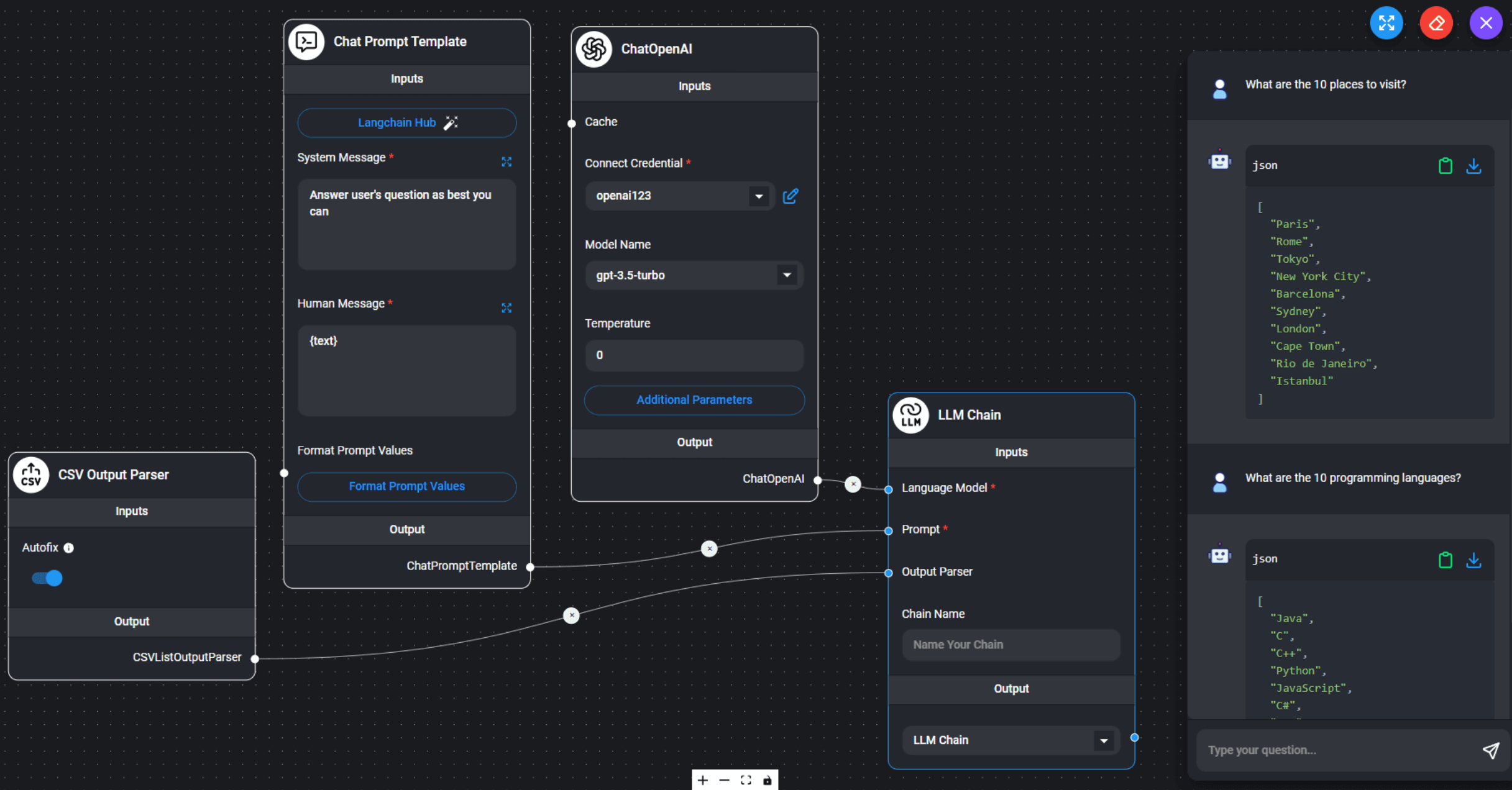
Steven's address is 2, 120 Jefferson St.

Type your question...

유스케이스- 고객 지원 티켓에서 후속 작업 요약 및 만들기

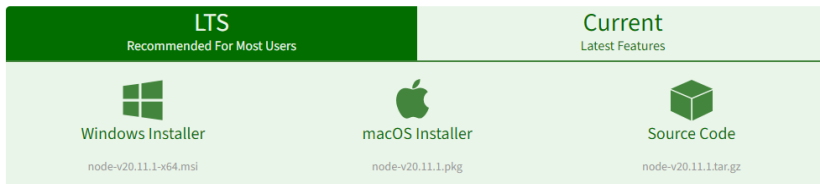


유스케이스- 다운스트림 작업에 사용할 구조화된 데이터 생성 또는 추출



Flowise 설치

NodeJS 설치 : <https://nodejs.org/en/download>



Flowise 설치

`npm install -g flowise`

Flowise 시작

`npx flowise start`

Flowise 사용

<http://localhost:3000> 접속

Developers (선택사항)

1. Yarn 설치

`npm i -g yarn`

2. Repository 복제

`git clone https://github.com/FlowiseAI/Flowise.git`

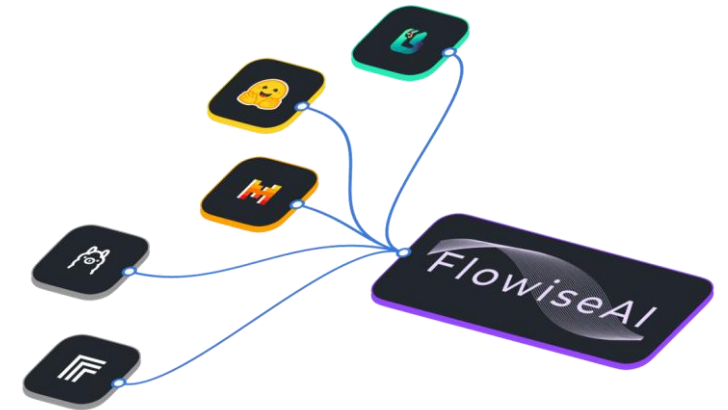
3. 모듈 설치

`cd Flowise`
`yarn install`
`yarn build`

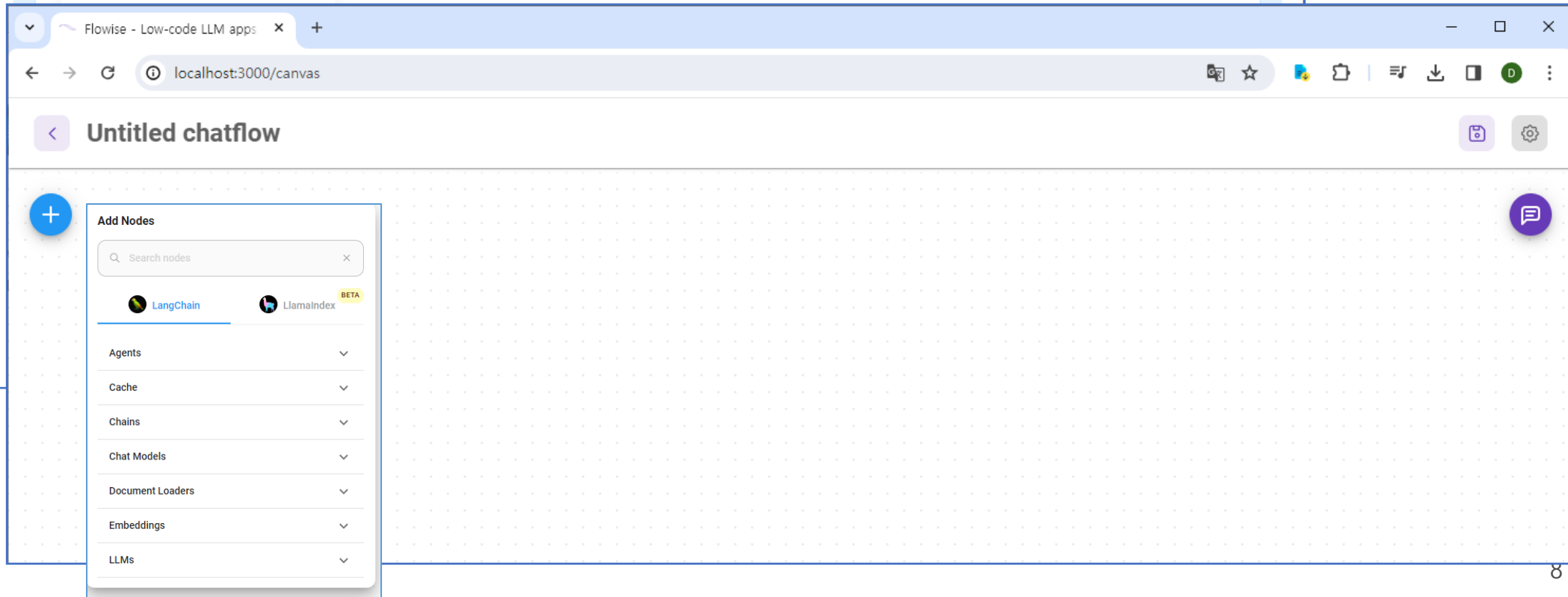
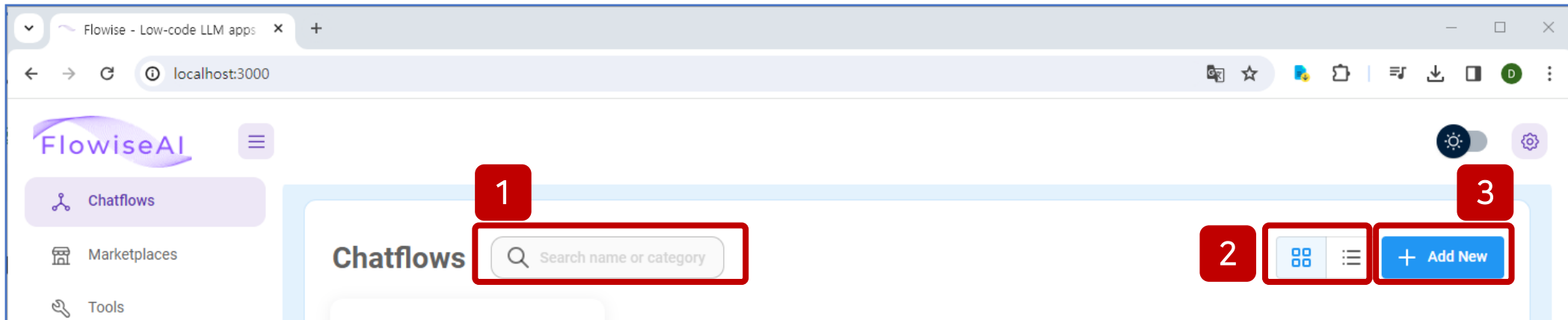
4. App 실행

`yarn start`

6. <http://localhost:3000> 접속



Chatflows



Marketplace

The image displays two screenshots of the FlowiseAI interface, showing the Marketplace and Chatflows sections.

Top Screenshot (Marketplace):

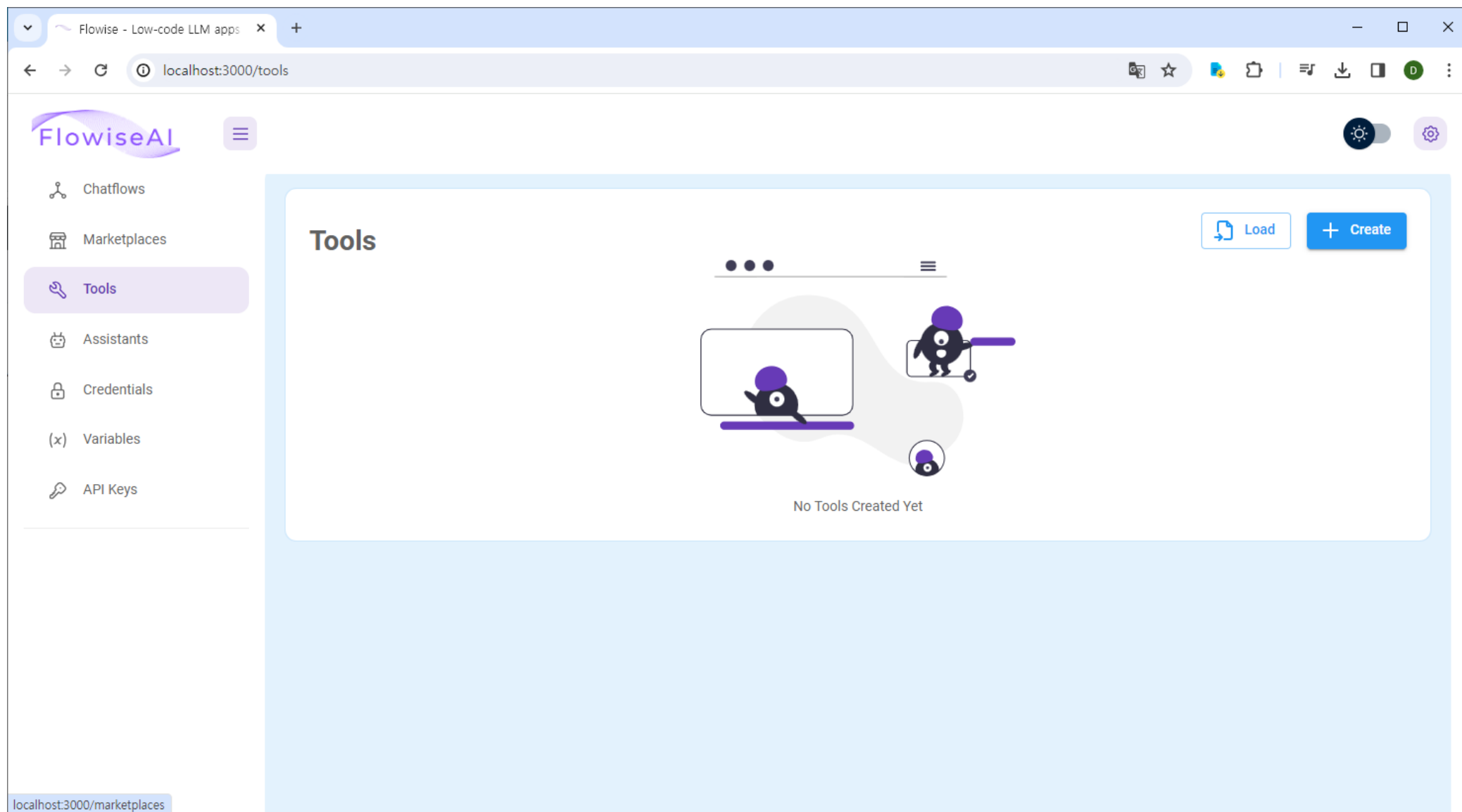
- URL: localhost:3000/marketplaces
- Search bar: Search name or description or node name
- Buttons: Show Filters, Use Template
- Templates displayed:
 - Add Hubspot Contact**: Add new contact to Hubspot
 - Advanced Structured Output Parser** (NEW): Return response as a JSON structure as specified by a Zod
 - Antonym**: Output antonym of given user input using few-shot prompt template built with examples
 - API Agent**: Given API docs, agent automatically decides call, generating url, request from conversation

Bottom Screenshot (Chatflows):

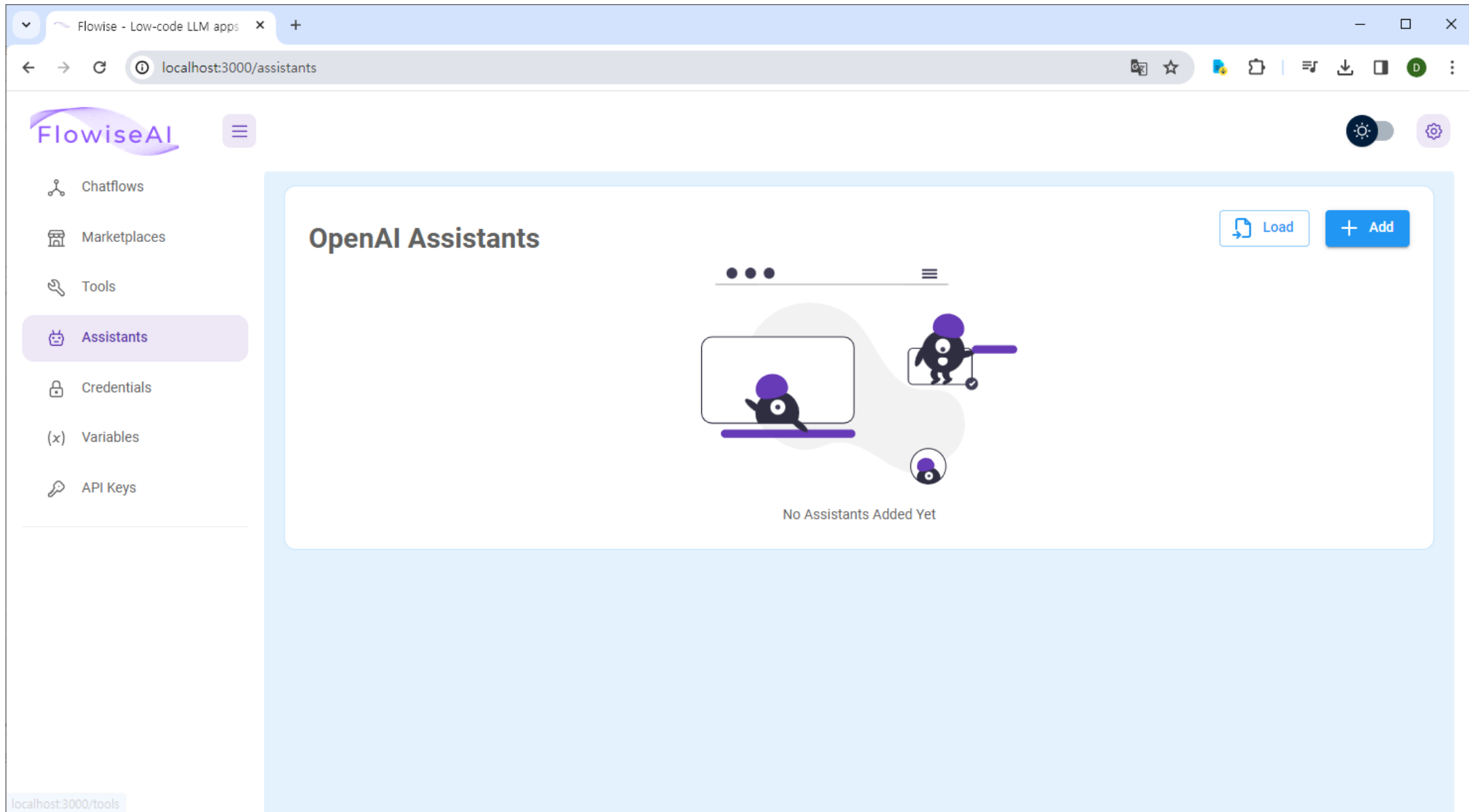
- URL: localhost:3000
- Search bar: Search name or category
- Buttons: Add New
- Table of Chatflows:

Name	Category	Nodes	Last Modified Date	Actions
Demo		M/L, GPT, QA, LLM, + 1 More	March 10th, 2024	Options

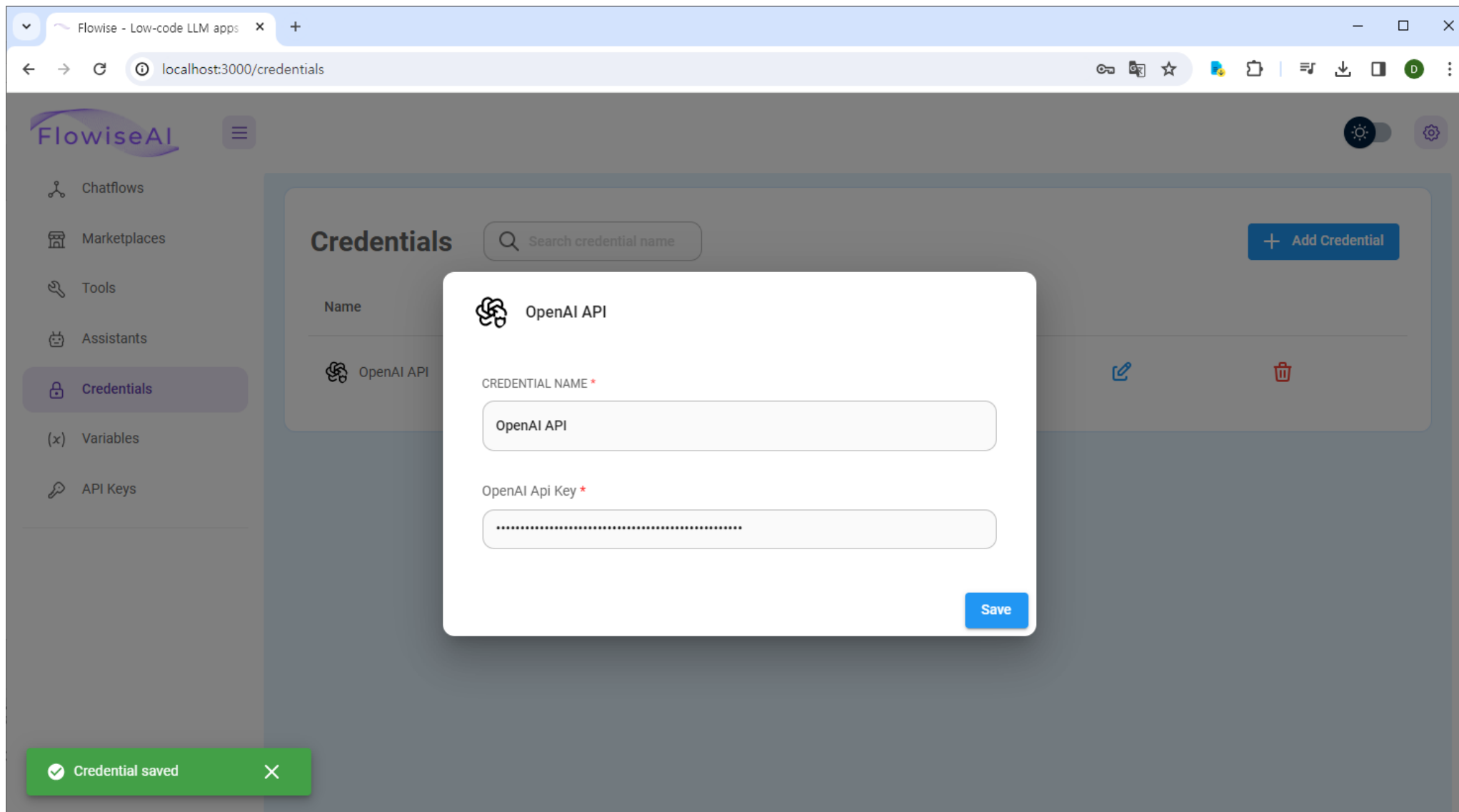
Tools



Assistants



Credentials



Food Recipe App

<https://platform.openai.com/api-keys>

The screenshot displays the Flowise AI interface for building a low-code LLM application. The workflow is titled "FoodRecipe" and is running on a local host. The workflow consists of three main steps:

- Step 1: LLM Chain** - A chain to series against the LLM.
- Step 2: OpenAI** - A wrapper around OpenAI large language models.
- Step 3: Prompt Template** - A schema to represent a basic prompt for an LLM.

The Prompt Template node contains the following text:

당신은 숙련된 요리사입니다.
주 재료 {ingredient} 를 사용하여 맛있는 레시피를 만들어 주세요.

The OpenAI node is configured with the following settings:

- Cache: (empty)
- Connect Credential: OpenAI
- Model Name: gpt-3.5-turbo-instruct
- Temperature: 0.7

The LLM Chain node is configured with the following settings:

- Language Model: (empty)
- Prompt: (empty)
- Output Parser: (empty)
- Input Moderation: (empty)
- Chain Name: chef

A "Select Variable" dialog is open, showing the following variables:

- question: User's question from chatbox
- chat_history: Past conversation history between user and AI

The "Format Prompt Values" dialog is also open, showing the selected variable "question".

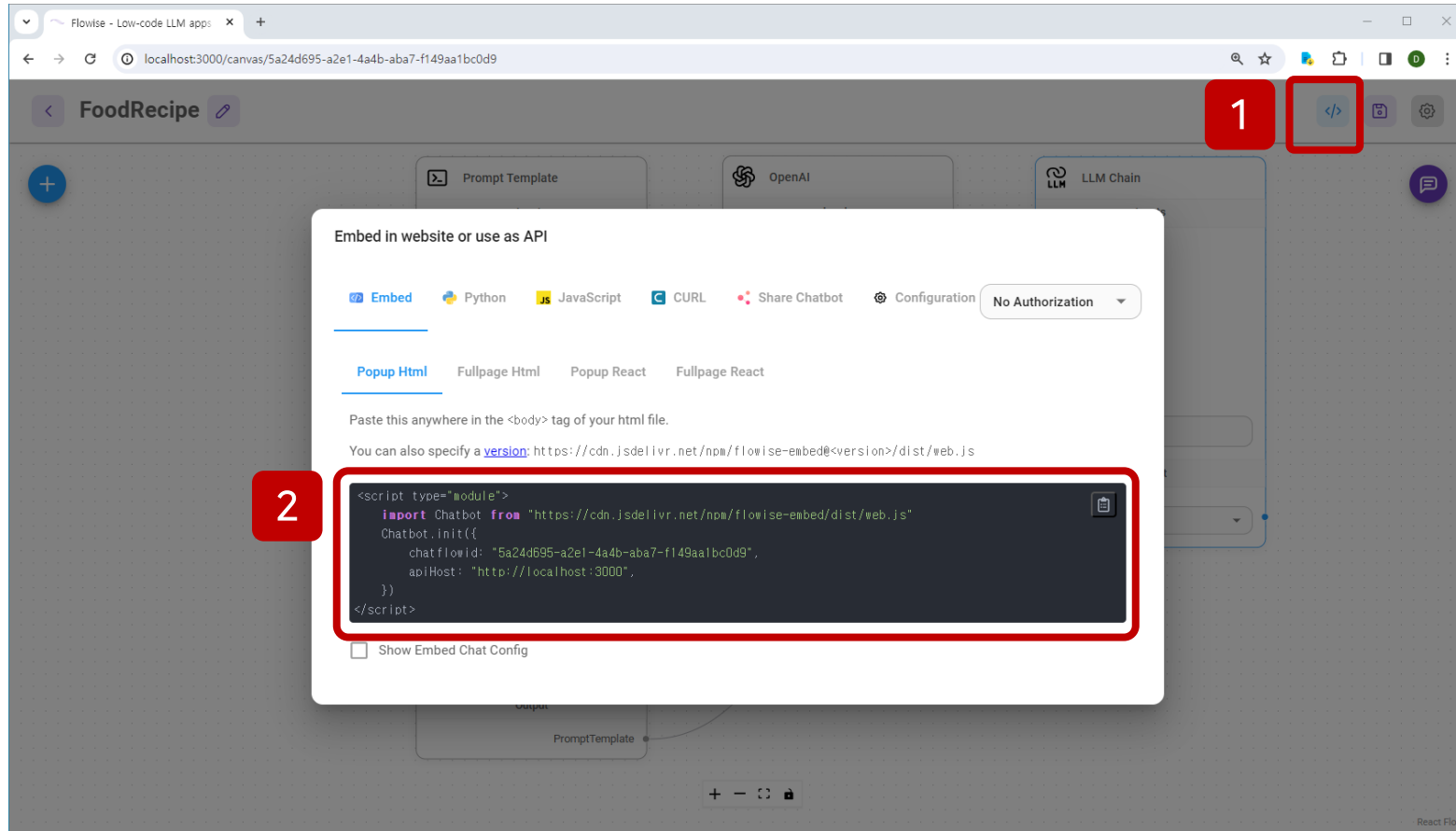
The final output of the workflow is a list of recipes for potatoes:

감자는 다양한 방식으로 요리할 수 있기 때문에 여러 가지 레시피를 소개해 드리겠습니다.

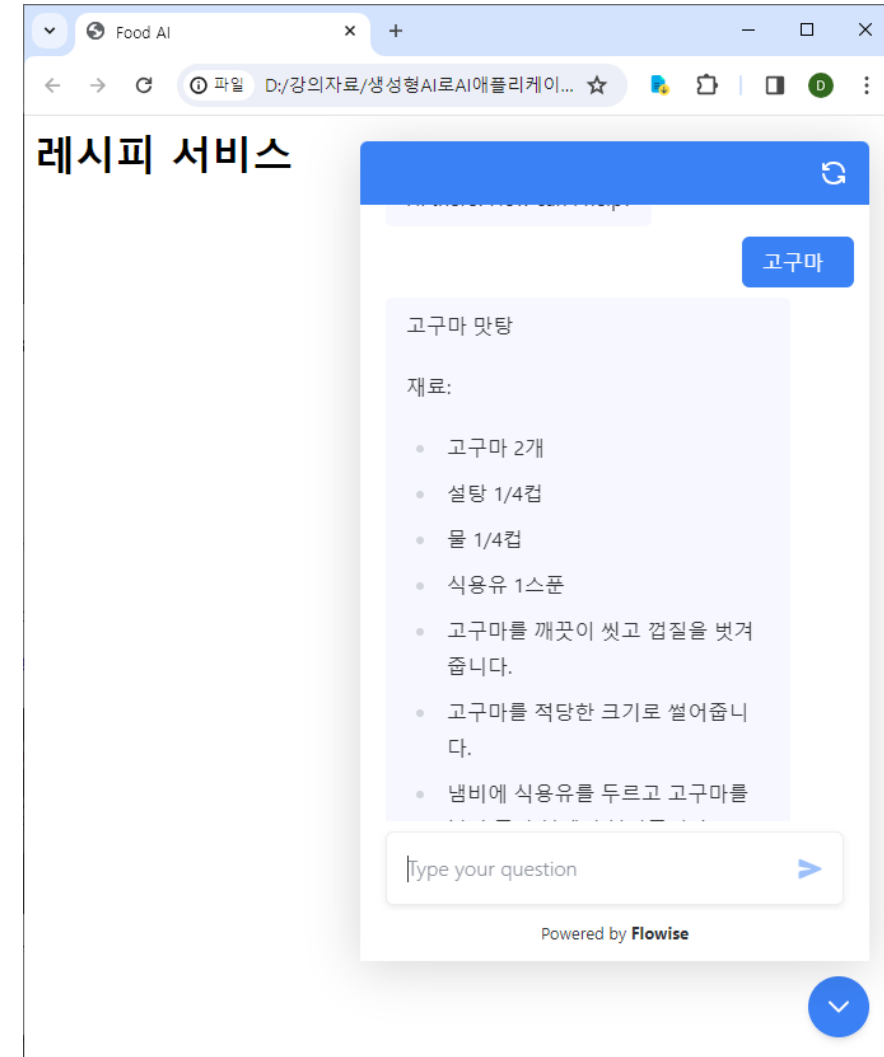
1. 감자조림

- 감자를 깨끗이 씻어 껍질을 벗긴 후 2cm 두께로 썬다.
- 양파 1/2개와 당근 1개는 비슷한 크기로 썰어 준다.
- 냄비에 참기름을 두르고 양파와 당근을 넣어 볶은 후 감자를 넣고 함께 볶는다.
- 물 1컵, 간장 2큰술, 설탕 1큰술을 넣어 감자가 잠길 정도로 물을 부어 뚜껑을 닫고 약한 불에서 10분 정도 끓인다

Food Recipe App



index.html



Food Recipe Critic App

The screenshot displays the Flowise Low-code LLM app interface for a project named "FoodRecipeCriticApp". The workflow is composed of several interconnected components:

- Prompt Template (Left):** Contains a template in Korean: "당신은 숙련된 요리사이십니다. 주 재료 (ingredient) 를 사용하여 맛있는 레시피를 만들어 주세요."
- OpenAI:** Configured with "Cache", "Connect Credential" (OpenAI), "Model Name" (gpt-3.5-turbo-instruct), and "Temperature".
- LLM Chain (Left):** Configured with "Language Model", "Prompt", "Output Parser", "Input Moderation", and "Chain Name" (chef). The output is labeled "1".
- Prompt Template (Middle):** Contains a template in English: "You are a harsh and rude food critic. Write a review about the following recipe in korean: {recipe}"
- ChatOpenAI:** Configured with "Cache", "Connect Credential" (OpenAI), "Model Name" (gpt-3.5-turbo-1106), "Temperature" (0.9), and "Allow Image Uploads".
- LLM Chain (Right):** Configured with "Language Model" and "Prompt". The output is labeled "2".

Two "Format Prompt Values" popups are visible:

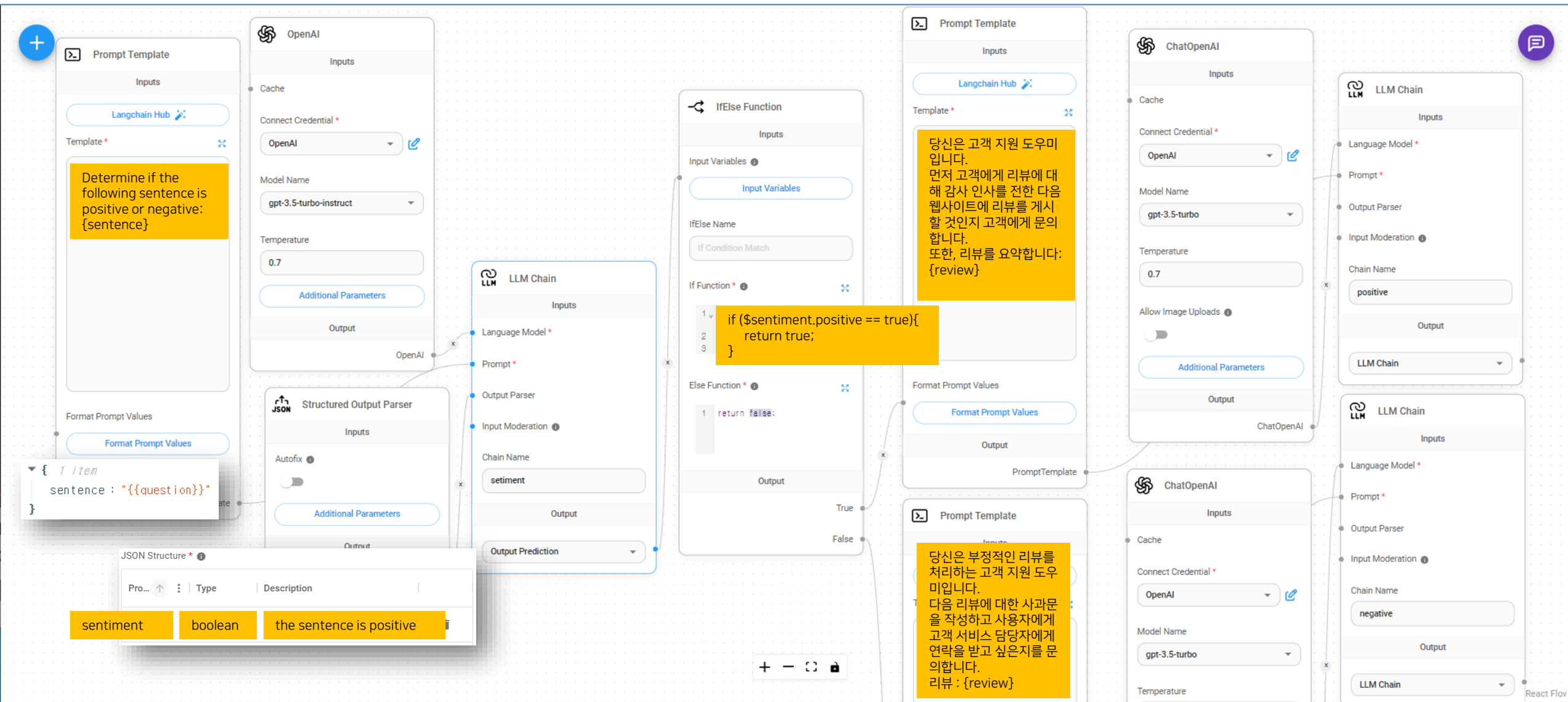
- Format Prompt Values (Left):** Shows a JSON structure:

```
{ 1 item recipe: {{llmChain_0.data.instance}}}
```
- Format Prompt Values (Middle):** Shows a "Select Variable" dialog with options: "question" (User's question from chatbox), "chat_history" (Past conversation history between user and AI), and "chef" (Output Prediction from LLM Chain).


On the right side, a chat interface shows a conversation in Korean. The user asks for a recipe, and the AI provides a detailed recipe for "고구마" (Chestnut). The AI then provides a harsh and rude review of the recipe in Korean.


The chat interface includes a text input field at the bottom: "Type your question..." and a microphone icon.


Review AI App




Review AI App

 제품의 가성비가 좋은 거 같아요.

 고객님, 안녕하세요! 리뷰를 남겨 주셔서 감사드립니다. 고객님의 소중한 의견을 전달해주셔서 감사합니다. 제품의 가성비가 좋다고 말씀해주셔서 저희도 기쁘게 생각합니다. 웹사이트에 리뷰를 게시해도 괜찮을까요? 게시를 원하시면 언제든지 말씀해주세요. 감사합니다!

 배송 제품에 스크래치가 있어요.

 고객님,
저희 제품을 구매해 주셔서 감사합니다. 그러나 제품이 스크래치가 있어서 심려를 끼쳐드려 죄송합니다. 제품이 완벽하지 않아서 정말 죄송합니다.

고객 서비스 담당자가 귀하의 문제를 즉시 해결하도록 도와드릴 수 있습니다. 귀하의 문제를 빠르게 해결해 드릴 수 있도록 도와드릴 수 있도록, 고객 서비스 담당자에게 연락을 받고 싶으신

```
C:\Windows\system32\cmd.exe
2024-03-10 19:58:08 [INFO]: POST /api/v1/internal-prediction/7f981a57-89d2-4e33-99bf-e382023913b5

*****OUTPUT PREDICTION*****
{ json: { positive: true } }
2024-03-10 19:58:09 [INFO]: [server]: Chatflow 7f981a57-89d2-4e33-99bf-e382023913b5 added into ChatflowPool

*****FINAL RESULT*****
고객님, 안녕하세요! 리뷰를 남겨 주셔서 감사드립니다. 고객님의 소중한 의견을 전달해주셔서 감사합니다. 제품의 가성비가 좋다고 말씀해주셔서 저희도 기쁘게 생각합니다. 웹사이트에 리뷰를 게시해도 괜찮을까요? 게시를 원하시면 언제든지 말씀해주세요. 감사합니다!
2024-03-10 19:58:38 [INFO]: POST /api/v1/internal-prediction/7f981a57-89d2-4e33-99bf-e382023913b5

*****OUTPUT PREDICTION*****
{ json: { positive: false } }
2024-03-10 19:58:38 [INFO]: [server]: Chatflow 7f981a57-89d2-4e33-99bf-e382023913b5 added into ChatflowPool

*****FINAL RESULT*****
고객님,
저희 제품을 구매해 주셔서 감사합니다. 그러나 제품이 스크래치가 있어서 심려를 끼쳐드려 죄송합니다. 제품이 완벽하지 않아서 정말 죄송합니다.
고객 서비스 담당자가 귀하의 문제를 즉시 해결하도록 도와드릴 수 있습니다. 귀하의 문제를 빠르게 해결해 드릴 수 있도록 도와드릴 수 있도록, 고객 서비스 담당자에게 연락을 받고 싶으신가요?
다시 한번 사과의 말씀을 전하며, 고객님의 양해를 부탁드립니다.
```

실습 - Conversational Chatbot

<https://upstash.com/>

The screenshot shows the Upstash Redis console interface. At the top, a navigation bar includes a logo, a 'Personal' dropdown, and a menu with 'Redis' (highlighted with a red box and number 1), 'Kafka', 'Vector' (marked 'NEW'), and 'QStash'. Below the navigation bar, the 'Usage For Current Billing' section shows 'COMMANDS 0' and 'AVERAGE STORAGE 0B'. A search bar is present. The main content area features a 'Create a Redis Database' section with the text 'We manage the database for you and you only'. A green 'Create database' button is highlighted with a red box and number 2. A modal window titled 'Create Database' is open, showing the following fields and options: 'Name' (labeled with number 3) containing 'ChatHist', 'Type' (with 'Regional' selected and 'Global' as an option), 'Region' (labeled with number 4) set to 'N. Virginia, USA (us-east-1)', and checkboxes for 'TLS (SSL) Enabled' and 'Eviction'. The 'Create' button at the bottom right of the modal is highlighted with a red box and number 5. An arrow points from the 'Create database' button to the modal. A separate inset window titled 'REST API' is shown, explaining that the REST API enables access to the Upstash database (labeled with number 6). It displays environment variables 'UPSTASH_REDIS_REST_URL' (labeled with number 7) and 'UPSTASH_REDIS_REST_TOKEN', and provides a cURL command for accessing the database.

실습 - Conversational Chatbot

Flowise - Low-code LLM apps

localhost:3000/canvas/73a737cf-72d0-4f6d-9ec7-db2ac1ffd679

* Conversational Chatbot

Simple Prompt Moderation

Inputs

Chat Model

Deny List

나쁜 강아지

Error Message

Cannot Process! Input violates content moderation policies.

Upstash Redis-Backed Chat Memory

Inputs

Connect Credential

Upstash Redis

Upstash Redis REST URL

https://us1-equal-tuna-40994.upstash

Additional Parameters

Output

UpstashRedisBackedChatMemory

ChatOpenAI

Inputs

Cache

Connect Credential

OpenAI

Session Id

chat001

Session Timeouts

Allow Image Uploads

Additional Parameters

Output

ChatOpenAI

Conversation Chain

Inputs

Chat Model

Memory

Chat Prompt Template

Input Moderation

Additional Parameters

Output

ConversationChain

Buffer Memory

Inputs

Memory Key

chat_history

Input Key

Upstash Redis Memory API

Refer to [official guide](#) on how to create redis instance and get redis REST Token

CREDENTIAL NAME *

Upstash Redis

Upstash Redis REST Token *

Add

내 이름은 oo

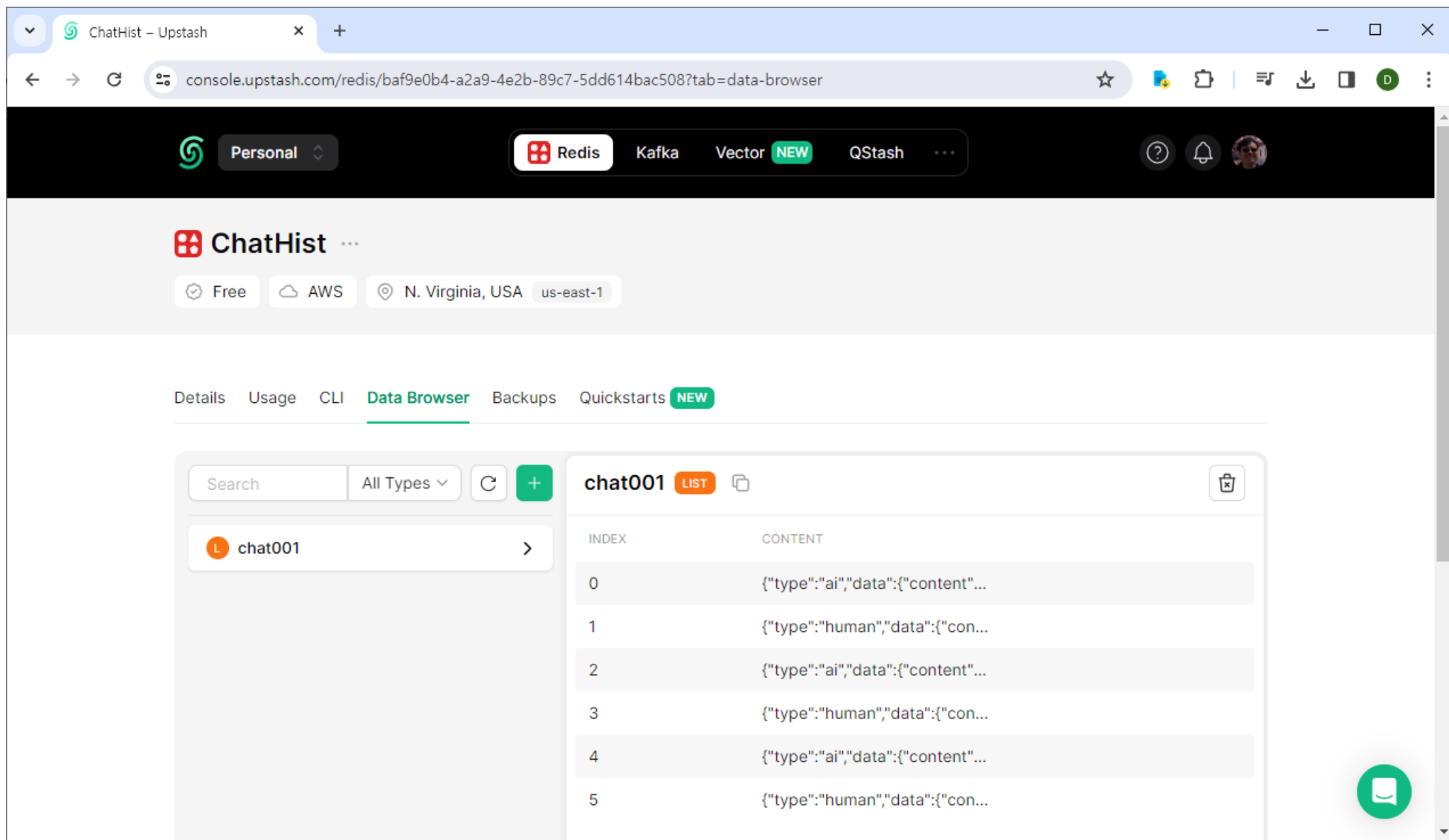
반가워요, 대니님! 좋은 하루 보내세요. 어떤 정보가 필요하신가요?

니 이름은?

나쁜 강아지

인스타그램에 "동네 산책"이라는 주제로 게시를 작성하려고 해. 일상의 기쁨, 소소한 감동, 여유, 힐링 등의 느낌이 물씬 나도록 구어체로 작성해줘. 에피소드도 하나 추가하고, 이모지와 해시태그를 사용해.

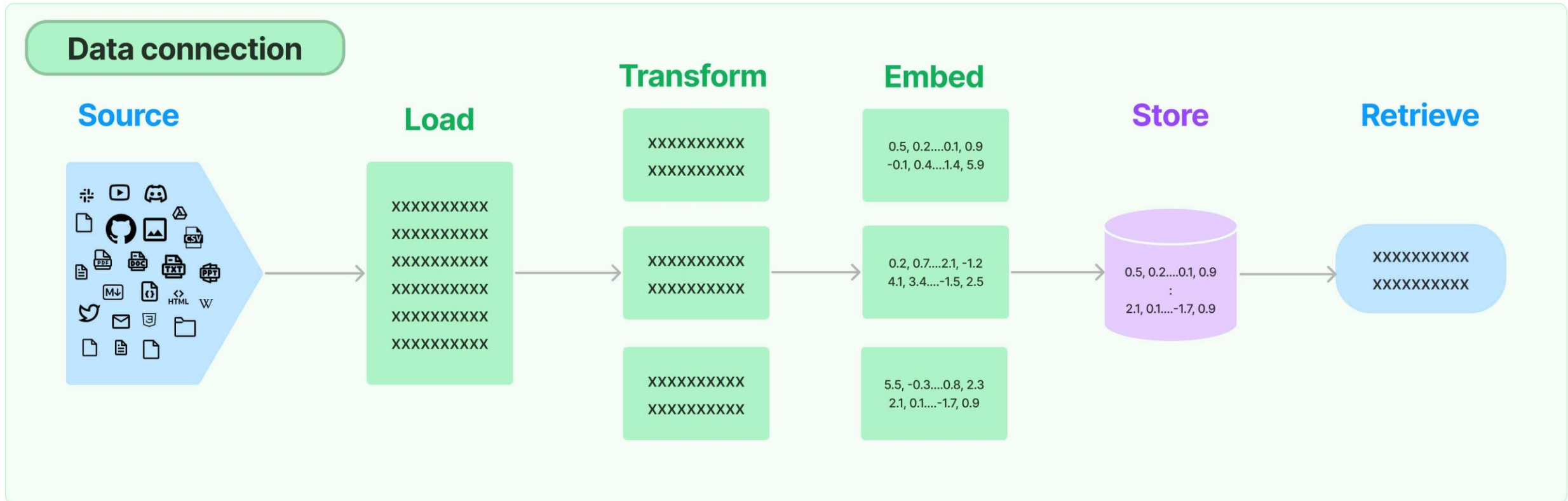
실습 - Conversational Chatbot



The screenshot shows the ChatHist Upstash console interface. The browser address bar displays the URL: `console.upstash.com/redis/baf9e0b4-a2a9-4e2b-89c7-5dd614bac508?tab=data-browser`. The top navigation bar includes the ChatHist logo, a 'Personal' dropdown, and tabs for 'Redis', 'Kafka', 'Vector' (marked 'NEW'), and 'QStash'. Below the navigation bar, the 'ChatHist' header is followed by filters: 'Free', 'AWS', 'N. Virginia, USA', and 'us-east-1'. The main content area features a tabbed interface with 'Details', 'Usage', 'CLI', 'Data Browser' (selected), 'Backups', and 'Quickstarts' (marked 'NEW'). The 'Data Browser' tab displays a search bar, a filter dropdown set to 'All Types', and a list of chat items. The first item, 'chat001', is selected. To the right, a table shows the chat history for 'chat001'.

INDEX	CONTENT
0	<code>{"type":"ai","data":{"content"...</code>
1	<code>{"type":"human","data":{"con...</code>
2	<code>{"type":"ai","data":{"content"...</code>
3	<code>{"type":"human","data":{"con...</code>
4	<code>{"type":"ai","data":{"content"...</code>
5	<code>{"type":"human","data":{"con...</code>

RAG Chatbot



RAG Chatbot

Flowise - Low-code LLM apps

localhost:3000/canvas/f7724231-4aae-4e2a-94b4-db93fef0f5e8

RAGChatbot

Recursive Character Text Splitter

Inputs

Chunk Size: 1000

Chunk Overlap: 50

Additional Parameters

Output: RecursiveCharacterTextSplitter

Pdf File

Inputs

Text Splitter

손익계산서.pdf

Upload File

Usage: One document per page

Additional Parameters

Output: Document

In-Memory Vector Store

Inputs

Document

Embeddings *

Top K: 4

Output: Memory Retriever

OpenAI Embeddings

Inputs

Connect Credential: OpenAI

Model Name: text-embedding-ada-002

Additional Parameters

ChatOpenAI

Inputs

Cache

Connect Credential: OpenAI

Model Name: gpt-3.5-turbo

Temperature: 0.4

Allow Image Uploads

Additional Parameters

Output: ChatOpenAI

Conversational Retrieval QA Chain

Inputs

Chat Model *

Vector Store Retriever *

Memory

Return Source Documents

Additional Parameters

Output: ConversationalRetrievalQAChain

Upset Vector Store

- Recursive Character Text Splitter (recursiveCharacterTextSplitter_0)
- Pdf File (pdfFile_0)
- OpenAI Embeddings (openAIEmbeddings_0)
- In-Memory Vector Store (memoryVectorStore_0)

Show API

Upset

Hi there! How can I help?

2020년 영업수익이 얼마야?

2020년 영업수익은 실액원으로 23,917입니다.

2023년 당기순이익은?

2023년 당기순이익은 실액원으로 989입니다.

2020년 영업수익이 얼마야?

2023년 당기순이익은?

RAG Chatbot

Flowise - Low-code LLM apps

localhost:3000/canvas/6384ca1b-c034-481e-a588-3e3fbd640e0

RAGWebChatbot

Recursive Character Text Splitter

Inputs

Chunk Size: 1000

Chunk Overlap: 50

Additional Parameters

Output: RecursiveCharacterTextSplitter

Cheerio Web Scraper

Inputs

Text Splitter

URL: <https://docs.flowiseai.com/configuration/deployment/render>

Manage Links

Additional Parameters

Output: Document

In-Memory Vector Store

Inputs

Document

Embeddings

Output: 4

Memory Retriever

ChatOpenAI

Inputs

Cache

Connect Credential: OpenAI

Model Name: gpt-3.5-turbo

Temperature: 0.4

Allow Image Uploads

Additional Parameters

Output: ChatOpenAI

Conversational Retrieval QA Chain

Inputs

Chat Model

Vector Store Retriever

Memory

Return Source Documents

Additional Parameters

Output: ConversationalRetrievalQA

OpenAI Embeddings

Inputs

Connect Credential: OpenAI

Model Name: text-embedding-ada-002

Additional Parameters

Scraped Links

<https://docs.flowiseai.com/sitemap.xml>

Fetch Links

Scraped Links

<https://docs.flowiseai.com/>

<https://docs.flowiseai.com/getting-started>

<https://docs.flowiseai.com/contributing>

<https://docs.flowiseai.com/using-flowise>

<https://docs.flowiseai.com/using-flowise/api>

<https://docs.flowiseai.com/using-flowise/streaming>

<https://docs.flowiseai.com/using-flowise/embed>

Cancel Save

Web Crawl

Scrape XML Sitemap

Scrape relative links from XML sitemap URL

10

Selector (CSS)

Metadata

By following these steps, you should be able to

6. Connect your GitHub account.
7. Choose your forked Flowise repo and click "Connect".
8. Fill in your preferred Name and Region.
9. Select Docker as your Runtime.
10. Optionally, add app level authorization by clicking "Advanced" and adding Environment Variables like FLOWISE_USERNAME, FLOWISE_PASSWORD, and NODE_VERSION with the value 18.18.1.
11. Click on "Create Web Service".
12. Navigate to the deployed URL to access your deployed Flowise application.

How can I deploy Flowise to Render?

How can I deploy Flowise to AWS?

How can I set up Flowise as a developer?

React Flow

RAG Chatbot

<https://www.pinecone.io/>

The screenshot shows the Pinecone Console interface for creating a new index. The left sidebar contains navigation links: PROJECTS > Starter, Indexes, Collections, API Keys, and Members. The main content area is titled 'Create a new Index' and includes a 'Back to Indexes' link. A 'Name' input field contains 'flowise' (annotated with a red box and '1'). Below it, the 'Configure your Index' section shows 'Dimensions' set to '1,536' (annotated with a red box and '2') and 'Metric' set to 'cosine'. The 'Capacity Mode' section has tabs for 'SERVERLESS', 'PODS', and 'STARTER' (selected). At the bottom, a 'Starter' section shows 'You are currently on the Free Plan' and a 'Create Index' button (annotated with a red box and '3'). A modal window on the right displays the index configuration for 'flowise' (Starter) with a green status dot. It contains two tables: one with METRIC (cosine), DIMENSIONS (1536), POD TYPE (starter), and HOST (https://flowise-d2c940b.svc.gcp-starter.pinecone.io); and another with CLOUD (gcp-starter), REGION (Iowa (us-central1)), and ENVIRONMENT (gcp-starter). A blue arrow points from the 'Create Index' button to the modal.

Pinecone Console

app.pinecone.io/organizations/-N_Zrnd0e-zx4U9wfG9X/projects/us-west1-gcp-free:d2c940b/create-index/starter

Unlock \$100 in serverless credits KGP

< Back to Indexes

Create a new Index

Name

flowise

1

Configure your Index

The dimensions and metric depend on the model you select.

Dimensions

1,536

2

Metric

cosine

Capacity Mode

SERVERLESS PODS **STARTER**

Starter

You are currently on the Free Plan

Cancel Create Index

3

flowise Starter

METRIC	DIMENSIONS	POD TYPE	HOST
cosine	1536	starter	https://flowise-d2c940b.svc.gcp-starter.pinecone.io

CLOUD	REGION	ENVIRONMENT
gcp-starter	Iowa (us-central1)	gcp-starter

RAG Chatbot

Flowise - Low-code LLM apps

localhost:3000/canvas/6384ca1b-c034-481e-a588-3e3fdbd640e0

RAGWebChatbot

Recursive Character Text Splitter

Inputs

Chunk Size: 1000

Chunk Overlap: 50

Additional Parameters

Output: RecursiveCharacterTextSplitter

Cheerio Web Scraper

Inputs

Text Splitter

URL: https://python.langchain.com/docs/expression_language/

Manage Links

Additional Parameters

Output: Document

Pinecone

Inputs

Document

Embeddings: Pinecone API

Pinecone Index: flowise

Additional Parameters

Output: Pinecone Retriever

ChatOpenAI

Inputs

Cache

Connect Credential: OpenAI

Model Name: gpt-3.5-turbo

Pinecone API

CREDENTIAL NAME: Pinecone API

Pinecone Api Key

Add

Conversational Retrieval QA Chain

Inputs

Chat Model

Vector Store Retriever

Memory

Return Source Documents: ☒

What is a LCEL?

LangChain Expression Language, or LCEL, is a declarative way to easily compose chains together. It was designed to support putting prototypes into production without code changes, from simple chains to complex ones with hundreds of steps. Some reasons to use LCEL include streaming support, optimized parallel execution, async support, retries and fallbacks configuration, access to intermediate results, and seamless integration with LangServe and LangSmith.

/docs/expresio...

Type your question: What is a LCEL?

Chatflow 분석

The screenshot displays the Flowise web interface in a browser window. The address bar shows the URL `localhost:3000/canvas/6384ca1b-c034-481e-a588-3e3fbd640e0`. The main workspace contains a chatflow titled "RAGWebChatbot" with two visible nodes: "Recursive Character Text Splitter" and "Cheerio Web Scraper". The "Recursive Character Text Splitter" node has inputs for "Chunk Size" (1000) and "Chunk Overlap" (50), and its output is "RecursiveCharacterTextSplitter". The "Cheerio Web Scraper" node has an input for "URL" (`https://python.langchain.com/docs/e...`) and its output is "Document".

An "Analyse Chatflow" modal is open in the center, showing configuration for the "LangSmith" credential. The modal includes a toggle switch set to "ON", a "Connect Credential" dropdown menu with "Langsmith" selected, a "Project Name" field with "RAG Web Chatbot", and an "On/Off" toggle switch. Below these are links for "LangFuse" (`https://langfuse.com`) and "Lunary" (`https://lunary.ai`). A "Save" button is at the bottom right of the modal.

On the right side of the interface, a menu is open with options: "Starter Prompts", "View Messages", "Speech to Text", "Duplicate Chatflow", "Load Chatflow", "Export Chatflow", "Analyse Chatflow" (highlighted with a red box and a blue arrow), and "Delete Chatflow". The background chatflow configuration shows inputs for "Conversational Retrieval QA Chain", "Store Retriever", and "Source Documents", and an output for "ConversationalRetrievalQAChain".

실습 - Chatflow 분석

smith.langchain.com/o/99048fc5-5e46-5edd-8a00-98665c3d24e7/projects/p/534faf7c-a1df-4f31-9d47-5793da9a8aa2?columnVisibilityModel=%7B"outputs"%3Afalse%2C"feedback_stats"%3Afalse%2C"reference_...

Personal > Projects > RAG Web Chatbot

RAG Web Chatbot

Traces LLM Calls All Runs Monitor Setup

Filters Last 7 days

		✓	Name	Input	Start Time	Latency	Dataset	Annotation Queue	Tokens	Cost
<input type="checkbox"/>	>	✓	RunnableSequence	Explain about Async s...	2024. 3. 11. 오후 10:4...	3.03s	<input type="checkbox"/>		1,837	\$0.0009985
<input type="checkbox"/>	>	✓	RunnableSequence	Show an example of u...	2024. 3. 11. 오후 10:4...	2.73s	<input type="checkbox"/>		1,735	\$0.0008845
<input type="checkbox"/>	>	✓	RunnableSequence	LCEL 주요 기능을 요약...	2024. 3. 11. 오후 10:4...	4.01s	<input type="checkbox"/>		1,693	\$0.0008635
<input type="checkbox"/>	>	✓	RunnableSequence	What is a LCEL?	2024. 3. 11. 오후 10:4...	4.53s	<input type="checkbox"/>		1,570	\$0.000883

Trace Collapse Stats Most relevant

RunnableSequence SUCCESS 4.53s 1,570

map:key:context 2.03s

FindDocs 2.02s

RetrievalChain... 2.02s

CondenseQ... 0.93s

ChatOpenAI 0.93s

VectorStoreRetr... 1.09s

ChatOpenAI 2.49s

Some runs have been hidden. [Show 14 hidden runs](#)

Runnable...

Run Feedback Metadata

Input

```
1 question:
  What is a
  LCEL?
2 chat_histo
  ry:
3 -
  message:
    What is a
    LCEL?
4 type:
  userMessag
  e
5 -
  message:
    LangChain
    Expression
```

YAML

Output

Flowise AI (2024) Tutorial

<https://youtube.com/playlist?list=PL4HikwTaYE0H7wBxhvQqxYcKOkZ4O3zXh&si=Z8RHj9IlyjoBQDm>



Build AI Apps WITHOUT Coding:
Flowise Tutorial #1



Creating Chatflows & LLM Chains -
FlowiseAI Tutorial #2



Combining Multiple Chains (Prompt
Chaining) - FlowiseAI Tutorial #3



Output Parsers & IfElse Function -
FlowiseAI Tutorial #4



Building Chatbots with Long-Term
Memory - FlowiseAI Tutorial #5



Chatting With Your Own Data! Chat,
Predict, & Analyze - FlowiseAI Tutoria...



Analysing Chatflows using LangSmith -
FlowiseAI Tutorial #7

Thank you 😊