

Bubble.io API Connector를 사용한 OpenAI 챗봇 구현

Bubble.io에서 OpenAI API를 호출하여
챗봇 UI를 만드는 방법

목차

1. 구현 플로우
요

2. API Connector
플러그인 설정

3. OpenAI API 요
청 설정

4. Chatbot UI 만
들기

5. Workflow 설
정

6. 테스트 및 디
버깅

1. 구현 플로우

- Bubble.io에서 OpenAI API를 활용하여 간단한 챗봇을 구현



- API Connector를 사용하여 OpenAI API 호출



- 사용자 입력을 받아 챗봇이 응답하는 UI 구현



- Workflow를 구성하여 데이터 흐름 제어



2. API Connector 플러그인 설정

1. Bubble.io 프로젝트 열기

2. Plugins 탭에서 'API Connector' 플러그인 설치

3. API Connector에서 새로운 API 추가

4. API 설정:

- Authentication: Private Key in Header

- Key Name: Authorization

- Key Value: Bearer YOUR_OPENAI_API_KEY

- Shared Headers: Content-Type: application/json

API Connector plugin설치

The screenshot shows the 'Install new plugins' interface in the Bubble Editor. On the left, there's a sidebar with filters and a search bar. The main area displays a list of plugins, with 'API Connector' at the top. Below it are 'Google Material Icons' and 'Ionic Elements'. Each plugin entry includes its icon, name, description, number of apps using it, and an 'Install' button. The 'API Connector' plugin is highlighted with a blue border.

Install new plugins

Filters

Sort by name **usage** rating date price

Search for a plugin...

Types deselect all

- ☒ Action
- ☒ Api
- ☒ Background Services
- ☒ Element
- ☒ Event
- ☒ Login Service
- ☒ Storage

Categories deselect all

- ☒ Artificial Intelligence (AI)
- ☒ Analytics

API Connector

The API connector lets you define your own API calls directly in the Bubble Editor and use them in your app.

By Bubble .b

1,240,155 apps Plugin page **Install**

Google Material Icons

Material design system library is a free library of icons built by Google, using Material Design principles.

By Bubble .b

1,172,143 apps **Install**

Ionic Elements

PAID PLAN Plugin subscriptions are available to Starter customers and above.

API Connector 설정



API Connector

referen

The API connector lets you define your own API calls directly in the Bubble Editor and use them in your app.

Learn about how to set up an API Connection [here](#) and see our guides for commonly built connections like [OpenAI](#).

You can also convert your connections into plugins that can be used in more than one app, and even shared with other Bubblers! See [your personal plugins page](#) for more details.

Add another API

클릭

Uninstall this plugin

API Connector 설정


아래와 같이 설정한다. OpenAi API Key 설정시 맨 앞에는 “Bearer”를 반드시 붙인다

API Name

Open AI

Authentication

Private key in header



Key name

Authorization

Private key

Bearer sk-proj-kBxKqFxqrftZS8b55cUeLCa6

Development key value

(optional)

Shared headers for all calls

Add a shared header

Shared parameters for all calls

Add a shared parameter

아래에서 오른쪽 위의 expand를 클릭한다

Name

API Call

expand

API Connector 설정

다음 OpenAI platform 주소에서 아래 스크립트를 복사해놓는다
<https://platform.openai.com/docs/api-reference/making-requests>

Making requests

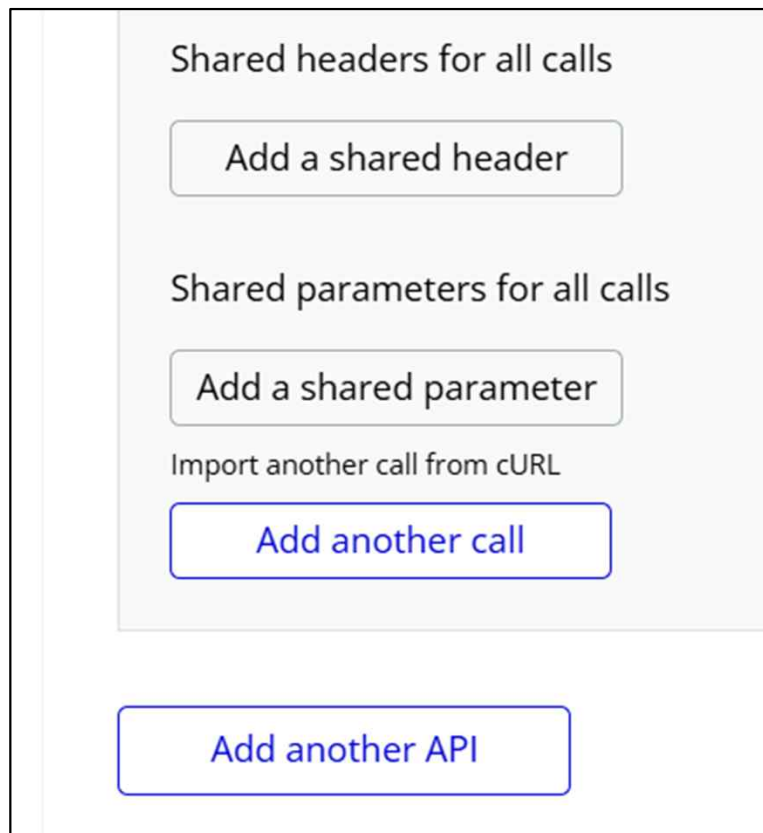
You can paste the command below into your terminal to run your first API request. Make sure to replace `$OPENAI_API_KEY` with your secret API key. If you are using a legacy user key and you have multiple projects, you will also need to [specify the Project Id](#). For improved security, we recommend transitioning to project based keys instead.

```
1 curl https://api.openai.com/v1/chat/completions \  
2   -H "Content-Type: application/json" \  
3   -H "Authorization: Bearer $OPENAI_API_KEY" \  
4   -d '{  
5     "model": "gpt-4o-mini",  
6     "messages": [{"role": "user", "content": "Say this is a test!"}],  
7     "temperature": 0.7  
8   }'
```



API Connector 설정

아래 Import another call from cURL를 클릭하고 복사한 내용을 붙여 넣고 [Import] 버튼을 클릭한다



The image shows the main interface of the API Connector tool. It has a light gray background with a white sidebar on the left. The sidebar contains four buttons: 'Add a shared header', 'Add a shared parameter', 'Add another call', and 'Add another API'. The main area has three sections: 'Shared headers for all calls' with an 'Add a shared header' button, 'Shared parameters for all calls' with an 'Add a shared parameter' button, and 'Import another call from cURL' with an 'Add another call' button.

Shared headers for all calls

Add a shared header

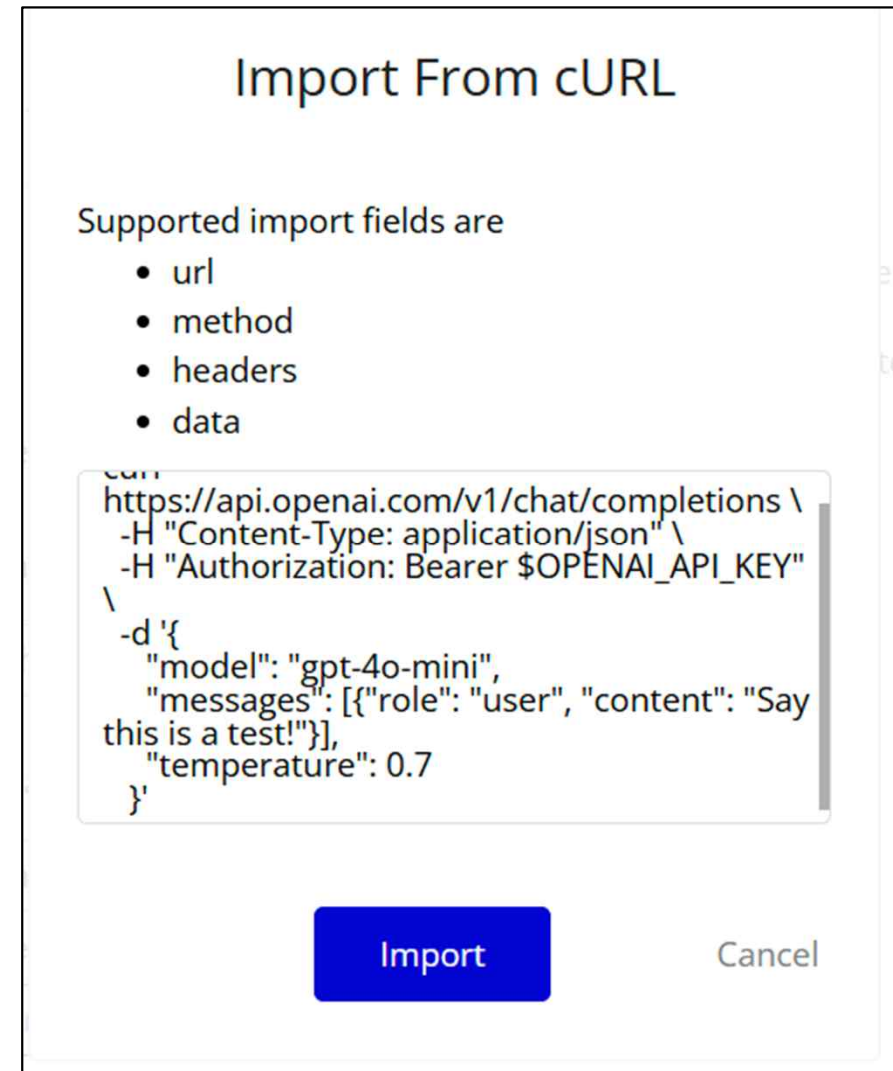
Shared parameters for all calls

Add a shared parameter

Import another call from cURL

Add another call

Add another API



The image shows the 'Import From cURL' dialog box. It has a title bar 'Import From cURL'. Below the title bar, it says 'Supported import fields are' followed by a list: url, method, headers, and data. Below the list is a text area containing a cURL command and its JSON body. At the bottom, there are two buttons: 'Import' and 'Cancel'.

Import From cURL

Supported import fields are

- url
- method
- headers
- data

```
curl https://api.openai.com/v1/chat/completions \
-H "Content-Type: application/json" \
-H "Authorization: Bearer $OPENAI_API_KEY" \
-d '{
  "model": "gpt-4o-mini",
  "messages": [{ "role": "user", "content": "Say
this is a test!" }],
  "temperature": 0.7
}'
```

Import Cancel

API Connector 설정

아래와 같이 나온다

NamecURL CallUse asDataData typeJSON

POSThttps://api.openai.com/v1/chat/completions(use [] for params)

Headers

KeyContent-TypeValueapplication/jsonPrivateOptional

KeyAuthorizationValueBearer \$OPENAI_API_KEYPrivateOptional

Add header

Body typeJSON

Parameters

Add parameter

Body (JSON object, use <> for dynamic values)

```
1 {
2   "model": "gpt-4o-mini",
3   "messages": [
4     {
5       "role": "user",
6       "content": "Say this is a test!"
7     }
8   ],
9   "temperature": 0.7
10 }
```

API Connector 설정

아래와 같이 “Say this is a test!”를 복사한 다음 <content>로 바꾸고 아래 Value에 붙여 넣고 Private은 체크 해제해준다

Body (JSON object, use <> for dynamic values)

```
1 {  
2   "model": "gpt-4o-mini",  
3   "messages": [  
4     {  
5       "role": "user",  
6       "content": <content>  
7     },  
8   ],  
9   "temperature": 0.7  
10 }
```

Key Value Private ☐ Allow blank ☐

Include errors in response and allow workflow actions to continue ☐

Capture response headers ☐

 You need to initialize this call before it will work.

[Initialize call](#)

Manually enter API response

3. OpenAI API 요청 설정

- 1. add Another API 에서 Name : OpenAI API Call
- 2. 설정값 입력:
 - - Use as: Action
 - - Data Type: JSON
 - - Method: POST
 - - API URL: <https://api.openai.com/v1/chat/completions>
- 3. JSON Body 설정:
- ```
curl https://api.openai.com/v1/chat/completions \
```
- ```
-H "Content-Type: application/json" \
```
- ```
-H "Authorization: Bearer $OPENAI_API_KEY" \
```
- ```
-d '{
```
- ```
 "model": "gpt-4o-mini",
```
- ```
  "messages": [{"role": "user", "content": "Say this is a test!"}],
```
- ```
 "temperature": 0.7
```
- ```
}'
```

API Connector 설정

Name을 OpenAI API Call , Uses as를 Action으로 변경한다

The screenshot displays the configuration interface for an API Connector. The 'Name' field is set to 'OpenAI API Call'. The 'Use as' dropdown is set to 'Action'. The 'Data type' dropdown is set to 'JSON'. The HTTP method is set to 'POST' and the URL is 'https://api.openai.com/v1/chat/completions'. The 'Headers' section contains two entries: 'Content-Type' with value 'application/json' and 'Authorization' with value 'Bearer \$OPENAI_API_KEY'. Both headers have 'Private' and 'Optional' checkboxes. An 'Add header' button is located below the headers. The 'Body type' dropdown is set to 'JSON'. The 'Parameters' section is partially visible at the bottom.

Name: OpenAI API Call Use as: Action ▼ Data type: JSON ▼

POST ▼ https://api.openai.com/v1/chat/completions (use [] for params)

Headers

Key	Content-Type	Value	application/json	Private	<input type="checkbox"/>	Optional	<input type="checkbox"/>
Key	Authorization	Value	Bearer \$OPENAI_API_KEY	Private	<input type="checkbox"/>	Optional	<input type="checkbox"/>

Add header

Body type: JSON ▼

Parameters

API Connector 설정

상단의 [Add a shared header]를 클릭한다

The screenshot displays the API Connector configuration interface. It includes the following elements:

- API Name:** A text input field containing "Open AI".
- Key name:** A text input field containing "Authorization".
- Authentication:** A dropdown menu with "Priv" selected.
- Private key:** A dropdown menu with "Bearer sk-" selected.
- Development key value:** A text input field containing "(optional)".
- Shared headers for all calls:** A section with a button labeled "Add a shared header".
- Shared parameters for all calls:** A section with a button labeled "Add a shared parameter".

API Connector 설정

하단에 있는 Key와 Value값을 아래와 같이 복사해서 붙여 넣는다

API Name

Open AI

Authentication

Private key in header

Key name

Authorization

Private key

Bearer sk-proj-kBxKqFxqrftZS8b55cUeLCat

Development key value

(optional)

Shared headers for all calls

Key

Content-Type

Value

application/json

Add a shared header

Shared parameters for all calls

Add a shared parameter

API Connector 설정

하단의 [Initialize call]을 클릭하면 왼쪽과 같은 창이 나타난다

Include errors in response and allow workflow actions to continue ☐

Capture response headers ☐

⚠ You need to initialize this call before it will work.

[Initialize call](#) [Manually enter API response](#)

Import another call from cURL

[Add another call](#)

Returned values - OpenAI API Call

You can modify the data types that are returned by the call. This affects how you can use the data in Bubble. If you chose 'Ignore field', the fields won't be shown in the dropdowns.

id <small>chatcmpl-B1T4C07U5re2q6HtNFot6MjlrN7m1</small>	text
object <small>chat.completion</small>	text
created <small>1739689760</small>	number
model <small>gpt-4o-mini-2024-07-18</small>	text
choices (list) <small>(see fields below)</small>	OpenAI API Call choice
index <small>0</small>	number
message role <small>assistant</small>	text
message content <small></small>	text

[SAVE](#) [Cancel](#)

API Connector 설정

스크롤 해서 “message content”를 찾아보면 아래에 희미하게 응답 메시지를 확인 할 수 있다 “This is a tets! How can I assist you further?”
[Save] 버튼을 클릭해준다

Returned values - OpenAI API Call

message role assistant	text
message content This is a test! How can I assist you further?	text
message refusal (no sample data)	text
logprobs (no sample data)	text
finish_reason stop	text

SAVE

Cancel

API Connector 설정

Reinitialize call 버튼 오른쪽의 Manually enter API response를 클릭하면 우측과 같은 Response 값들을 확인 할 수 있다

Include errors in response and allow workflow actions to continue

Capture response headers

Reinitialize call

Manually enter API response

Import another call from cURL

Add another call

Manually enter API response

```
1 {
2   "id": "chatcmpl-B1T5fn1soSSiaHoavchkVvIO0WHfD",
3   "object": "chat.completion",
4   "created": 1739689851,
5   "model": "gpt-4o-mini-2024-07-18",
6   "choices": [
7     {
8       "index": 0,
9       "message": {
10        "role": "assistant",
11        "content": "This is a test! How can I assist you further?",
12        "refusal": null
13      },
14      "logprobs": null,
15      "finish_reason": "stop"
16    }
17  ],
18  "usage": {
19    "prompt_tokens": 13,
20    "completion_tokens": 13,
21    "total_tokens": 26,
22  }
23 }
```

SAVE

Cancel

4. Chatbot UI 만들기

강의 중 진행되는 앱을 참조하여 실습한다

- 1. 'Add New Page'로 새 페이지 추가 (페이지 이름: chatbot)
- 2. 화면 구성 요소 추가:
 - - Group (채팅창) - 'Group_Chat'
 - - Group (입력 영역) - 'Group_Input'
- 3. 요소 배치:
 - - Repeating Group (채팅 목록 표시)
 - - Input (질문 입력창) - 'Input 질문입력'
 - - Button (보내기) - 'Button 보내기 '

<https://www.youtube.com/watch?app=desktop&v=0k9N4myrLpg&t=0s>

[앱 실행 화면]


Open AI

챗GPT에게 무엇이든 물어보세요

하늘은 왜 파란가요?

질문 보내기

대화 목록 보기



답변

하늘이 파란 이유는 대기 중의 공기 분자와 작은 입자들이 태양광을 산란시키기 때문입니다. 태양빛은 여러 색깔의 빛으로 이루어져 있으며, 이들 색은 서로 다른 파장을 가지고 있습니다. 파란색 빛은 파장이 짧아서 대기 중의 분자와 잘 상호작용하여 산란됩니다.

태양이 하늘의 높은 위치에 있을 때, 파란색 빛이 더 많이 산란되어 하늘이 파랗게 보이게 됩니다. 반면, 해가 지거나 떠 있는 각도가 낮을 때는 붉은색과 주황색 빛이 더 많이 산란되기 때문에 저녁이나 아침 하늘은 붉은 색조로 물들게 됩니다. 이렇게 하늘의 색깔은 시간과 태양의 위치에 따라 변할 수 있습니다.

Open AI

챗GPT에게 무엇이든 물어보세요

하늘은 왜 파란가요?

질문 보내기

대화 목록 보기

답변

하늘이 파란 이유는 대기 중의 공기 분자와 작은 입자들이 태양광을 산란시키기 때문입니다. 태양빛은 여러 색깔의 빛으로 이루어져 있으며, 이들 색은 서로 다른 파장을 가지고 있습니다. 파란색 빛은 파장이 짧아서 대기 중의 분자와 잘 상호작용하여 산란됩니다.

태양이 하늘의 높은 위치에 있을 때, 파란색 빛이 더 많이 산란되어 하늘이 파랗게 보이게 됩니다. 반면, 해가 지거나 떠 있는 각도가 낮을 때는 붉은색과 주황색 빛이 더 많이 산란되기 때문에 저녁이나 아침

(모바일)

5. Workflow 설정

- 1. 버튼 클릭 시 API 호출 Workflow 추가
 - - Action: ChatGPT_Request (API Connector에서 설정한 API 호출)
 - - Parameters: <YOUR_INPUT_TEXT> → 'Input 질문입력's value'
- 2. API 응답을 화면에 표시
 - - API 응답 값을 State에 저장
 - - 'Group_Chat' 내 'Repeating Group' 업데이트
- 3. Custom State 활용:
 - - State Name: ChatHistory
 - - Type: List of text
 - - Value: Page chatbot's ChatHistory :plus item API Response's message

6. 테스트 및 디버깅

1. 페이지 실행 후 입력 → 버튼 클릭 테스트

2. API 응답 정상 출력 확인

3. Bubble.io Debugging 도구 활용하여 API 응답 검토

4. UI 최적화 및 사용자 경험 개선

감사합니다