Hello World

API Reference

The API is organized around REST (Django-rest-framework). Our API has predictable resource-oriented URLs, accepts form-encoded request bodies, returns JSON-encoded responses, and uses standard HTTP response codes.

The API doesn't support bulk updates. You can work on only one object per request.

Authentication

It doesn't require authentication and it's free forever, but Openai requires a key, which you need to put in the .env file as shown below.

```
OPENAI_API_KEY=your_api_key
```

The keys are prefixed with sk-.

Your API keys carry a lot of privileges, so make sure to keep them safe! Do not share your secret API keys in publicly accessible areas such as GitHub, client-side code, etc.

Errors

API uses conventional HTTP response codes to indicate the success or failure of an API request. In general: Codes in the 2xx range indicate success.

Codes in the 4xx range indicate an error that failed given the information provided .

Codes in the 5xx range indicate an error with servers (these are rare).

Application submission method

You can use it to see the answer to your question.

```
POST /api/req/
```

which includes your response after data processing.

The method of sending information must be of POST type, otherwise you will encounter an error.

Formatting Your Request

Request Body
When submitting data to a resource via POST, you must submit your payload in JSON.

```
POST  http://127.0.0.1:8000/api/req/  HTTP/1.1
Content-Type: application/json
```

```
{

    " question": " your question"

 }
```

Content-Type Header

All responses are returned in JSON format. We specify this by sending the Content-Type header.

```
POST  http://127.0.0.1:8000/api/req/  HTTP/1.1
Content-Type: application/json
```

```
HTTP/1.1 200 OK

Content-Type: application/json

 {

     "message": "message"

 }
```

Hamidreza Ebrahimi

https://github.com/hamidp30/Interns-Tasks-2-and-3