🧠 Code Base Genius

Generate rich, human-readable repository documentation with **Gemini 2.0 Flash AI summaries**, clear structure maps, and code insights.

🔗 Enter GitHub Repository URL:

https://github.com/openai/tiktoken

🚀 Generate Documentation

🧭 Repository Summary (AI)

This is the OpenAI Python library for accessing the OpenAI API (Python 3.8+). It offers both synchronous and asynchronous clients generated from the OpenAI OpenAPI specification. Install with `pip install openai`. Key features include: text generation via the Responses/Chat Completions APIs, vision support, streaming responses, and a Realtime API for low-latency conversations. The library supports automatic pagination for list methods, file uploads, webhook verification, and detailed error handling. Advanced features include configurable HTTP clients, retries, timeouts, and access to raw responses. For Azure OpenAI, use the `AzureOpenAI` class.

› 📁 Repository Structure

📖 Full Markdown Documentation

📘 Repository Documentation

Generated: 2025-11-01 22:45:58

This is the OpenAI Python library for accessing the OpenAI API (Python 3.8+). It offers both synchronous and asynchronous clients generated from the OpenAI OpenAPI specification. Install with `pip install openai`. Key features include: text generation via the Responses/Chat Completions APIs, vision support, streaming responses, and a Realtime API for low-latency conversations. The library supports automatic pagination for list methods, file uploads, webhook verification, and detailed error handling. Advanced features include configurable HTTP clients, retries, timeouts, and access to raw responses. For Azure OpenAI, use the `AzureOpenAI` class.

~~~
# The OpenAI Python Library: A Story

Imagine you're a software developer, and you want to build an application that uses

That's where the OpenAI Python library comes in!  It's like a friendly translator t

</details>
<details><summary><b>What Does It Do?</b></summary>

The OpenAI Python library provides a convenient way to access the OpenAI REST API t

*   **Authentication:** Securely handles your API key so you don't have to worry ab
*   **Request Formatting:** Converts your Python objects into the correct format fo
*   **Response Parsing:** Transforms the API's responses into Python objects that a
*   **Error Handling:** Gracefully handles errors and provides informative error me
*   **Asynchronous Operations:**  Supports both synchronous (blocking) and asynchro
*   **Realtime API access:** Supports low-latency, multi-modal conversational exper

</details>
<details><summary><b>How Does It Work?</b></summary>

Under the hood, the library uses a powerful HTTP client called `httpx` (or optiona

Here's a simplified diagram illustrating the process:

```mermaid
graph LR
    A[Your Python Code] --> B(OpenAI Python Library);
    B --> C{OpenAI API};
    C --> B;
    B --> A;
    style C fill:#f9f,stroke:#333,stroke-width:2px
~~~

Or as a Markdown table:

Step	Action	Component Involved
1	Code calls the library	Your Python Code

Step	Action	Component Involved
2	Library prepares and sends the API request	OpenAI Python Library
3	API receives and processes the request	OpenAI API
4	API sends response back	OpenAI API
5	Library receives and parses the response	OpenAI Python Library
6	Code gets structured data back	Your Python Code

▶ Use Cases

▶ Getting Started

▶ Key Features

▶ 🧩 Installation

▶ Installation

▶ ▶️ Usage

▶ Usage

▶ 📚 API Reference

▶ 🧠 Code Analysis

▶ 🤖Gemini AI Code Summary

▶ Codebase Analysis Summary: openai-python

🪶 Rendered Mermaid Diagrams

Diagram 1

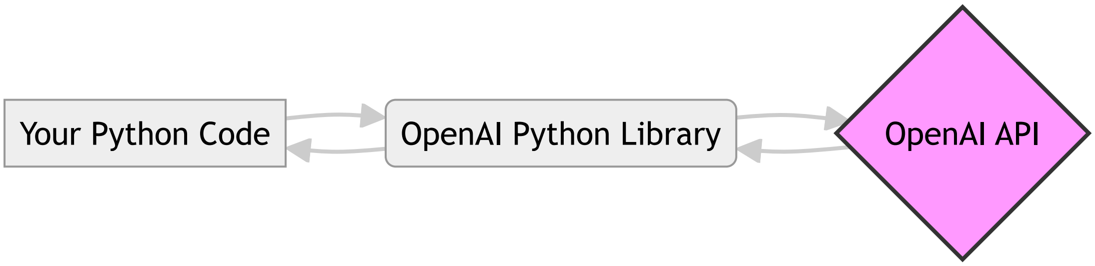
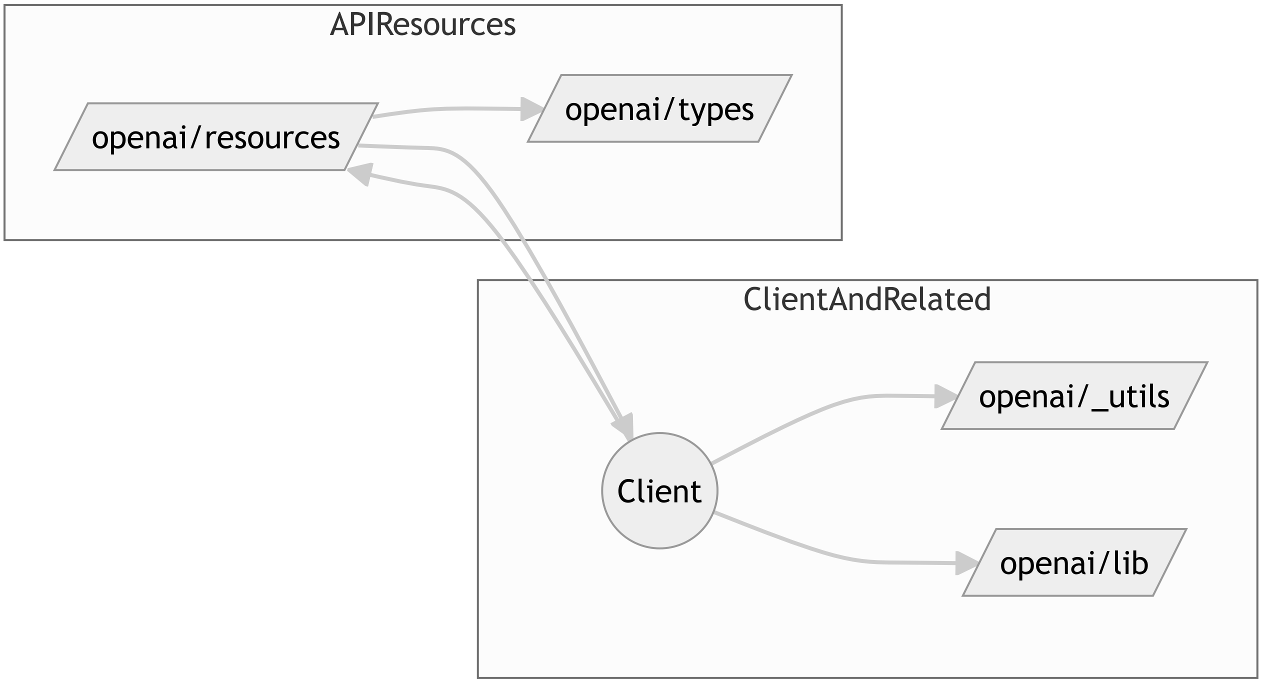


Diagram 2



✅ Documentation generated successfully!

Code Base Genius — AI documentation orchestrated by Jac + Gemini 2.0 Flash.