



# 19 PYTHON LIBRARIES EVERY AI ENGINEER SHOULD KNOW

USES

ADVANTAGES

LIMITATIONS

SLIDE TO EXPLORE



# PYDANTIC

## USED FOR

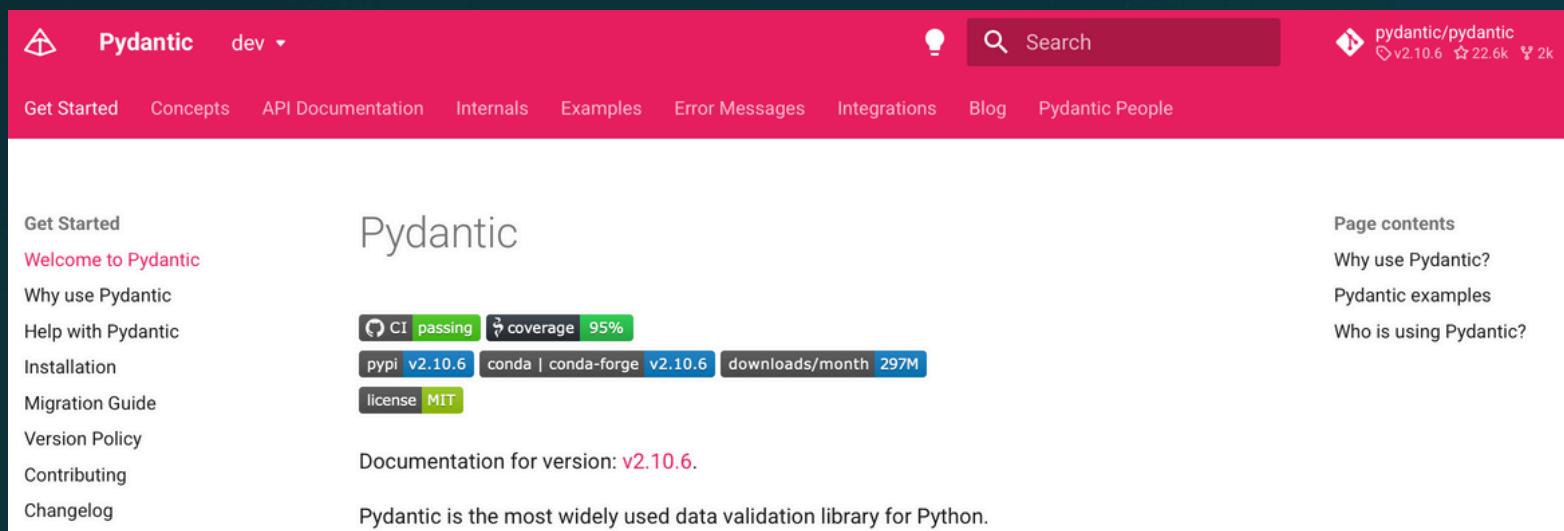
Used for data validation and settings management using Python type annotations.

## ADVANTAGE

Provides automatic data validation, serialization, and documentation generation.

## LIMITATION

Can have a steeper learning curve for developers unfamiliar with Python type hints.



The screenshot shows the official Pydantic website at [pydantic-docs.helpmanual.io](https://pydantic-docs.helpmanual.io/). The header includes a logo, navigation links for 'Get Started', 'Concepts', 'API Documentation', 'Internals', 'Examples', 'Error Messages', 'Integrations', 'Blog', and 'Pydantic People', and a search bar. The main content area features the title 'Pydantic' in large font, followed by a 'Page contents' sidebar with links to 'Why use Pydantic?', 'Pydantic examples', and 'Who is using Pydantic?'. The central content area displays CI status (CI passing), coverage (95%), package versions (pypi v2.10.6, conda | conda-forge v2.10.6), downloads/month (297M), and license (MIT). Below this, a note says 'Documentation for version: v2.10.6.' and a summary states 'Pydantic is the most widely used data validation library for Python.'

# PYDANTIC SETTINGS

## USED FOR

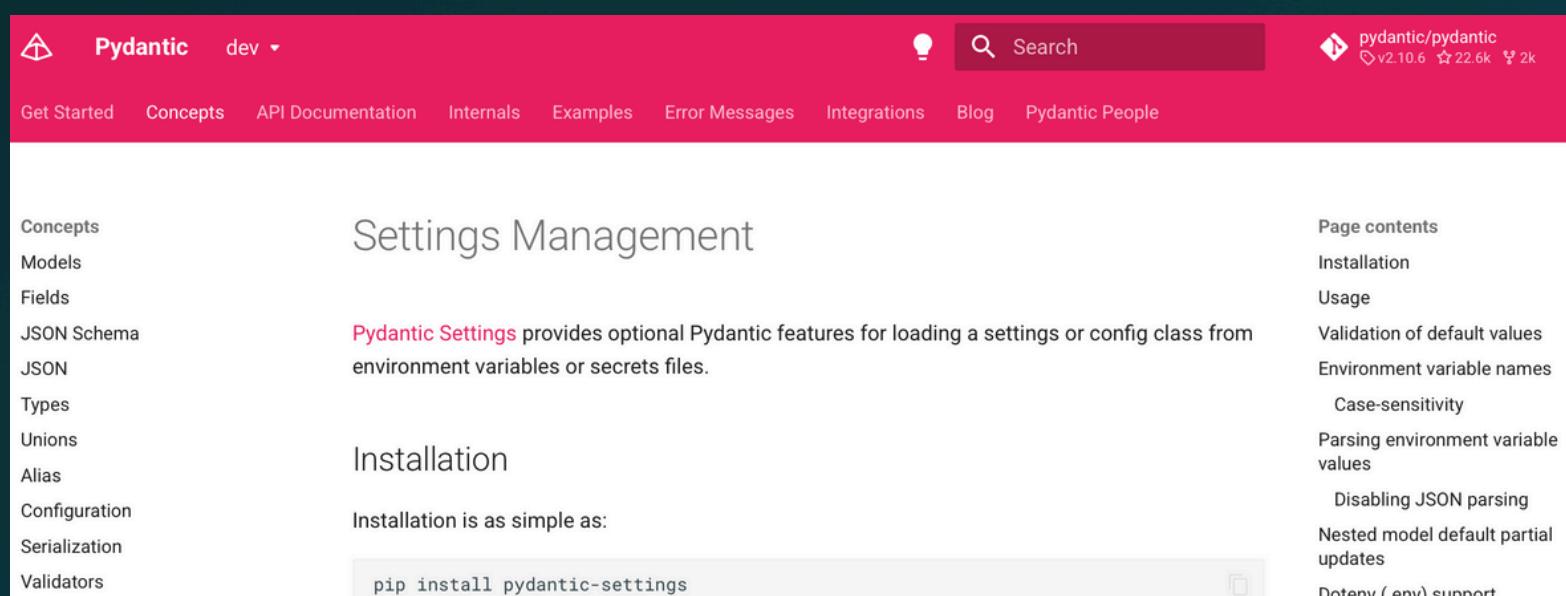
Used for loading and validating configuration from various sources like environment variables and files.

## ADVANTAGE

Simplifies configuration management with type checking and automatic environment variable parsing.

## LIMITATION

May add complexity to smaller projects that don't require extensive configuration management.



The screenshot shows the Pydantic Settings documentation page. The top navigation bar includes links for 'Get Started', 'Concepts', 'API Documentation', 'Internals', 'Examples', 'Error Messages', 'Integrations', 'Blog', and 'Pydantic People'. The main content area features a sidebar with links to 'Concepts', 'Models', 'Fields', 'JSON Schema', 'JSON', 'Types', 'Unions', 'Alias', 'Configuration', 'Serialization', and 'Validators'. The main content area has two sections: 'Settings Management' and 'Installation'. The 'Settings Management' section describes Pydantic Settings as providing optional features for loading settings or config classes from environment variables or secrets files. The 'Installation' section explains that installation is simple, showing the command 'pip install pydantic-settings'. A 'Page contents' sidebar on the right lists various topics such as Installation, Usage, Validation of default values, Environment variable names, Case-sensitivity, Parsing environment variable values, Disabling JSON parsing, Nested model default partial updates, and Dotenv (.env) support.

# PYTHON-DOTENV

## USED FOR

Used for loading environment variables from a .env file into the application.

## ADVANTAGE

Simplifies management of environment-specific settings and secrets.

## LIMITATION

Requires careful handling to avoid accidentally committing sensitive information to version control.

The screenshot shows the GitHub documentation for the `python-dotenv` project. The top navigation bar includes links for Home, Changelog, Contributing, Reference, License, and a GitHub icon. The main content area is titled "dotenv" and contains the `dotenv_values` function documentation. The function signature is `dotenv_values(dotenv_path=None, stream=None, verbose=False, interpolate=True, encoding='utf-8')`. A note below the function says "Parse a .env file and return its content as a dict." To the right, there is a "Table of contents" sidebar listing methods: `dotenv_values()`, `find_dotenv()`, `get_cli_string()`, `get_key()`, `load_dotenv()`, `set_key()`, and `unset_key()`.

# FastAPI

## USED FOR

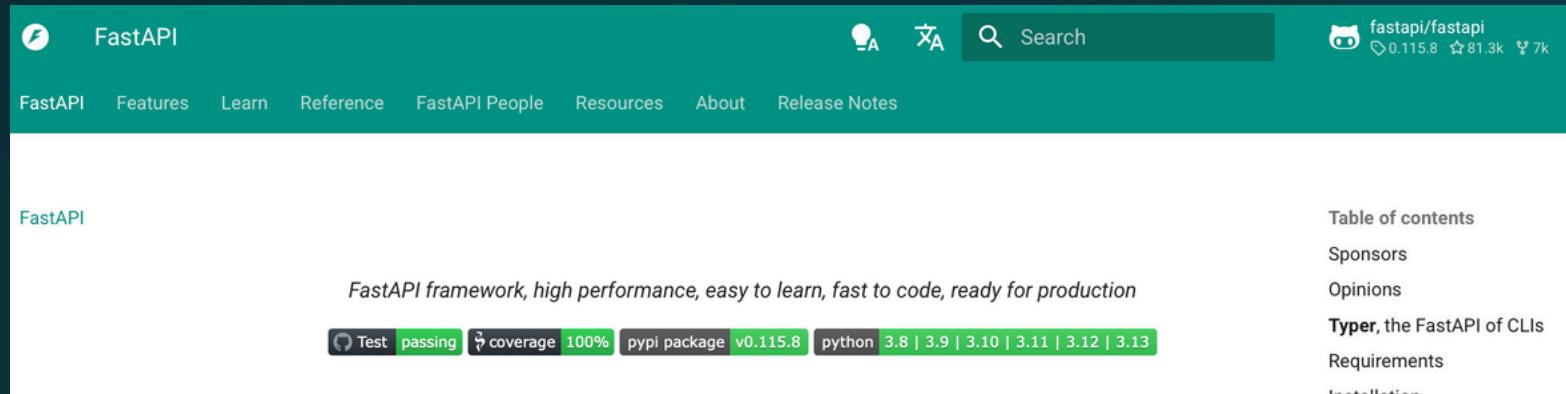
Used for building high-performance web APIs with Python.

## ADVANTAGE

Offers automatic API documentation, data validation, and asynchronous support.

## LIMITATION

May have a steeper learning curve for developers new to asynchronous programming.



The screenshot shows the official FastAPI website. At the top, there's a navigation bar with links for 'FastAPI', 'Features', 'Learn', 'Reference', 'FastAPI People', 'Resources', 'About', and 'Release Notes'. To the right of the navigation are icons for a microphone, a document, a search bar, and a GitHub repository link: `fastapi/fastapi`, `0.115.8`, `81.3k`, and `7k`. Below the navigation, the main content area has a heading 'FastAPI' and a sub-heading: 'FastAPI framework, high performance, easy to learn, fast to code, ready for production'. It also features several green buttons for 'Test passing', 'coverage 100%', 'pypi package v0.115.8', and 'python 3.8 | 3.9 | 3.10 | 3.11 | 3.12 | 3.13'. On the right side of the content area, there are links for 'Table of contents', 'Sponsors', 'Opinions', 'Typer, the FastAPI of CLIs', 'Requirements', and 'Installation'.

# CELERY

## USED FOR

Used for distributed task queues and asynchronous job processing.

## ADVANTAGE

Enables scalable background task execution and scheduled jobs.

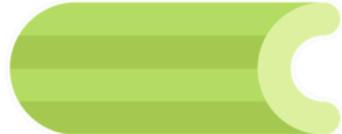
## LIMITATION

Can be complex to set up and debug, especially in distributed environments.

[Celery 5.4.0 documentation](#) » [Celery - Distributed Task Queue](#)

[next](#) | [modules](#) | [index](#)

This document describes the current stable version of Celery (5.4). For development docs, [go here](#).



## Celery - Distributed Task Queue

Celery is a simple, flexible, and reliable distributed system to process vast amounts of messages, while providing operations with the tools required to maintain such a system.

It's a task queue with focus on real-time processing, while also supporting task scheduling.

Celery has a large and diverse community of users and contributors, you should come

Star 25,638

# PyOPG

## for PostgreSQL

### USED FOR

Used for interacting with PostgreSQL databases in Python applications.

### ADVANTAGE

Provides efficient and Pythonic access to PostgreSQL features.

### LIMITATION

Limited to PostgreSQL databases, lacking support for other database systems.

The screenshot shows the PyPI project page for `psycopg2`. At the top, there's a navigation bar with a search bar, help links, and user authentication options. The main title is `psycopg2 3.2.5`, with a green "Latest version" button. Below the title, there's a "pip install psycopg2" button and a release date of "Released: Feb 22, 2025". A brief description at the bottom states "PostgreSQL database adapter for Python".

# PyMongo

## for MongoDB

### USED FOR

Used for working with MongoDB databases in Python applications.

### ADVANTAGE

Offers a native Python interface to MongoDB with support for its document-oriented model.

### LIMITATION

Specific to MongoDB, not suitable for relational database interactions.

## PyMongo

pypi v4.11.1 python 3.9 | 3.10 | 3.11 | 3.12 | 3.13 downloads/month 41M docs passing

### About

The PyMongo distribution contains tools for interacting with MongoDB database from Python. The `bson` package is an implementation of the [BSON format](#) for Python. The `pymongo` package is a native Python driver for MongoDB. The `gridfs` package is a [gridfs](#) implementation on top of `pymongo`.

PyMongo supports MongoDB 4.0, 4.2, 4.4, 5.0, 6.0, 7.0, and 8.0.

# SQLAlchemy

## USED FOR

Used for database abstraction and ORM (Object-Relational Mapping) in Python.

## ADVANTAGE

Provides a flexible and powerful toolkit for working with various SQL databases.

## LIMITATION

Can have a steep learning curve due to its extensive feature set and abstractions.

## PyMongo

pypi v4.11.1 python 3.9 | 3.10 | 3.11 | 3.12 | 3.13 downloads/month 41M docs passing

### About

The PyMongo distribution contains tools for interacting with MongoDB database from Python. The `bson` package is an implementation of the [BSON format](#) for Python. The `pymongo` package is a native Python driver for MongoDB. The `gridfs` package is a [gridfs](#) implementation on top of `pymongo`.

PyMongo supports MongoDB 4.0, 4.2, 4.4, 5.0, 6.0, 7.0, and 8.0.

# ALEMBIC

## USED FOR

Used for database migration management in SQLAlchemy-based applications.

## ADVANTAGE

Simplifies schema evolution and version control for database structures.

## LIMITATION

Primarily designed for use with SQLAlchemy, which may limit its applicability in non-SQLAlchemy projects.



## Quick Start

Flask-SQLAlchemy simplifies using SQLAlchemy by automatically handling creating, using, and cleaning up the SQLAlchemy objects you'd normally work with. While it adds a few useful features, it still works like SQLAlchemy.

This page will walk you through the basic use of Flask-SQLAlchemy. For full capabilities and customization, see the rest of these docs, including the API docs for the `SQLAlchemy` object.

### Contents

- [Quick Start](#)
- [Check the SQLAlchemy Documentation](#)
- [Installation](#)
- [Initialize the Extension](#)
- [About the `SQLAlchemy` object](#)
- [Configure the Extension](#)
- [Define Models](#)
- [Create the Tables](#)

## Check the SQLAlchemy Documentation

Flask-SQLAlchemy is a wrapper around SQLAlchemy. You should follow the [SQLAlchemy Tutorial](#) to learn about how to use it, and consult its documentation for detailed information about its features. These docs show how to set up Flask-SQLAlchemy itself, not how to use SQLAlchemy. Flask-SQLAlchemy sets up the engine and scoped session automatically, so you can skip those parts of the SQLAlchemy tutorial.

This guide assumes you are using SQLAlchemy 2.x, which has a new API for defining models and better support for Python type hints and data classes. If you are using SQLAlchemy 1.x, see [Legacy Quickstart](#).

# PANDAS

## USED FOR

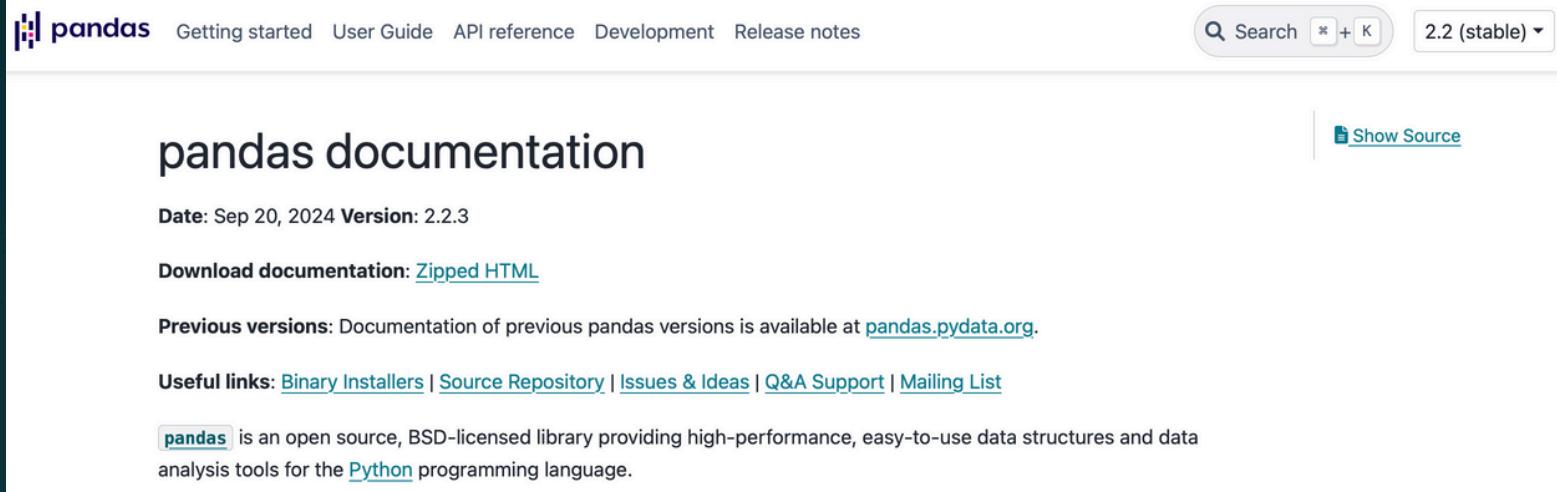
Used for data manipulation and analysis in Python.

## ADVANTAGE

Offers powerful data structures and tools for working with structured data.

## LIMITATION

Can be memory-intensive for very large datasets, potentially impacting performance.



The screenshot shows the official pandas documentation homepage. At the top, there's a navigation bar with links for "Getting started", "User Guide", "API reference", "Development", and "Release notes". To the right of the navigation is a search bar and a link for "2.2 (stable)". Below the navigation, the title "pandas documentation" is displayed, along with the date "Date: Sep 20, 2024 Version: 2.2.3". There are links for "Download documentation: Zipped HTML" and "Show Source". Further down, there are sections for "Previous versions", "Useful links", and a brief description of what pandas is: "pandas is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language."

# OpenAI API

## USED FOR

Used for integrating AI models and capabilities into applications.

## ADVANTAGE

Provides access to state-of-the-art language models and AI functionalities

## LIMITATION

Requires careful management of API usage and costs, especially for high-volume applications.

The screenshot shows the OpenAI Platform API Reference website. The top navigation bar includes a search bar and a user icon. The left sidebar has a 'API REFERENCE' section with links to 'Introduction', 'Authentication', 'Making requests', and 'Streaming'. The main content area is titled 'Introduction' and contains text about the RESTful, streaming, and realtime APIs available for interacting with the OpenAI platform via HTTP. It also mentions language-specific SDKs for various programming languages.

OpenAI Platform

API REFERENCE

Introduction Authentication Making requests Streaming

Introduction

This API Reference describes the RESTful, streaming, and realtime APIs you can use to interact with the OpenAI platform. REST APIs are usable via HTTP in any environment that supports HTTP requests.

On top of the core REST APIs, we also offer language-specific SDKs for a variety of popular backend programming languages like JavaScript, Python, and C#. You can see the full list of supported languages [on the libraries reference page](#).

# INSTRUCTOR

## USED FOR

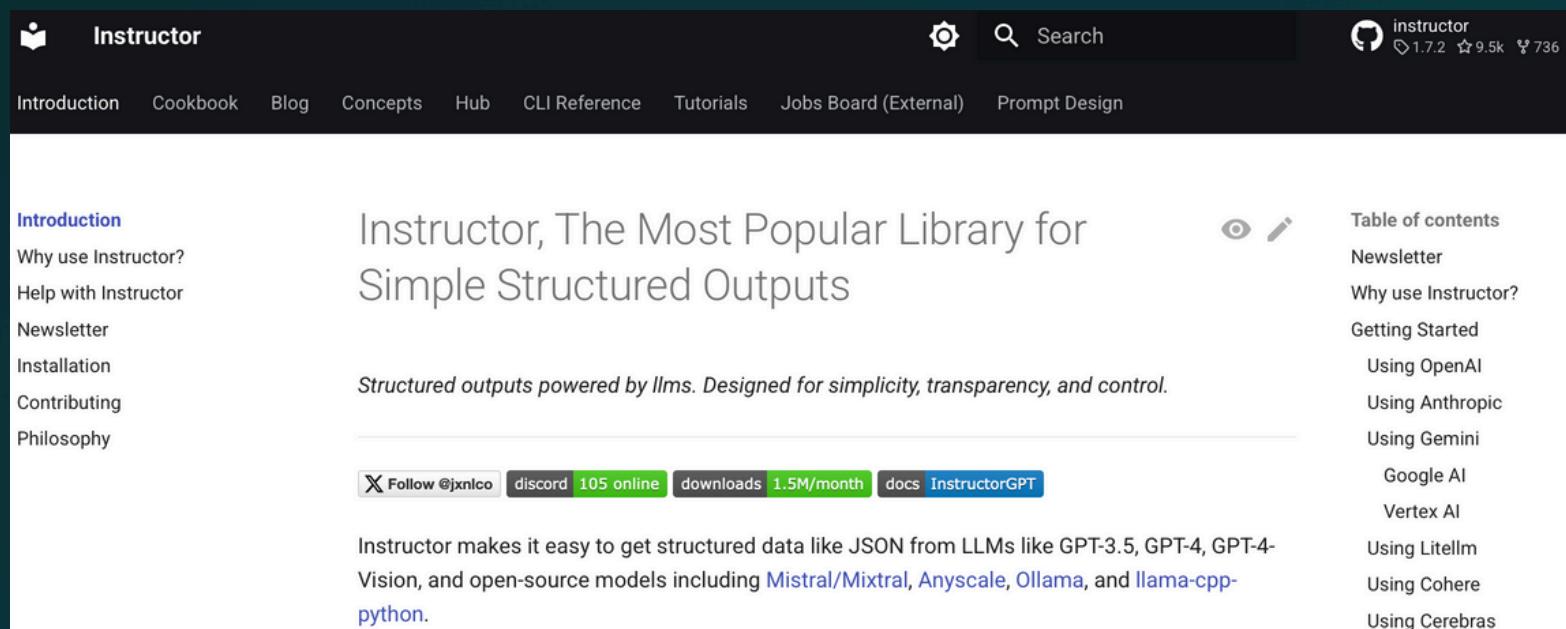
Used for enhancing interactions with language models and obtaining structured outputs.

## ADVANTAGE

Improves reliability and consistency of AI model outputs through structured prompts.

## LIMITATION

May add an extra layer of complexity to AI integrations compared to direct API calls.



The screenshot shows the Instructor website homepage. At the top, there's a dark header bar with the word "Instructor" and a navigation menu including "Introduction", "Cookbook", "Blog", "Concepts", "Hub", "CLI Reference", "Tutorials", "Jobs Board (External)", and "Prompt Design". To the right of the menu are search and settings icons. Below the header, the main content area has a white background. On the left, there's a sidebar with links to "Introduction", "Why use Instructor?", "Help with Instructor", "Newsletter", "Installation", "Contributing", and "Philosophy". The main content area features a large title "Instructor, The Most Popular Library for Simple Structured Outputs" with a "Edit" icon to its right. Below the title is a subtitle "Structured outputs powered by llms. Designed for simplicity, transparency, and control." At the bottom of the content area, there's a footer with social media links ("Follow @jxnlico", "discord 105 online", "downloads 1.5M/month", "docs InstructorGPT"), a brief description of the library's purpose, and a list of supported models like "Mistral/Mixtral", "Anyscale", "Ollama", and "Llama-cpp-python". To the right of the main content, there's a sidebar titled "Table of contents" containing links to various sections such as "Newsletter", "Why use Instructor?", "Getting Started", "Using OpenAI", "Using Anthropic", "Using Gemini", "Google AI", "Vertex AI", "Using Liteml", "Using Cohere", and "Using Cerebras".

# LangChain or Llamaindex

## USED FOR

Used for building applications with large language models (LLMs).

## ADVANTAGE

Simplifies the process of chaining together multiple AI operations and managing context.

## LIMITATION

Can introduce additional dependencies and complexity to AI projects.

The screenshot shows the LangChain website's navigation bar at the top, featuring links for Integrations, API Reference, More, v0.3, a search bar, and a GitHub button. The main content area has a sidebar on the left with links to Introduction, Tutorials, How-to guides, Conceptual guide, Ecosystem (LangSmith, LangGraph), Versions (v0.3, v0.2), and Pydantic compatibility. The main content area displays the "Introduction" page, which includes a brief description of LangChain as a framework for developing applications powered by large language models, a statement about simplifying the LLM application lifecycle, and a bulleted list of features: Development, Productionization, and Deployment. A sidebar on the right lists various guides and resources, such as Architecture, Guides, Tutorials, How-to guides, Conceptual guide, Integrations, API reference, Ecosystem (LangSmith, LangGraph), Additional resources, Versions, Security, and Contributing.

# VECTOR DATABASES

Pinecone, Weaviate, Qdrant, or pgvector

## USED FOR

Used for efficient storage and retrieval of high-dimensional vector data.

## ADVANTAGE

Enables fast similarity search and AI-powered information retrieval.

## LIMITATION

May require additional infrastructure and expertise to set up and optimize effectively.

## Pinecone

 Open in Colab

 Open on GitHub

Pinecone is a vector database with broad functionality.

This notebook shows how to use functionality related to the Pinecone vector database.

### Setup #

To use the PineconeVectorStore you first need to install the partner package, as well as the other packages used throughout this notebook.

# LANGFUSE OR LANGSMITH

## USED FOR

Used for monitoring and observability of AI and LLM applications.

## ADVANTAGE

Provides insights into AI model performance, usage, and potential issues.

## LIMITATION

Adds another component to manage in the AI application stack.

The screenshot shows the Langfuse website homepage. At the top, there's a navigation bar with links for Docs, Pricing, Changelog, Blog, Demo, a search bar, and social media icons for GitHub, LinkedIn, and X. A 'Sign Up' button is also present. On the left, a sidebar has sections for Docs, Self Hosting, Guides, FAQ, Overview (which is highlighted), Tracing, and Tracing Features. The main content area has a heading 'Langfuse' and a paragraph describing it as an open-source LLM engineering platform. Below this are three expandable sections: 'Why Langfuse?', 'Challenges of building LLM applications', and 'Langfuse features along the development lifecycle'. A note at the bottom says 'Find an overview of all Langfuse features below. For details, refer to the individual documentation pages.'

# DSPy

## USED FOR ADVANTAGE LIMITATION

Used for optimizing and automating prompt engineering in AI applications.

Helps improve AI model performance through automated prompt optimization.

May require significant computational resources for effective prompt optimization.



## DSPy Programming—not prompting—LMs



DSPy is the framework for *programming—rather than prompting—language models*. It allows you to iterate fast on **building modular AI systems** and offers algorithms for **optimizing their prompts and weights**, whether you're building simple classifiers, sophisticated RAG pipelines, or Agent loops.

DSPy stands for Declarative Self-improving Python. Instead of brittle prompts, you write compositional *Python code* and use DSPy to **teach your LM to deliver high-quality outputs**. This [lecture](#) is a good conceptual introduction. Meet the community, seek help, or start contributing via our [GitHub repo](#) and [Discord server](#).

# PyMuPDF or PyPDF2

## USED FOR

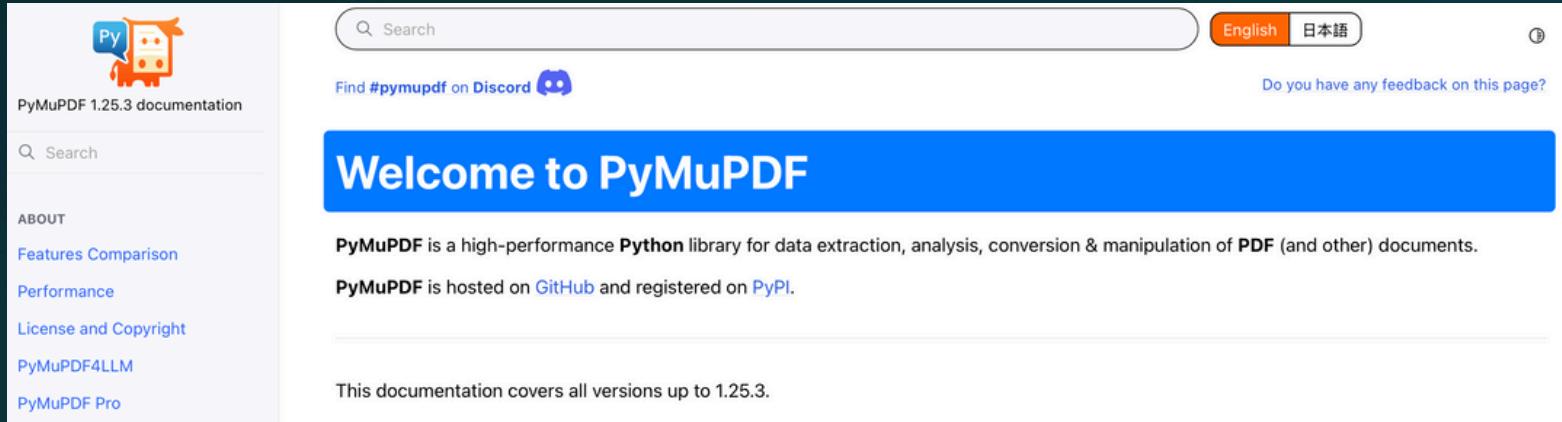
Used for PDF manipulation and data extraction in Python.

## ADVANTAGE

Enables programmatic interaction with PDF documents for various use cases.

## LIMITATION

May have limitations with complex or non-standard PDF structures.



The screenshot shows the homepage of the PyMuPDF 1.25.3 documentation. At the top left is the PyMuPDF logo (a stylized orange house with a blue 'Py' icon). To its right is the text "PyMuPDF 1.25.3 documentation". A search bar with a magnifying glass icon is positioned at the top center. To the right of the search bar are language selection buttons for "English" and "日本語" (Japanese), and a small help icon. Below the search bar is a link "Find #pymupdf on Discord" with a blue Discord icon. To the right of the search bar is a link "Do you have any feedback on this page?" with a pencil icon. A large blue header bar contains the text "Welcome to PyMuPDF". Below the header, there is a brief introduction: "PyMuPDF is a high-performance **Python** library for data extraction, analysis, conversion & manipulation of **PDF** (and other) documents." and "PyMuPDF is hosted on [GitHub](#) and registered on [PyPI](#)". At the bottom of the main content area, a note states: "This documentation covers all versions up to 1.25.3."

# AISUITE

## USED FOR

Provides a unified interface for interacting with multiple Generative AI providers using a standardized API.

## ADVANTAGE

Simplifies integration, enables easy LLM swapping, supports multiple providers, and ensures stability using HTTP endpoints or SDKs.

## LIMITATION

Currently focused on chat completions, limited coverage of advanced AI use cases, and dependent on provider-specific constraints.

### aisuite

[pypi v0.1.10](#) [code style black](#)

Simple, unified interface to multiple Generative AI providers.

`aisuite` makes it easy for developers to use multiple LLM through a standardized interface. Using an interface similar to OpenAI's, `aisuite` makes it easy to interact with the most popular LLMs and compare the results. It is a thin wrapper around python client libraries, and allows creators to seamlessly swap out and test responses from different LLM providers without changing their code. Today, the library is primarily focussed on chat completions. We will expand it cover more use cases in near future.

# GRAPHITI

## USED FOR

Simplifies graph-based reasoning and decision-making with LLMs.

## ADVANTAGE

Structured knowledge representation, enhanced reasoning, better interpretability.

## LIMITATION

Requires proper graph structuring, may have scalability challenges.

### Graphiti

#### Temporal Knowledge Graphs for Agentic Applications

 Zep 1038 members  Lint with Ruff passing  Unit Tests passing  MyPy Type Check passing

 Open in GitHub Codespaces

 Help us reach more developers and grow the Graphiti community. Star this repo!

Graphiti builds dynamic, temporally aware Knowledge Graphs that represent complex, evolving relationships between entities over time. Graphiti ingests both unstructured and structured data, and the resulting graph may be queried using a fusion of time, full-text, semantic, and graph algorithm approaches.

# Interested in more content like this?

Follow me :  
**OM NALINDE**

