```
import requests
from bs4 import BeautifulSoup
from IPython.display import Markdown, display
```

- **import requests**

Thư viện requests dùng để gửi yêu cầu HTTP, giúp tải nội dung của trang web.

- **from bs4 import BeautifulSoup**

BeautifulSoup là một thư viện dùng để phân tích cú pháp HTML/XML, hỗ trợ trích xuất dữ liệu từ trang web.

- **from IPython.display import Markdown, display**

Markdown và display được sử dụng trong Jupyter Notebook để hiển thị nội dung dưới dạng Markdown.

```
OLLAMA_API = "http://localhost:11434/api/chat"
HEADERS = {"Content-Type": "application/json"}
MODEL = "llama3.2"
```

- **OLLAMA_API = "http://localhost:11434/api/chat"**

Đây là địa chỉ API của Ollama chạy trên máy tính cục bộ (localhost).
Cổng 11434 là cổng mặc định mà Ollama sử dụng.
/api/chat là endpoint để gửi và nhận tin nhắn từ mô hình AI.

- **HEADERS = {"Content-Type": "application/json"}**

Đây là tiêu đề HTTP (Headers) khi gửi yêu cầu API.
"Content-Type": "application/json" chỉ định rằng dữ liệu gửi đi sẽ ở dạng JSON.

- **MODEL = "llama3.2"**

Xác định mô hình AI cần sử dụng, ở đây là "llama3.2".
Nếu Ollama có nhiều mô hình, bạn có thể thay đổi tên mô hình tùy theo nhu cầu.

```
messages = [ {"role": "user", "content": "Describe some of the business
applications of Generative AI"} ]
payload = { "model": MODEL, "messages": messages, "stream": False }
```

messages = [...]

- Danh sách chứa các tin nhắn gửi đến mô hình AI.
- role: "user" → Chỉ định rằng đây là tin nhắn từ người dùng.
- content: "Describe some of the business applications of Generative AI" → Nội dung câu hỏi mà người dùng gửi đến AI.

payload = {...}

- Dữ liệu gửi đi trong yêu cầu API.
- "model": MODEL → Chỉ định mô hình AI cần sử dụng (llama3.2).
- "messages": messages → Chứa danh sách tin nhắn trao đổi giữa người dùng và AI.
- "stream": False →
  Nếu True, AI sẽ trả lời theo kiểu streaming, gửi dữ liệu từng phần.
  Nếu False, AI trả về toàn bộ kết quả một lần.

```
response = requests.post(OLLAMA_API, json=payload, headers=HEADERS)
print(response.json()['message']['content'])
```

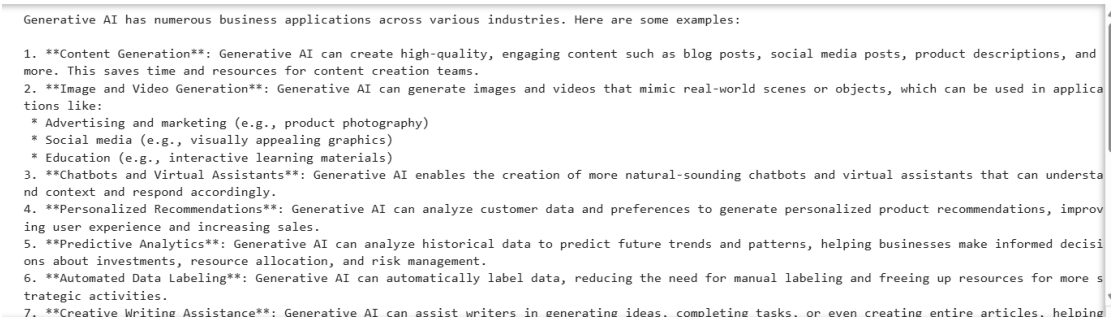
response = requests.post(OLLAMA_API, json=payload, headers=HEADERS)

- Gửi yêu cầu POST đến API của Ollama tại địa chỉ OLLAMA_API.
- Dữ liệu (payload) được gửi dưới dạng JSON.
- Tiêu đề HTTP (HEADERS) đảm bảo yêu cầu có đúng định dạng.

print(response.json()['message']['content'])

- response.json(): Chuyển đổi phản hồi từ API (ở dạng JSON) thành một đối tượng Python (dict).
- ['message']['content']: Trích xuất nội dung câu trả lời từ phản hồi của AI.
- In kết quả ra màn hình.

Kết quả:




```
import ollama

response = ollama.chat(model=MODEL, messages=messages)
print(response['message']['content'])
```

import ollama

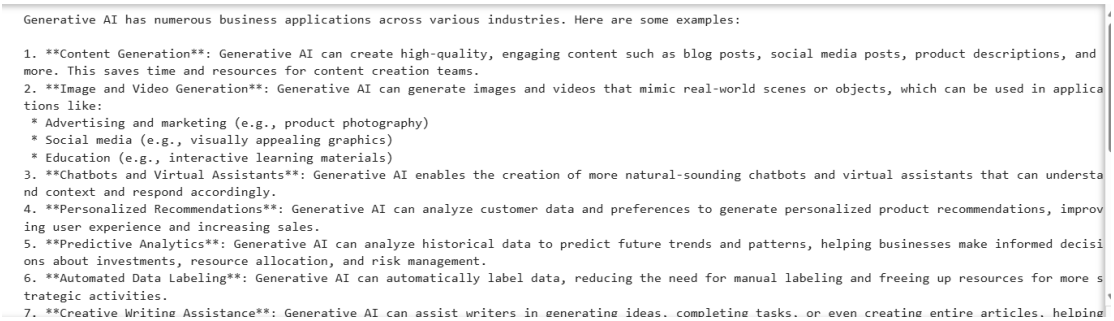
- Import thư viện Ollama, cho phép tương tác trực tiếp với mô hình AI cục bộ mà không cần dùng requests.

response = ollama.chat(model=MODEL, messages=messages)

- Gửi một tin nhắn đến mô hình AI cục bộ.
- model=MODEL → Xác định mô hình AI đang sử dụng (llama3.2 hoặc một mô hình khác).
- messages=messages → Danh sách tin nhắn trao đổi giữa người dùng và AI.

print(response['message']['content'])

- response chứa phản hồi từ AI dưới dạng dictionary.
- response['message']['content'] → Trích xuất nội dung từ phản hồi của AI và in ra màn hình.



```
from openai import OpenAI
ollama_via_openai = OpenAI(base_url='http://localhost:11434/v1',
api_key='ollama')

response = ollama_via_openai.chat.completions.create(
    model=MODEL,
    messages=messages
)

print(response.choices[0].message.content)
```

from openai import OpenAI

- Import thư viện OpenAI, có thể dùng để kết nối với Ollama hoặc OpenAI API (như GPT-4).

ollama_via_openai = OpenAI(base_url='http://localhost:11434/v1', api_key='ollama')

- Tạo một client OpenAI để kết nối với Ollama API cục bộ.
- base_url='http://localhost:11434/v1' → Đây là API của Ollama, được thiết kế để tương thích với OpenAI API.
- api_key='ollama' → Ollama không yêu cầu API Key thực sự, nhưng cần đặt một giá trị tùy ý (ở đây là "ollama").

response = ollama_via_openai.chat.completions.create(...)

- Gửi yêu cầu chat đến Ollama thông qua OpenAI API wrapper.
- model=MODEL → Xác định mô hình AI cần sử dụng (llama3.2, mistral, gemma, v.v.).
- messages=messages → Danh sách tin nhắn trao đổi giữa người dùng và AI.

print(response.choices[0].message.content)

- API trả về danh sách các lựa chọn (choices), mỗi lựa chọn là một phản hồi từ AI.
- choices[0].message.content → Trích xuất nội dung tin nhắn từ phản hồi đầu tiên của AI.

So sánh 3 đoạn code gửi tin nhắn đến mô hình AI chạy trên Ollama và nhận phản hồi

Cách gọi API	Cách giao tiếp	Độ phức tạp	Hiệu suất	Độ linh hoạt
requests	Gửi request HTTP	Trung bình	Trung bình	Rất cao
ollama	Gọi trực tiếp thư viện Ollama	Dễ	Nhanh	Thấp
openai	Gọi API theo chuẩn OpenAI	Dễ	Nhanh	Cao