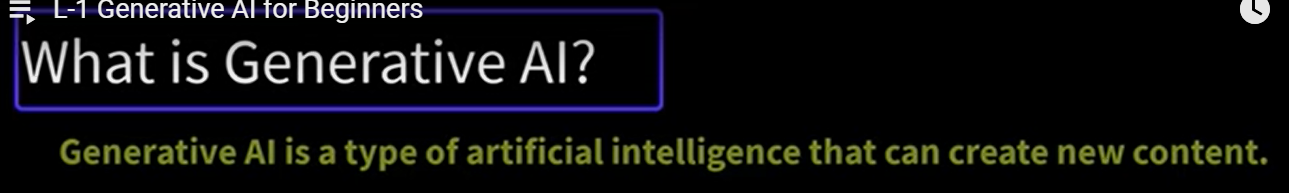
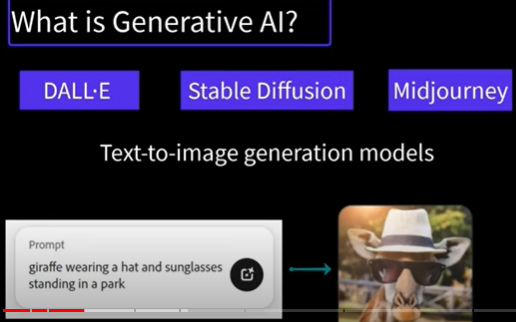
<https://www.youtube.com/watch?v=dI8EQU0PFzc&list=PLv8Cp2NvcY8AzNCATbDWMr8vqbJBYbxFW>

GEN AI



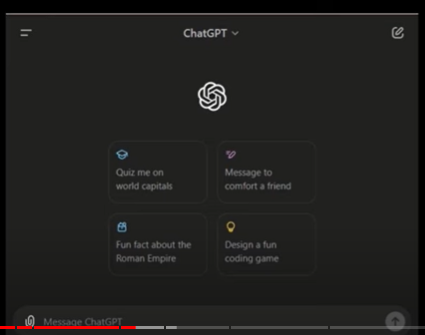


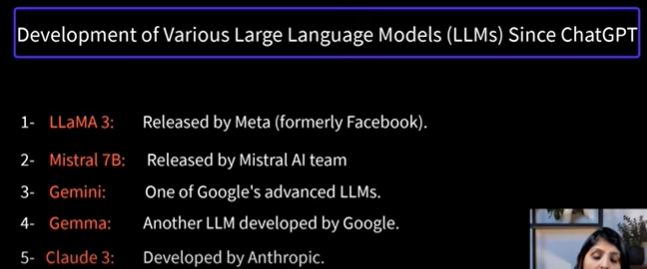


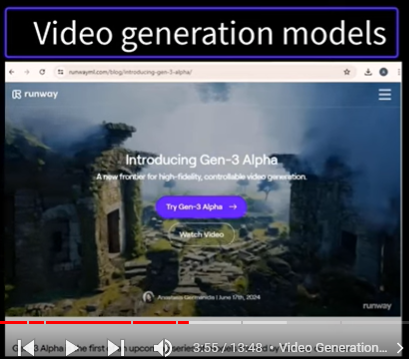


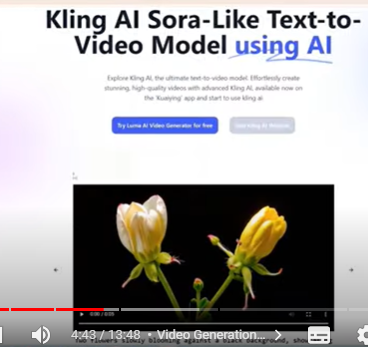
Chatgpt is open Gen AI is created by LLM

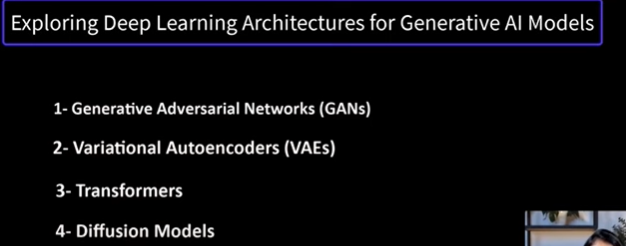
LLM stands for Large Language Model, which is a type of artificial intelligence (AI) model that uses machine learning to understand and generate human language. LLMs are trained on massive amounts of text and other content

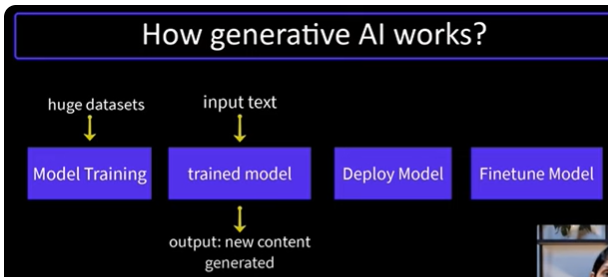


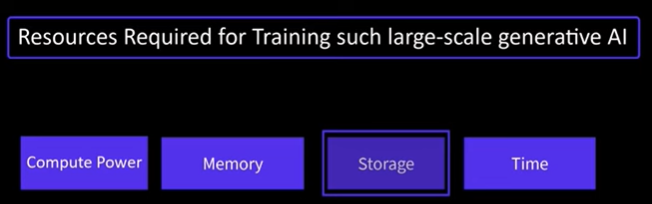


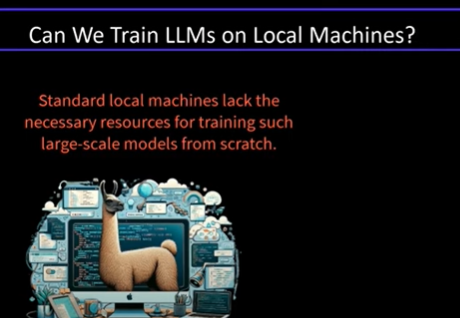


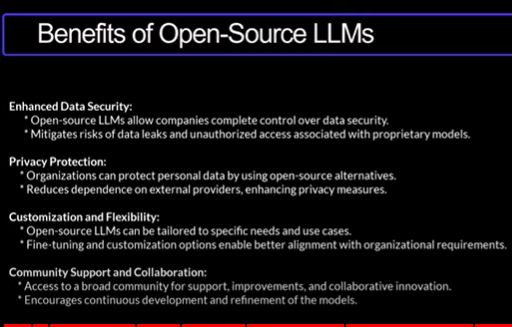


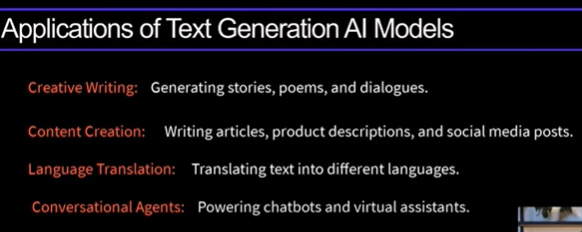










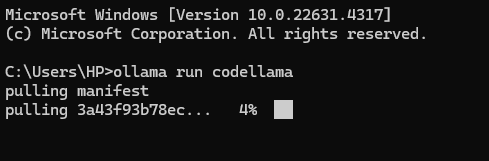


Chapter 2

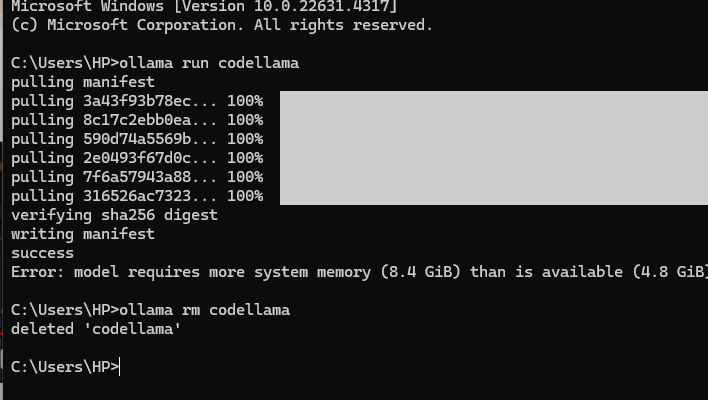
Ollama.com open this website

And download ollama, then download complited and then install It then after install

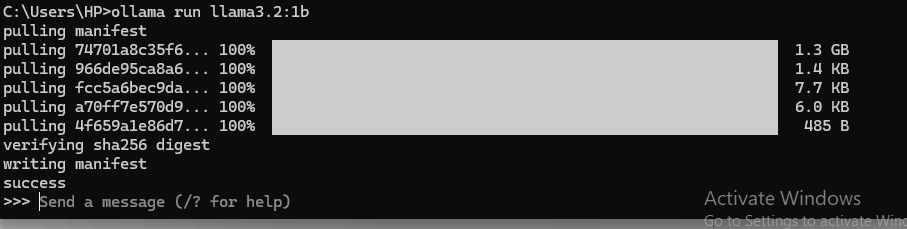
Open cmd and type

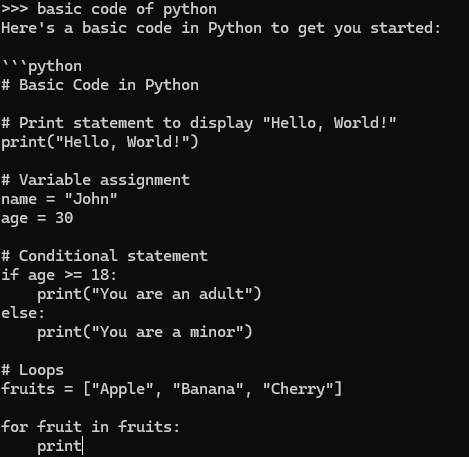


Above take 100 %



Then Use it





Check the ollama file

<https://github.com/ollama/ollama>

| **Model** | **Parameters** | **Size** | **Download** |
| --- | --- | --- | --- |
| Llama 3.2 | 3B | 2.0GB | ollama run llama3.2 |
| Llama 3.2 | 1B | 1.3GB | ollama run llama3.2:1b |
| Llama 3.2 Vision | 11B | 7.9GB | ollama run llama3.2-vision |
| Llama 3.2 Vision | 90B | 55GB | ollama run llama3.2-vision:90b |
| Llama 3.1 | 8B | 4.7GB | ollama run llama3.1 |
| Llama 3.1 | 70B | 40GB | ollama run llama3.1:70b |
| Llama 3.1 | 405B | 231GB | ollama run llama3.1:405b |
| Phi 3 Mini | 3.8B | 2.3GB | ollama run phi3 |
| Phi 3 Medium | 14B | 7.9GB | ollama run phi3:medium |
| Gemma 2 | 2B | 1.6GB | ollama run gemma2:2b |
| Gemma 2 | 9B | 5.5GB | ollama run gemma2 |
| Gemma 2 | 27B | 16GB | ollama run gemma2:27b |
| Mistral | 7B | 4.1GB | ollama run mistral |
| Moondream 2 | 1.4B | 829MB | ollama run moondream |
| Neural Chat | 7B | 4.1GB | ollama run neural-chat |
| Starling | 7B | 4.1GB | ollama run starling-lm |
| Code Llama | 7B | 3.8GB | ollama run codellama |
| Llama 2 Uncensored | 7B | 3.8GB | ollama run llama2-uncensored |
| LLaVA | 7B | 4.5GB | ollama run llava |
| Solar | 10.7B | 6.1GB | ollama run solar |

**Chapter 3**

Open

[www.langchain.com](http://www.langchain.com)

LangChain is an open-source framework that allows developers to build applications using large language models (LLMs):

* **What it does**

LangChain provides tools and abstractions to help developers create applications that use LLMs to perform tasks like answering questions, creating images, and more.

* **How it works**

LangChain offers a variety of features, including:

* + A centralized development environment
  + A module-based approach
  + The ability to compare different prompts and foundation models
  + The ability to use multiple LLMs in a program
  + A standard interface for agents, along with LangGraph for building custom agents
  + A prompts library to help parameterize common prompt text
  + Indexes that serve as databases, organizing and storing information in a structured manner
* **What it's used for**

LangChain can be used to build applications like:

* + Chatbots
  + Virtual agents
  + Intelligent search
  + Question-answering
  + Summarization services

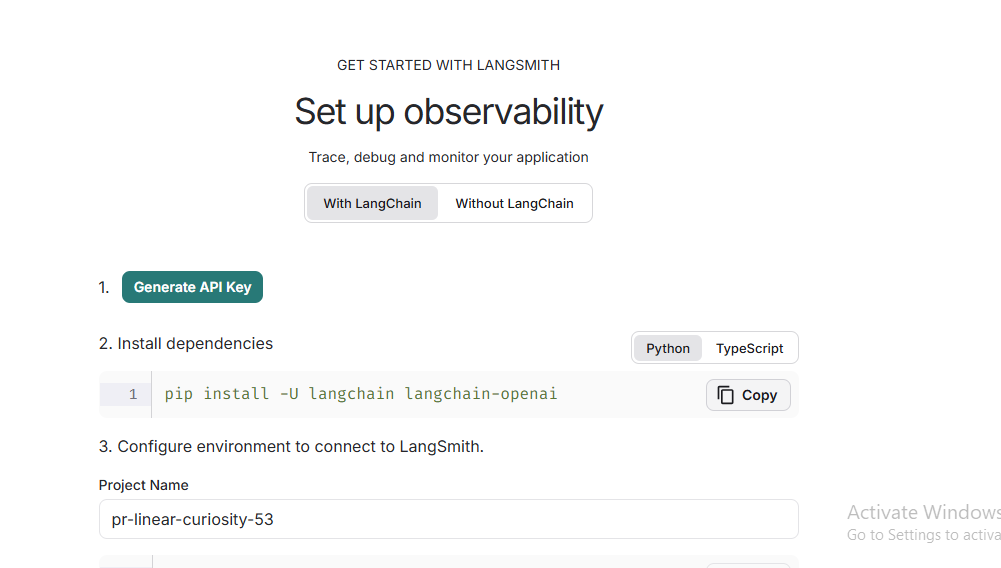
LangChain is available in both Python- and Javascript-based libraries

Step1:

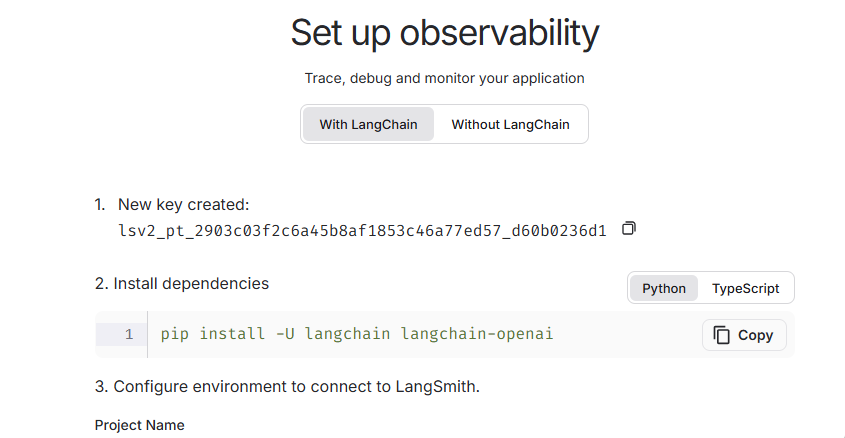
Open the langchain website then click on signup

Login with gmail

Get started



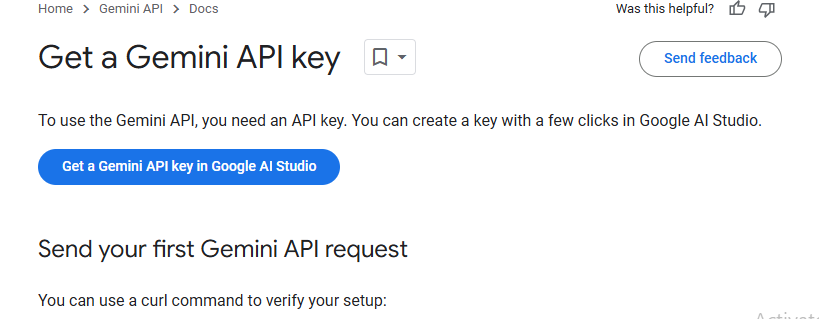
Click on generate API Key



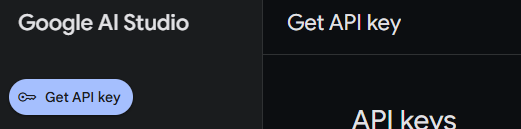
Step 2

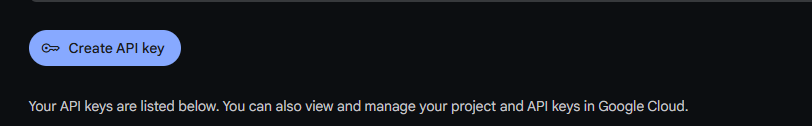
Get a Google API key:

    Head to https://ai.google.dev/gemini-api/docs/api-key



Click on get a Gemini APLI key in google Ai Studio



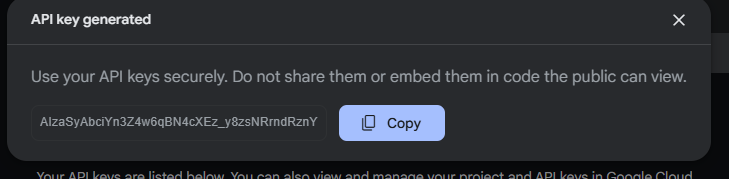


Then

Tick 1

Tick 2

Accept



Step3 **### Environment setup:**

Step 3

D:\datascience\GEN AI WITH Aarhi\First\_genai>**py -3.10 -m venv myvenv**

D:\datascience\GEN AI WITH Aarhi\First\_genai>**myvenv\Scripts\activate**

(myvenv) D:\datascience\GEN AI WITH Aarhi\First\_genai>**python -m pip install --upgrade pip**

Requirement already satisfied: pip in d:\datascience\gen ai with aarhi\first\_genai\myvenv\lib\site-packages (22.3.1)

Collecting pip

Downloading pip-24.3.1-py3-none-any.whl (1.8 MB)

━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.8/1.8 MB 8.9 MB/s eta 0:00:00

Installing collected packages: pip

Attempting uninstall: pip

Found existing installation: pip 22.3.1

Uninstalling pip-22.3.1:

Successfully uninstalled pip-22.3.1

Successfully installed pip-24.3.1

(myvenv) D:\datascience\GEN AI WITH Aarhi\First\_genai>**pip install --upgrade --quiet langchain-google-genai pillow**

(myvenv) D:\datascience\GEN AI WITH Aarhi\First\_genai>

(myvenv) D:\datascience\GEN AI WITH Aarhi\First\_genai>**pip install streamlit**

Collecting streamlit

Downloading streamlit-1.40.1-py2.py3-none-any.whl.metadata (8.5 kB)

Collecting altair<6,>=4.0 (from streamlit)

Downloading altair-5.4.1-py3-none-any.whl.metadata (9.4 kB)

Collecting blinker<2,>=1.0.0 (from streamlit)

Using cached blinker-1.9.0-py3-none-any.whl.metadata (1.6 kB)

Requirement already satisfied: cachetools<6,>=4.0 in d:\datascience\gen ai with aarhi\first\_genai\myvenv\lib\site-packages (from streamlit) (5.5.0)

Collecting click<9,>=7.0 (from streamlit)

Using cached click-8.1.7-py3-none-any.whl.metadata (3.0 kB)

Collecting numpy<3,>=1.20 (from streamlit)

Using cached numpy-2.1.3-cp310-cp310-win\_amd64.whl.metadata (60 kB)

Requirement already satisfied: packaging<25,>=20 in d:\datascience\gen ai with aarhi\first\_genai\myvenv\lib\site-packages (from streamlit) (24.2)

Collecting pandas<3,>=1.4.0 (from streamlit)

Using cached pandas-2.2.3-cp310-cp310-win\_amd64.whl.metadata (19 kB)

Requirement already satisfied: pillow<12,>=7.1.0 in d:\datascience\gen ai with aarhi\first\_genai\myvenv\lib\site-packages (from streamlit) (11.0.0)

Requirement already satisfied: protobuf<6,>=3.20 in d:\datascience\gen ai with aarhi\first\_genai\myvenv\lib\site-packages (from streamlit) (5.28.3)

Collecting pyarrow>=7.0 (from streamlit)

Downloading pyarrow-18.0.0-cp310-cp310-win\_amd64.whl.metadata (3.4 kB)

Requirement already satisfied: requests<3,>=2.27 in d:\datascience\gen ai with aarhi\first\_genai\myvenv\lib\site-packages (from streamlit) (2.32.3)

Collecting rich<14,>=10.14.0 (from streamlit)

Downloading rich-13.9.4-py3-none-any.whl.metadata (18 kB)

Requirement already satisfied: tenacity<10,>=8.1.0 in d:\datascience\gen ai with aarhi\first\_genai\myvenv\lib\site-packages (from streamlit) (9.0.0)

Collecting toml<2,>=0.10.1 (from streamlit)

Downloading toml-0.10.2-py2.py3-none-any.whl.metadata (7.1 kB)

Requirement already satisfied: typing-extensions<5,>=4.3.0 in d:\datascience\gen ai with aarhi\first\_genai\myvenv\lib\site-packages (from streamlit) (4.12.2)

Collecting gitpython!=3.1.19,<4,>=3.0.7 (from streamlit)

Downloading GitPython-3.1.43-py3-none-any.whl.metadata (13 kB)

Collecting pydeck<1,>=0.8.0b4 (from streamlit)

Downloading pydeck-0.9.1-py2.py3-none-any.whl.metadata (4.1 kB)

Collecting tornado<7,>=6.0.3 (from streamlit)

Downloading tornado-6.4.1-cp38-abi3-win\_amd64.whl.metadata (2.6 kB)

Collecting watchdog<7,>=2.1.5 (from streamlit)

Downloading watchdog-6.0.0-py3-none-win\_amd64.whl.metadata (44 kB)

Collecting jinja2 (from altair<6,>=4.0->streamlit)

Using cached jinja2-3.1.4-py3-none-any.whl.metadata (2.6 kB)

Collecting jsonschema>=3.0 (from altair<6,>=4.0->streamlit)

Downloading jsonschema-4.23.0-py3-none-any.whl.metadata (7.9 kB)

Collecting narwhals>=1.5.2 (from altair<6,>=4.0->streamlit)

Downloading narwhals-1.14.1-py3-none-any.whl.metadata (7.4 kB)

Requirement already satisfied: colorama in d:\datascience\gen ai with aarhi\first\_genai\myvenv\lib\site-packages (from click<9,>=7.0->streamlit) (0.4.6)

Collecting gitdb<5,>=4.0.1 (from gitpython!=3.1.19,<4,>=3.0.7->streamlit)

Downloading gitdb-4.0.11-py3-none-any.whl.metadata (1.2 kB)

Collecting python-dateutil>=2.8.2 (from pandas<3,>=1.4.0->streamlit)

Using cached python\_dateutil-2.9.0.post0-py2.py3-none-any.whl.metadata (8.4 kB)

Collecting pytz>=2020.1 (from pandas<3,>=1.4.0->streamlit)

Using cached pytz-2024.2-py2.py3-none-any.whl.metadata (22 kB)

Collecting tzdata>=2022.7 (from pandas<3,>=1.4.0->streamlit)

Using cached tzdata-2024.2-py2.py3-none-any.whl.metadata (1.4 kB)

Requirement already satisfied: charset-normalizer<4,>=2 in d:\datascience\gen ai with aarhi\first\_genai\myvenv\lib\site-packages (from requests<3,>=2.27->streamlit) (3.4.0)

Requirement already satisfied: idna<4,>=2.5 in d:\datascience\gen ai with aarhi\first\_genai\myvenv\lib\site-packages (from requests<3,>=2.27->streamlit) (3.10)

Requirement already satisfied: urllib3<3,>=1.21.1 in d:\datascience\gen ai with aarhi\first\_genai\myvenv\lib\site-packages (from requests<3,>=2.27->streamlit) (2.2.3)

Requirement already satisfied: certifi>=2017.4.17 in d:\datascience\gen ai with aarhi\first\_genai\myvenv\lib\site-packages (from requests<3,>=2.27->streamlit) (2024.8.30)

Collecting markdown-it-py>=2.2.0 (from rich<14,>=10.14.0->streamlit)

Downloading markdown\_it\_py-3.0.0-py3-none-any.whl.metadata (6.9 kB)

Collecting pygments<3.0.0,>=2.13.0 (from rich<14,>=10.14.0->streamlit)

Downloading pygments-2.18.0-py3-none-any.whl.metadata (2.5 kB)

Collecting smmap<6,>=3.0.1 (from gitdb<5,>=4.0.1->gitpython!=3.1.19,<4,>=3.0.7->streamlit)

Downloading smmap-5.0.1-py3-none-any.whl.metadata (4.3 kB)

Collecting MarkupSafe>=2.0 (from jinja2->altair<6,>=4.0->streamlit)

Using cached MarkupSafe-3.0.2-cp310-cp310-win\_amd64.whl.metadata (4.1 kB)

Collecting attrs>=22.2.0 (from jsonschema>=3.0->altair<6,>=4.0->streamlit)

Downloading attrs-24.2.0-py3-none-any.whl.metadata (11 kB)

Collecting jsonschema-specifications>=2023.03.6 (from jsonschema>=3.0->altair<6,>=4.0->streamlit)

Downloading jsonschema\_specifications-2024.10.1-py3-none-any.whl.metadata (3.0 kB)

Collecting referencing>=0.28.4 (from jsonschema>=3.0->altair<6,>=4.0->streamlit)

Downloading referencing-0.35.1-py3-none-any.whl.metadata (2.8 kB)

Collecting rpds-py>=0.7.1 (from jsonschema>=3.0->altair<6,>=4.0->streamlit)

Downloading rpds\_py-0.21.0-cp310-none-win\_amd64.whl.metadata (4.2 kB)

Collecting mdurl~=0.1 (from markdown-it-py>=2.2.0->rich<14,>=10.14.0->streamlit)

Downloading mdurl-0.1.2-py3-none-any.whl.metadata (1.6 kB)

Collecting six>=1.5 (from python-dateutil>=2.8.2->pandas<3,>=1.4.0->streamlit)

Using cached six-1.16.0-py2.py3-none-any.whl.metadata (1.8 kB)

Downloading streamlit-1.40.1-py2.py3-none-any.whl (8.6 MB)

━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 8.6/8.6 MB 10.5 MB/s eta 0:00:00

Downloading altair-5.4.1-py3-none-any.whl (658 kB)

━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 658.1/658.1 kB 12.4 MB/s eta 0:00:00

Using cached blinker-1.9.0-py3-none-any.whl (8.5 kB)

Using cached click-8.1.7-py3-none-any.whl (97 kB)

Downloading GitPython-3.1.43-py3-none-any.whl (207 kB)

Using cached numpy-2.1.3-cp310-cp310-win\_amd64.whl (12.9 MB)

Using cached pandas-2.2.3-cp310-cp310-win\_amd64.whl (11.6 MB)

Downloading pyarrow-18.0.0-cp310-cp310-win\_amd64.whl (25.1 MB)

━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 25.1/25.1 MB 10.7 MB/s eta 0:00:00

Downloading pydeck-0.9.1-py2.py3-none-any.whl (6.9 MB)

━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 6.9/6.9 MB 12.9 MB/s eta 0:00:00

Downloading rich-13.9.4-py3-none-any.whl (242 kB)

Downloading toml-0.10.2-py2.py3-none-any.whl (16 kB)

Downloading tornado-6.4.1-cp38-abi3-win\_amd64.whl (438 kB)

Downloading watchdog-6.0.0-py3-none-win\_amd64.whl (79 kB)

Downloading gitdb-4.0.11-py3-none-any.whl (62 kB)

Using cached jinja2-3.1.4-py3-none-any.whl (133 kB)

Downloading jsonschema-4.23.0-py3-none-any.whl (88 kB)

Downloading markdown\_it\_py-3.0.0-py3-none-any.whl (87 kB)

Downloading narwhals-1.14.1-py3-none-any.whl (220 kB)

Downloading pygments-2.18.0-py3-none-any.whl (1.2 MB)

━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.2/1.2 MB 10.0 MB/s eta 0:00:00

Using cached python\_dateutil-2.9.0.post0-py2.py3-none-any.whl (229 kB)

Using cached pytz-2024.2-py2.py3-none-any.whl (508 kB)

Using cached tzdata-2024.2-py2.py3-none-any.whl (346 kB)

Downloading attrs-24.2.0-py3-none-any.whl (63 kB)

Downloading jsonschema\_specifications-2024.10.1-py3-none-any.whl (18 kB)

Using cached MarkupSafe-3.0.2-cp310-cp310-win\_amd64.whl (15 kB)

Downloading mdurl-0.1.2-py3-none-any.whl (10.0 kB)

Downloading referencing-0.35.1-py3-none-any.whl (26 kB)

Downloading rpds\_py-0.21.0-cp310-none-win\_amd64.whl (218 kB)

Using cached six-1.16.0-py2.py3-none-any.whl (11 kB)

Downloading smmap-5.0.1-py3-none-any.whl (24 kB)

Installing collected packages: pytz, watchdog, tzdata, tornado, toml, smmap, six, rpds-py, pygments, pyarrow, numpy, narwhals, mdurl, MarkupSafe, click, blinker, attrs, referencing, python-dateutil, markdown-it-py, jinja2, gitdb, rich, pydeck, pandas, jsonschema-specifications, gitpython, jsonschema, altair, streamlit

Successfully installed MarkupSafe-3.0.2 altair-5.4.1 attrs-24.2.0 blinker-1.9.0 click-8.1.7 gitdb-4.0.11 gitpython-3.1.43 jinja2-3.1.4 jsonschema-4.23.0 jsonschema-specifications-2024.10.1 markdown-it-py-3.0.0 mdurl-0.1.2 narwhals-1.14.1 numpy-2.1.3 pandas-2.2.3 pyarrow-18.0.0 pydeck-0.9.1 pygments-2.18.0 python-dateutil-2.9.0.post0 pytz-2024.2 referencing-0.35.1 rich-13.9.4 rpds-py-0.21.0 six-1.16.0 smmap-5.0.1 streamlit-1.40.1 toml-0.10.2 tornado-6.4.1 tzdata-2024.2 watchdog-6.0.0

(myvenv) D:\datascience\GEN AI WITH Aarhi\First\_genai>**pip install python-dotenv**

Collecting python-dotenv

Using cached python\_dotenv-1.0.1-py3-none-any.whl.metadata (23 kB)

Using cached python\_dotenv-1.0.1-py3-none-any.whl (19 kB)

Installing collected packages: python-dotenv

Successfully installed python-dotenv-1.0.1

(myvenv) D:\datascience\GEN AI WITH Aarhi\First\_genai>

**Then Code**

**gemini\_app\_qa.py**

import streamlit as st

from dotenv import load\_dotenv

from langchain\_core.prompts import ChatPromptTemplate

from langchain\_google\_genai import ChatGoogleGenerativeAI

from langchain\_core.output\_parsers import StrOutputParser

load\_dotenv()

# Now we can instantiate our model object and generate chat completions:

llm = ChatGoogleGenerativeAI(

    model="gemini-1.5-pro",

    temperature=0,

    max\_tokens=None,

    timeout=None,

    max\_retries=2,

)

prompt=ChatPromptTemplate.from\_messages(

    [

        ("system","You are a chatbot"),

        ("human","Question:{question}")

    ]

)

st.title('Langchain Demo With Gemini')

input\_text=st.text\_input("Enter your question here")

output\_parser=StrOutputParser()

chain=prompt|llm|output\_parser

if input\_text:

    st.write(chain.invoke({'question':input\_text}))

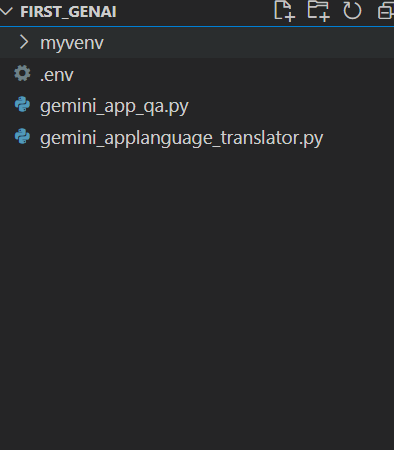
# To run this code, write-  streamlit run gemini\_app\_qa.py

**.env code**

GOOGLE\_API\_KEY='AIzaSyAbciYn3Z4w6qBN4cXEz\_y8zsNRrndRznY'

LANGCHAIN\_API\_KEY="lsv2\_pt\_2903c03f2c6a45b8af1853c46a77ed57\_d60b0236d1"

LANGCHAIN\_PROJECT="geminiChatbottutorial"

****

**gemini\_applanguage\_translator.py**

'''

https://python.langchain.com/v0.2/docs/integrations/chat/google\_generative\_ai/

'''

import getpass

import os

import streamlit as st

from dotenv import load\_dotenv

from langchain\_core.output\_parsers import StrOutputParser

load\_dotenv()

# Now we can instantiate our model object and generate chat completions:

from langchain\_google\_genai import ChatGoogleGenerativeAI

llm = ChatGoogleGenerativeAI(

    model="gemini-1.5-pro",

    temperature=0,

    max\_tokens=None,

    timeout=None,

    max\_retries=2,

    # other params...

)

# We can chain our model with a prompt template like so:

from langchain\_core.prompts import ChatPromptTemplate

prompt = ChatPromptTemplate.from\_messages(

    [

        (

            "system",

            "You are a helpful assistant that translates {input\_language} to {output\_language}.",

        ),

        ("human", "{input}"),

    ]

)

st.title('Langchain Demo With Gemini (language translator)')

input\_text=st.text\_input("Write the sentence in english and it will be translated in german")

# chain = prompt | llm

output\_parser=StrOutputParser()

chain=prompt|llm|output\_parser

if input\_text:

    st.write(chain.invoke(

    {

        "input\_language": "English",

        "output\_language": "German",

        "input": input\_text,

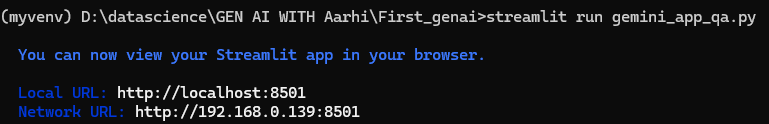
        #"input": "I love programming.",

    }

))

# Run app- streamlit run gemini\_applanguage\_translator.py

**To Run Program**

****