

# Langchain Tutorial for the LOST

LLM নিয়ে যারা কাজ করছেন, তারা নিশ্চয়ই ল্যাংচেইন এর নাম শুনেছেন। তবে এত বিচিত্র সব LLM, প্রায় সবাই ক্লেইম করে যাচ্ছেন, তাদের LLMই নাকি সবচেয়ে ভাল ! এতকিছুর মাঝে আবার ডেভেলপারদের জন্য অনেক কোম্পানি অনেক কিছু নিয়ে আসছেন, সেগুলি শিখতেও আমাদের দিগবিদিশূণ্য অবস্থা (অন্তত আমার মত যারা আমজনতা) ।

এরকম যাদের অবস্থা, Langchain এর নাম শুনেছে, কিন্তু এক্সপ্লোর করে দেখা হয়নি, জিনিসটা আসলে কি, তাদের জন্যই এই ছোট্ট প্রয়াস।

বিশেষত, বাংলা ভাষায় কন্টেন্ট তৈরি করতে চাওয়ার মত দুঃসাহস হয়েছে, অধমের ভুল্ভ্রান্তি ক্ষমাসুন্দর (সাথে ফিডব্যাকও থাকুক) চোখে দেখবেন :)

## Talk of the town: But what it is?

খুব সহজে বলতে চাইলে, **Langchain** একটা ফ্রেমওয়ার্ক, যেটা LLM কে আপনার এপ্লিকেশান এর সাথে যুক্ত করতে সাহায্য করবে। LLM গুলির API এখন মার্কেটে বেশ এভেইলেবল, Langchain সেগুলি ব্যবহার করে আপনাকে বেশ সুবিধাজনক একটা ইন্টারফেইস দেবে, যাতে আপনি নিজের কাজে, গবেষণায় এমনকি ডেভেলপমেন্টের কাজে LLM গুলিকে ব্যবহার করতে পারেন

## Do I need Langchain?

আপনি যদি শুধু ইউজার হিসেবে LLM ব্যবহার করতে চান, মাঝেমধ্যে দুয়েকটা প্রশ্ন করতে চান, তাহলে আপনার Langchain এর কোন প্রয়োজন নেই। সরাসরি ওয়েবসাইট ব্যবহার করলেই সই।

কিন্তু আপনি যদি:

১। গবেষক, যাকে প্রচুর পরিমাণে LLM এর API গুলি ব্যবহার করতে হয়, মূলত জেনারেশানের কাজে, এবং আপনি চান সেগুলি কিছুটা স্ট্রাকচারড উপায়ে ব্যবহার করতে

২। ডেভেলপার, যিনি চান তার তৈরি এপ্লিকেশানগুলি লেটেস্ট ফিচারে সমৃদ্ধ থাকুক, এপ ব্যবহারকারিরা খুব সহজেই AI-integrated feature পেয়ে যাক

তবে, অবশ্যই ! Langchain আপনার জন্য ভীষণ দরকারি হতে যাচ্ছে।

## Let's install langchain first

```
In [1]: ▶ # নিচের ইন্টেলেশান কমান্ড টি আনকমেন্ট করে দিন:  
#!pip install langchain
```

```
In [2]: ➤ import warnings
warnings.filterwarnings('ignore')
```

```
In [3]: ➤ import langchain
```

```
In [4]: ➤ langchain.__version__
```

```
Out[4]: '0.1.0'
```

NB: ভার্সন এর ব্যাপারটা খুব গুরুত্বপূর্ণ। ল্যাংচেইন এর অনেক ডেভেলপমেন্ট হচ্ছে, নতুন নতুন ফিচার আসছে। নতুন ভার্সনে অনেক কিছাই হয়ত থাকবে না !

## Prompting

যারা টুকুটাকি LLM ব্যবহার করেছেন, তারা জানেন, Prompting এর ব্যাপারটা কতটা পাওয়ারফুল। ল্যাংচেইন এই কাজটাই বেশ স্মার্টলি করতে শেখায়

আমরা openai gpt-3.5 turbo ব্যবহার করব LLM হিসেবে। দুঃসংবাদ হল, আপনাকে openai key পেতে হবে এর জন্য। key পেতে কোন সমস্যা হলে আমাকে জানাতে পারেন।

```
In [4]: ➤ import os
os.environ["OPENAI_API_KEY"] = "*****"
```

এস্টেরিস্ক এর জায়গায় আপনার api key দিয়ে দিন !

```
In [6]: ➤ from langchain.llms import OpenAI
```

```
In [7]: ➤ llm = OpenAI(temperature=0.6, model="gpt-3.5-turbo-instruct")
```

```
/home/sadat/.local/lib/python3.8/site-packages/langchain_core/_api/deprecation.py:117: LangChainDeprecationWarning: The class `langchain_community.llms.openai.OpenAI` was deprecated in langchain-community 0.0.10 and will be removed in 0.2.0. An updated version of the class exists in the langchain-openai package and should be used instead. To use it run `pip install -U langchain-openai` and import as `from langchain_openai import OpenAI`.
warn_deprecated(
```

একটা খুব সিম্পল প্রম্পট দিয়ে শুরু করা যাক।

```
In [8]: ▶ name = llm("I have expertise on t-shirt painting. Suggest a cool startup idea")
print(name)
```

```
/home/sadat/.local/lib/python3.8/site-packages/langchain_core/_api/deprecation.py:117: LangChainDeprecationWarning: The function `__call__` was deprecated in LangChain 0.1.7 and will be removed in 0.2.0. Use invoke instead.
warn_deprecated(
```

A personalized t-shirt subscription service that allows customers to choose from a variety of unique designs and have them hand-painted onto high-quality t-shirts by local artists. Customers can also submit their own designs to be painted, making each t-shirt truly one-of-a-kind. This startup would offer a fun and creative way for people to express themselves through fashion while supporting local artists.

আপনি চাইলে এই মজার খেলাটা খেলতে পারেন, অর্থাৎ expertise হিসেবে নানান জিনিস যেমন, ইলেক্ট্রিক্যাল ইঞ্জিনিয়ার, ফাইন্যান্স, বাংলা ভাষা, মধ্যযুগীয় সাহিত্য, তন্দুরি রান্না, ইত্যাদি। কিন্তু বারবার পুরো জিনিস লেখার চাইতে, আরেকটু ইফিসিয়েন্ট উপায় দরকার, তাই না?

```
In [9]: ▶ from langchain.prompts import PromptTemplate
```

```
In [10]: ▶ prompt_template_skill = PromptTemplate(
    input_variables = ["skill"],
    template = "I have expertise on {skill}. Suggest a cool startup idea"
)
```

```
In [11]: ▶ # That's what the prompt looks like
prompt_template_skill.format(skill="data science and machine learning")
```

```
Out[11]: 'I have expertise on data science and machine learning. Suggest a cool startup idea.'
```

```
In [12]: ▶ # A glimpse of chain ;)
from langchain.chains import LLMChain
```

```
In [13]: ▶ prompt_template_skill_chain = LLMChain(llm=llm, prompt=prompt_template)
prompt_template_skill_chain.run("data science and machine learning")
```

```
/home/sadat/.local/lib/python3.8/site-packages/langchain_core/_api/deprecation.py:117: LangChainDeprecationWarning: The function `run` was deprecated in LangChain 0.1.0 and will be removed in 0.2.0. Use invoke instead.
warn_deprecated(
```

```
Out[13]: "\n\nA personalized virtual wardrobe assistant that uses machine learning to suggest outfit combinations based on the user's style preferences, weather, and occasion. The assistant would also keep track of the user's wardrobe and suggest items to purchase to fill any gaps in their wardrobe. It could also offer styling tips and inspiration from fashion influencers. This idea combines both data science and machine learning to provide a unique and personalized fashion experience for users."
```

```
In [14]: ▶ # Let's try another fun one
prompt_template_skill_chain.run("sleeping and eating")
```

```
Out[14]: "\n\nA sleep and nutrition tracking app that uses artificial intelligence to create personalized sleep and meal plans for individuals. The app would track sleep patterns and eating habits, and provide recommendations for improving both based on the user's goals and preferences. It could also offer features such as guided meditations for better sleep and healthy recipe suggestions. The app could partner with fitness trackers and meal delivery services to offer a comprehensive wellness experience."
```

আপনি চাইলে এমন বেশ কিছু ইন্ট্রিস্টিং প্রম্পট দিয়ে চেক করতে পারেন। জিপটি মাঝেমাঝে বেশ অদ্ভুত কিছু উত্তর দেয়। তেমন কিছু পেলে শেয়ার করতে ভুলবেন না! :D

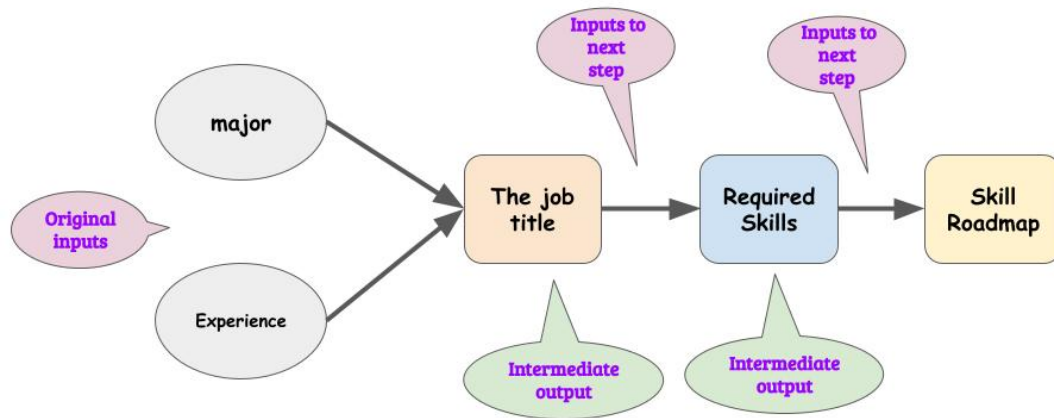
## Chains

ল্যাংচেইনের এই ফিচারটির জন্য এর ডেভেলপারদের দাঁড়িয়ে স্যালুট দিতে হয় ! চেইনের কনসেপ্টটা অনেকটা এমনঃ আমরা জানি, LLM স্টেপ বাই স্টেপ কোনকিছু বেশ পছন্দ করে। একবারে একগাদা আদেশ না দিয়ে একটার পর একটা কাজ করতে বললে অনেকক্ষেত্রেই বেশি উপকার মেলে। চেইন সেই কাজটাই স্ট্রাকচারাল উপায়ে করতে সাহায্য করে।

```
In [15]: ▶ from langchain.chains import SequentialChain
```

ধরা যাক, কোন এক পড়ন্ত বিকেলে আপনার মনে হল, নাহ, এভাবে আর চলে না। আমার যোগ্যতার পূর্ণ মূল্যায়ন হচ্ছে না! এবার আপনি চান, মনের মত কিছু করতে। কিন্তু কি করা যায়? অন্যসময়ে বন্ধুদের ফোন দেয়া যেত, কিন্তু আজকাল সবাই বড় ব্যস্ত। আপনি এবার দারস্থ হতে চান কৃত্রিম বুদ্ধিমত্তার। LLM ব্যবহার করে ক্যারিয়ার অপশান নিয়ে ব্রেইনস্টোর্ম করতে চান। ধরা যাক, এমন একটা মডেল চান, যেটা আপনার পড়াশুনার মেজর, আর অভিজ্ঞতা ইনপুট হিসেবে নিয়ে একটা ভাল জবের সাজেশান দেবে। আপনি চান, সেই জবের জন্য প্রয়োজনীয় স্কিলগুলি কী কী, সেটাও বলে দেবে LLM। সেই সাথে, স্কিল অর্জনের জন্য রোডম্যাপটা কেমন হতে হবে, সেই তথ্যও জানাবে LLM.

হ্যাঁ, আপনার বেস্ট ফ্রেন্ড হতে যাচ্ছে, Sequential Chain



```

In [16]: ▶ # prompt template 1: Ask the question of job title
job_title_prompt = PromptTemplate.from_template(
    "Assume the role of a career coach. Suggest me the title of the sing
    Also consider the experience level of {experience}\n"
)
# chain JT: input= major and experience and output= job_title
chain_JT = LLMChain(llm=llm, prompt=job_title_prompt,
    output_key="job_title"
)
  
```

```
In [17]: ▶ # prompt template 2: Ask the question of Required skills
reqd_skills_prompt = PromptTemplate.from_template(
    "What are the name of top 3 required skills for the job title: {job_title}"
)
# chain_RS: input= job_title and output= reqd_skills
chain_RS = LLMChain(llm=llm, prompt=reqd_skills_prompt,
                    output_key="reqd_skills"
                    )
```

```
In [18]: ▶ # prompt template 3: Ask the ways of roadmap for the skills
skill_roadmap_prompt = PromptTemplate.from_template(
    "Tell me the very brief roadmap of getting the following skills: {reqd_skills}"
)
# chain_SR: input= reqd_skills and output= skill_roadmap
chain_SR = LLMChain(llm=llm, prompt=skill_roadmap_prompt,
                    output_key="skill_roadmap"
                    )
```

```
In [19]: ▶ # overall_chain: input= {major} and {experience}
# and output= Job title, required skills, skill roadmap
overall_chain = SequentialChain(
    chains=[chain_JT, chain_RS, chain_SR],
    input_variables=["major", "experience"],
    output_variables=["job_title", "reqd_skills", "skill_roadmap"],
    verbose=True
)
```

এই তিনটা চেইনকে একসাথে বেঁধে শেকল পড়ালেই হয়ে যাবে আপনার আল্টিমেট ক্যারিয়ার ব্রেইনস্টর্মিং চেইন!

```
In [20]: ▶ Response = overall_chain({"major": "Data Scientist", "experience": "5 years"})
```

```
/home/sadat/.local/lib/python3.8/site-packages/langchain_core/_api/deprecation.py:117: LangChainDeprecationWarning: The function `__call__` was deprecated in LangChain 0.1.0 and will be removed in 0.2.0. Use invoke instead.
warn_deprecated()
```

> Entering new SequentialChain chain...

> Finished chain.

```
In [21]: ▶ print(Response["job_title"])
```

"Senior Data Scientist: Maximizing Impact through Advanced Analytics and Strategic Insights"

```
In [22]: ▶ print(Response["reqd_skills"])
```

1. Advanced Analytics
2. Strategic Insights
3. Data Science

```
In [23]: ▶ print(Response["skill_roadmap"])
```

#### 1. Advanced Analytics:

- Start by gaining a strong foundation in statistics, mathematics, and programming.
- Familiarize yourself with data analysis tools such as Excel, SQL, and Python.
- Learn advanced analytical techniques such as regression analysis, predictive modeling, and data mining.
- Gain hands-on experience by working on real-world projects or participating in data analytics competitions.
- Stay updated with the latest trends and developments in the field through online courses, workshops, and conferences.

#### 2. Strategic Insights:

- Develop a strong understanding of business strategy and decision-making processes.
- Familiarize yourself with data analysis tools and techniques, as well as market research methods.
- Gain experience in synthesizing and interpreting data to provide actionable insights.
- Learn how to effectively communicate insights to stakeholders through presentations and reports.
- Develop a strategic mindset and continuously seek opportunities to apply your skills in a business context.

#### 3. Data Science:

- Start by building a strong foundation in statistics, mathematics, and computer science.
- Learn programming languages such as Python and R, as well as data manipulation and visualization tools.
- Gain knowledge of machine learning algorithms and techniques.
- Familiarize yourself with big data technologies such as Hadoop and Spark.
- Gain hands-on experience through data science projects and internships

উপরের উদাহরণটিতে বেশ সিম্পল একটা চেইন দেখানো হয়েছে। আপনি চাইলে আরো বেশ কমপ্লেক্স চেইন তৈরি করতে পারেন। সেটাকে নন লিনিয়ার বানাতে পারেন, আরো ইনপুট যোগ করতে পারেন, ইত্যাদি। আপনার মাথায় কি এমন কোন আইডিয়া আছে ?

## Agent

এজেন্ট Langchain এর একটা বৈশিষ্ট্যবিশিষ্ট ধারণা। LLM নিয়ে সবারই একটা কমন অভিযোগ ছিল, যেইসব ডাটা LLM দেখেনি, সেইসব ক্ষেত্রে বেচারি একেবারেই নির্বোধ এর মত আচরণ করে। এজেন্ট সেই গ্যাপ টাই ফিল আপ করে। অর্থাৎ এজেন্ট মূলত বাইরের নলেজ বেসের সাথে LLM এর সক্ষমতার একটা সেতু তৈরি করে।

একটা সিম্পল উদাহরণ দিয়ে শুরু করি। আমরা Open ai এর LLM ইউজ করছি, যার জ্ঞান কেবলই ২০২১ এর সেপ্টেম্বর পর্যন্ত। এর মানে এর পরের ঘটনা সে কিছু জানে না। যেমন, অভিনেতা Matthew Perry (Chandler Bing of F.R.I.E.N.D.S) গত হয়েছেন কয়েক মাস আগে। দেখি, LLM সেটা জানে কিনা

```
In [39]: ▶ response = llm("When did Matthew Perry die and how?")
print(response)
```

Matthew Perry is still alive as of 2021. He was born on August 19, 1969, and is 51 years old. There have been no reports of his death.

জানেনা, যা ভেবেছিলাম ! এবার তাহলে দেখা যাক, agent কি করতে পারে

```
In [42]: ▶ ## You may have to install wikipedia

#!pip install wikipedia
```

```
In [43]: ▶ from langchain.agents import AgentType, initialize_agent, load_tools
from langchain.llms import OpenAI
```

```
In [48]: ▶ tools = load_tools(["wikipedia"], llm=llm)

agent = initialize_agent(
    tools,
    llm,
    agent = AgentType.ZERO_SHOT_REACT_DESCRIPTION,
    verbose= False
)
```



```
In [49]: ▶ agent.run("When did Matthew Perry die and how?")
```

```
Out[49]: 'Matthew Perry (the actor) died on October 28, 2023. The cause of his death is not specified. Matthew Calbraith Perry (the naval officer) died on March 4, 1858.'
```

সেইইইই!

এজেন্ট আমাদেরকে উইকিপিডিয়া আর LLM এর মধ্যে সংযোগ স্থাপন করিয়ে দিয়েছে, যার ফলাফল, সঠিক উত্তর।

```
In [51]: ▶ from langchain.tools import BaseTool, StructuredTool, Tool, tool, DuckDuckGoSearchRun
from langchain import LLMMathChain
```

এবার আরো কিছু এজেন্ট এর সাথে পরিচয় হয়ে যাক। duckduckgo নামে একটা সার্চ ইঞ্জিন আছে, সেটার সাথে গাণিতিক সক্ষমতা যোগ করে আমরা একটা এজেন্ট তৈরি করতে চাই।

```
In [52]: ▶ search = DuckDuckGoSearchRun()

search_tool = Tool.from_function(
    func=search.run,
    name="Search",
    description="useful for when you need to search the internet for info"
)

llm_math_chain = LLMMathChain(llm=llm, verbose=True)

math_tool = Tool.from_function(
    func=llm_math_chain.run,
    name="Calculator",
    description="Useful for when you are asked to perform math calculations"
)

tools = [search_tool, math_tool]
```

```
In [64]: ▶ agent2 = initialize_agent(
    tools=tools,
    llm=llm,
    agent = AgentType.ZERO_SHOT_REACT_DESCRIPTION,
    verbose= True
)
```

```
In [65]: ▶ agent2.run("What is the average income of US families? Also, multiply it
```

> Entering new AgentExecutor chain...

I need to find the average income of US families and then multiply it by 80.

Action: Search

Action Input: "average income of US families"

Observation: Introduction This report presents estimates on income, earnings, and inequality in the United States for calendar year 2022, based on information collected in the 2023 and earlier Current Population Survey Annual Social and Economic Supplements (CPS ASEC) conducted by the Census Bureau. Median annual family income in Canada 2020, by province ; ... Percentage distribution of household income in the United States in 2022 [Graph], US Census Bureau, September 12, 2023. [Online]. The median family income in the United States grew to 92,750 U.S. dollars in 2022, an increase on the previous year. Family income is the total income earned by all family members who have been ... U.S. Census Bureau, Median Family Income in the United States [MEFAINUSA646N], retrieved from FRED, Federal Reserve Bank of St. Louis; <https://fred.stlouisfed.org/series/MEFAINUSA646N>, (<https://fred.stlouisfed.org/series/MEFAINUSA646N>,) January 11, 2024. RELEASE TABLES Income and Poverty in the United States Personal and Family Income by Census Region, Annual RELATED DATA AND CONTENT A further 19.3 percent of families in the United States had an annual income between 100,000 and 149,999 U.S. dollars. ... Median annual family income in Canada 2020, by earning situation ...

Thought: I should look for the median annual family income in the United States.

Action: Search

Action Input: "median annual family income United States"

Observation: This report presents estimates on income, earnings, and inequality in the United States for calendar year 2022, based on information collected in the 2023 and earlier Current Population Survey Annual Social and Economic Supplements (CPS ASEC) conducted by the Census Bureau. 2022: 92,750 (+ more) Updated: Sep 12, 2023 Units: Current Dollars, Not Seasonally Adjusted Frequency: Annual 1Y | 5Y | 10Y | Max to Edit Graph Share Links Account Tools NOTES Source: U.S. Census Bureau Release: Income and Poverty in the United States Units: Current Dollars, Not Seasonally Adjusted Frequency: Annual Updated on Sept. 13 with the latest U.S. Census Bureau data. The national median household income is \$74,580, according to the most recent U.S. Census Bureau Current Population Survey data... U.S. household income distribution 2022 Published by Statista Research Department , Nov 3, 2023 In 2022, just over 50 percent of Americans had an annual household income that was less... For an indication of the comparability of medians calculated using Pareto interpolation with medians calculated using linear interpolation, see Series P-60, Number 114, Money Income in 1976 of Families and Persons in the United States ([www2.census.gov/prod2/popscan/p60-114.pdf](http://www2.census.gov/prod2/popscan/p60-114.pdf)). Suggested Citation:

Thought: I should use the median annual family income of \$92,750.

Action: Calculator

Action Input: 92750 \* 80

> Entering new LLMMathChain chain...

92750 \* 80``text

92750 \* 80

````

```
...numexpr.evaluate("92750 * 80")...
```

Answer: **7420000**  
> Finished chain.

Observation: **Answer: 7420000**  
Thought: **I now know the final answer.**  
**Final Answer: The average income of US families is \$92,750 and when multiplied by 80, it is \$7,420,000.**

> Finished chain.

Out[65]: 'The average income of US families is \$92,750 and when multiplied by 80, it is \$7,420,000.'

দেখতেই পাচ্ছেন, এজেন্ট নিজে নিজেই সিদ্ধান্ত নিয়েছে, প্রথমে ইনকাম সংক্রান্ত তথ্য খুঁজে বের করবে। এরপরে গাণিতিক টুল ব্যবহার করে বাকি হিসেবটুকু করে দিয়েছে !

এরকম আরো অনেক দরকারি টুল রয়েছে, যেটা দিয়ে এজেন্ট বানিয়ে অবিশ্বাস্য সব এক্সপেরিমেন্ট করা যায়। সময় থাকলে অবশ্যই চেক করে দেখবেন !

## Memory

একটা ব্যাপার শুরুতেই পরিষ্কার করে নেই: **\*\*চ্যাটজিপিটি নামে কোন LLM বাজারে নেই\*\***, বরং বলা যায়, চ্যাটজিপিটি একটি বা দুটি LLM এর সমন্বয়ে, এবং আরো কিছু উপাদানের সংমিশ্রণে তৈরি একটি ওয়েব এপ্লিকেশন। LLM এর নিজস্ব কোন মেমোরি নেই, কনভার্সেশন মনে রাখার সক্ষমতাও ওর নেই। আমরা আগে যেই LLMchain দেখেছি, সেটির ক্ষেত্রেও এই কথাটি প্রযোজ্য।

বিশ্বাস হয়নি ? নিচের উদাহরণটি দেখুন:

In [66]: `llm("Can you tell me who was the president of USA in 1996? ")`

Out[66]: '\n\nThe president of the USA in 1996 was Bill Clinton.'

In [67]: `llm("Which party was in power back then? ")`

Out[67]: '\n\nIt is not specified which country or time period is being referred to, so it is impossible to determine which party was in power. '

দেখলেন তো, বোকা LLM জানেই না, কী নিয়ে কথা হচ্ছে !

কাজেই, মেমোরি নিয়ে কথা বলা যাক

```
In [137]: ➤ from langchain.chains import ConversationChain
```

```
In [138]: ➤ convo = ConversationChain(llm=llm)
```

```
In [143]: ➤ prompt_template_president = PromptTemplate(
    input_variables = [],
    template = "Can you tell me who was the president of USA in 1996?"
)

prompt_template_party = PromptTemplate(
    input_variables = [],
    template = "Which party was in power back then?"
)
```

```
In [144]: ➤ convo.run(prompt_template_president.format())
```

```
Out[144]: ' In 1996, the president of the USA was Bill Clinton. He was the 42nd
president of the United States, serving from 1993 to 2001. He was a me
mber of the Democratic Party and is known for his economic policies, i
ncluding the North American Free Trade Agreement and the Balanced Budg
et Act of 1997.'
```

```
In [145]: ➤ convo.run(prompt_template_party.format())
```

```
Out[145]: " The Democratic Party was in power during Bill Clinton's presidency i
n 1996. The Democratic Party is one of the two major political parties
in the United States, along with the Republican Party. They have histo
rically supported social welfare programs and progressive policies."
```

এই হল মেমোরির জাদু!

```
In [147]: ➤ ## Here's the memory content
convo.memory
```

```
Out[147]: ConversationBufferMemory(chat_memory=ChatMessageHistory(messages=[Huma
nMessage(content='Can you tell me who was the president of USA in 199
6?'), AIMessage(content=' In 1996, the president of the USA was Bill C
linton. He was the 42nd president of the United States, serving from 1
993 to 2001. He was a member of the Democratic Party and is known for
his economic policies, including the North American Free Trade Agreeeme
nt and the Balanced Budget Act of 1997. '), HumanMessage(content='Which
party was in power back then?'), AIMessage(content=" The Democratic Pa
rty was in power during Bill Clinton's presidency in 1996. The Democra
tic Party is one of the two major political parties in the United Stat
es, along with the Republican Party. They have historically supported
social welfare programs and progressive policies."))])
```

In [149]: `print(convo.memory.buffer)`

Human: Can you tell me who was the president of USA in 1996?

AI: In 1996, the president of the USA was Bill Clinton. He was the 42nd president of the United States, serving from 1993 to 2001. He was a member of the Democratic Party and is known for his economic policies, including the North American Free Trade Agreement and the Balanced Budget Act of 1997.

Human: Which party was in power back then?

AI: The Democratic Party was in power during Bill Clinton's presidency in 1996. The Democratic Party is one of the two major political parties in the United States, along with the Republican Party. They have historically supported social welfare programs and progressive policies.

## Conclusion

এই হল ল্যাংচেইন নিয়ে আমার ক্ষুদ্র (মারবেন না) প্রয়াস। এই ব্যাপারটি না বললেই নয় যে, ল্যাংচেইন আরো বিশদ, আরো অজস্র ফিচারের সমাহার। আমি চেষ্টা করেছি খুব সহজে একটা সূচনা কীভাবে দেয়ার ব্যাপারে। ল্যাংচেইন দিয়ে আরো নানারকমের এপ্লিকেশান বানানো যায়, এক্সপেরিমেন্ট করা যায়। কেউ চাইলে চেষ্টা করে দেখতে পারেনঃ :

<https://python.langchain.com/docs/modules/> (<https://python.langchain.com/docs/modules/>)

N.B. Langchain is still evolving, they are adding many new functionality, and changing things rapidly. It's strongly encourage to keep an eye on their official page

## References

1. <https://www.youtube.com/watch?v=2xxziIWmaSA&list=PLqZXAkV1bPNQER9mLmDbntNfSpzdDIU5&index=3> (<https://www.youtube.com/watch?v=2xxziIWmaSA&list=PLqZXAkV1bPNQER9mLmDbntNfSpzdDIU5&index=3>)
2. <https://www.youtube.com/watch?v=nAmC7SoVLd8> (<https://www.youtube.com/watch?v=nAmC7SoVLd8>)
3. <https://www.deeplearning.ai/short-courses/langchain-for-llm-application-development/> (<https://www.deeplearning.ai/short-courses/langchain-for-llm-application-development/>)

In [ ]: