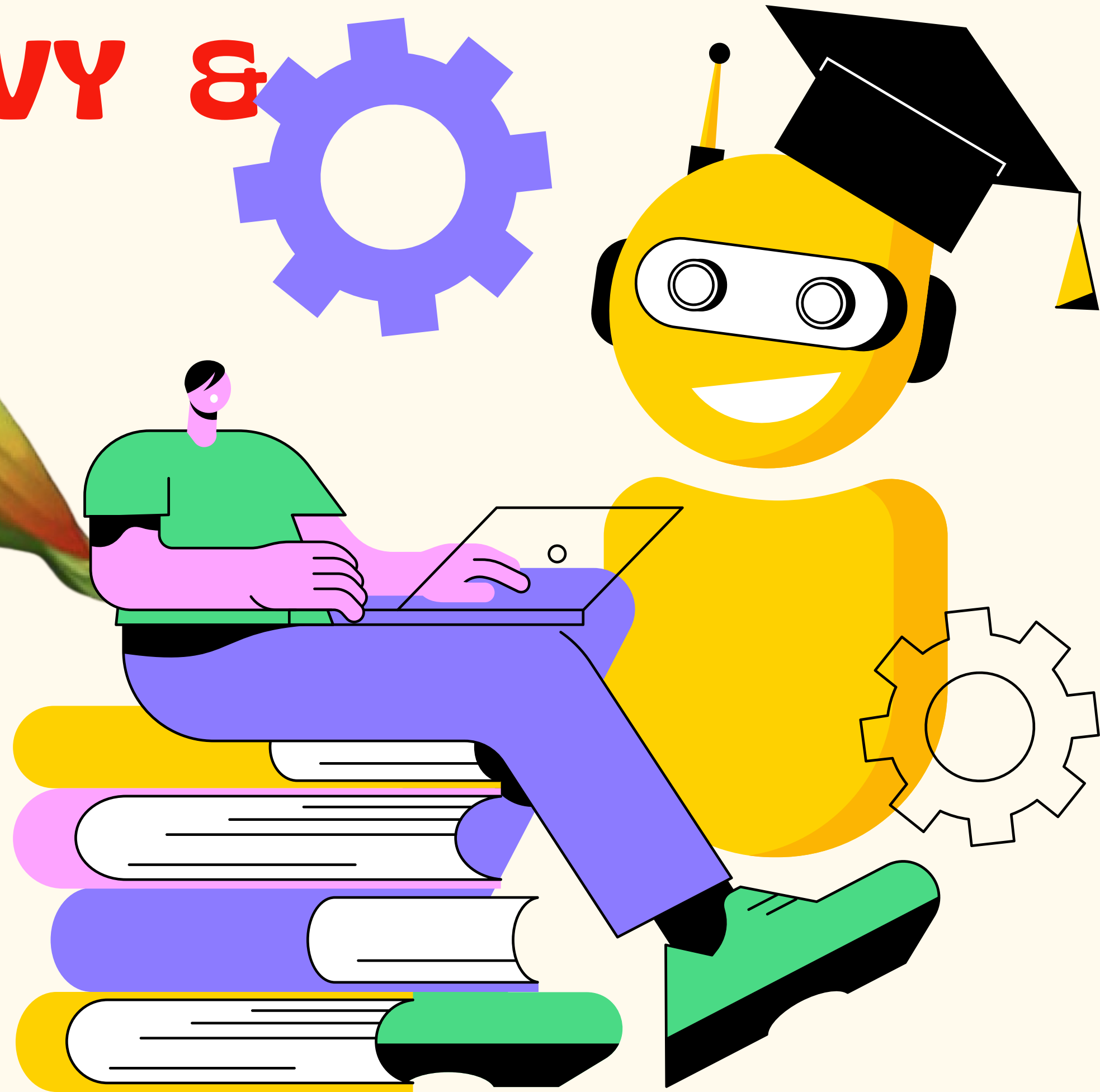
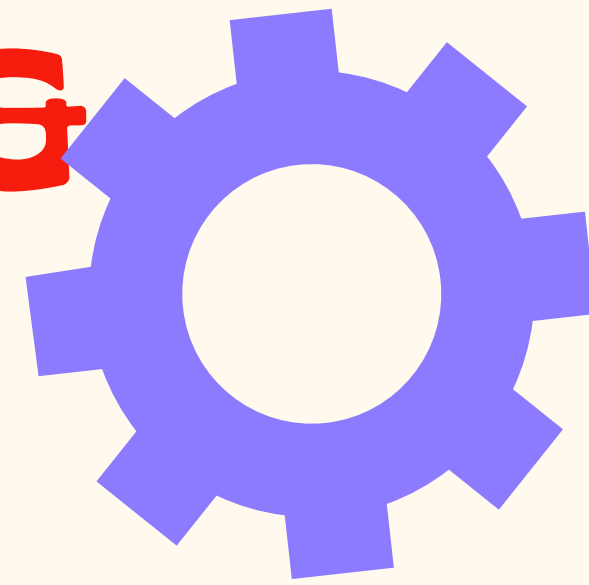


APACHE-2 LICENSED CHATBOT GPT4ALL-J GROOVY & LANGCHAIN



GPT4ALL-J : CHALLENGE SOLVED

- **QUANTIZED MODEL OF GPT-J THAT CAN RUN ON CPU. GPT-J IS RELEASED BY ELEUTER-AI. WHILE NOMIC HAS RELEASED QUANTIZED VERSION.**
- **PERFORMANCE IS COMPARABLE TO THE GPT-3, AS ITS INSTRUCTION TUNED WITH FORMATTED DATA**
- **CAN BE INCORPORATED WITH LLM_CHAIN WITH COMPLEX PROMPT TEMPLATES**

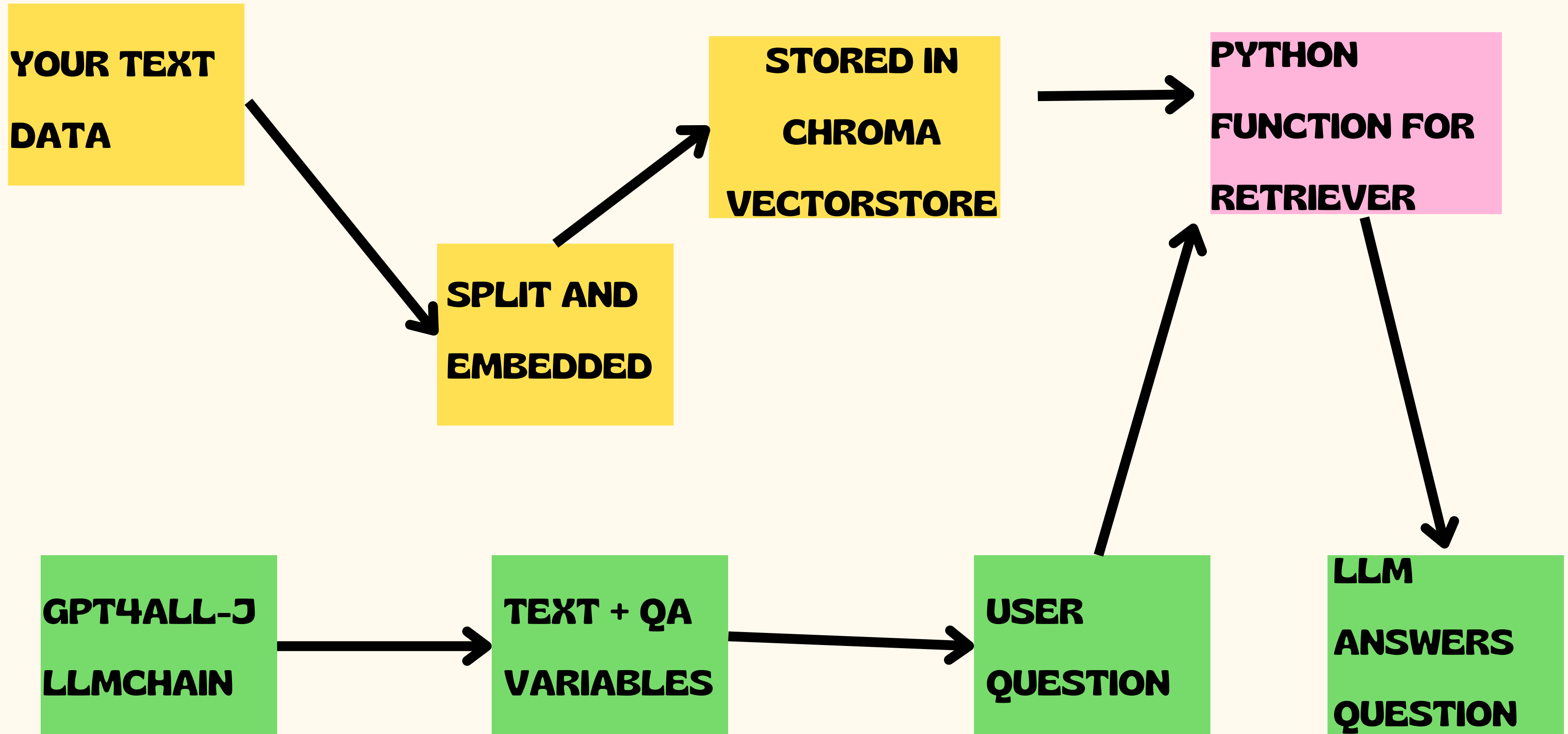
NOTABLE ISSUES:

- **REQUIRES ATLEAST 12GB OF RAM**
- **WITH LONGER PROMPTS TAKES MORE TIME TO RETURN ANSWERS**
- **CRASHES THE EXECUTION IF WRONG EMBEDDING MODEL IS USED IN RETRIEVER**

NOTABLE DEVELOPMENTS:

- **CHAIN OF THOUGHT PROMPTING CAN BE IMPLEMENTED**
- **USED EFFICIENTLY FOR TEXT QUESTION ANSWERING**
- **TOKENIZER MODEL USED FOR EMBEDDING DOCUMENTS CAN BE ANY SENTENCE-TRANSFORMER MODEL**

GPT4ALL-J QA PIPELINE



IMPLEMENTING QA CHAIN WITH GPT4-ALL

```
FROM LANGCHAIN.CHAINS IMPORT RETRIEVALQA

FROM LANGCHAIN.EMBEDDINGS IMPORT LLAMACPPEMBEDDINGS

FROM LANGCHAIN.CALLBACKS.STREAMING_STDOUT IMPORT

STREAMINGSTDOUTCALLBACKHANDLER

FROM LANGCHAIN.VECTORSTORES IMPORT CHROMA

FROM LANGCHAIN.LLMS IMPORT GPT4ALL, LLAMACPP

IMPORT OS


CALLBACKS = [STREAMINGSTDOUTCALLBACKHANDLER()]

LLM = GPT4ALL(MODEL="/CONTENT/GGML-GPT4ALL-J-V1.3-GROOVY.BIN",

    N_CTX=1000,

    BACKEND='GPTJ',

    CALLBACKS=CALLBACKS,

    VERBOSE=FALSE)

FROM LANGCHAIN IMPORT PROMPTTEMPLATE, LLMCHAIN

TEMPLATE = """USE BELOW TEXT AND ANSWER THE QUESTION: {QUESTION}

{TEXT}

ANSWER: LET'S THINK STEP BY STEP."""

PROMPT = PROMPTTEMPLATE(TEMPLATE=TEMPLATE,

    INPUT_VARIABLES=["QUESTION", "TEXT"])
```

```
DEF GET_TEXT(QUESTION):

    SIMILAR_DATA = VECTORSTORE.SIMILARITY_SEARCH(QUESTION)

    TEXT = ''

    FOR ELEM IN SIMILAR_DATA:

        TEXT = TEXT + ELEM.PAGE_CONTENT

    PRINT(TEXT)

    RETURN TEXT
```

```
DEF RETRIEVE_ANSWER(QUESTION):

    SUPPORT_TEXT = GET_TEXT(QUESTION)

    ANSWER = LLM_CHAIN.RUN(QUESTION=QUESTION,

        TEXT= SUPPORT_TEXT)

    RETURN ANSWER
```

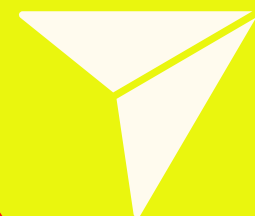
```
RETRIEVE_ANSWER("WHAT IS THE DINOSAUR MOVIE?")
```

THANKS FOR WATCHING

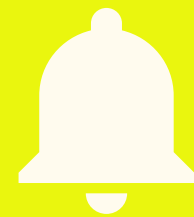
REMEMBER TO PRACTICE WITH EXAMPLES



LIKE



SHARE



SUBSCRIBE