



# LangChain 101

Austin LangChain Users Group

# Agenda

- Mixer
- Welcome Message
- News & Announcements
- Lecture
- Labs



# Welcome - Austin LangChain

- Local Austin LangChain User Group
    - Twitter - [@AustinLangChain](https://twitter.com/AustinLangChain)
    - Github - [https://github.com/colinmcnamara/austin\\_langchain](https://github.com/colinmcnamara/austin_langchain)
    - Meetup - <https://www.meetup.com/austin-lanchain-ai-group/>
    - Discord - <https://discord.gg/fjQfpwcsZX>
  - Monthly in person meetings & workshops
  - Low stress, learning and sharing
  - Connect with other early adopters
  - Learn, Share, Grow
- 
- CoC - **Be cool** to each other, no gross behavior



# What we are building

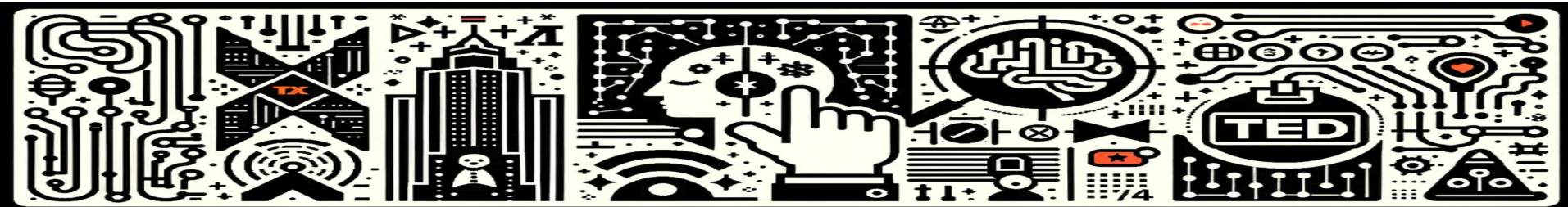
- Community of builders centered around austin
- Users of LangChain
- Developers of LangChain
- Investors in AI Applications, People, and Platforms
- Contributors to Hackathons
- Friendly, Fun, Growth oriented environment





# News and Announcements

- LangChain 102 - November 14th @ 1204 San Antonio 6-9 [here](#)
  - User Showcase!
  - LangChain 102 - Access from Google Drive + LangSmith Debug (possible LangServe...)
- LangServe AI Middleware Released <https://blog.langchain.dev/langserve-hub/>
- LangServe Playground Released  
<https://weblangchain.fly.dev/chat/playground/>
- LangServe Webinar 11/2  
<https://www.crowdcast.io/c/zcl2p0y0wrvd?ref=blog.langchain.dev>





# LangChain 101

Lecture & Labs

# About Me - Colin McNamara

- **Home:** Austin, East Side
- **Work:** Managing Partner, Engineering & Finance @ Always Cool Brands
- **Background:** Hyperscale Cloud Engineering & Operations
- **FOSS:** Linux, OpenStack, OpenDaylight, ...
- **Using LangChain:** Business operations
- **Goals:** grow skills, grow community, automate operations
- **Socials:**
  - WWW <http://www.colinmcnamara.com>
  - X(Twitter) [@colinmcnamara](https://twitter.com/colinmcnamara)
  - LinkedIn <https://www.linkedin.com/in/colinmcnamara/>





# Always Cool Brands - LangChain Use Case - Due Diligence

## Our Business

- Product Design - Brand & Private Label
  - D2C, Retail, Grocery, Club
- Engineering, Sourcing & Financing
- Regulatory - SQF, HASSUP, CAPA, SEC

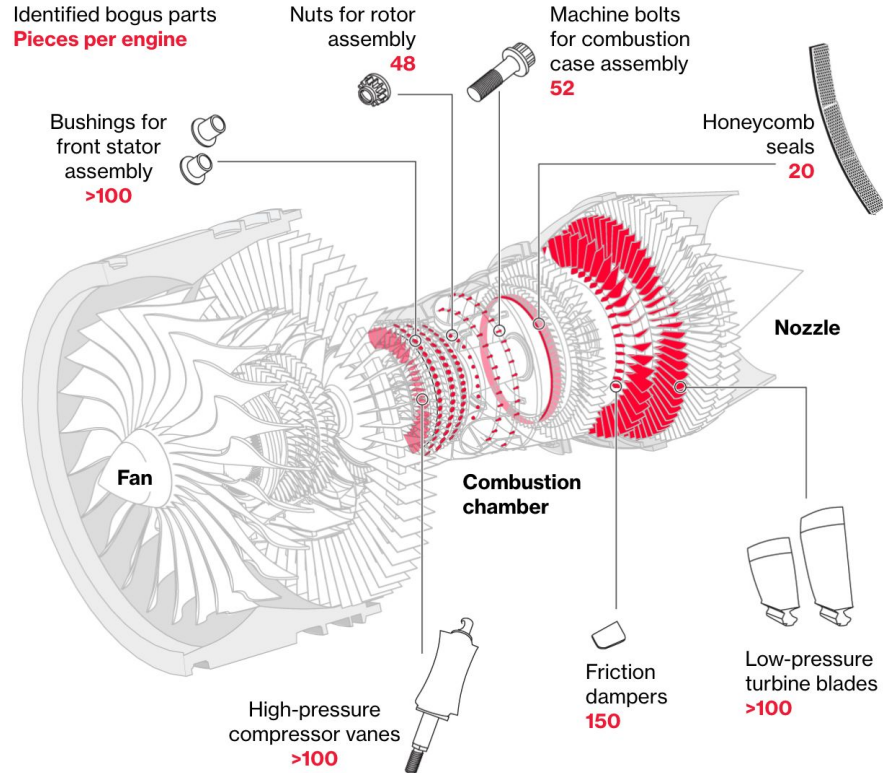
## RAG Solution for Due Diligence

- Docs + Workshops -> Vector Stores
- Python -> Markdown -> Git
- **Identified fraud** in our supply chain



# How safe is your next flight?

- London distributor AOG Technics - fake parts with forged documents.
- 90+ fake certificates and bogus parts on 126 engines.
- Fraud exposed weaknesses in parts market.
- Airlines urgently removing fake parts - Costing millions



# How do we scale this?

- **Creation of API layer**
  - Langserve
- **Integrations into key apps**
  - Hubspot
  - Email / Chat
  - Gdocs
  - Zapier
  - Kafka
  - Product websites
- **Control Cost at Scale**
  - 40MM unit orders normal for retail
- **Protect against Data Leakage**
  - API endpoints vs public chat
- **Address Bias in reports**
  - Ragas, Human as Agent
- **Audit and Control**
  - Constitutional AI in pipelines
  - Safety Controls at user interface



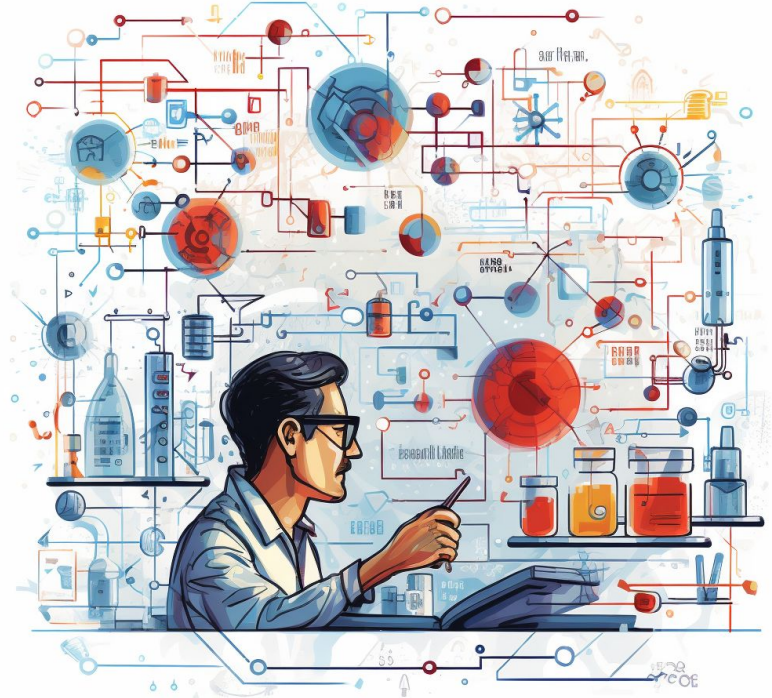
# Lab Series - Our journey into production

- **LC101 - Intro to AI Microservices (Colab, Python, Streamlit)**
- LC102 - Enrich with data (Retrievers, Loaders, Ragas) - **Maybe LangServe...**
- LC103-9 - (Tools, Retrievers, Agents, Logic, Chains, Vector Stores)
- LC110 - Logging - (STD\_OUT, LangSmith)
- LC201 - Local LangChain (Docker, Nginx)
- LC202 - Cloud Langchain (K8s, Netlify, Streamlit, GCE)
- LC203 - GitOps workflows(AI Microservices from repo commit)
- LC204 - Pipelines (git actions, netlify)
- LC205 - Reporting (Loki, SigNoz, LangSmith)



# Course Structure

- Conceptual Overview (lecture)
- Labs, run in browser on Google Colab
- Conceptual Reinforcement
- More labs
- ... etc





# What is LangChain?

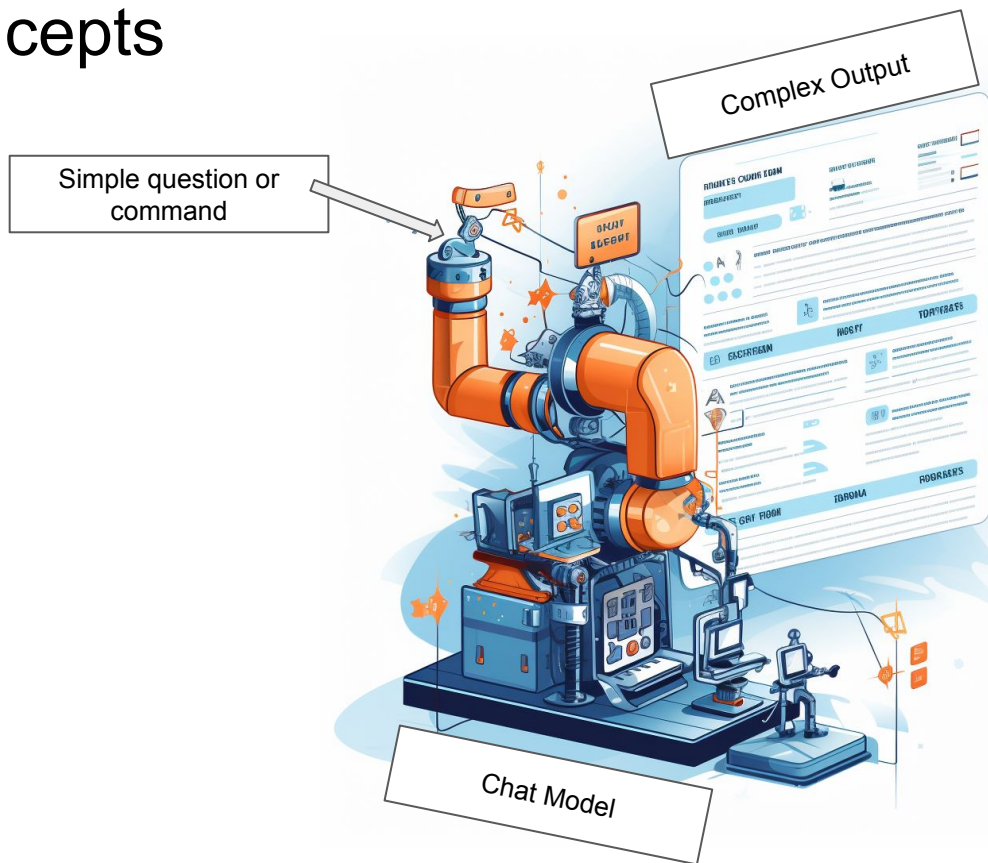
- **Open-Source** library for building LLM applications
- Founded by Harrison Chase
- **Python** and **Typescript** packages
- Focused on composition and modularity
  - Modular components & Implementations
  - Common use cases that combine components together





# LangChain Key Concepts

- Models
  - Chat (Today's Labs)
  - LLM (Future Labs)



# LangChain Key Concepts

- Models
  - LLM Integrations: 20+
  - Text Embeddings Models: 10+

# LangChain Key Concepts

- Prompts

- Features

- Separate instructions & user input
    - Supports variable placeholders (e.g., {variable\_name})
    - Composable for text & chat prompts

- Types

- PromptTemplate (text)
    - ChatPromptTemplate (chat)
    - Chain & Message templates

## Chat Prompt Template

```
from langchain.chat_models import ChatOpenAI

from langchain.prompts import HumanMessagePromptTemplate
from langchain.schema.messages import SystemMessage

chat_template = ChatPromptTemplate.from_messages(
    [
        SystemMessage(
            content=(
                "You are a helpful assistant that re-writes the user's text to "
                "sound more upbeat."
            )
        ),
        HumanMessagePromptTemplate.from_template("{text}"),
    ]
)

llm = ChatOpenAI()
llm(chat_template.format_messages(text="i dont like eating tasty things."))
```

```
AIMessage(content='I absolutely love indulging in delicious treats!')
```

# LangChain Key Concepts

- Prompts

- Features

- Separate instructions & user input
    - Supports variable placeholders (e.g., {variable\_name})
    - Composable for text & chat prompts

- Types

- PromptTemplate (text)
    - ChatPromptTemplate (chat)
    - Chain & Message templates

## Prompt Template (Similar to sed/awk)

```
from langchain.prompts import PromptTemplate

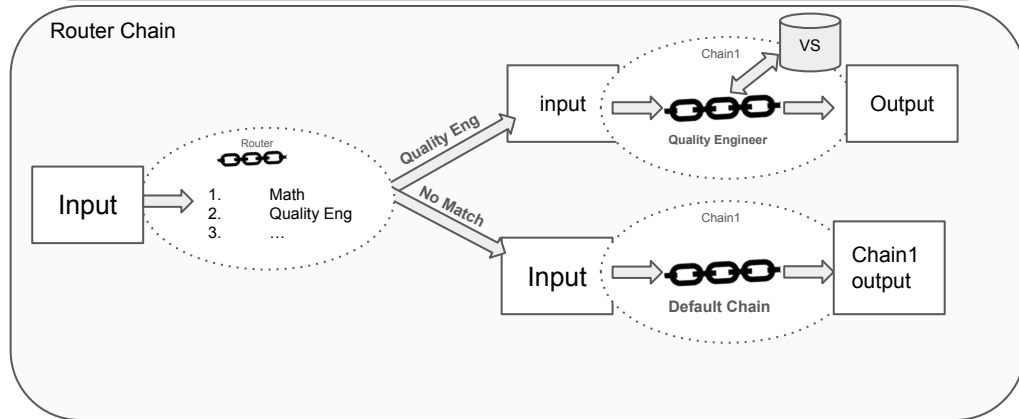
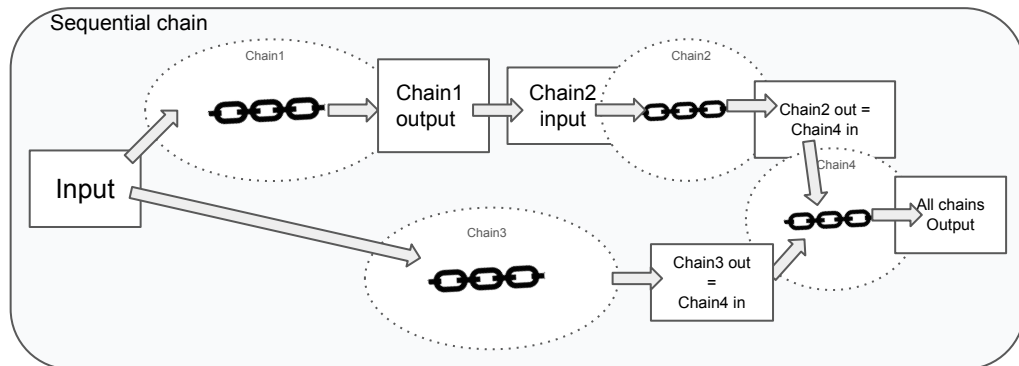
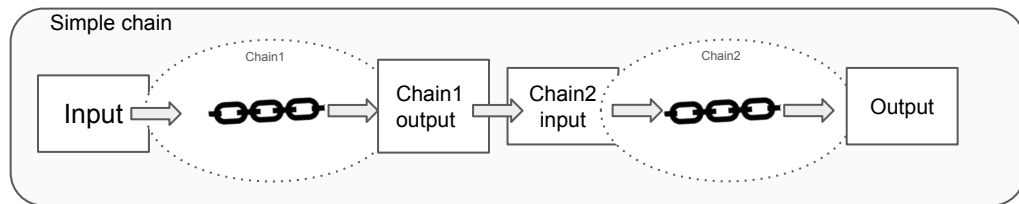
prompt_template = PromptTemplate.from_template(
    "Tell me a {adjective} joke about {content}."
)
prompt_template.format(adjective="funny", content="chickens")
```

```
'Tell me a funny joke about chickens.'
```

# LangChain Key Concepts

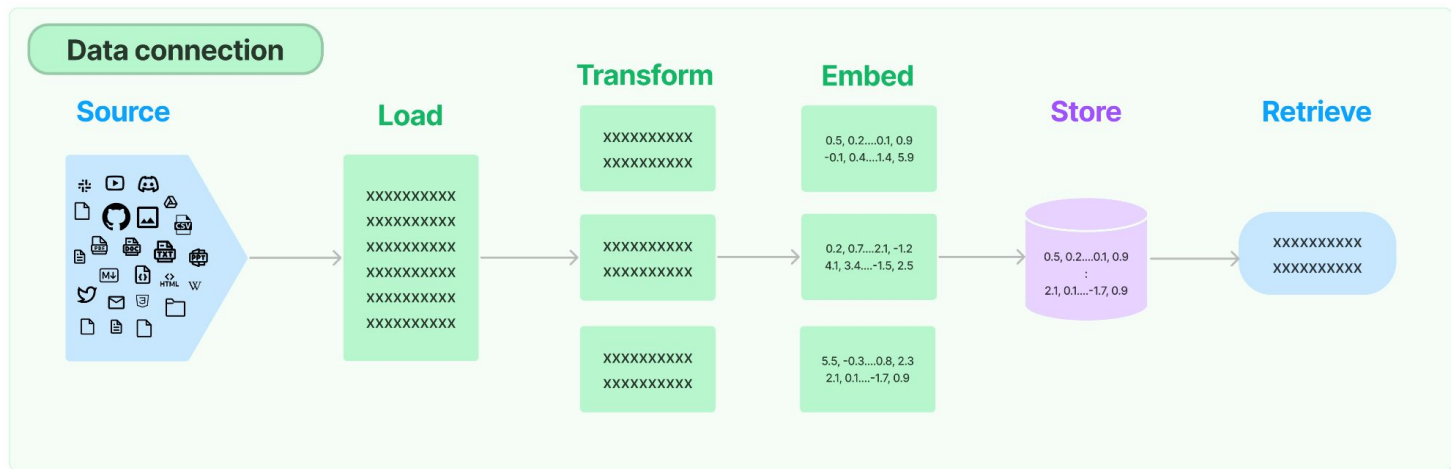
- Chains

- Simple Chain
- Sequential Chain
- Router Chain
- Transformation Chains
- Building blocks for other chains
- Application specific chains: 20+



# LangChain Key Concepts

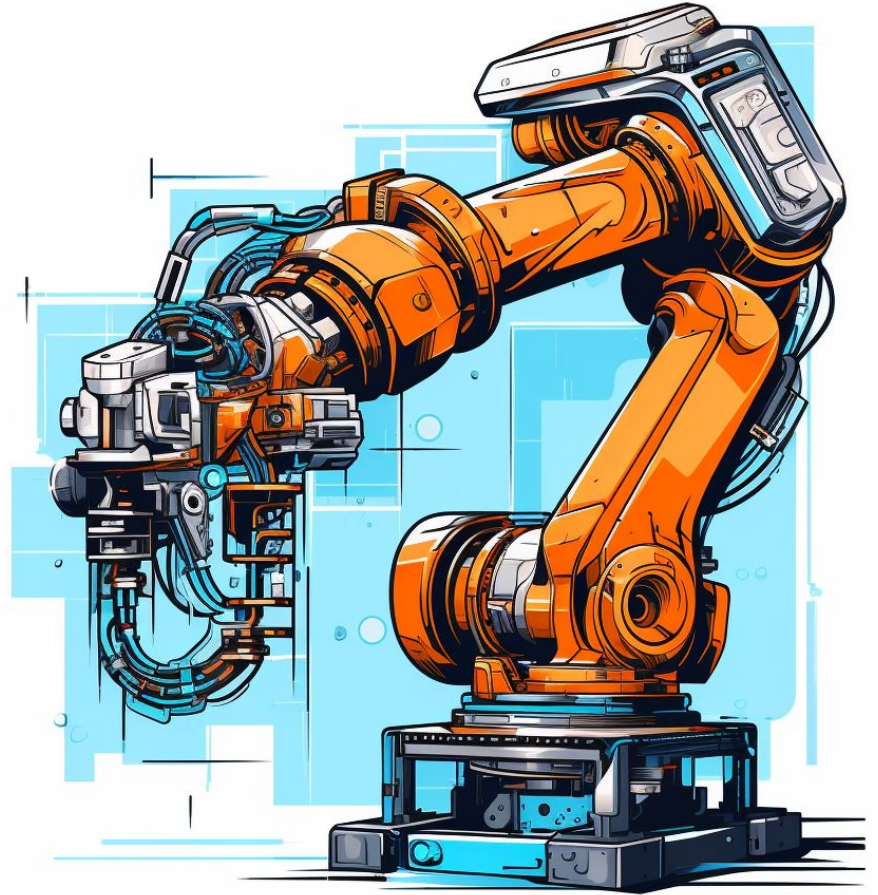
- Indexes
  - Document Loaders: 50+
  - Text Splitters: 10+
  - Vector Stores: 10+
  - Retrievers: 5+





# LangChain Key Concepts

- Tools
  - Set of functions available to agents
  - Internet Search
  - Multiple Vector Stores
  - 50+ more

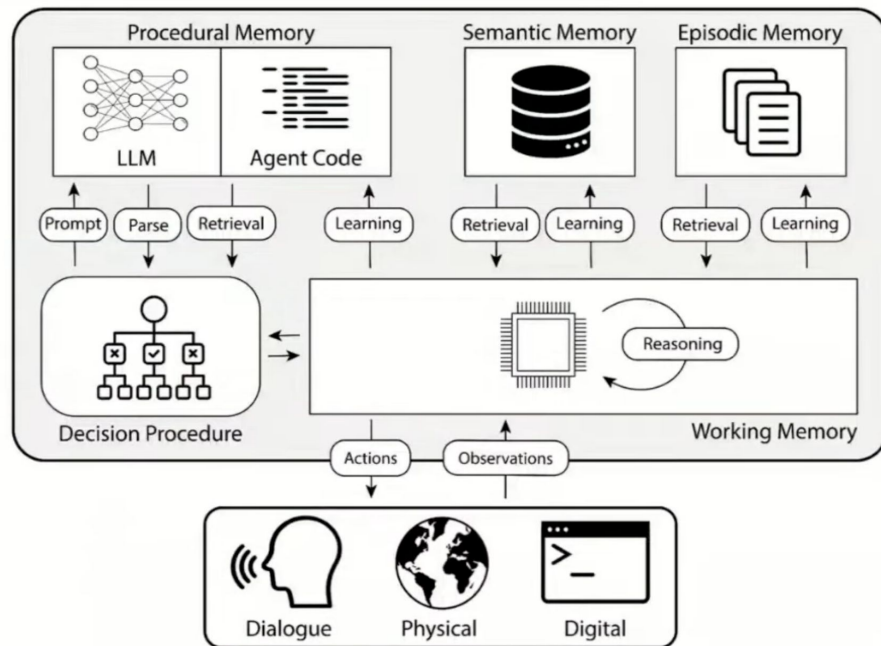


# LangChain Key Concepts

- Agents

- Independent Entity
- Has access to tools
- Reasons, Acts (ReAct)
- Combination of
  - LLM
  - Code
  - Memory
  - Tools

## The CoALA framework





**Labs!!!**



# Lab - get your OpenAI key

1. Go to <https://openai.com/>
2. Click on Menu > Developers > Overview
3. Click on your Profile image (top right) > View API keys
4. Click on `+ Create new secret key`
5. Enter an optional `Name` for the API key for future reference
6. Save this key using a password manager

**Colin McNamara**

colin@2cups.com

ColinMcNamara

Manage account

View API keys

Invite team

Visit ChatGPT

Visit DALL-E

Help

Pricing

Terms &amp; policies

Log out

# Open in Notebook in Colab

The screenshot shows the GitHub interface for the repository `austin_langchain`. The file `101-1-streamlit_streaming.ipynb` is selected. A callout box with the text "Click Here" and an arrow points to the "Open in Colab" button.

**Repository Header:**

- File display: `marla / austin_langchain` (Public)
- Notifications
- Fork: 0
- Star: 1

**Navigation Bar:**

- <> Code
- Issues
- Pull requests
- Actions
- Projects
- Security
- Insights

**Files Panel (Left):**

- main
- Go to file
- labs
  - LangChain\_101
    - 101-1-streamlit\_streaming.ipynb
    - 101-2-streamlit\_document\_sea...
    - streamlit\_document\_search.ipynb
  - LangChain\_102
    - .DS\_Store
    - LICENSE.txt
  - resources
    - .DS\_Store
    - .gitignore
    - README.md

**File View (Right):**

`austin_langchain / labs / LangChain_101 / 101-1-streamlit_streaming.ipynb`

colinmcnamara fixed 101-1 link 2f0a932 · 1 minute ago History

Preview Code Blame 1 lines (1 loc) · 4.57 KB Raw Copy Download

**Code Content:**

```
In [1]: !pip install -q langchain openai streamlit

WARNING: You are using pip version 24.0; however, version 23.2.1 is available.
You should consider upgrading via: 'python -m pip install --upgrade pip' command.
Note: you may need to restart the terminal to use updated packages.

In [2]: %%writefile streaming_app.py
from langchain.callbacks.base import BaseCallbackHandler
from openai import OpenAI
page = ...

def main():
    initial_text = ""
```

# LAB101-1 - Accessing Chat Model via API

- Required: OpenAI API Key: (keep this a secret)
- Required: Google account
- Use free Colab instance: <https://colab.research.google.com>
- Repo: [https://github.com/colinmcnamara/austin\\_langchain/](https://github.com/colinmcnamara/austin_langchain/)
- Notebook:  
[https://colab.research.google.com/github/colinmcnamara/austin\\_langchain/blob/main/labs/LangChain\\_101/101-1-streamlit\\_streaming.ipynb](https://colab.research.google.com/github/colinmcnamara/austin_langchain/blob/main/labs/LangChain_101/101-1-streamlit_streaming.ipynb)
- 





# LAB101-1 - Accessing Chat Model via API

- Connect to google colab
- Run
- Runall
- Runtime Commands (Delete and restart
- Save to Google Drive
- Interact with streamlit app - Streaming Chat Interface

# LAB101.3 - Chat w/ internet search

- Required: OpenAI API Key: (keep this a secret)
- Required: Google account
- Use free Colab instance: <https://colab.research.google.com>
- Repo: [https://github.com/colinmcnamara/austin\\_langchain/](https://github.com/colinmcnamara/austin_langchain/)
- Notebook:  
[https://colab.research.google.com/github/colinmcnamara/austin\\_langchain/blob/main/labs/LangChain\\_101/101-3-search-chat.ipynb](https://colab.research.google.com/github/colinmcnamara/austin_langchain/blob/main/labs/LangChain_101/101-3-search-chat.ipynb)



# LAB101-3 Agents Accessing the internet

- Model = gpt-3.5-turbo
- Streaming = True
- Tools = DuckDuckGoSearchRun
- Memory
- Return Steps

# LAB101.2 - Document Search

- Required: OpenAI API Key: (keep this a secret)
- Required: Google account
- Use free Colab instance: <https://colab.research.google.com>
- Repo: [https://github.com/colinmcnamara/austin\\_langchain/](https://github.com/colinmcnamara/austin_langchain/)
- Notebook:  
[https://colab.research.google.com/github/colinmcnamara/austin\\_langchain/blob/main/labs/LangChain\\_101/101-3-search-chat.ipynb](https://colab.research.google.com/github/colinmcnamara/austin_langchain/blob/main/labs/LangChain_101/101-3-search-chat.ipynb)

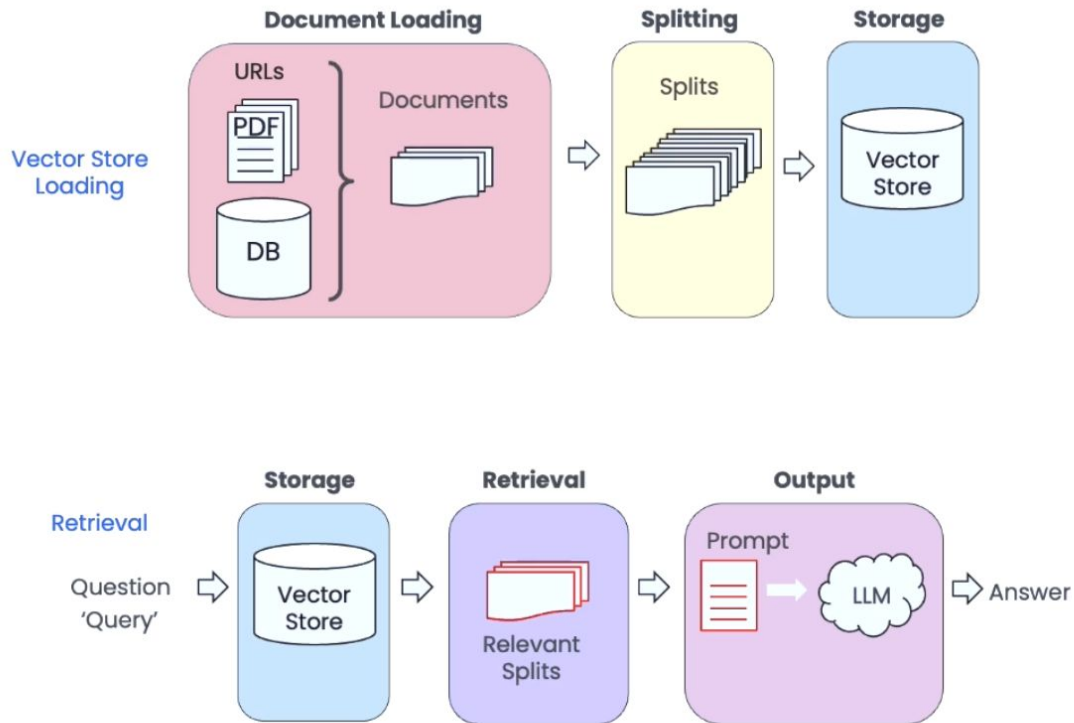


# LAB101-2 pdf Search

- Download  
[https://www.cisa.gov/sites/default/files/ncirp/National\\_Cyber\\_Incident\\_Response\\_Plan.pdf](https://www.cisa.gov/sites/default/files/ncirp/National_Cyber_Incident_Response_Plan.pdf)
- Play with temperature (0.7)



# Retrieval Augment Generation





# Thank you - Please Join

- Local Austin LangChain User Group
  - Twitter - [@AustinLangChain](https://twitter.com/AustinLangChain)
  - Github - [https://github.com/colinmcnamara/austin\\_langchain](https://github.com/colinmcnamara/austin_langchain)
  - Meetup - <https://www.meetup.com/austin-lanchain-ai-group/>
  - Discord - <https://discord.gg/SDwD254V>
  - Monthly in person meetings & workshop
  - Monthly remote lab repeat
- Low stress, learning and sharing
- Connect with other early adopters
- Learn, Share, Grow