**LLM Based *Chatbot* with Google’s Gemini API**

* Get Google’s Gemini API, get the keys, and deploy it in the Streamlit cloud.

**Q. To get API keys?**

1. search for “google ai studio”

2. left navigation bar “get an API key” – copy it (keys are limited)

3. create .env file – to load or mention keys and other configurations.

**Q. Libraries required?**

* Python
* Google generative
* Streamlit

**Q Steps?**

1. Install requirements.
2. Create folder streamlt – inside it configuration are mentioned that are required when we deploy it in streamlit cloud.
3. create .env file – to load or mention keys.

**Q. how to run?**

* streamlit run filename.py

**Q. how to deploy?**

1. Upload on GitHub, publicly available
2. “share.streamlit.io” – create an account
3. New app – deploy it

IT IS FREE OF COST

**Q. Why are we storing the API key in .env file?**

* better security practices.
* separating configuration from code - easier to manage different env (like development and production)
* sensitive information is not inadvertently included in the codebase.
* it simplifies the process of updating or changing credentials without modifying the actual code.

**Q. What is .toml file?**

* configuration file format that stands for "Tom's Obvious, Minimal Language."
* designed to be easy to read and write, making it a popular choice for application configuration settings.
* structure of a `.toml` file is straightforward, using key-value pairs, tables, and arrays to organize data.
* often used in various programming env for managing settings, dependencies, and project configurations.
* Compared to other formats like JSON or XML, TOML is intended to be more human-readable and writable, which can simplify the process of configuration management.

READ CONFIG FILE.

**Q. what is "from dotenv import load\_dotenv"?**

* imports the load\_dotenv ***function*** from the dotenv ***module***.
* used to load environment variables from a .env file into the environment of the running application.
* *Usage: After calling load\_dotenv(), any variables defined in the .env file will be accessible through os.getenv() or similar methods.*