# Prerequisites

* Python
  + https://www.python.org/downloads/
* Packages from requirements.txt installed
  + Open terminal and type in the following
    - pip install -r requirements.txt
* Api\_key requested and set as environment variable
  + OpenAI
    - <https://help.openai.com/en/articles/4936850-where-do-i-find-my-openai-api-key>
    - <https://medium.com/@lorenzozar/how-to-get-your-own-openai-api-key-f4d44e60c327>
    - This is for extracting expected .json file to requested format
    - OPENAI\_API\_KEY
  + LandingAI
    - <https://docs.landing.ai/landinglens/api-key>
    - This is for extracting .md file from the .pdf file supplied
    - VISION\_AGENT\_API\_KEY
  + LangSmith
    - <https://docs.smith.langchain.com/administration/how_to_guides/organization_management/create_account_api_key>
    - This is to debug issues and understand what is happening under the hood for the RAG system
    - Debugging purposes

# Processing Workflow High level

A diagram of a company

AI-generated content may be incorrect.

Run through each file in the following folder in specified order

* 1\_run\_rotation\_fix.py
* 2\_pdf\_json\_agentic.py
* 3\_convert\_llm\_json.py
* 4\_qc\_and\_flagging.py
* 5\_build\_knowledge\_base.py

To run the file

* Open terminal and type in the following
  + python file.py

To run the RAG application, do the following after the previous 5 workflow are completed

* streamlit run app.py

# Expected dependencies & output

1\_run\_rotation\_fix.py

* dependencies
  + pdf files in source\_pdfs
  + no nested or individual folder but rather all the files
* output
  + new folder corrected\_pdfs
  + rotated pdf files

2\_pdf\_json\_agentic.py

* dependencies
  + 1\_run\_rotation\_fix.py ran
* Output
  + New folder generic\_json\_outputs
  + Text extracted .json files per each .pdf input

3\_convert\_llm\_json.py

* dependencies
  + 2\_pdf\_json\_agentic.py ran
* Output
  + New folder final\_pdf\_json\_outputs
  + Expected .json files with schema format specified per each .pdf input

4\_qc\_and\_flagging.py

* dependencies
  + 3\_convert\_llm\_json.py ran
* Output
  + New folder generic\_json\_outputs
  + Text extracted .json files per each .pdf input

5\_build\_knowledge\_base.py

* dependencies
  + 4\_qc\_and\_flagging.py ran
* Output
  + New folder vector\_db