In [1]:

```python
import os
from dotenv import load_dotenv
from langchain.chat_models import ChatOpenAI
from langchain.prompts import PromptTemplate
from langchain.chains import LLMChain
from langchain.agents import Tool

# 🔐 Load API keys from .env
load_dotenv(dotenv_path=".env")
openai_api_key = os.getenv("OPENAI_API_KEY")

# 🔸 Step 1: Initialize the LLM
llm = ChatOpenAI(model="gpt-3.5-turbo", temperature=0)

# 🔸 Step 2: Create a prompt template
prompt = PromptTemplate.from_template("Answer clearly: {question}")

# 🔸 Step 3: Build the LLMChain
qa_chain = LLMChain(llm=llm, prompt=prompt)

# 🔸 Step 4: Wrap LLMChain as a Tool
qa_tool = Tool(
    name="Simple QA",
    func=qa_chain.run,
    description="A basic LLM chain that answers clear questions"
)

# 🔸 Step 5: Use the Tool directly
query = "What is LangGraph in LangChain?"
answer = qa_tool.run(query)

# 🖨️ Output
print("\nUser Question:", query)
print("\nTool Answer:", answer)
```

```
Tool Answer: LangGraph in LangChain is a graph database that stores and manag
es language data, including words, phrases, and their relationships. It is a
key component of the LangChain platform, allowing for efficient storage and r
etrieval of language information for various applications.
```

```
In [ ]:
```