

# **AI Agent Developer — Technical Test (2-Hour Limit)**

 Estimated Duration: 1.5–2 hours

## **Deliverables: GitHub repo or zipped folder with:**

- Working code
  - README with setup and brief explanation
  - (Optional) Notes on what you'd improve with more time
- 

## **Objective:**

Build a simple prototype that accepts a user question, retrieves relevant data, and uses a language model (LLM) to respond with a concise, context-aware answer.

---

## **Requirements:**

### 1. Input Interface

- A simple CLI or script that accepts a natural language question.  
Example: *"What's the property tax rate in Travis County?"*

### 2. Basic Retrieval Logic

- Load a small sample dataset (you may use a CSV, JSON, or array of text objects).
- Implement basic retrieval, either:
  - Keyword-based search, or
  - Vector similarity search using sentence-transformers or OpenAI embeddings.

### 3. LLM Integration

- Use an LLM (e.g., OpenAI API) to generate an answer using the retrieved context.
  - Prompt the model using the format:  
*"Answer the following question based only on the provided context."*  
Context: [...]  
Question: [...]
- 



#### **Example Use Case:**

Your script reads a dataset of local tax policies. The user asks a question. The agent retrieves the most relevant paragraph and generates a helpful, clear answer based on it.

---



#### **Bonus (Optional, Not Required):**

- Add a second sample dataset to demonstrate multi-source retrieval.
  - Add a follow-up mode for basic conversational memory.
- 



#### **Submission Instructions:**

- Send your completed test via GitHub repo or zip file.
- Include a short README.md with:
  - Setup instructions
  - A short summary of your approach
  - What you'd do next with more time