Assignment Tasks (What the Candidate Must Build)

Task 1: AI Agent Development (Core)

Build an AI agent that can:

- Accept a **user query**
- Decide whether:
    - It can answer directly using LLM, OR
    - It needs to fetch information from provided documents
- Return a **clear, structured response**

Requirements:

- Use **Azure OpenAI / OpenAI API**
- Implement:
    - Prompt engineering
    - Tool calling (at least one tool)
    - Basic agent memory (session-based is fine)

Example Use Case:

> "Answer user questions about a company's internal policy documents."

Task 2: RAG (Retrieval-Augmented Generation)

- Provide **3–5 sample documents** (PDF or text) yourself
  Example topics:
    - Company policies
    - Product FAQs

    - Technical documentation

Requirements:

- Convert documents to embeddings
- Store embeddings in:
    - Azure AI Search / FAISS / Pinecone (any one)
- Retrieve relevant chunks
- Pass them to the LLM as context

Task 3: Backend API

Create a backend service using **Python**:

- Framework: **FastAPI** (preferred)
- API Endpoint:

```
POST /ask
{
  "query": "string",
  "session_id": "optional"
}
```

Response:

```
{
  "answer": "string",
  "source": ["doc1", "doc2"]
}
```

Task 4: Azure Deployment (Mandatory)

Deploy the application on **Azure**:

- Azure App Service / Azure Functions (preferred)
- Use **Azure OpenAI**
- Environment variables for secrets
- Application should be accessible via a public URL

Bonus (Not Mandatory):

- Dockerized deployment
- Azure Monitor / basic logging

Task 5: Documentation

Provide a **README.md** with:

1. Architecture overview (diagram optional)
2. Tech stack used
3. Setup instructions (local + Azure)
4. Design decisions
5. Limitations & future improvements