ReAct Web Research Agent — Report

i. Reasoning with LLM: How the Reasoning Step Works

In this project, the **Gemini 1.5 Flash** model serves as the reasoning engine within the ReAct (Reasoning + Acting) architecture. The **reasoning step** is implemented during the **planning phase**, where the agent uses the LLM to generate relevant, structured research questions from a given user-defined topic.

The reasoning process works as follows:

- The user inputs a topic (e.g., "Artificial Intelligence in Education").
- A prompt is constructed and sent to the LLM asking it to generate 5–6 diverse research questions on the topic.
- The LLM returns well-structured questions that span various dimensions (ethical, pedagogical, technological, etc.), allowing for a comprehensive investigation.

This LLM-powered planning step enables the agent to **intelligently scope the research** before any action is taken, ensuring that the web search phase is purpose-driven rather than random.

---

ii. Code and Flow Explanation

The notebook is structured into modular and well-commented code blocks, following a **planning → acting → reporting** architecture. Here's the breakdown:

1. Installation and API Configuration

- Required Python libraries (`google-generativeai`, `tavily-python`) are installed.
- API keys for Gemini and Tavily are securely loaded from Colab `userdata`.

2. Planning (Reasoning with Gemini 1.5 Flash)

- Function: `generate_questions_gemini(topic)`
    - Uses the Gemini API to generate 5–6 relevant questions for a given topic.
    - Returns a list of questions as output.
    - Example: For "AI in Education", it may ask about equity, teacher roles, personalization, and long-term impact.

3. Acting (Web Search with Tavily)

- Function: `search_web(query)`
    - Sends each question to the Tavily API using the `search_depth="advanced"` setting.
    - Retrieves the top 3 results for each question with title and content.

4. ReAct Agent Class

- Class: `ResearchAgent`
    - `__init__(self, topic)`: Initializes with topic and empty lists for questions and answers.
    - `plan()`: Calls the Gemini function to generate research questions.
    - `act()`: Uses Tavily to gather search results for each question.
    - `generate_report()`: Compiles a Markdown report including all the questions and summarized answers.

5. Final Report Generation

- Report structure includes:
    - Title and Introduction
    - One section per research question

- Each section lists summarized insights from Tavily search results
- Conclusion

✅ Summary

This ReAct agent demonstrates how **reasoning (planning via LLM)** and **acting (via search tools)** can be combined in an agentic loop to automate web research tasks. The modular design makes it adaptable to any topic and scalable for more advanced workflows.