📌 Task Overview

1. **Analyze the provided log file to determine table structures.**
2. **Identify 'START' and 'END' blocks that define tables in the log file.**
3. **Figure out whether there is only one table or multiple tables in the file.**
4. **Use ChatGPT/Gemini to generate effective prompts for extracting table data.**
5. **Store the extracted tables in an Excel file with proper formatting.**
6. **Document the process of how patterns were identified and how extraction was performed.**
7. **Upload the project to a public GitHub repository.**
8. **Share the GitHub repository link with us.**

📝 Test Instructions

Step 1: Log File Analysis

- Inspect the given **log file (.txt)**.
- Identify **START** and **END** markers that define structured tables.
- Determine whether **one or multiple tables** exist in the file.
- Describe:
    - **How you identified the table structure** from patterns in the file.
    - **How many tables exist** in the log file.
    - **The column titles of each table** (based on observed patterns).

Step 2: Writing Prompts for ChatGPT

- Craft effective **ChatGPT prompts** to extract data from the log file.
- The prompts should help in:
    - Identifying and isolating data within **START and END** blocks.
    - Extracting relevant **columns and values** from each table.
    - Formatting the output as a structured table.
- **Modify and improve prompts** based on the complexity of the log file.

Step 3: Extracting Data & Converting to Excel

- Use **ChatGPT output** to extract table data accurately.
- Save the extracted tables in an **Excel file (.xlsx)**.
- Ensure:
    - Correct column headers.
    - Proper data formatting (no missing or misaligned values).

Step 4: Documenting the Extraction Process

- In a **README.md** file, describe:

- **How you identified START/END blocks** and determined table structures.
- **How many tables you found** and their **titles**.
- **How your code extracts data** effectively and why your approach works.
- **How you designed your ChatGPT prompts** and refined them for accuracy.

Step 5: GitHub Upload & Submission

- Create a **public repository** on **GitHub**.
- Upload:
  - 📄 **Log file (log_file.txt)**
  - 📄 **Extracted data (extracted_data.xlsx)**
  - 📄 **README.md (explanation of process & analysis)**
  - 📄 **Any script/code used (optional but recommended)**
- **Share your GitHub repository link** after completion.

📂 Expected Project Structure

```
📁 your-github-repo
│── 📄 log_file.txt          # Provided log file
│── 📄 extracted_data.xlsx    # Extracted tables in Excel
│── 📄 README.md              # Documentation (process, patterns, tables)
│── 📄 extraction_script.py   # Any Python code used for extraction
```

📌 Evaluation Criteria

1. **Pattern Identification** – Did you correctly analyze the log file structure?
2. **Prompt Quality** – Are the ChatGPT prompts well-structured and effective?
3. **Data Extraction Accuracy** – Is the extracted table data correct and well-formatted?
4. **Excel Organization** – Are multiple tables properly stored in separate sheets (if applicable)?
5. **Documentation Clarity** – Is the README file clear in explaining the process?
6. **GitHub Usage** – Is the repository well-organized and public?

📌 Deadline

- **Submission Deadline:** Sunday 2nd of March 2025
- **Submit GitHub Link:** Share your GitHub public project repository link.

Good luck! 🚀