## 

## **Key Enhancements for the Interactive Implementation**

1. **Continuous Session Loop**:
   * The main function now runs in a continuous loop until the user explicitly types 'exit' or 'quit'
   * Each iteration is tracked as a session with an incremental counter
2. **Context Preservation**:
   * The KnowledgeBase instance persists across all sessions
   * Session history (prompts and results) is stored in the knowledge base
   * This allows for potential reference to previous interactions
3. **Improved User Experience**:
   * Clear session boundaries with numbered sessions
   * Explicit options to continue or exit
   * Handling of empty prompt sets
4. **Preserved Concurrency**:
   * The original concurrent processing is maintained
   * Multiple prompts within a session are still processed in parallel

This implementation maintains all the asynchronous benefits of your current system while adding the continuous interaction capability. The core processing flow is preserved:

1. Users can still enter multiple prompts per session
2. All prompts are processed concurrently
3. Results are displayed in a consistent manner
4. The KnowledgeBase maintains state across sessions

In future enhancements, you could leverage the stored session history to implement features like:

* Referring to previous results ("use the result from my last request")
* Context-aware responses that build on earlier interactions
* Session management commands (list previous sessions, replay, etc.)

**Update General Knowledge Function**:

* + Replace the existing answer\_general\_question function in utils/general\_knowledge.py with the new context-aware version I provided.
  + This will enable the system to answer questions about previous sessions.

1. **Fix Location Parsing**:
   * Update the create\_comprehensive\_model function in core/functions\_registery.py with the fixed version I provided.
   * This will ensure locations are properly parsed and not split into individual characters.
2. **Knowledge Base Enhancement** (optional):
   * If you want to further enhance the knowledge base's ability to store information between sessions, you could add a method to explicitly save its state to disk and load it when the program starts.

## **How These Changes Work**

1. **Context Awareness**: The enhanced answer\_general\_question function:
   * Detects when a question is about previous sessions using keyword matching
   * Retrieves session history from the knowledge base
   * Extracts relevant information based on the question type
   * Provides this context to the LLM for a more informed response
2. **Location Parsing Fix**: The updated create\_comprehensive\_model function:
   * Properly handles both string and list inputs for locations
   * Normalizes location names by stripping whitespace and capitalizing
   * Handles comma-separated lists correctly
   * Applies the same normalization to generation types and carriers

## **Expected Results**

After implementing these changes:

1. When you ask about previous math questions like "what was the last math question?", the system will provide details from the previous session.
2. When creating energy models, locations will be properly processed as whole names (e.g., "France, Germany, Spain") instead of being split into individual characters.