AccuRanker

Time Series Data Analysis with ClickHouse

Description:
Develop a command-line (CLI) program that interacts with a ClickHouse database containing time series event data. Your program will retrieve data into a Pandas DataFrame, perform time-based aggregations and groupings, and then display the results in a neatly formatted table using PrettyTable.

Task Overview:

1. **Docker & ClickHouse Setup:**
    - Use Docker Compose to set up a ClickHouse instance.
    - On the first run, initialize the database by creating an `events` table with the following schema:
        - `event_time`: The timestamp when the event occurred.
        - `event_type`: A category label for the event (e.g., `'click'`, `'view'`, `'purchase'`).
        - `value`: A numerical metric associated with the event.
    - Populate the `events` table with sample data spanning at least **3 days**. It is up to you how to do this, as long as the following constraints are met:
        - There are multiple events per hour.
        - Various `event_type` values are represented.
        - The `value` field has non-uniform data to allow meaningful aggregation.

2. **Data Retrieval & Transformation:**
   In your CLI program, perform the following steps:
    - **Data Retrieval:**
        - Connect to ClickHouse eg using the clickhouse_connect library.
        - Query *all* data from the `events` table and load it into a Pandas DataFrame.
        - Parse the `event_time` column as a datetime object.
    - **Time-based Aggregations:**
        - **Hourly Aggregation:**
          Resample the data to compute for each hour:
            - Total number of events.
            - Sum of the `value` field.
            - Average of the `value` field.
        - **Event Type Grouping:**
          Group the data by `event_type` (regardless of time) to compute:
            - Total count of events for each type.
            - Total sum of the `value` field for each type.

3. **Output Formatting:**
    - Use PrettyTable to print the above-aggregated metrics.
    - Ensure the tables are well-formatted and easy to read

4. **Program Execution & Exit:**
    - The CLI program should run once, perform the data processing and output the tables, then exit gracefully.

Technical Requirements:

- The solution must be written in Python.
- Use the following libraries (additional libraries are allowed if necessary):
    - Pandas
    - PrettyTable
- The entire application (both ClickHouse and your CLI program) should be orchestrated via Docker Compose.
- Your Docker Compose setup should ensure that the CLI program starts only after ClickHouse is confirmed healthy.