Project Title

"Stack Overflow Post Analysis: A SQL Portfolio Project"

Dataset:
https://www.kaggle.com/datasets/stackoverflow/stackoverflow/data?select=post_history

Project Objective

The objective is to analyze the history of Stack Overflow posts, including edits, comments, and other changes, to gain insights into user activity and content evolution while mastering SQL skills.

Tables in the Dataset

1. **`badges`**
    - Tracks badges earned by users.
    - **Key Fields:**
        - `id`, `user_id`, `name` (badge name), `date` (earned date).
2. **`comments`**
    - Contains comments on posts.
    - **Key Fields:**
        - `id`, `post_id`, `user_id`, `creation_date`, `text`.
3. **`post_history`**
    - Tracks the history of edits, comments, and other changes made to posts.
    - **Key Fields:**
        - `id`, `post_history_type_id`, `post_id`, `user_id`, `text`, `creation_date`.
4. **`post_links`**
    - Links between related posts.
    - **Key Fields:**
        - `id`, `post_id`, `related_post_id`, `link_type_id`.
5. **`posts_answers`**
    - Contains questions and answers.

- **Key Fields:**
  - `id`, `post_type_id` (question or answer), `creation_date`, `score`, `view_count`, `owner_user_id`.

6. **`tags`**
   - Information about tags associated with posts.
   - **Key Fields:**
     - `id`, `tag_name`.
7. **`users`**
   - Details about Stack Overflow users.
   - **Key Fields:**
     - `id`, `display_name`, `reputation`, `creation_date`.
8. **`votes`**
   - Tracks voting activity on posts.
   - **Key Fields:**
     - `id`, `post_id`, `vote_type_id`, `creation_date`.
9. **`posts`**

Tasks and Concepts

Part 1: Basics

1. **Loading and Exploring Data**
   - Explore the structure and first 10 rows of each table.
   - Identify the total number of records in each table.
2. **Filtering and Sorting**
   - Find all posts with a comment_count greater than 2
   - Display comments made in 2012, sorted by `creation_date` (`comments` table).
3. **Simple Aggregations**
   - Count the total number of badges (`badges` table).
   - Calculate the average score of posts grouped by `post_type_id` (`posts`_answer table).

Part 2: Joins

1. **Basic Joins**
    - Combine the `post_history` and `posts` tables to display the `title` of posts and the corresponding changes made in the post history.
    - Join the `users` table with `badges` to find the total badges earned by each user.
2. **Multi-Table Joins**
    - Fetch the titles of posts (`posts`), their comments (`comments`), and the users who made those comments (`users`).
    - Combine `post_links` with `posts` to list related questions.
    - Join the `users`, `badges`, and `comments` tables to find the users who have earned badges and made comments.

Part 3: Subqueries

1. **Single-Row Subqueries**
    - Find the user with the highest reputation (`users` table).
    - Retrieve posts with the highest score in each `post_type_id` (`posts` table).
2. **Correlated Subqueries**
    - For each post, fetch the number of related posts from `post_links`.

Part 4: Common Table Expressions (CTEs)

1. **Non-Recursive CTE**
    - Create a CTE to calculate the average score of posts by each user and use it to:
        - List users with an average score above 50.
        - Rank users based on their average post score.
2. **Recursive CTE**
    - Simulate a hierarchy of linked posts using the `post_links` table.

Part 5: Advanced Queries

1. **Window Functions**
    - Rank posts based on their score within each year (`posts` table).
    - Calculate the running total of badges earned by users (`badges` table).

New Insights and Questions

- Which users have contributed the most in terms of comments, edits, and votes?
- What types of badges are most commonly earned, and which users are the top earners?
- Which tags are associated with the highest-scoring posts?
- How often are related questions linked, and what does this say about knowledge sharing?

Deliverables

- A report or notebook containing:
    - SQL scripts for each task.
    - Key insights derived from the queries.
    - Visualizations (if using tools like Tableau or Power BI). (Optional)