CS 485 01 – Fall 2019
Group Project
Assigned: November 12, 2019
Due: December 5, 2019, on the server at class time

Note: You can do this project as a team of 2 students (you pick your partner) or you can do it alone. If you work as a team, one of the two team members needs to email me about the members of the team and which account is used for the project.

You need to implement a form of NoSQL using MySQL (!).

1 – (10 points) create a table named threads with 3 columns: id, tags, thread. The data types are as follows:

- **id is an auto increment int. It is also the primary key.**
- **tags is a string of 50 or fewer characters.**
- **thread is a string that can have up to 4GB characters (you will need to google what data type that translates into)**

2 – (30 points) Using an HTML form (insertThread.html) and a corresponding PHP script (insertThread.php), allow the user to start a thread with the following data:

Body of the post (this is a user input in your HTML form)
Name of the user (this is a user input in your HTML form)
Up to 5 tags (these are user input in your HTML form)

Insert the thread in the threads table:
As you know, the id column's type is auto increment.
In the tags column, the value is a JSON array made up of the tags entered by the user
In the thread column, the value is a JSON array made up of one object (since this is the first post of a thread). The object should have the following keys (and appropriate values): body, name (of user), datePosted (you can retrieve it in your PHP)

2 – (30 points) Using an HTML form (insertPost.html) and a corresponding PHP script (insertPost.php), allow the user to post in a thread with the following data:

Body of the post (this is a user input in your HTML form)
Name of the user (this is a user input in your HTML form)
Id of the thread (obviously, it needs to be in the table; otherwise, there would be a primary key constraint violation)

Insert the post inside the corresponding thread in the threads table:
In the thread column, you need to "insert" the post so that you still have a valid JSON array (with valid JSON objects inside)

4 – (30 points) Using an HTML form (selectThreads.html) and a corresponding PHP script (selectThreads.php), allow the user to retrieve threads based on a search tag entered by the user (this is a user input in your HTML form):

Your PHP script should retrieve the appropriate threads (no duplicate) and display them in an HTML table. The HTML table should have as many columns as there are threads retrieved.

For only 20 points: the HTML table only has 2 rows, one for the thread id and one for the whole thread changed as follows: delete the outside square brackets.

For the full 30 points: Each row of the HTML table should show a single post (no curly braces, no commas). Table headers should be the thread ids.