

SQL CONNECT FORM

Vishal Kumar Sah¹, Shubham Banerjee², Vishal Deo Mahto³, Akshay U⁴, Ms. Vijayalakshmi.K⁵

¹B. Tech Student, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Tamilnadu, India

²B. Tech Student, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Tamilnadu, India

³B. Tech Student, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Tamilnadu, India

⁴B. Tech Student, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Tamilnadu, India

⁵Assistant Professor, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Tamilnadu, India

ABSTRACT

The project My SQL connect form is one of the most powerful MySQL manager and an administration tool which combines the feature of SQL browser, administrator, PHP and other MySQL front end features. It is a complete Unicode support and has table diagnostics and also has the provision of dropping a database. There is also presence of a multitab query editor and result set editor. Administer MySQL accounts as easily as a toy. Create, modify databases and visually create and edit tables, create exact copies of an existing table, within the same database or in another one. Table contents are editable in a grid, switchable columns and sorting orders. The results of various data grids are saved to HTML and XML files. It holds various filtering mechanisms. Effort has been spent on holding numerous MySQL queries as options to implement. These queries are accessed by dragging or selecting the queries from a drop down section. These components along with several easy queries make up the GUI for of the connect form. The system also allows for creating multiple tables at once, adhering by the concurrent data property of database management. Before creating the GUI, different strategies have been formulated and analysis of alternative solution strategies were compared. The final GUI is the simplified version of all its predecessors. Java is the primary language used for creating the GUI. Several in-built Java functions have made the formation of this connect form easier. Although user-defined functions are used to maintain the integrity of our application.

Keyword : - SQL, JAVA, GUI, MySQL, Table

1. INTRODUCTION

The MySQL GUI is the most powerful MySQL manager and admin tool, combining the features of MySQL Query Browser, Administrator, phpMyAdmin and various other MySQL Front Ends.

- Complete Unicode/UTF8 Support
- MySQL 5.x objects support
- Relationship/Foreign Key Manager
- Table diagnostics
- Creating/dropping database
- Creating/Dropping/Altering tables
- Multi-tab Query Editor and Result-set Editor
- Multiple Query Execution

It allows you to manage and browse your databases and tables from an intuitive Windows and Linux interface. One of the most wanted, needed features was a GUI for creating and editing triggers, available in MySQL 5.0.2 and above. Now implemented and hopefully as usable as it can be. Very simple compared to editing tables or procedures. The recently refactored "Table tools" dialog has a new tab now: "Bulk table editor". What is bulk editing? Move all tables of one or more databases to another database. Create and edit stored procedures and functions. The very first dialog which asks for hostname and credentials. The list of wanted databases can be limited and sorted alphabetically.

Administer MySQL accounts as easily as a toy. A pull-down showing assigned database objects reveals everything, details about tables and views. Switchable column headers is used to rename, delete, empty and modify tables. Create, modify databases and visually create and edit tables, create exact copies of an existing table, within the same database or in another one. Table contents are editable in a grid. Switchable columns and sorting orders. Various filtering mechanisms. SQL query editing: Customizable syntax highlighting, snippet files, completion proposal pull-down and editable results. Pump database structures and data into a single dump file, one file per table or directly to another host. Save the result of any data grid to HTML and XML files. Find text in a table, a database or even on the whole server. Send any list of tables, columns, variables etc. to your printer. Browse through the topics in MySQL's built-in help system. Create and edit stored procedures and functions.

2. EXISTING AND PROPOSED SYSTEM

2.1 Existing System

In the present scenario, for database transactions the queries need to be written manually. Also multiple queries cannot be executed at a time. The creation of triggers, stored procedures are not in an easier way in the existing system. No import or export facility is available in existing system. So this existing system cannot satisfy the customer needs. Most of these sites are mainly focused on limited databases.

2.2 Proposed System and its advantages

In our proposed system the creation of databases and data transactions are simpler compared to the existing system. Here the creation of stored procedures, functions and triggers can be done in an easier way.



Fig -1: GUI Samples

The advantages of proposed system are:

- GUI for creating and editing triggers
- Create and edit stored procedures and functions
- Create exact copies of an existing table, within the same database or in another one.
- Table contents editable in a grid. Switchable columns and sorting orders. Various filtering mechanisms
- Pump database structures and data into a single dump file, one file per table or directly to another host. Save the result of any data grid to HTML and XML or PDF files
- Send any list of tables, columns, variables etc. to your printer.

2.2 FEASIBILITY STUDY :

The main aim of feasibility study is to determine whether it would be functionally and technically feasible to develop the product. The feasibility study involves the analysis of the problem and the collection of relevant information relating to the product such as the different data item which would be the input to the system, the processing required to be carried out on these data, the output data required to be produced by the system, as well as various constraints on the behavior of the system. A feasibility study is a rest of the system proposal according to its working, impact on the organization, ability to meet users and effective use of resources. The objective of the feasibility study is acquiring the sense of scope of the system.

The collected data are analyzed to arrive following:

- An abstract problem definition: An abstract problem definition is a rough description of the problem, which considers only the important requirements and ignores the rest.
- Formulation of different strategies.
- Operational Feasibility
- Technical Feasibility
- Economical Feasibility

3. SYSTEM ARCHITECTURE

The architecture of Microsoft SQL Server is broadly divided into three components: SQLOS which implements the basic services required by SQL Server, including thread scheduling, memory management and I/O management; the Relational Engine, which implements the relational database components including support for databases, tables, queries and stored procedures as well as implementing the type system; and the Protocol Layer which exposes the SQL Server functionality.

3.1 RELATION ENGINE

The relational engine implements the relational data store using the capabilities provided by SQLOS, which is exposed to this layer via the private SQLOS API. It implements the type system, to define the types of the data that can which is the component that retrieves data. SQL queries specify what data to retrieve, and the query processors optimizes and translates the query into the sequences of operations needed to retrieve the data .The operation are then performed by worker threads, which are scheduled for execution by SQLOS.

3.1 DATA STORAGE

The main units of data storage is a database , which is a collection of tables with typed columns .SQL Server supports different data types, including primary types such as Integer, Float, Decimal, Char (including character strings), Varchar (variable length character strings), binary (for instructed blobs of data),Text(for textual data) among others. It also allows user-defined composite types (UDTs) to be defined and used.

SQL Server also makes server statistics available as virtual tables and views (called Dynamic Management Views or DMVs). A database can also contain other objects including views, stored procedures, indexes and constraints, in addition to tables, along with a transaction log. An SQL Server database can contain a maximum of 2A3 I objects, and can span multiple OS-level files with a maximum filesize of 2A2OTB

3.1 DATA RETRIEVAL

The main mode of retrieving data from an SQL Server database is querying for it. The query is expressed using a variant of SQL called T-SQL, a dialect Microsoft SQL Server shares with Sybase SQL Server due to its legacy. The query declaratively specifies what is to be retrieved. It is processed by the query processor, which figures out the sequence of steps that will be necessary to retrieve the requested data. The sequence

of actions necessary to execute a query is called a query plan. There might be multiple ways to process the same query. For example, for a query that contains a join statement and a select statement, executing join on both the tables and then executing select on the results would give the same result as selecting from each table and then executing the join, but result in different execution plans.

The overall system design contains the following modules:

3.1.1 Sign in module

This module will allow user to login to SQL connect form where he/she can do further work. User are asked for username and password while installing and while opening the app it will ask for Username and password.



Fig -2: Sign In modules

3.1.2 Main Form

It is the first screen after opening the after opening the application. Here user can select what type of data he/she wants to create.

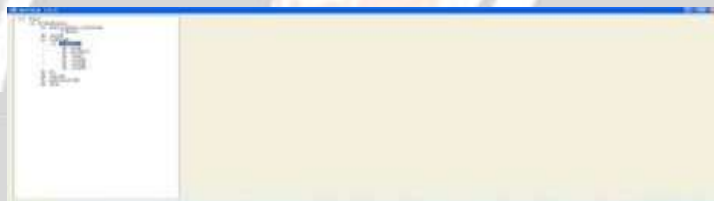


Fig -3: Main Form

3.1.3 Create Database form:

Here users are asked to enter the filename by which data to be saved.



Fig -4 : Create Database form

3.1.4 Create Table form:

In this module user are asked to create column and datatypes . All are done graphically no needs of coding skills.



Fig -5: Create Table form

3.1.5 Query Builder

In this module user needs to enter all data that he wants to manage. Rows are filled according to users.

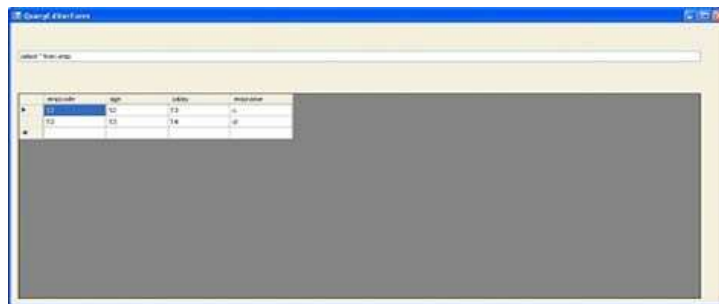


Fig -6: Query Builder

3.1.6 Import form

Using this we can import this file to any other like XML.



Fig -7: Import form

4. REQUIREMENT ANALYSIS

4.1 Hardware Requirement

- Processor : Pentium IV
- Memory : 128RAM20 GB HDD
- Monitor : 14VGA color monitor
- Mouse : 2/3button
- Keyboard : 104 keys

4.2 Software Requirement

- Windows XP or above
- .NET platform

- MYSQL 4 or above
- Visual C 2005.NET
- Microsoft Visual Studio 2005.NET

4. CONCLUSIONS

The project was completed within the time span allotted. Every effort has been made to present in the system in more user friendly manner. All the activities provide a feeling like an easy walk over to the user who is interfacing with the system. A trial run of the system has been made and is giving good results.

The software has been developed in .NET and a very user friendly GUI is designed using C. All the modules are tested separately and put together to form the main system in an attractive fashion. So, users with minimum knowledge about the computers operates easily.

As for the efficiency of the application several tests have been made concerning the time factor of the results of the queries. Any of the other previously known bugs, like crashing of the GUI, server creation error and more were also reported in these test runs. At the end, we can assure that the presence of bugs in this application is to a minimal. Assurance is also provided for the smooth running and execution of each queries and their resultant table formation

5. ACKNOWLEDGEMENT

Authors would like to warmly thanks Ms Vijayalakshmi K.,Assistant Professor(O.G),Dept. Of Computer Science And Engineering,SRM Institute of Science and Technology for her great cooperation during the development of this project.

6. REFERENCES

- [1] Fazal Mithani,Sahista Machchhar, Fernaz Jasdanwala, A research paper on ,Non First Normal Form Relations and Recursive Queries: IBM Scientific Center Heidelberg,Germany
- [2] Dr Anne Moorhead and Dr Raymond bond, A research paper on ,A Novel Approach for SQL Query Optimization ,MEF Group of Institutions Rajkot,India
- [3] Elena Baralis, Andrea Dalla Valle, Paolo Garza,Claudio Rossi and Francesco Scullino, A research paper on ,SQL versus NoSQL Databases for Geospatial Applications ,I Istituto Superiore Mario Boella (ISMB) Torino, Italy
- [4] Dejan Dundjerski, Stefan Lazi, Milo Tomaevi, and Dragan Boji, A research paper on ,Improving Schema Issue Advisor in the Azure SQL Database Serbia, Belgrade,
- [5] Yun Guo, A research paper on ,Localizing and Fixing Faults in SQL Predicates George Mason University