

# QUERY ANALYZER FOR SQL SERVER USING MSDE

PROJECT WORK DONE AT

Logic Software Solutions Pvt. Ltd.

## PROJECT REPORT

SUBMITTED IN PARTIAL FULFILLMENT OF THE  
REQUIREMENTS  
FOR THE AWARD OF THE DEGREE OF  
**MASTER OF COMPUTER APPLICATIONS**  
OF BHARATHIAR UNIVERSITY, COIMBATORE.

SUBMITTED BY

Sumy Thomas  
Reg.No.9938M0647

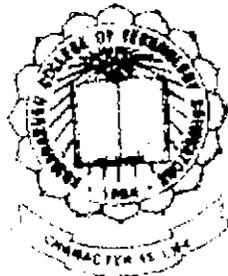
GUIDED BY

### EXTERNAL GUIDE

Mr. Jojan Antony MCA.,  
Project Manager,  
Logic Software Solutions Pvt.Ltd,  
Cochin.

### INTERNAL GUIDE

Mrs. N. Chithra Devi ME.,  
Lecturer,  
Department of Computer Science & Engg.  
Kumaraguru College of Technology,  
Coimbatore.



Department of Computer Science & Engineering  
**KUMARAGURU COLLEGE OF TECHNOLOGY**  
Coimbatore-641006

MAY 2002

---

---

***CERTIFICATE***

---

---



**LOGIC SOFTWARE SOLUTIONS PVT. LTD.**

Towers, St. Albert's College Lane, Banerji Road, Cochin - 682 018, Kerala, India.  
Tel: 484-393141, 393148, 398448, Fax: 91 - 484 - 396201

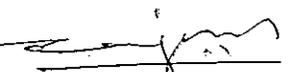
Email: [lss@eth.net](mailto:lss@eth.net), [www.lssindia.com](http://www.lssindia.com)



## **CERTIFICATE**

This is to certify that the project titled "**Query Analyzer for SQL Server using MSDE**" is a bonafide record of the project work done by **Ms. Sumy Thomas, MCA** student of **Kumaraguru College Of Technology, Coimbatore** during the period **December 2001 to April 2002**.

The project was successfully completed and tested for its functionality at **LOGIC SOFTWARE SOLUTIONS PVT. LTD.**

  
Jojan Antony  
Project Manager

  
Biju Joseph Jacob  
Director



Department of Computer Science & Engineering  
**KUMARAGURU COLLEGE OF TECHNOLOGY**  
(Affiliated to the Bharathiar University)  
Coimbatore-641006

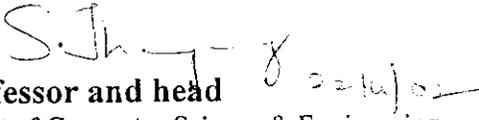
## CERTIFICATE

This is to certify that the project work entitled  
**QUERY ANALYZER FOR SQL SERVER USING MSDE**

Done by

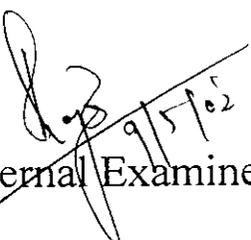
Sumy Thomas  
9938M0647

Submitted in partial fulfillment of the requirements for the award of the  
degree of  
**Master of Computer Applications of Bharathiar  
University.**

  
Professor and head  
Department of Computer Science & Engineering

  
Internal Guide

Submitted to university examination held on 9-05-2002

  
Internal Examiner

  
External Examiner

DECLARATION

# DECLARATION

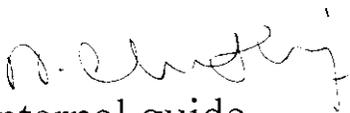
I here by declare that the project entitled 'QUERY ANALYZER FOR SQL SERVER USING MSDE ' submitted to **Bharathiar University** as the project work of Master of Computer Applications Degree, is a record of original work done by me under the supervision and guidance of **Mr. Jojan Antony, Project Manager, Logic Software Solutions Pvt.Ltd. , Cochin** and **Mrs. N Chithra Devi ME, Lecturer, Kumaraguru College of Technology, Coimbatore** and this project work has not found the basis for the award of any Degree /Diploma/ Associateship/ Fellowship or similar title to any candidate of any university.

Place: COCHIN

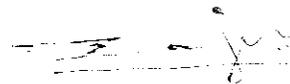


Sumy Thomas

Date: 18-04-2002



Internal guide



External Guide

*Dedicated to  
My parents , Teachers  
& Friends*

---

---

***ACKNOWLEDGEMENT***

---

---

## ACKNOWLEDGEMENT

*I am incredibly fortunate to have so many people helping me on this project. First I wish to thank **Mr. Jojan Antony**, Project Manager, Logic Software Solutions Pvt.Ltd. , Cochin for entrusting this project to me and providing all facilities that made the experience a pleasant one. In particular my deepest thanks goes to **Mr. Biju Joseph Jacob** , **Director**, for recognizing and approving this project and for his immense support, timely suggestions, and technical guidance throughout the project work.*

*I am extremely thankful To **Dr.S.Thangasamy**, Professor and Head of the Department for his kind suggestion he has given in every step throughout our studies and in this project work. I am grateful to my internal guide **Mrs.N Chithra Devi ME**, Lecturer, **Kumaraguru College Of Technology** who offered his guidance and always supported me with keen interest and constant encouragement suggestions in every step through out the project.*

*I wish to thank my parents for all their encouragement support and love. Without their encouragement I would never have gotten where I am.*

*Last but not the least I would like to thanks my Classmates and friends for their love and moral support. Finally thanking You God for your immense support and love*

---

---

*SYNOPSIS*

---

---

# SYNOPSIS

The project entitled 'QUERY ANALYSER FOR SQLSERVER USING MSDE' has been developed for Logic Software Solutions Pvt .Ltd., Cochin.

Query Analyzer is an interface for the Microsoft SQL Server. Using the Query Analyzer we can manage the SQL queries. The draw back in the existing system is that we should install Microsoft SQL Server in each and every client machine, which is very costly.

The new system, Query Analyzer is a visual interface for MSDE (Microsoft Data Engine) which is a freely supplied data engine by Microsoft. The new system, QUERY ANALYSER helps the user to manage the SQL Queries in a simple, reliable and efficient manner without installing MS SQL Server in the client machine of an Intranet. It uses queries to manage the SQL Server. In this system, we need to install SQL Server in any one of the machines in the Intranet. And the MSDE must be installed in each and every client machine. With the help of the MSDE, a client located anywhere in the Intranet can be connected to the server and thereby we can reduce the cost of installing the costly licensed SQL Server.

The project is divided into 6 modules namely Connection Module, Database Listing Module, Tables/Views/Stored Procedures Listing Module, Query Execution Module, Showing Execution Plan Module, and Parsing Module.

The system has been implemented in Visual C++ on working under the platform Windows NT.

---

---

## *CONTENTS*

---

---

# CONTENTS

Page No.

## 1. Introduction

- 1.1 Project Overview 1
- 1.2 Organization Profile 4

## 2. System Study and Analysis

- 2.1 Introduction 5
- 2.2 Existing System 6
- 2.3 Drawbacks Of The Existing System 6
- 2.4 Proposed System 7
- 2.5 Advantages Of Proposed System 7
- 2.6 Comparative System Study 8
- 2.7 Requirements on New System 8

## 3. Programming Environment

- 3.1 Hardware Configuration 9
- 3.2 Software Specification 9
- 3.3 Reasons for Choosing Software 10

## 4. System Design & Development

- 4.1 Design Methodology 11
- 4.2 Input Design 11
- 4.3 Output Design 15
- 4.4 Process Design 17
- 4.5 System Architecture 23
- 4.6 Flowcharts for Modules 24

## 5. System Implementation and Testing

- 5.1 System Implementation 30
- 5.2 System Testing 32

<b>6. Conclusion</b>	<b>34</b>
<b>7. Scope for future development</b>	<b>35</b>
<b>8. Bibliography</b>	<b>36</b>
<b>9. Appendices</b>	

---

---

## *INTRODUCTION*

---

---

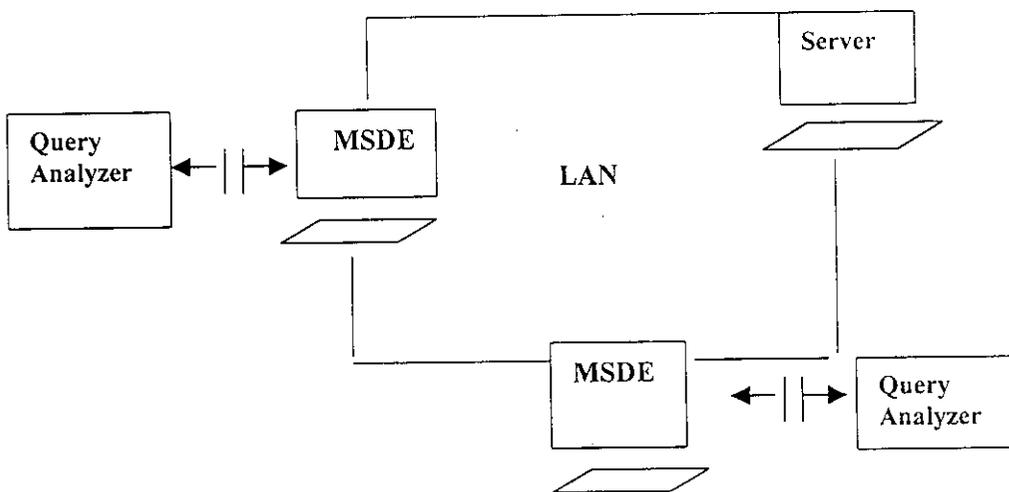
# 1.Introduction

## Project Overview

QUERY ANALYZER is a user interface of SQL Server. Usually each client machine should have a licensed SQL server, which has the interface Query Analyzer. But it is very costly. So Microsoft is now providing us with a free database engine called MSDE.

QUERY ANALYZER helps the user to manage the SQL Queries without installing MS SQL Server in the client machine of an Intranet in a simple, reliable and efficient manner. It uses queries to manage the SQL Server. By using queries we can perform any operations like creating databases, tables, stored procedure etc. MSDE Query Analyzer is an interface provided to MSDE (Microsoft Data Engine), which is used to connect to the server. With the help of MSDE we can connect to the Server from any of the Client Machines and thereby we can reduce the cost of installing the costly licensed SQL Server.

The picture below shows the Query Analyzer interface for an Intranet.



## **The product functions are:**

- Edit and author SQL queries.
- Execute multiple Transact-SQL scripts.
- Return query results to a text.
- Checks the syntax of the query statements
- Shows the execution plan of the query statements
- Shows the status information of the server connected

## **Advantages of Query Analyzer are:**

- Without installing MS SQL Server Client in any machine of an Intranet we will be able to manage the SQL Server.
- Functionality of the Query Analyzer can be performed on the client machine by installing MSDE.
- MSDE is free licensed software, so the cost of installing the clients is made null.
- Consumption of space is reduced to megabytes.
- This system is simple, reliable and efficient.

## **The main inputs acquired from the user are as follows.**

- SQL Server name
- User name and password: - Information required to logon to the server.
- Queries to execute

**The main outputs, which are to be handled in this soft ware, are as follows.**

- Display the name of all databases present in that particular server.
- Display all tables, stored procedures and views in the each database.
- Display the fields in each table and data stored in each fields.
- Display the forms for creating the database, tables, stored procedures, Views etc.
- Display the forms for connection and disconnection to and from the Server.

MSDE databases running on windows NT/ windows 2000 platforms are capable of utilizing windows 2000-integrated security to protect database objects. MSDE allows you to secure individual database objects such as tables or stored procedures this enhances database security.

## **Organizational Profile**

LOGIC is Cochin based Information Technology consulting firm specializing in the design, development and integration of custom-built business systems and solutions using advanced internet, electronic commerce and client server technology.

Since 1996, it has built a solid reputation as an enterprise dedicated to offering a complete and sound approach to systems development in the emerging paradigm of the networked world.

The clients are industry leaders who pursue innovative ways of exploiting information technology to create, manage and distribute information to support their business strategies. They seek a new approach to the delivery of custom solutions, backed by a team of dynamic and competent professionals. They want a partner with the capacity to effectively develop new systems to realize their corporate objectives whether they be developing stronger customer relations, leveraging human capital, or improving business efficiency.

The management team is structured to fully support and guide the clients through their information system initiatives. With their solid experience in the field of systems consulting, development and integration, the managers understand the human and business aspects of information technology and systems development as well as the technical issues. This assures the clients of a greater chance of achieving the desired results.

---

*SYSTEM STUDY AND  
ANALYSIS*

---

## **2.System Study and Analysis**

### **Introduction**

The objectives of the system analysis phase are the establishment of the requirements for the system to be acquired, developed and installed. Analyzing the project to understand the intricacy forms the vital part of the system study. Problematic areas are identified and information is collected. Fact finding or gathering is essential to analysis of requirements. It is necessary that the analyst familiarize himself with the objectives, activities and the functions of the organizations in which the system is to be implemented.

It involves in studying the ways an organization currently retrieves and processes data to produce information with the goal of determining how to make it better. System analysis includes investigation and possible changing to the existing system. System Analysis is for finding out what happens in the existing system, deciding on what changes and new features are required and defining exactly what the proposed system must be. The process of system analysis is largely concerned with determining, developing and agreeing to the user's requirements. It provides prime opportunity to communicate well with the user and conceive a joint understanding of what a system should be doing, together with a view of the relative importance of the system facilities using interactive techniques.

## **Existing System**

At present in intranet we need MS SQL Server in client side machines to control the server side MS SQL database. Since SQL Server package is very costly, It is very difficult to install it in each client machine. Moreover in most of the organization there is only one SQL Server. This machine stores all data required by the programs. Execution of queries takes place in this machine and only the results are passed to the client machine. Therefore time-sharing system is being used. This leads to the delay of the programs and hence they have to wait in a queue till their chance comes. In the existing system, we have to work on each and every client machine to access its SQL database.

## **Drawbacks Of the Existing System**

- Existing system needs costly licensed SQL Server in every client machine.
- Excessive usage of memory space
- High complexity

The major draw back in existing system is that for accessing SQL database of each machine we should work on the respective machine. Thus increasing the workload and effort.

## **Proposed System.**

MSDE Query Analyzer is used to manage the SQL Server without installing the MS SQL in client machines.

**This component has the major functions like: -**

- Edit and author SQL queries.
- Execute multiple Transact-SQL scripts.
- Return query results to a text.
- Checks the syntax of the query statements
- Shows the execution plan of the query statements
- Shows the status information of the server connected

## **Advantages Of Proposed System**

- Without installing MS SQL Server Client in any machine of an Intranet we will be able to manage the SQL Server.
- Functionality of the Query Analyzer can be performed on the client machine by installing MSDE.
- MSDE is free licensed software, so the cost of installing the clients is made null.
- Consumption of space is reduced to megabytes.
- This system is simple, reliable and efficient.

The major advantage of the new system is that we can access the SQL database of the server from any client machine. Thus decreasing the workload.

---

*PROGRAMMING  
ENVIRONMENT*

# 3.Programming Environment

## Hardware Configuration

The hardware configuration for MSDE Query Analyzer to perform smooth running is as

- **Server**

Processor : Pentium II, or above  
Memory : 64 MB RAM  
Hard Disk : 1.2 GB or above  
Monitor : 14" SVGA Monochrome or Above

- **Client**

Processor : Pentium II, or above  
Memory : 64 MB RAM  
Hard Disk : 1.2 GB or above  
Monitor : 14" SVGA Monochrome or Above

## Software Specification

The following are the list of software required

### Server

MS SQL Server 7.0 or Greater

Windows NT 4.0/ 2000

Programming Language Used: Microsoft Visual C++ 6.0

### Client

MSDE

Windows NT 4.0 / 98/ 2000

## **Reasons for choosing Visual C++**

- Visual C++ is an extremely powerful tool for Windows programming. Visual C++ is not actually one tool, it is a collection of tools, wrapped together into one dynamic, ready to use package.
- Using Visual C++ we can create small and efficient programs. A program, which is written using this language, can operate at nearly the same speed as one written in assembler, without any problems.
- One of the important features that Microsoft has added is better prototyping capability through the use of enhanced wizards. it reduce the time required to get an application started.
- Visual C++ is also an ideal programming environment for writing ActiveX controls, along with IIS-specific code like ISAPI (Internet Sever Application Programming Interface) extensions and ISAPI filters.
- Visual C++ not only makes the use of MFC, the Microsoft Foundation Class library, which is an extra ordinary package of prewritten, ready to use code but also makes Windows programming far easier by introducing many programming tools, such as menu editor for designing menus, and the dialog editor for designing the dialog boxes.
- Visual C++ provides us with one integrated design environment in which we can write our programs and run them.

---

**SYSTEM DESIGN AND  
DEVELOPMENT**

## 4. System Design and Development

### Design Methodology

System design is vital for efficient database management. In the present software, conceptualization of the system involves form layouts for inputs and reports for the output. Based on the proposed system objectives, the major modules are identified and the operations to be carried out are determined. In the design phase of this project inputs, outputs and screens are designed.

### Input Design

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data into a usable form for processing data entry. The activity of putting data into the computer for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling errors, avoiding delay, avoiding extra steps and keeping the process simple.

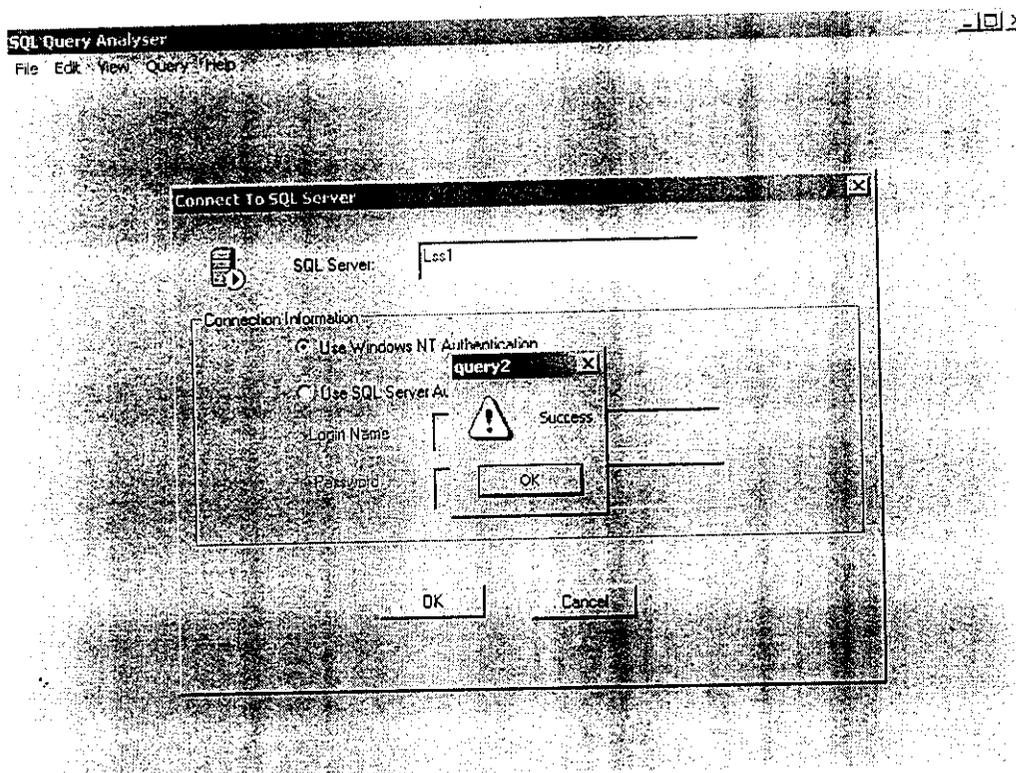
In Input design the following things are considered.

- What data should be given as input?
- How the data should be arranged or coded?
- The dialogue to guide the operating personnel in providing input
- The dialogue to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

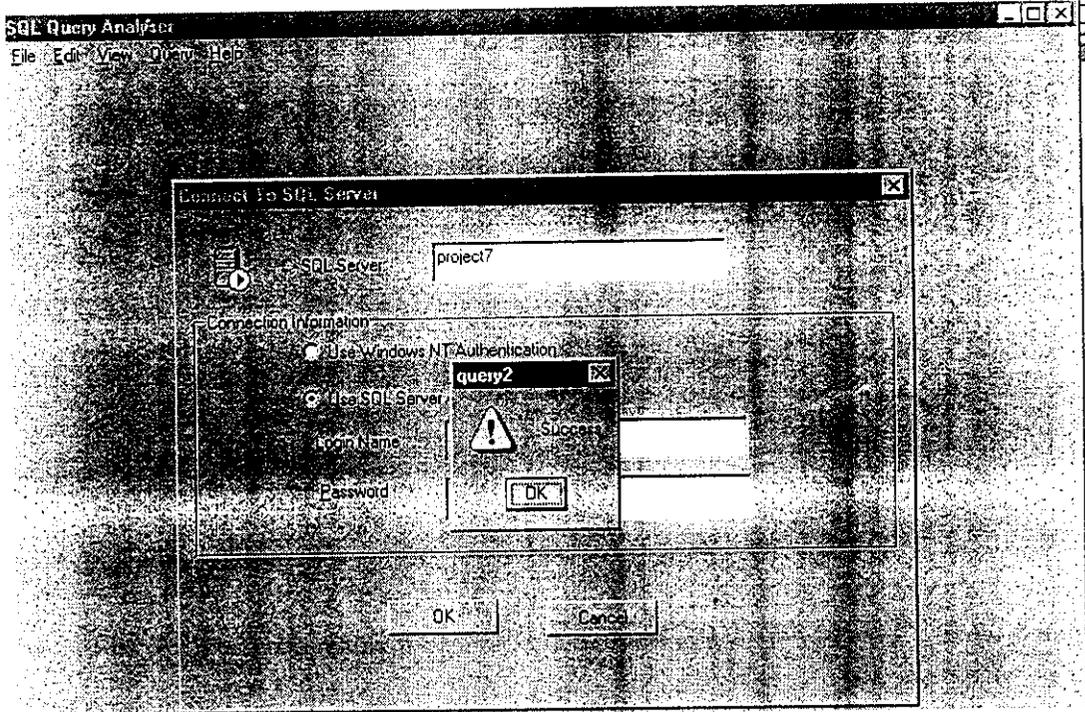
**The main inputs acquired from the user are as follows.**

- SQL Server name--Name of the SQL Server
- User name and password--Information required to logon to the server.
- Queries to execute

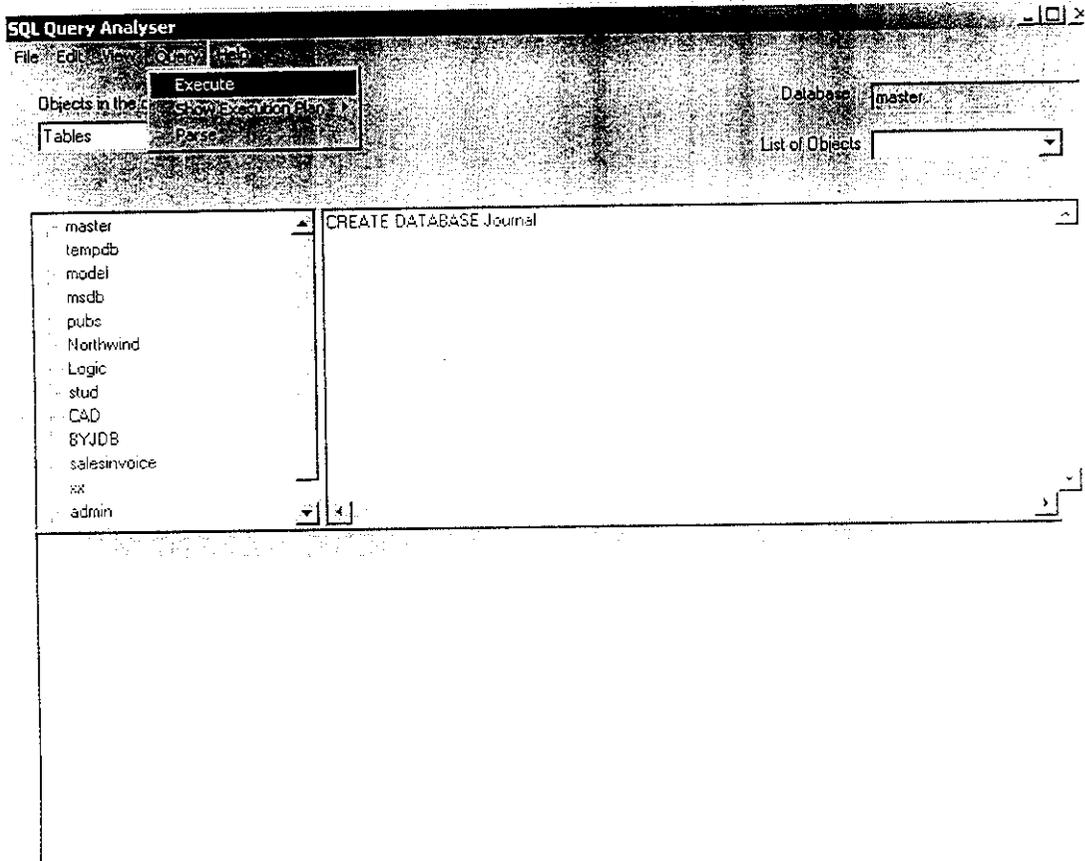
## Connection using the Windows NT authentication



# Connection using the SQL Server authentication



# Inputting the Query



## Security

MSDE databases running on Windows NT / windows 2000 platforms are capable of utilizing windows 2000-integrated security to protect database objects. MSDE allows you to secure individual database objects such as tables or stored procedures this enhances database security.

## Output Design

Computer output is the most important and direct information source to the user. Output design is a process that involves designing necessary outputs in the form of reports that should be given to the users according to the requirements. Efficient, intelligible output design should improve the system's relationship with the user and help in decision making. Since the reports are directing referred by the management for taking decisions and to draw conclusions they must be designed with almost care and the details in the reports must be simple, descriptive and clear to the user. So while designing output the following things are to be considered.

- Determine what information to present
- Arrange the presentation of information in an acceptable format
- Decide how to distribute the output to intended receipts

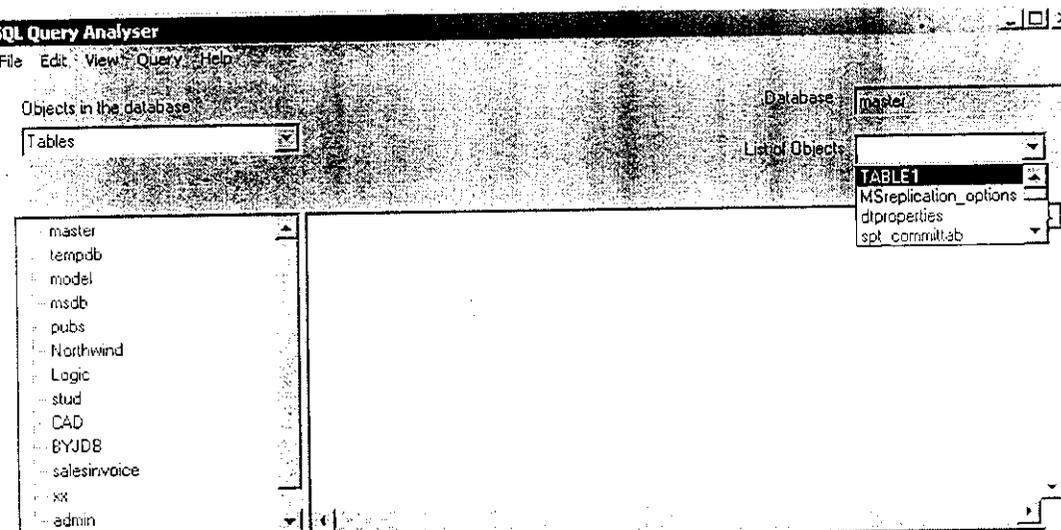
Depending on the nature and future use of output required, they can be displayed on the monitor for immediate need and for obtaining the hardcopy.

The main outputs, which are to be handled in this software, are as follows.

- Display the name of all databases present in that particular server.
- Display all tables, stored procedures and views in the each database.
- Display the fields in each table and data stored in each fields.

- Display the forms for creating the database, tables, stored procedures, Views etc.
- Display the forms for connection and disconnection to and from the Server.

## Output Screen





```

Void CmydIlg::OnConnect()
{
    try
    {
        _SQLServerPtr p1;
        HRESULT hr;
        UpdateData(TRUE);
        if SUCCEEDED(p1.CreateInstance(__uuidof(SQLServer)))
        { m_svr=m_server;
          if(m_authen==0)
          {
              p1->PutLoginSecure(TRUE);
          }
          else
          {
              m_user=m_login;
              m_pass1=m_pass;
              p1-> PutLoginSecure(FALSE);
          }
          if SUCCEEDED(hr=p1->Connect(m_svr, m_user, m_pass1))
              m_connect="True";
        }
    }
    catch(_com_error)
    { AfxMessageBox("Failed to connect to the server");
      m_server.Empty();
      m_pass.Empty();
      m_login.Empty();
      UpdateData(FALSE);
    }
}

```

After establishing connection with the server the next step is listing the databases. For this we pass the query “Select name from master.dbo.sysdatabases” to the server. This query is executed on the server. The result of the execution will have a set of results. By taking each result set and by extracting the column strings from each column which is extracted from each row. The database listing is done in the class named CQueryDlg class, which has the base class CDialog.

After listing the databases, we can list the tables, views or stored procedures of that particular server. The listing is done in the same way as database listing. The only difference is the query “Select name from ‘database name’.dbo.sysobjects where xtype= ‘U’ / ‘V’ / ‘P’”. This module is done in the class named CQueryDlg class, which has the base class CDialog.

The next module is the Query Execution module, which deals with the execution of the queries. We get the SQL commands from the input screen and pass it to the server. This query after execution will return the result set in a QueryResults object. This module is done in the class named CQueryDlg class, which has the base class CDialog. The SQL commands include ‘ Create’, ‘ Select’, ‘Insert’, ‘Update’, ‘Delete’, ‘Alter’, ‘Drop’.

The methods used in this module are:

### **ExecuteWithResults Method**

The ExecuteWithResults method executes a Transact-SQL command batch returning batch result sets in a QueryResults object.

#### *Syntax*

object.ExecuteWithResults( Command , [ Length ] ) as QueryResults

#### *Parts: -*

**Command**                      String that specifies a Transact-SQL or provider-specific command batch.

**Length (Optional)**        Long integer that indicates the statement batch length.

#### *Returns*

A QueryResults object.

## **QueryResults Object (SQL-DMO)**

The QueryResults object presents tabular data to the SQL-DMO application. SQL-DMO enumeration methods, such as the EnumLocks method of the Database object, return a QueryResults object to report their data. SQL-DMO statement execution methods, such as the ExecuteWithResults method of Database and SQL Server objects, also return a QueryResults object.

The QueryResults object is a reporting tool. All properties of the QueryResults object are read-only. With the QueryResults object, you can:

- Navigate data returned from a server running Microsoft® SQL Server™ as the result of statement execution.
- Retrieve specific data values in a data type usable by your application.
- Get the data result of a statement execution as a delimited string of values

## **CurrentResultSet Property (SQL-DMO)**

The CurrentResultSet property controls access to the result sets of a QueryResults object.

### *Syntax*

object.CurrentResultSet [= value]

## **GetColumnString Method (SQL-DMO)**

The GetColumnString method returns a QueryResults object result set member converted to a String value.

### *Syntax*

object.GetColumnString( Row , Column ) as String.

In the Showing Execution Plan Module, there are two ways to show the plan.

- ShowPlan Text
- ShowPlan All

This causes SQL Server not to execute SQL statements. Instead, SQL server returns detailed information about how the statements are executed.

#### *Syntax*

```
SET SHOWPLAN_ALL{ON/OFF}
```

```
SET SHOWPLAN_TEXT{ON/OFF}
```

When SET SHOWPLAN\_TEXT=ALL is ON the SQL Server returns execution information for each statement without executing it and this continues till the option is set OFF.

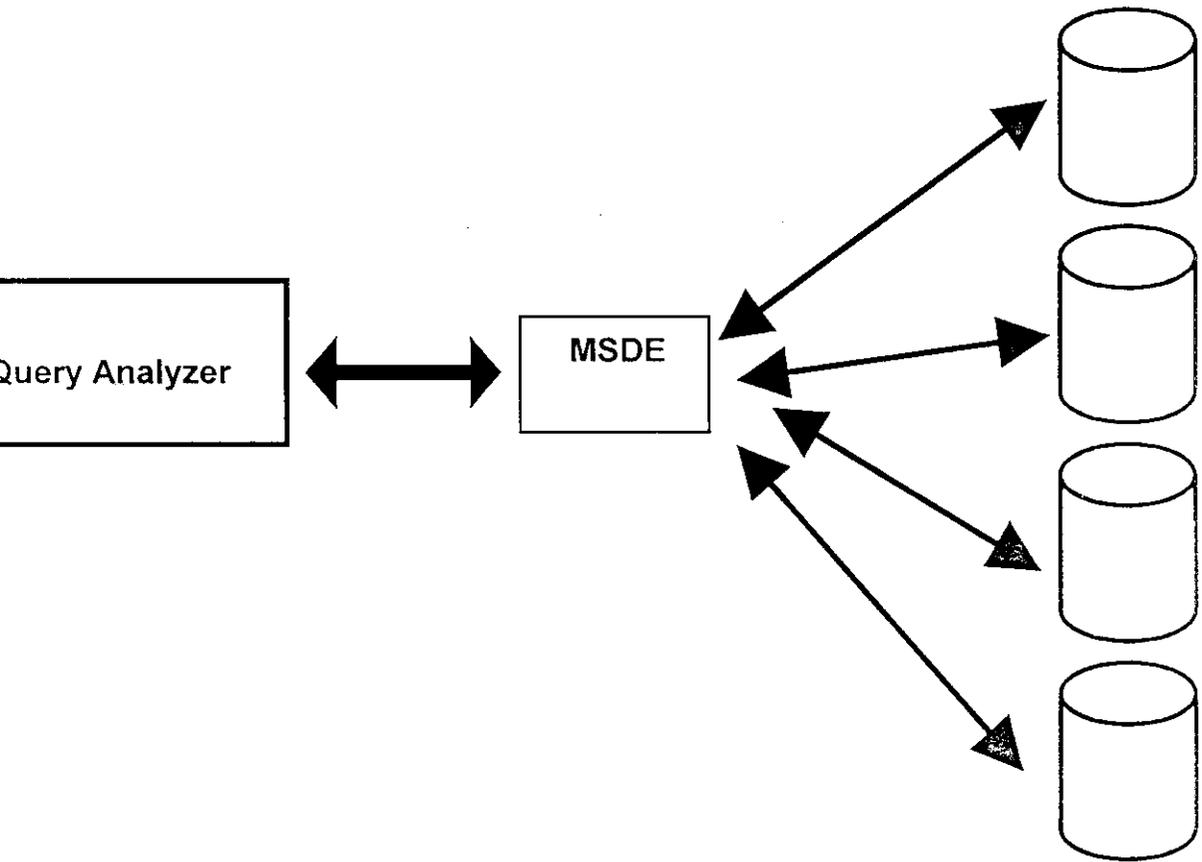
For SET SHOWPLAN\_TEXT{ON/OFF} output contains only the SQL statement column.

For SET SHOWPLAN\_ALL{ON/OFF} provides estimates of the resource requirements for the statements and the output contains the following columns:

- StmtText            Transact SQL statement
- StmtId             Number of the statement in the current batch
- NodeId             Id of the node in the current query
- Parent             NodeId of the parent step.
- PhysicalOp         physical implementation algorithm for the node.
- LogicalOp         relational algebraic operator this node represents.
- Argument           provides supplemental information about the operation being performed
- Defined values     Contains a comma separated list of values introduced by this operator
- Estimate Rows     Estimated number of rows output by this operator
- EstimateIO         Estimated IO cost for this operator
- Estimate CPU       Estimated CPU cost for this operator

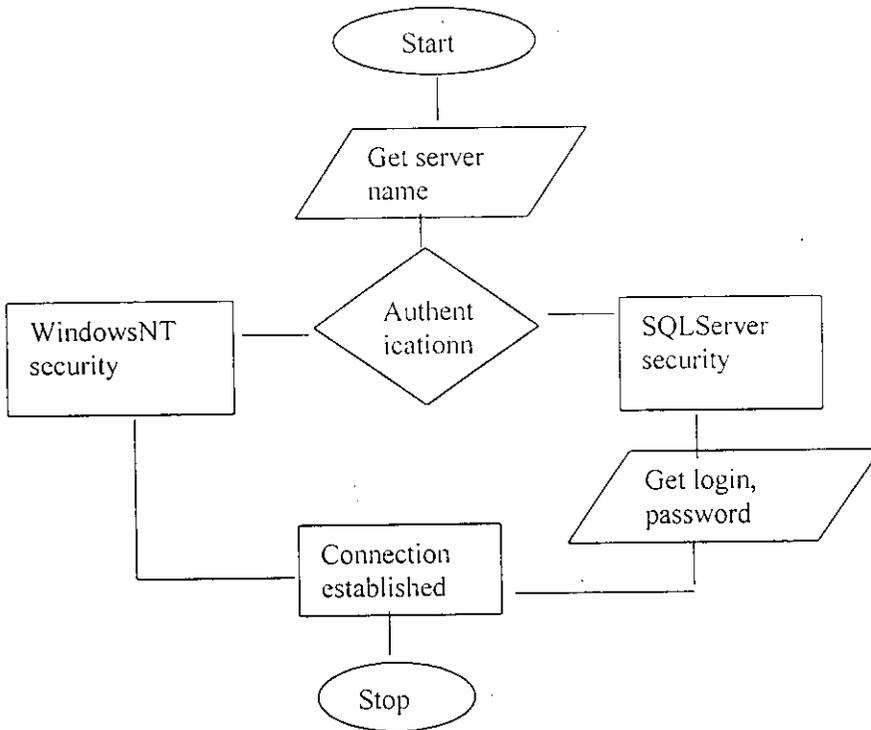
- AvgRowSize Estimated Average row size of the row being passed through this operator
- Outputlist contains a column being projected by the current operation
- Warnings warning messages relating to the current
- Type Node type

The parsing module checks the error of the query statement. Here if there is any error in the query statement, the exceptions are caught and appropriate messages are given.



## Flowcharts for Different Modules

### Connection Module



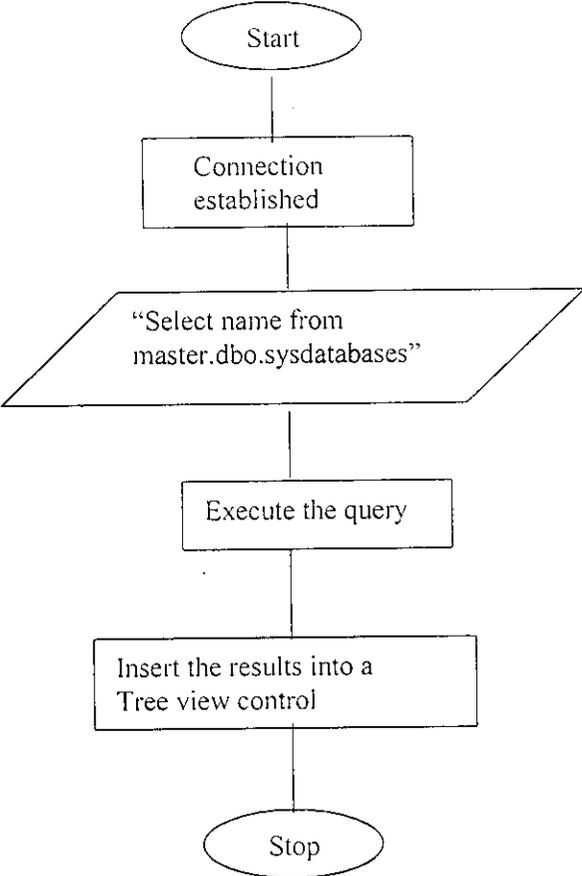
The Connection module deals with establishing the connection to the server machine.

There are 2 types of security

- Windows NT security
- SQL Server security

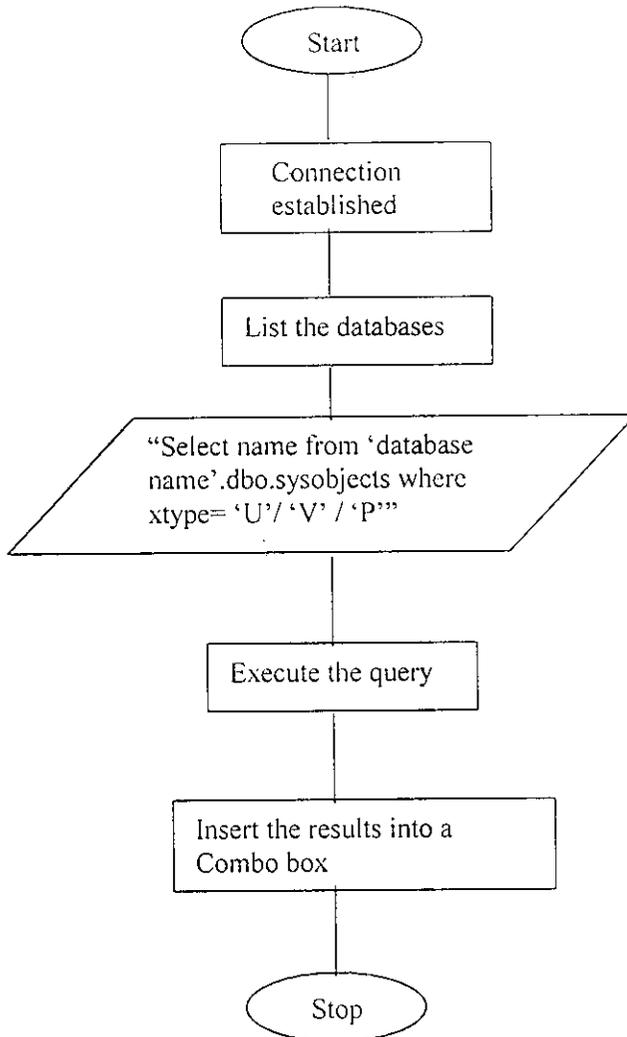
In Windows NT Security we need only to know the server name, System login Name and Password. But in SQL Server security in addition to the above requirements we need SQL Server user Name and password to access the database.

# Database Listing Module



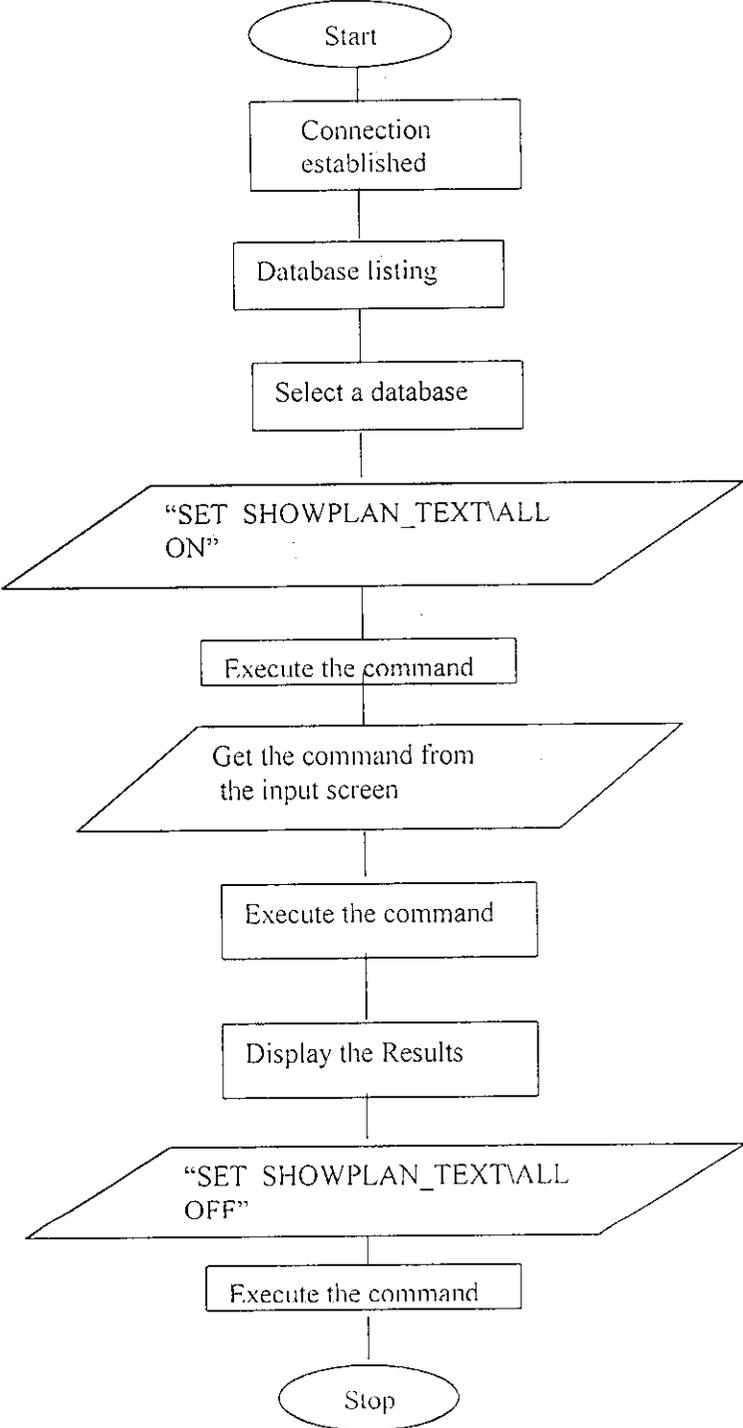
In Database Module, After establishing the connection with the server with the help of MSDE, the Databases corresponding to that particular Server is listed in a Tree view control.

## Tables /Views/Stored Procedures Listing Module



In this module, we can list Tables or Views or Stored Procedures of that database. For this we must connect to a server then select a database and then select the tables or views or the stored procedures as per the user needs.

# Showing Execution Plan Module



---

---

**SYSTEM  
IMPLEMENTATION AND TESTING**

---

---

## 5. System Testing and Implementation

### SYSTEM IMPLEMENTATION

Implementation is the stage of the project where the theoretical design is turned in to a working system. If the implementation is not carefully planned and controlled it can cause chaos and confusion. Implementation includes all those activities that take place to convert from the old system to new one. The new system may be totally new, replacing an existing manual or automated system or it may be a major modification to an existing system. Proper implementation is essential to provide a reliable system to meet the organization requirements.

The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from the old system to new system. The system can be implemented only after thorough testing is done and if it is found to be working according to specifications. The more complex the system being implemented, the more involved will be the system analysis and design effort required just for implementation.

The project “Query Analyzer for SQL Server Using MSDE” is to undergo the formal process of implementation in the same manner as every other system would undergo.

The implementation stage involves the following tasks:-

- Careful planning.
- Investigation of system and constraints.
- Design of methods to achieve the changeover.
- Training of the staff in the changeover phase.
- Evaluation of the changeover method.

Full implementation of the project requires the successful completion of the other modules as well. Change over is a stage of moving from existing system to proposed system. In Query Analyzer Direct change over method is used as a change over method for replacing the current system.

The method of implementation and the time scale to be adopted are found out initially. Next the system is tested properly and the users are trained in the new procedures. The system is successfully implemented at Logic Software Solutions Pvt. Ltd.

### **Post Implementation Review**

A study of the system has revealed that the employees due to the user friendliness have accepted the system, reduced the number of errors, increased accuracy and decreased cost of operations. The system also pays for efficient and speedy execution of operations compared to the earlier system.

## **SYSTEM TESTING**

System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before the live operation begins. During the development of a software project, errors of various types can occur at any stage. At each phase, different techniques are used to detect the errors. The first major hurdle in the process of implementation is the period of testing the system. The debugging process is the most unpredictable part of testing procedure.

To make the system developed here to be reliable and accepted, various testing methods were used:

### **Unit Testing:**

This test focuses verification effort on the smallest unit of software design

– The module. It is the set of tests performed by an individual programmer prior to the integration of the unit into a larger system. The tested modules must be assembled to form the complete software package. According to the test plan for the Query Analyzer, each and every module that comprises has been unit tested and the errors were eradicated.

Individual modules of the system were tested for functionality and each of the modules was found to perform the desired function.

### **Integration Testing**

Integration testing is a systematic technique for constructing the program structure at the same time conducting tests to uncover errors associated with interfacing. Unit tested modules were taken and a single program structure was built that has been dictated by design.

In Query Analyzer, the sub-modules are Connection Module, Database Listing Module, Tables/Views/Stored Procedures Listing Module, Query Execution Module, Showing Execution Plan Module, and Parsing Module. All these sub-modules are connected correctly into a whole module. The overall system was tested after the integration of the independently developed modules and they were found to complement each other in an efficient way. The functionality expected of the whole system was found to be met.

## **Password Testing**

The user of the Query Analyzer has to specify the user name and password for entering into the system. If the username and password does not match with the real one the access is denied, thereby providing a strong security.

## **Output Testing:**

The outputs generated or displayed by the system under consideration are tested by asking the users about the same required by them, Here, the output format on the screen is found to be correct as the format was designed in the system design phase according to the user needs.. Hence, output testing has not resulted in any correction in the system.

In all these aspects, the system was found to be reliable. The security measures were checked and found to be satisfied

The results of the testing phase are taken as hard copy and are attached in the appendix section of the report.

---

**CONCLUSION**

---

## 6. Conclusion

MSDE is the data engine supplied by Microsoft. It is a mere database engine without any interfaces such as the query analyzer or enterprise manager, which SQL server possesses. A database engine is often just a database management system without its graphical management tools. Thus with the MSDE in hand, if we can develop an environment which fully exploits the services provided by it, we can transform this engine into a complete DBMS product. To do this we require interfaces like the query analyzer of the SQL Server.

By the use of this Query Analyzer and another interface called Enterprise Manager we can get the full properties of the MS SQL. Enterprise Manager has fewer options when compared to the Query Analyzer. For eg. In Enterprise Manager we can create, modify and maintain a table, but in Query Analyzer, by the use of Query we can get data in different formats as we needed. With the help of Query Analyzer we can do every operation related to the database such as creation deletion, modification etc of database are some among them.

---

*SCOPE FOR FUTURE  
DEVELOPMENT*

## **7. SCOPE FOR FURTHER DEVELOPMENT**

- Service Manager can be included to control the SQL Server remotely.
- The font, size and color can be changed according to the users convenience.
- Printing functionalities can be added.

---

*BIBLIOGRAPHY*

---

## 8. BIBLIOGRAPHY

1. System Analysis and Design : Elias M Awad ,  
Galgotia Publications
2. Design and Implementation of MS SQL Server 7.0
3. Mastering Visual C++ 6 : Michael J Young ,  
BPB Publications
4. Mastering Windows NT Server 4 : Minasi ,  
BPB Publications
5. Visual C++ 6 in record time : Steven Holzner ,  
BPB Publications
6. Visual C++ 6 from the ground up : John Paul Mueller ,  
Tata Mc Graw Hill

---

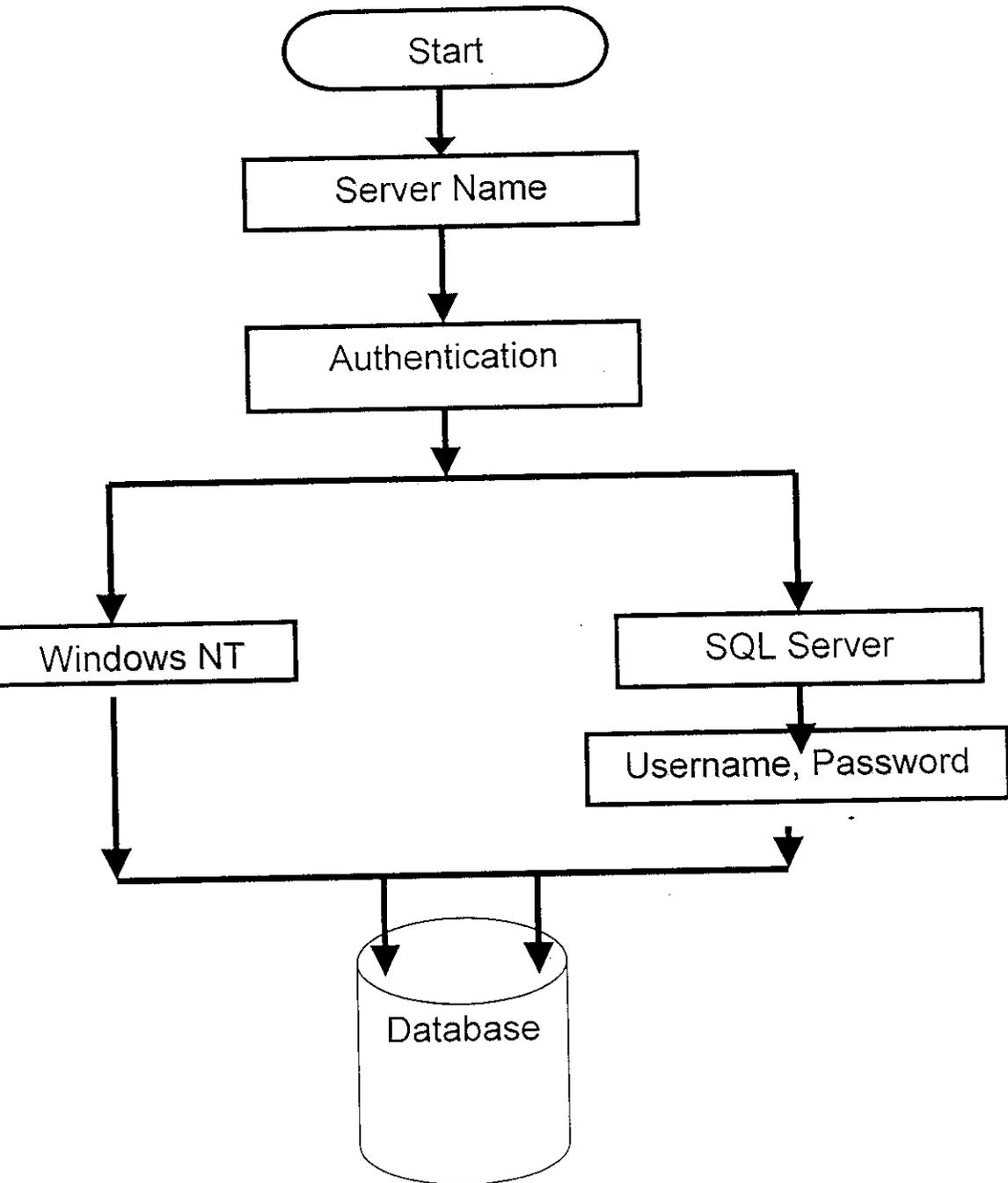
---

*APPENDIX*

---

---

# System Flow Chart



---

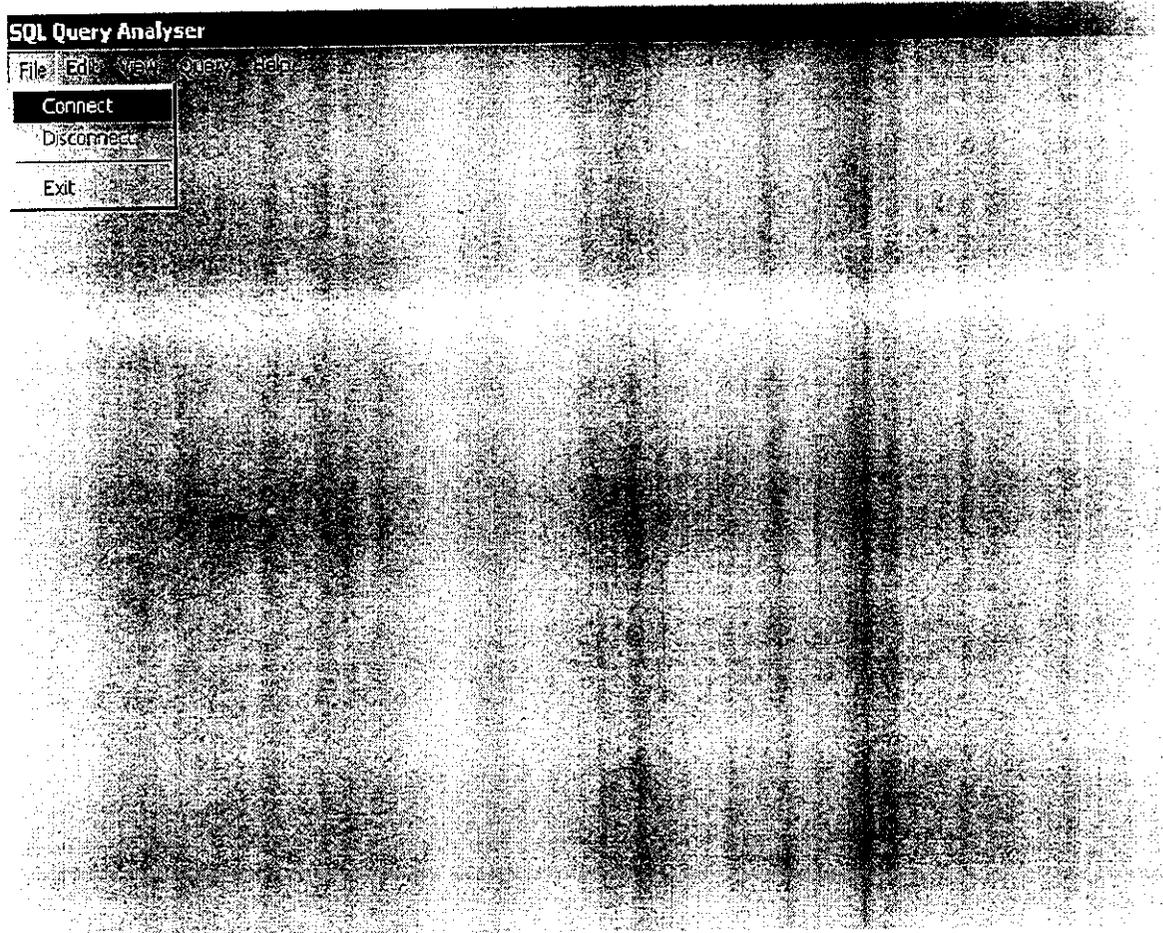
---

*SCREEN FORMATS*

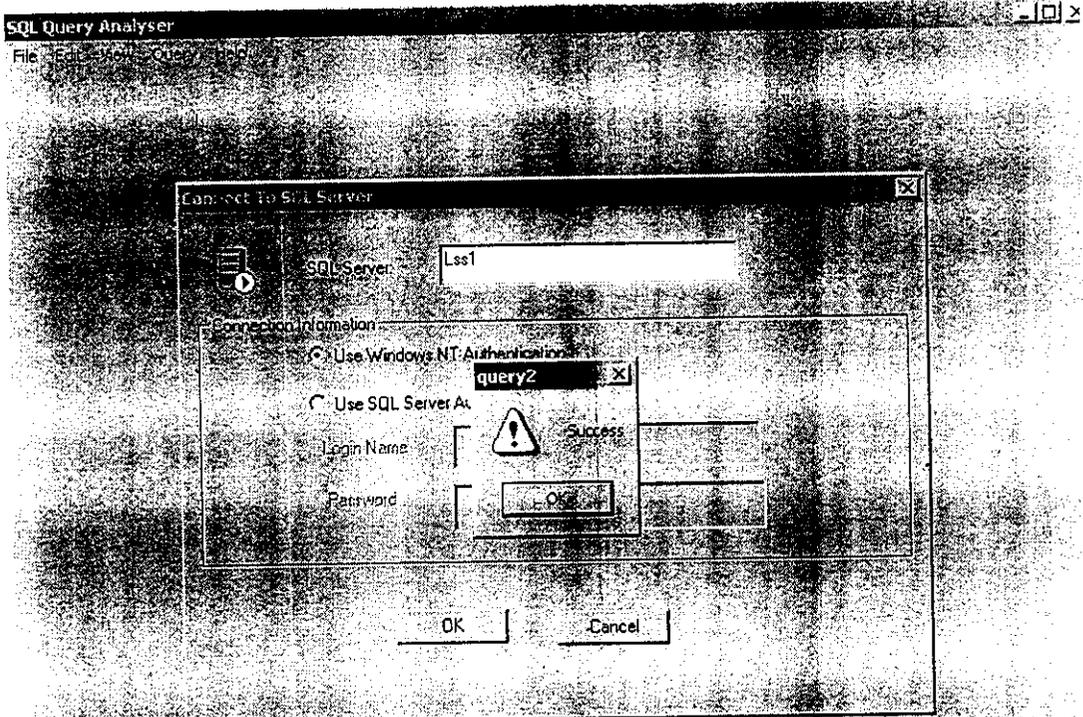
---

---

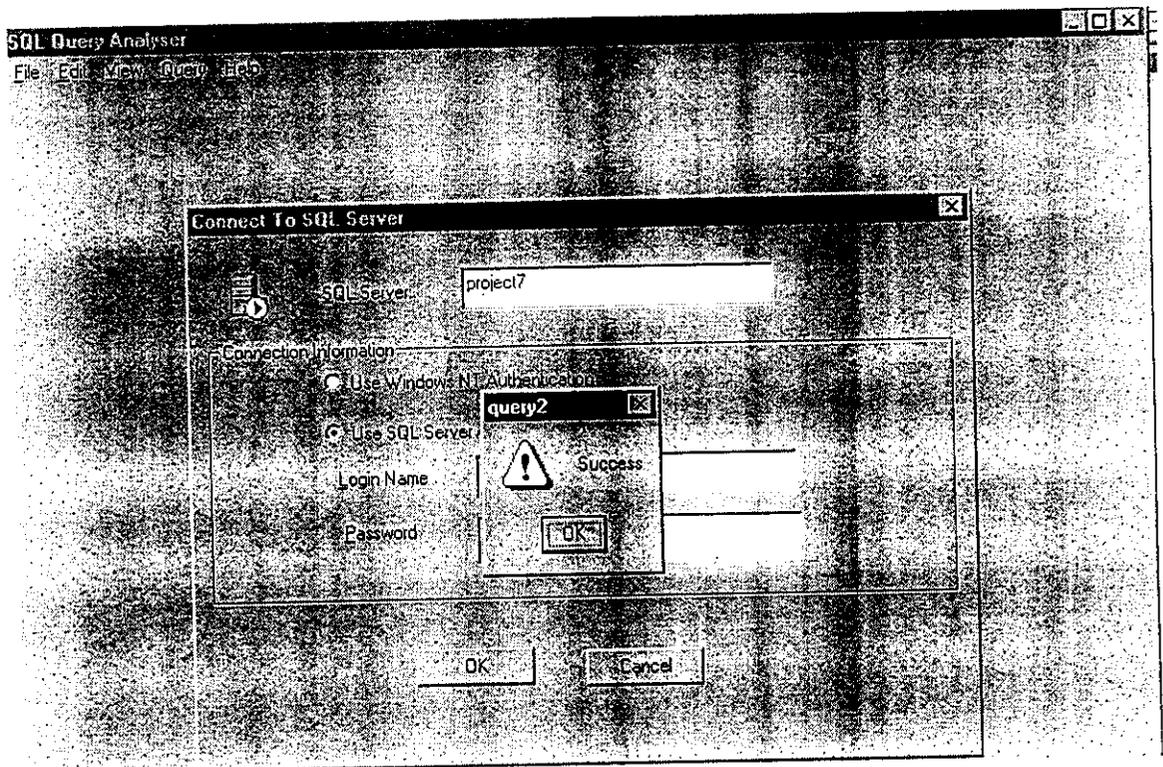
# Connect Menu



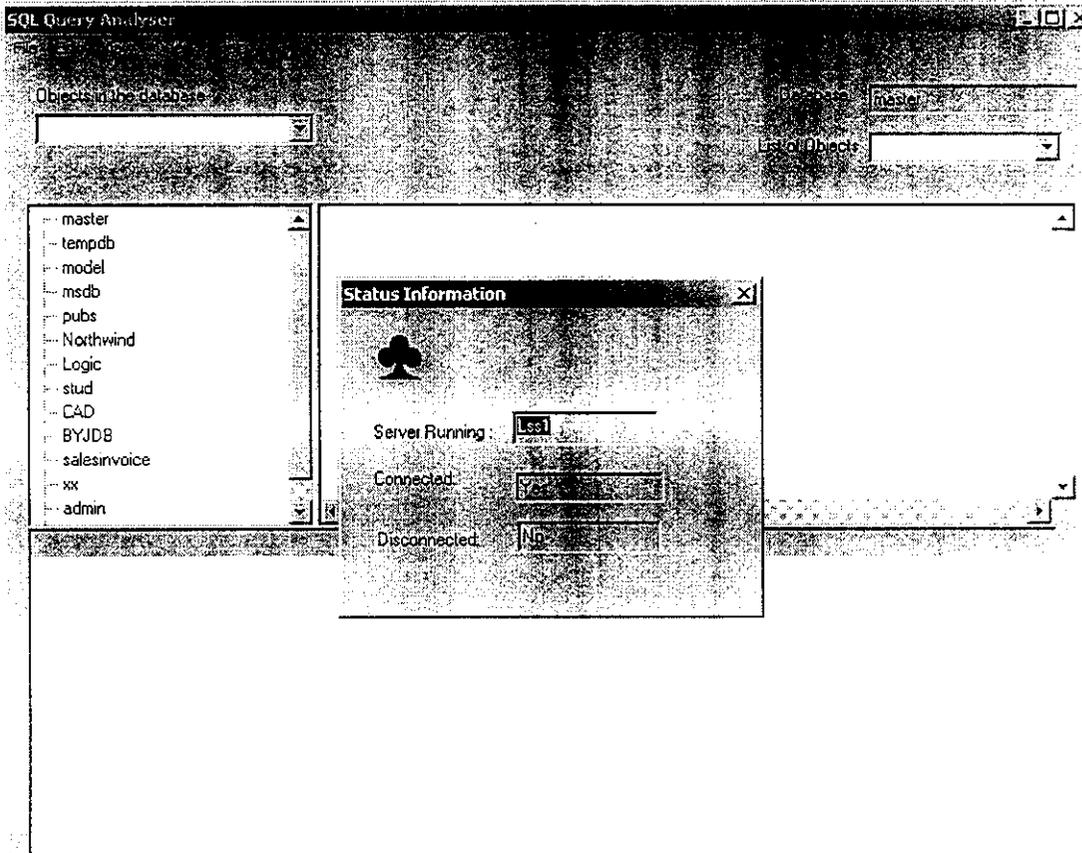
# Connect Dialog using Windows NT



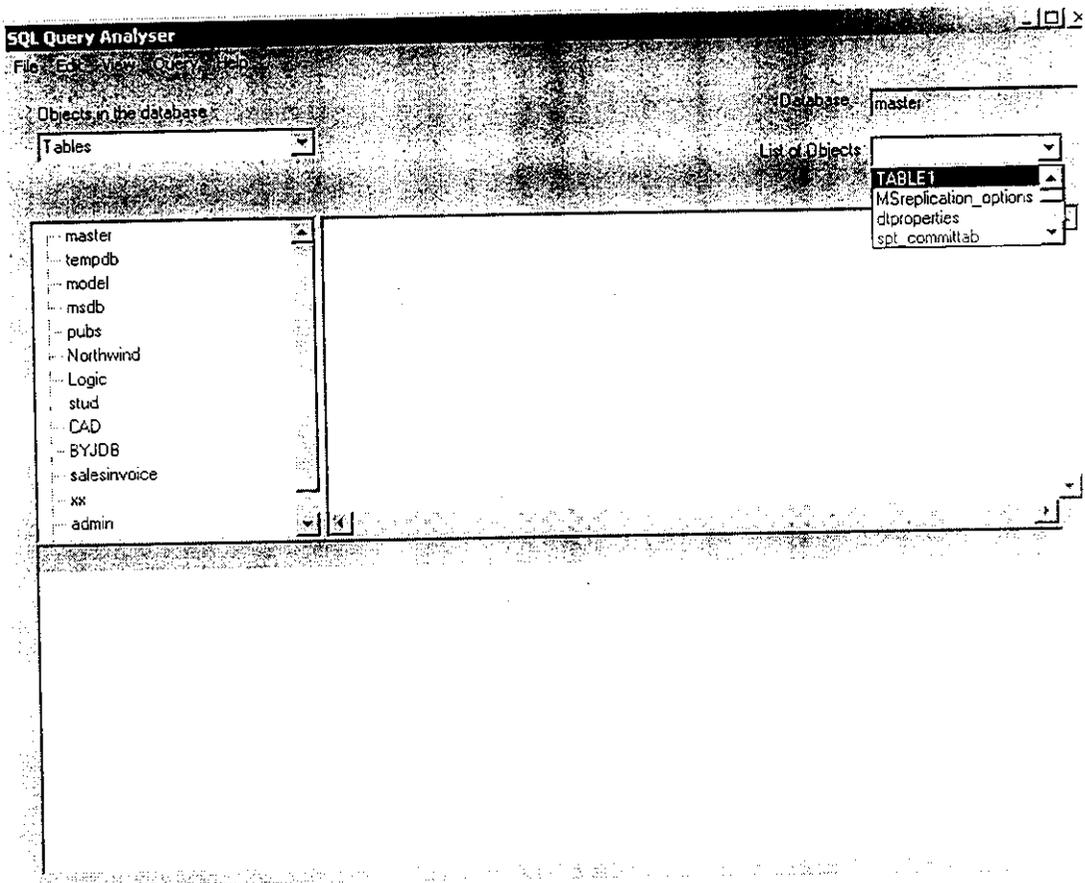
# Connect dialog using SQL Server authentication



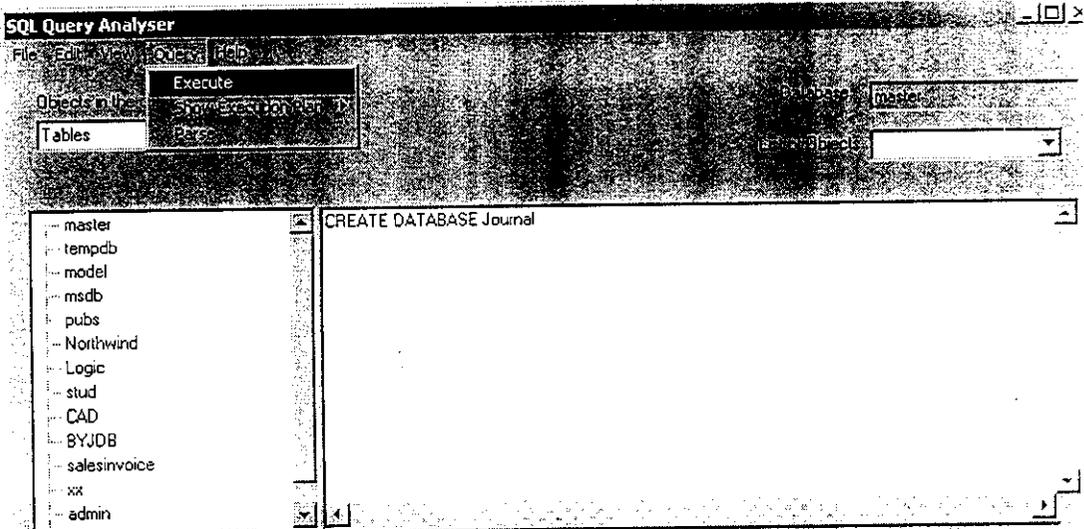
## Screen showing status information



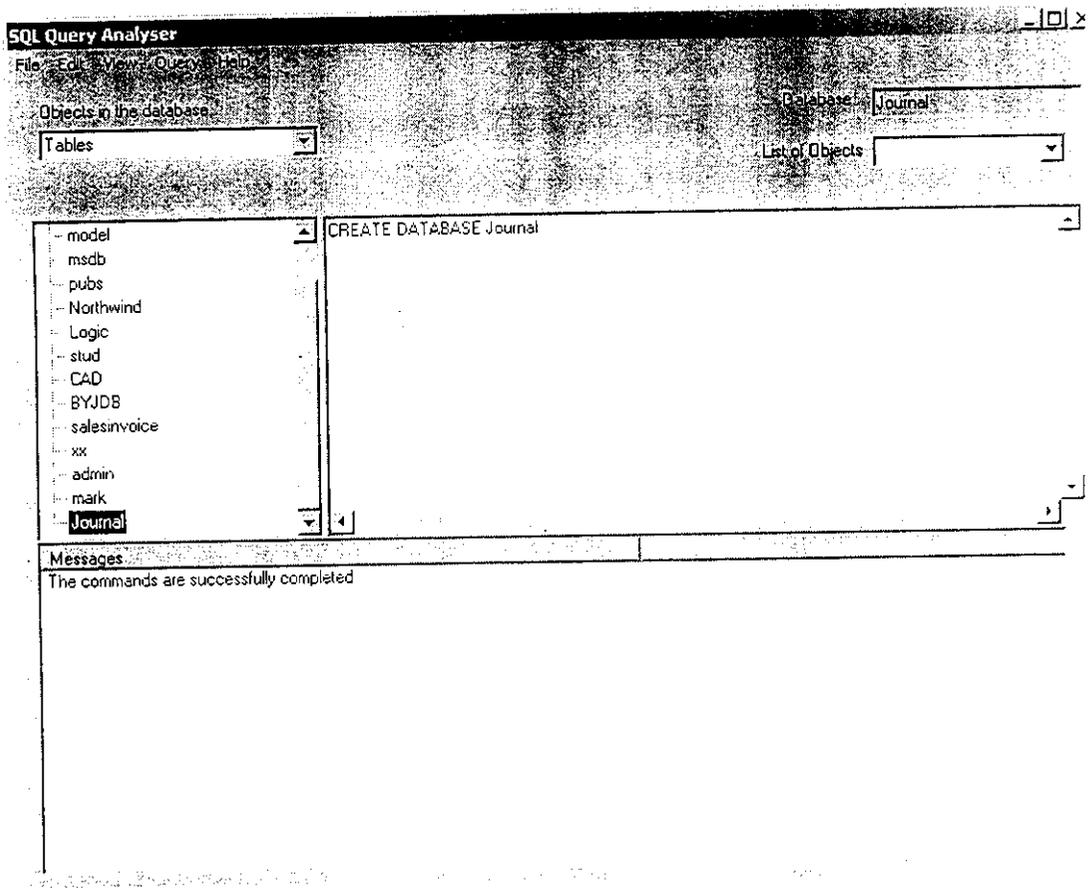
## Screen showing list of objects



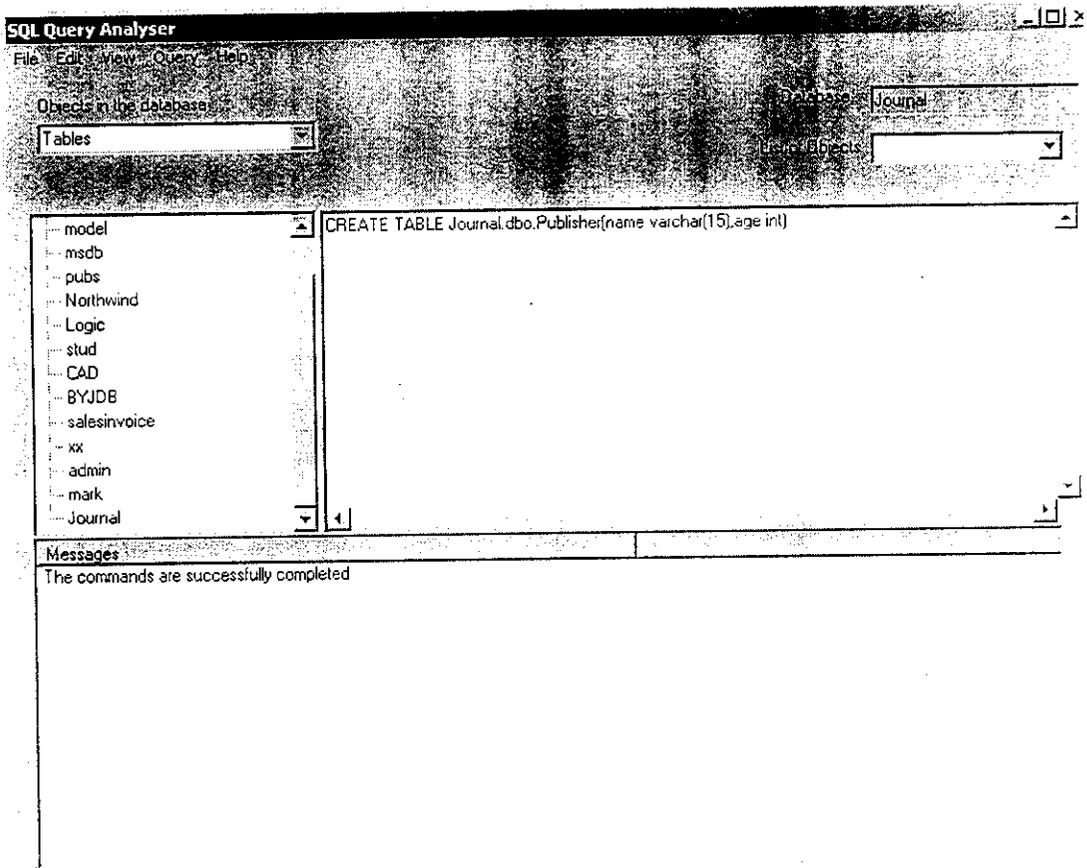
# Database Creation



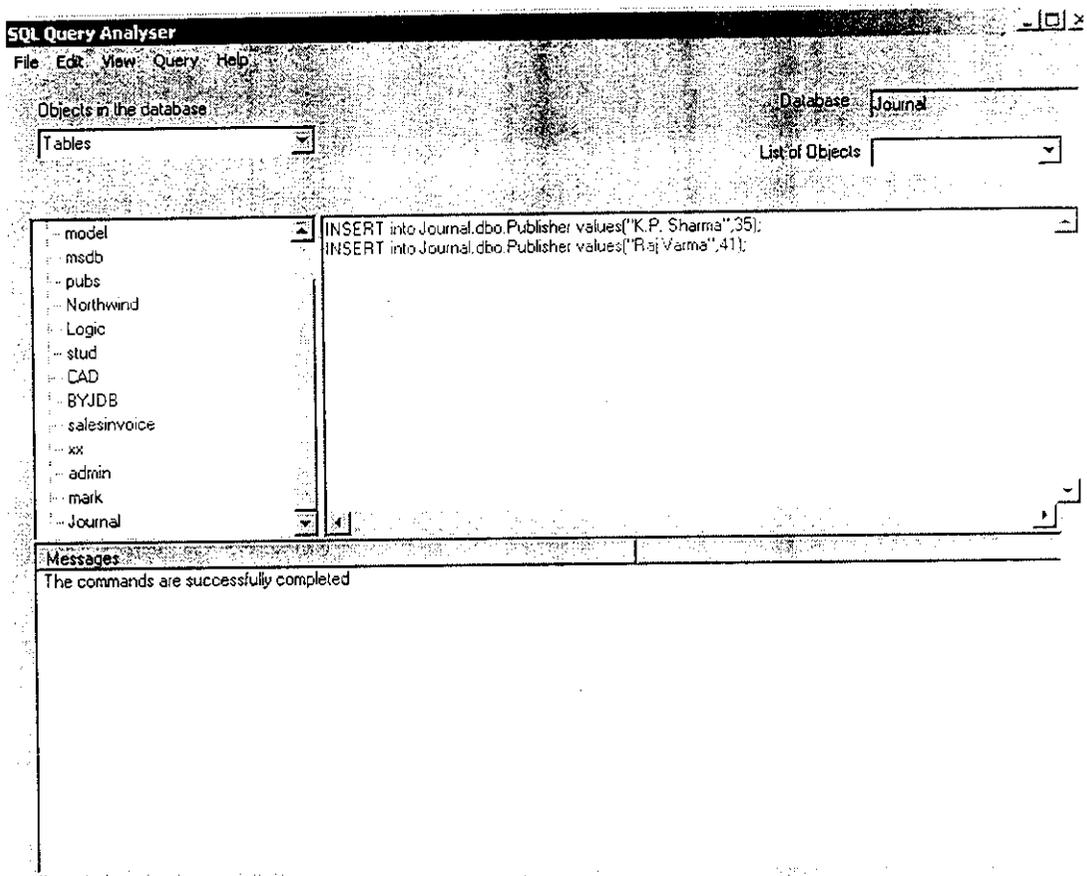
## Result of creation



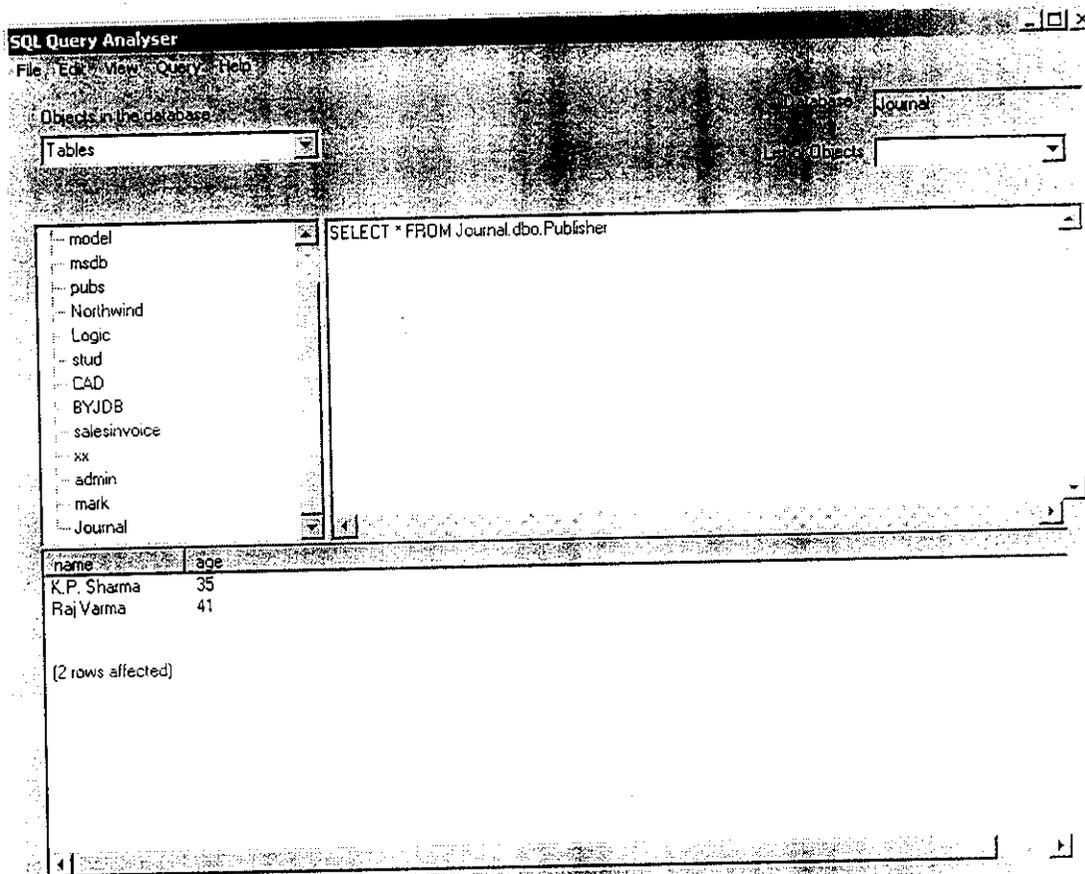
## Creation of table



## Insertion of values into the table using multiple statements



## Displaying the fields and contents of the table



The screenshot shows the SQL Query Analyzer interface. The 'Objects in the database' pane on the left lists various databases, with 'Journal' selected. The 'Tables' dropdown is also set to 'Journal'. The 'Query' pane contains the SQL statement: `SELECT * FROM Journal.dbo.Publisher`. The 'Results' pane at the bottom displays the query output as a table with two columns: 'name' and 'age'. The results are as follows:

name	age
K.P. Sharma	35
Raj Varma	41

[2 rows affected]

## Updation of the table

The screenshot displays the SQL Query Analyser interface. At the top, the title bar reads "SQL Query Analyser" with standard window controls. Below the title bar is a menu bar with "File", "Edit", "View", "Query", and "Help".

On the right side, there is a "Database" dropdown menu set to "Journal" and a "List of Objects" dropdown menu.

On the left side, under "Objects in the database", there is a "Tables" dropdown menu. Below it is a tree view of database objects:

- model
- msdb
- pubs
- Northwind
- Logic
- stud
- C&D
- BYJDB
- salesinvoice
- xx
- admin
- mark
- Journal

The main query window contains the following SQL command:

```
UPDATE Journal.dbo.Publisher SET age=42 WHERE name="Raj Varma"
```

At the bottom, a "Messages" pane displays the text: "The commands are successfully completed".

## Effect of updating

The screenshot shows the SQL Query Analyzer interface. The title bar reads "SQL Query Analyzer". The menu bar includes "File", "Edit", "View", "Query", and "Help".

At the top, there are two dropdown menus: "Objects in the database" set to "Tables" and "Database" set to "Journal".

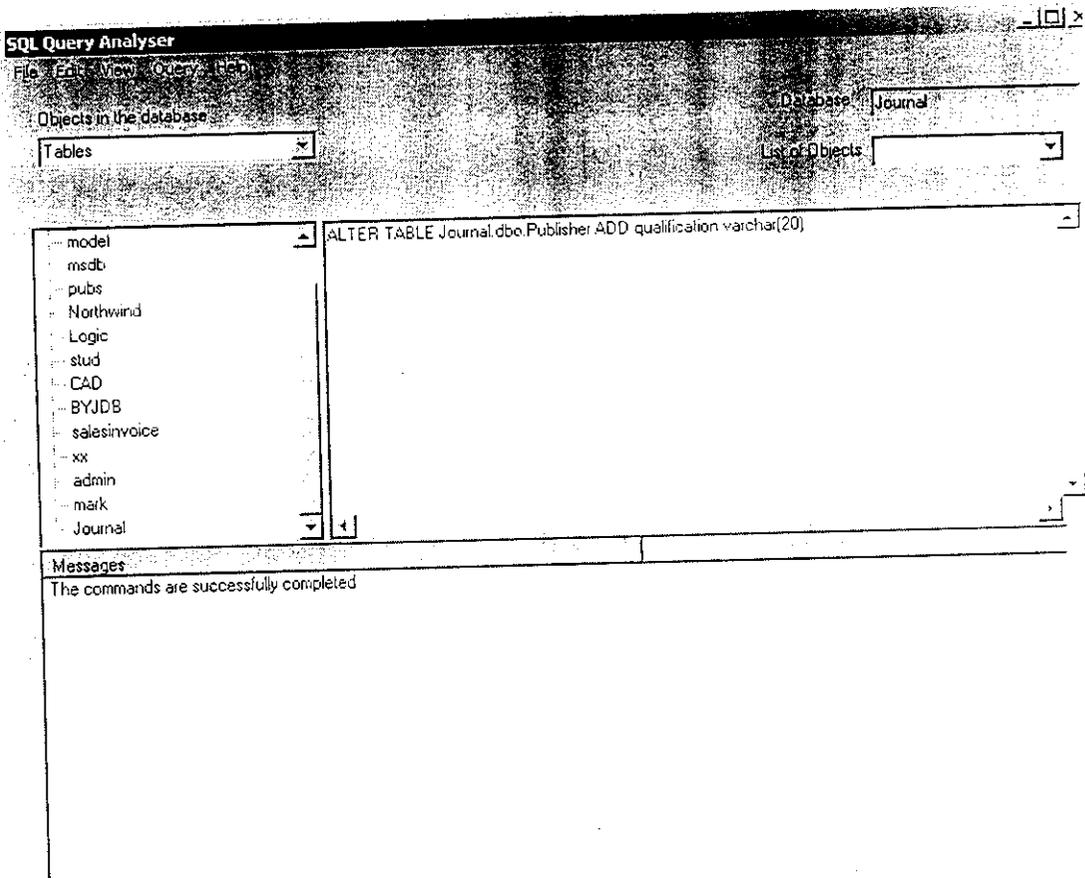
The main area is split into two panes. The left pane shows a tree view of database objects, including "model", "msdb", "pubs", "Northwind", "Logic", "stud", "CAD", "BYJDB", "salesinvoice", "xx", "admin", "mark", and "Journal". The right pane contains the SQL query: `SELECT * FROM Journal.dbo.Publisher`.

Below the query, the results are displayed in a table with two columns: "name" and "age".

name	age
K.P. Sharma	35
Raj Varma	42

Below the table, it indicates "(2 rows affected)".

## Altering the table



## Effect of Alteration

The screenshot shows the SQL Query Analyser interface. The title bar reads "SQL Query Analyser". The menu bar includes "File", "Edit", "View", "Query", and "Help". The "Objects in the database" section shows a tree view with "Tables" selected. The "Database" dropdown is set to "Journal". The "List of Objects" dropdown is empty. The query editor contains the text "SELECT \* FROM Journal.dbo.Publisher". The results pane displays a table with the following data:

name	age	qualification
K.P. Sharma	35	
Raj Varma	42	

(2 rows affected)

## Showing execution plan(Text)

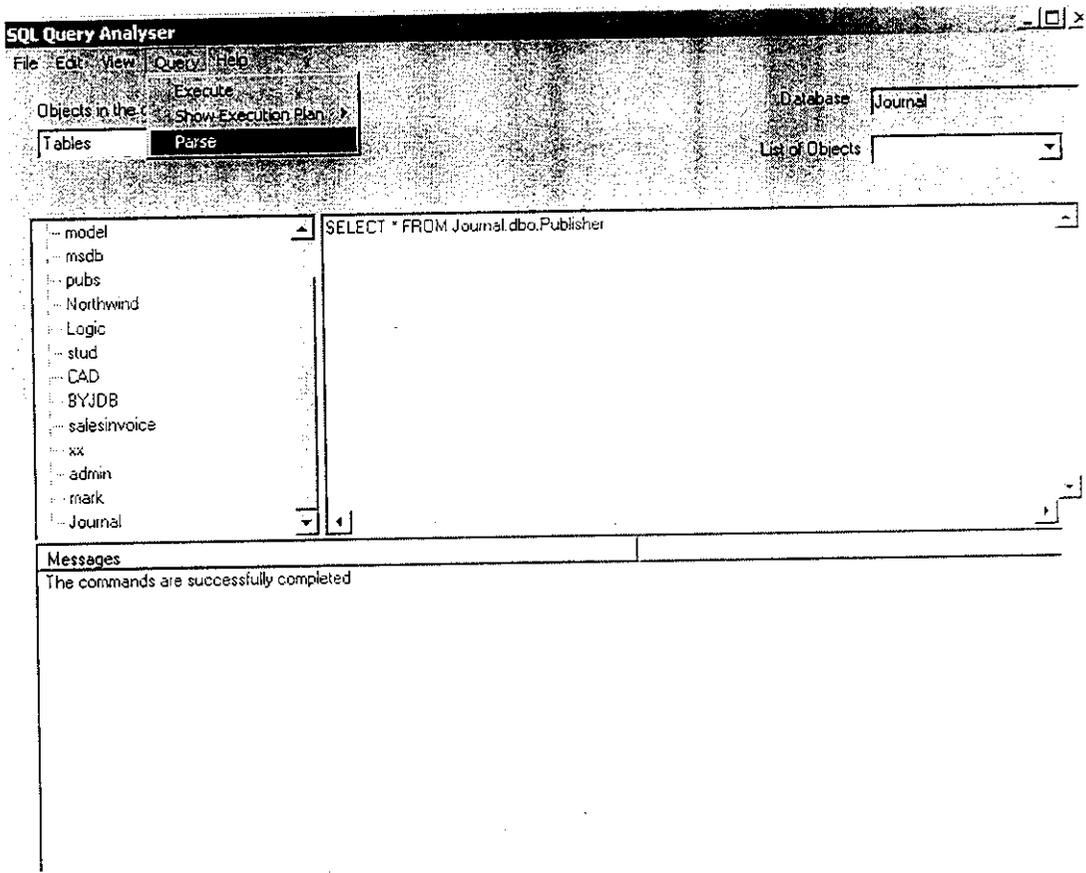
The screenshot shows the SQL Query Analyser interface. The menu bar includes File, Edit, View, Query, and Help. The 'Query' menu is open, showing options: Execute, Show Execution Plan (highlighted), ShowPlan Text, Parse, and ShowPlan All. The 'Database' dropdown is set to 'Journal'. The 'List of Objects' dropdown is also visible. The main query text area contains the query: `SELECT * FROM Journal.dbo.Publisher`. On the left, a tree view shows the database structure with 'Journal' selected. Below the query text, the 'Stmt Text' pane displays the execution plan: `1-Table Scan(OBJECT:([Journal].[dbo].[Publisher]))` followed by the query text `SELECT * FROM Journal.dbo.Publisher`.

## Showing execution plan (All)

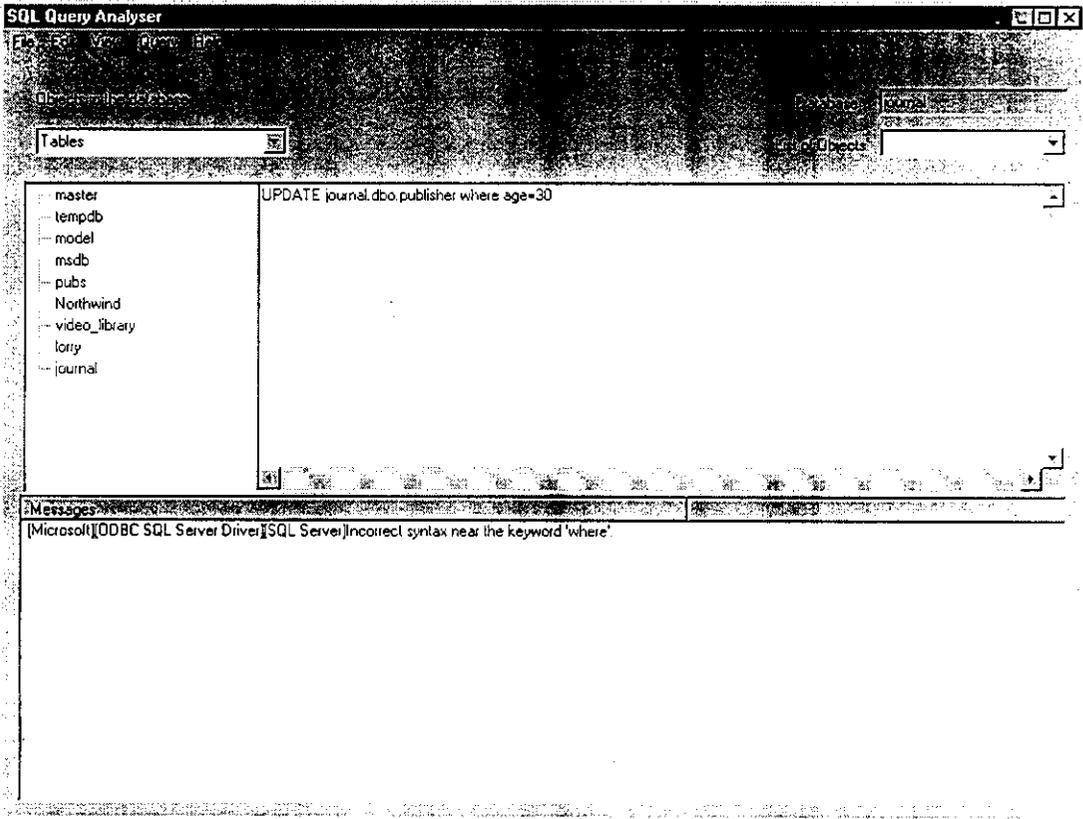
The screenshot shows the SQL Query Analyzer interface. The menu bar includes File, Edit, Views, Query, and Help. The 'Query' menu is open, showing options like 'Show Execution Plan', 'ShowPlan All', and 'ShowPlan Text'. The 'Tables' tab is selected in the 'Objects in the Server' pane. The query text is 'SELECT \* FROM Journal.dbo.Publisher'. The execution plan table below shows the following data:

Stmt Text	Stmt Id	Node Id	Parent	Physical Op	Logical Op	Argument
SELECT * FROM Journal.dbo.Publisher	11	1	0			
Table Scan(OBJECT:([Journal].[dbo].[Publish...])	11	3	1	Table Scan	Table Scan	OBJECT:([Journal].[dbo].[Publish...

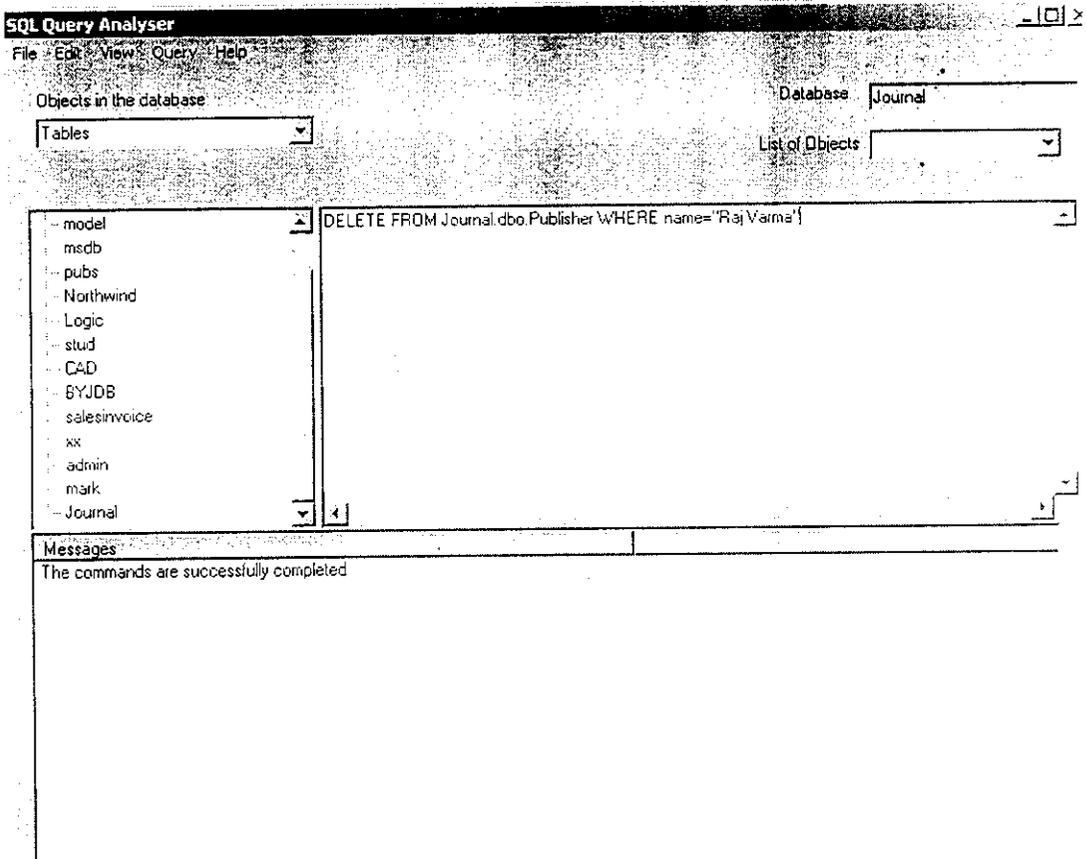
## Checking the syntax of the query



## Error in the Query Statement



## Deleting a row from the table



# Effect of deletion

The screenshot shows the SQL Query Analyser interface. The title bar reads "SQL Query Analyser". The menu bar includes "File", "Edit", "View", "Query", and "Help".

On the left, under "Objects in the database", there is a dropdown menu set to "Tables". The "Database" field is set to "Journal". Below this is a tree view of database objects:

- model
- msdb
- pubs
- Northwind
- Logic
- stud
- CAD
- BYJDB
- salesinvoice
- xx
- admin
- mark
- Journal

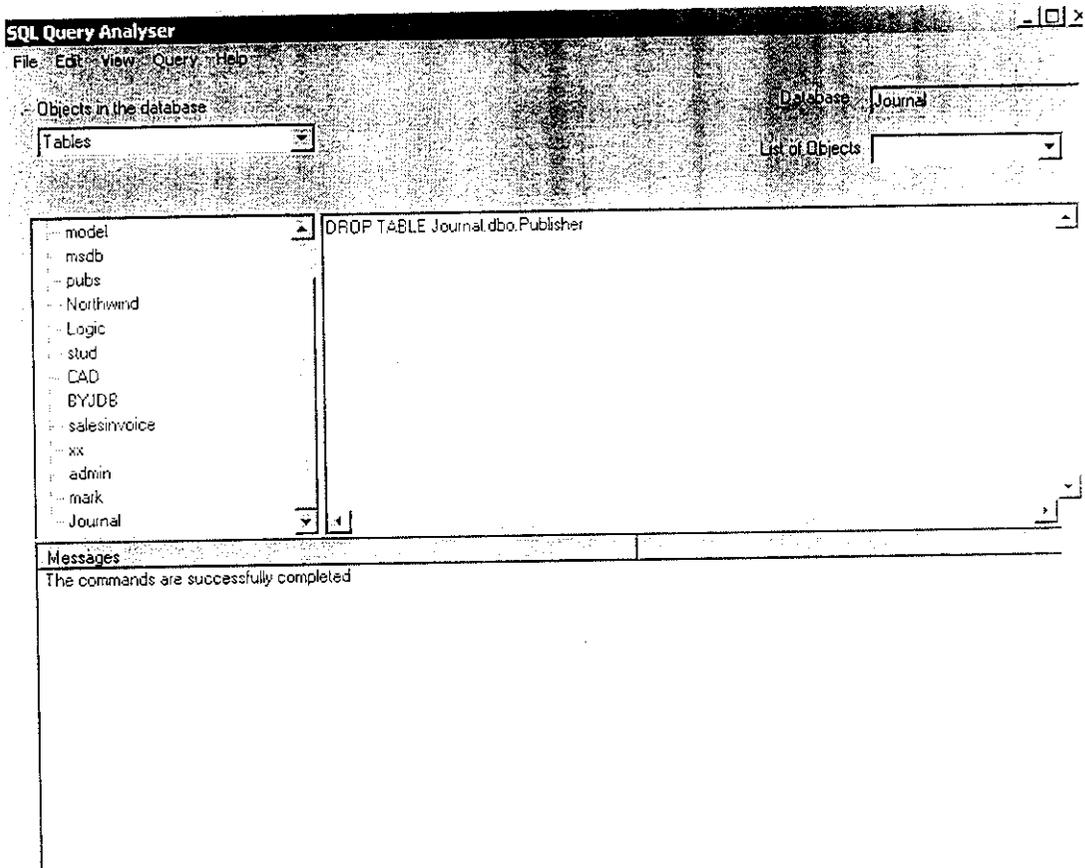
The central query editor contains the SQL statement: `SELECT * FROM Journal.dbo.Publisher`

The results pane at the bottom shows a table with the following data:

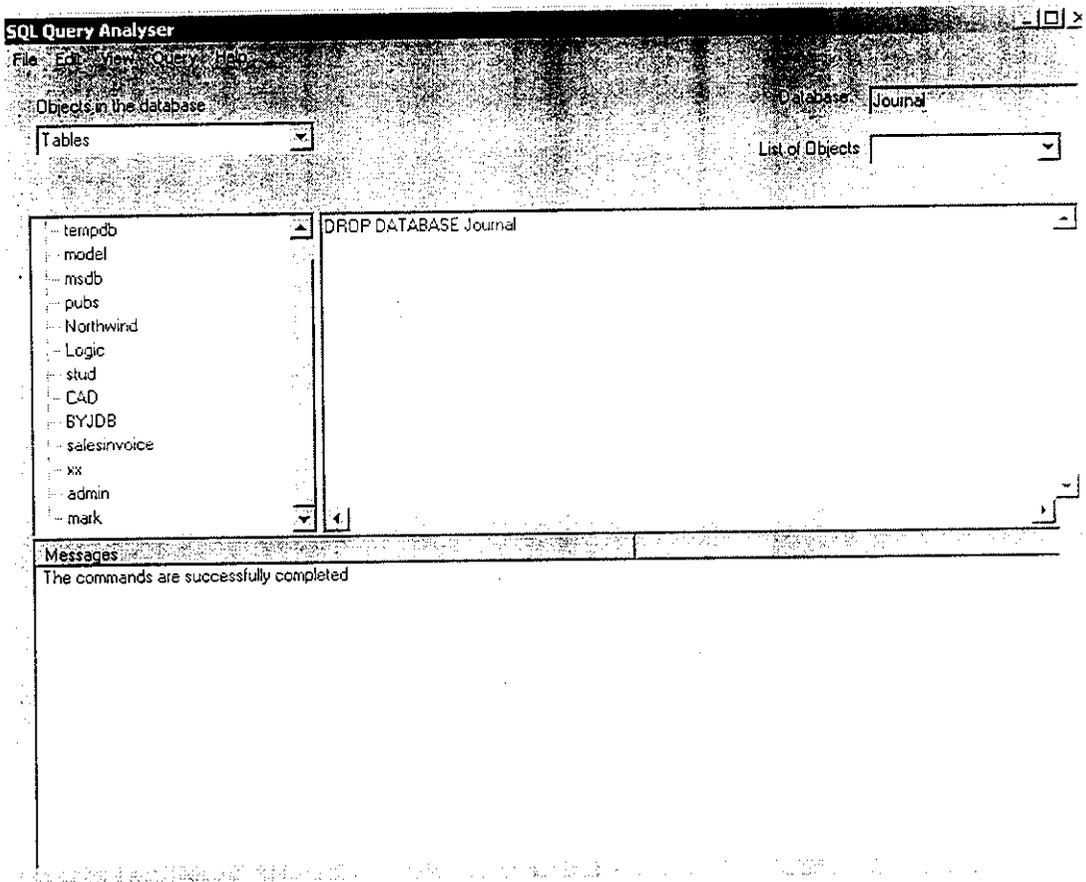
name	age	qualification
K.P. Sharma	35	

Below the table, it indicates "[1 rows affected]".

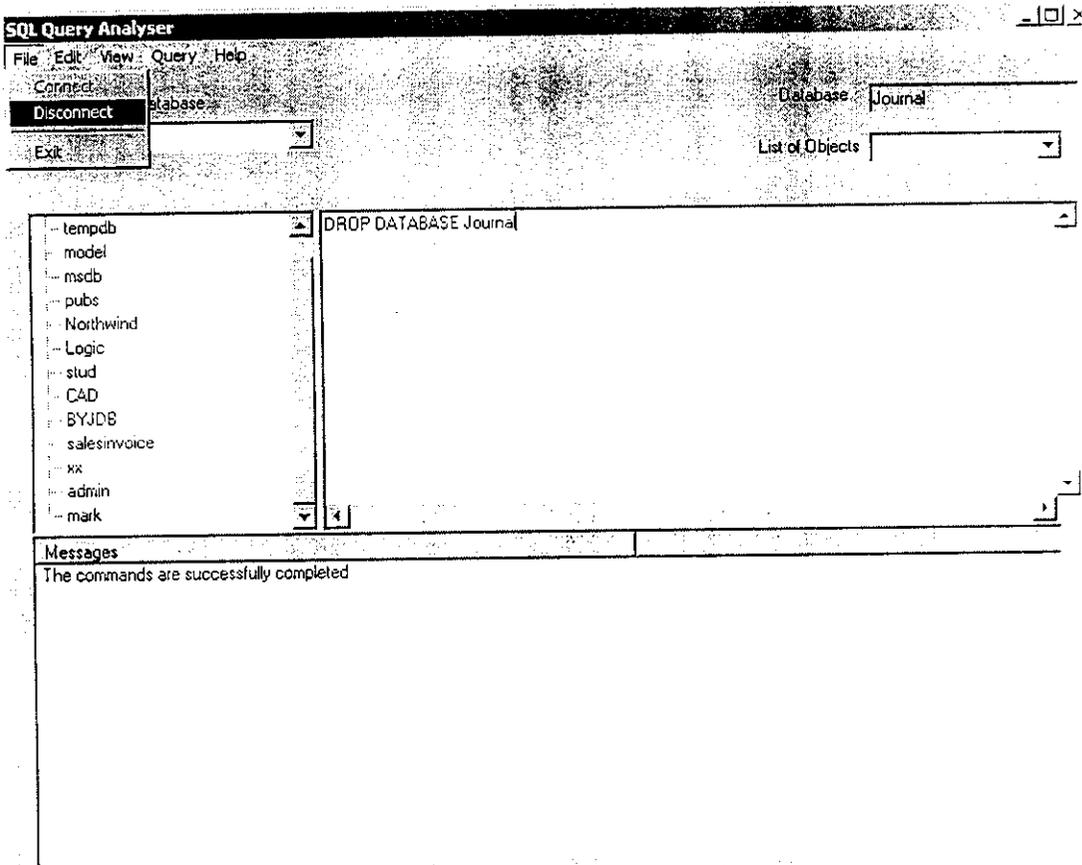
## Dropping a table



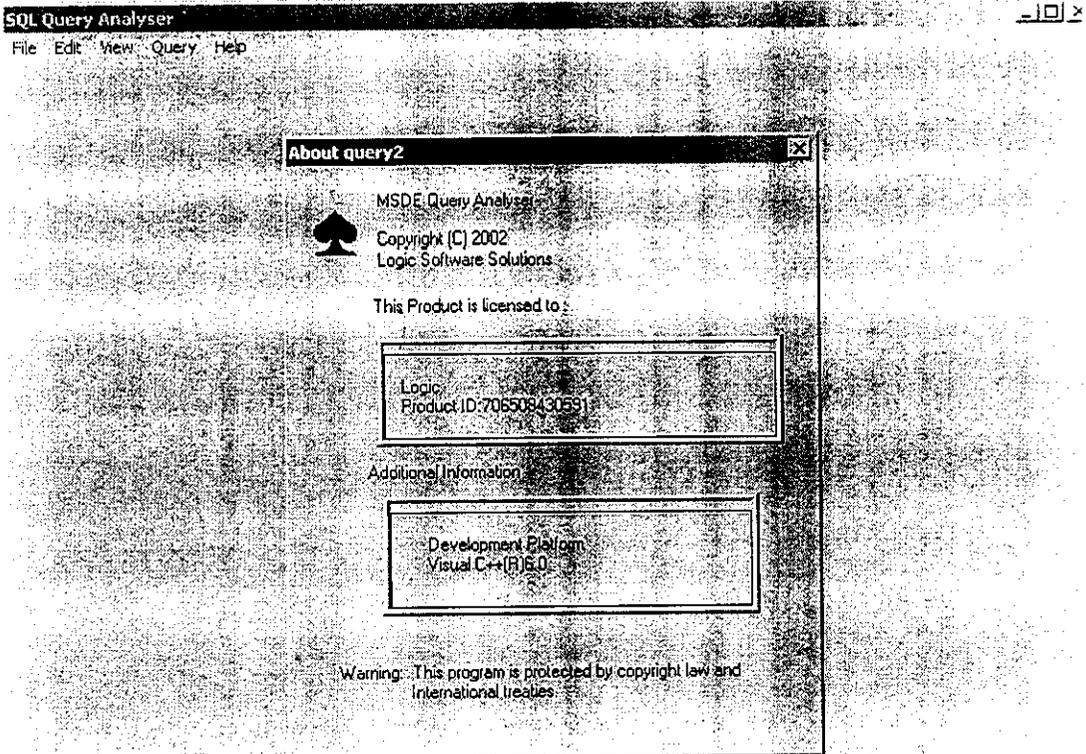
## Dropping a database



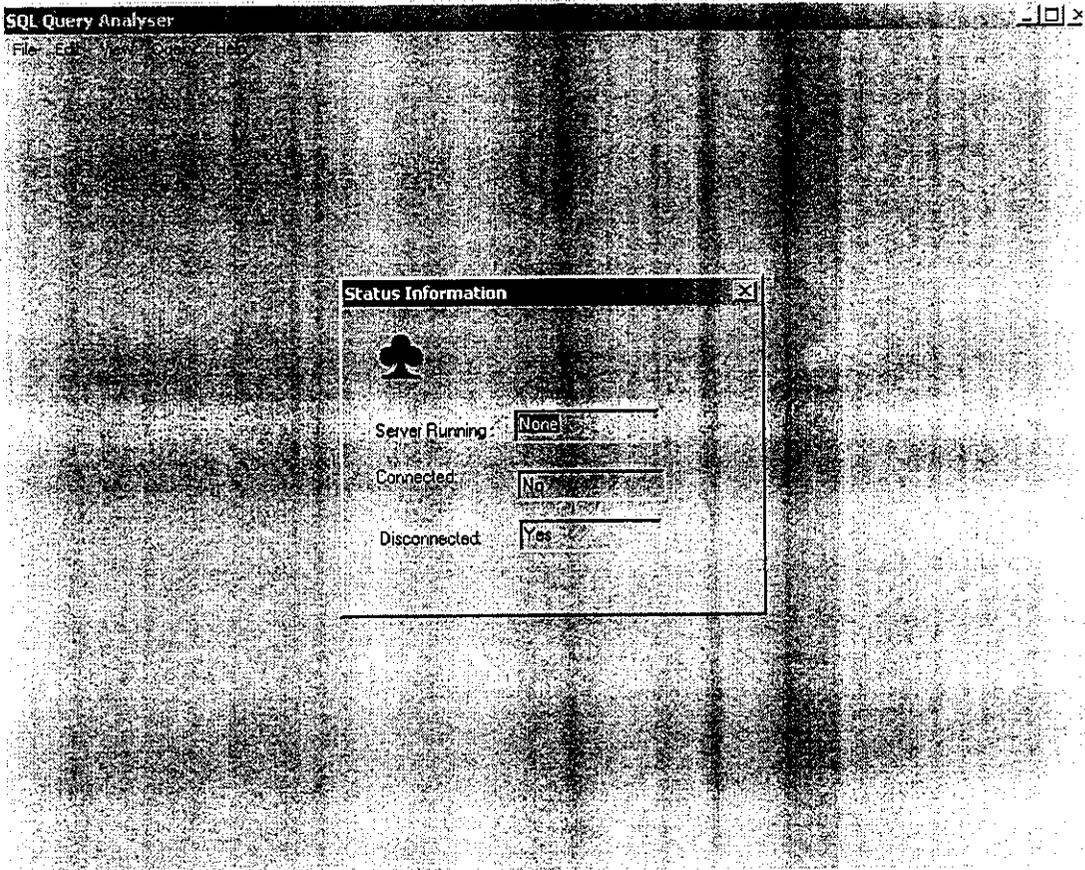
## Selecting the disconnect option



# About query analyzer



## Status information after disconnection



Exit from the application

