Module-7

28. Database.

A database is a separate application that stores a collection of data. Each database has one or more distinct APIs for creating, accessing, managing, searching and replicating the data it holds.

Other kinds of data stores can also be used, such as files on the file system or large hash tables in memory but data fetching and writing would not be so fast and easy with those type of systems.

Nowadays, we use relational database management systems (RDBMS) to store and manage huge volume of data. This is called relational database because all the data is stored into different tables and relations are established using primary keys or other keys known as Foreign Keys.

MYSQL:

MySQL is the most popular Open Source Relational SQL Database Management System. MySQL is one of the best RDBMS being used for developing various web-based software applications. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company.

A relational database is a type of database that stores and provides access to data points that are related to one another. Relational databases are based on the relational model, an intuitive, straightforward way of representing data in tables.

Keys are very important part of Relational database model. They are used to establish and identify relationships between tables and also to uniquely identify any record or row of data inside a table.

A Key can be a single attribute or a group of attributes, where the combination may act as a key.

Super Key is defined as a set of attributes within a table that can uniquely identify each record within a table. Super Key is a superset of Candidate key.

A super key could include student_id, (student_id, name), phone etc.

The first one is pretty simple as student_id is unique for every row of data, hence it can be used to identity each row uniquely.

Next comes, (student_id, name), now name of two students can be same, but their student_id can't be same hence this combination can also be a key.

Similarly, phone number for every student will be unique, hence again, phone can also be a key.

So they all are super keys.

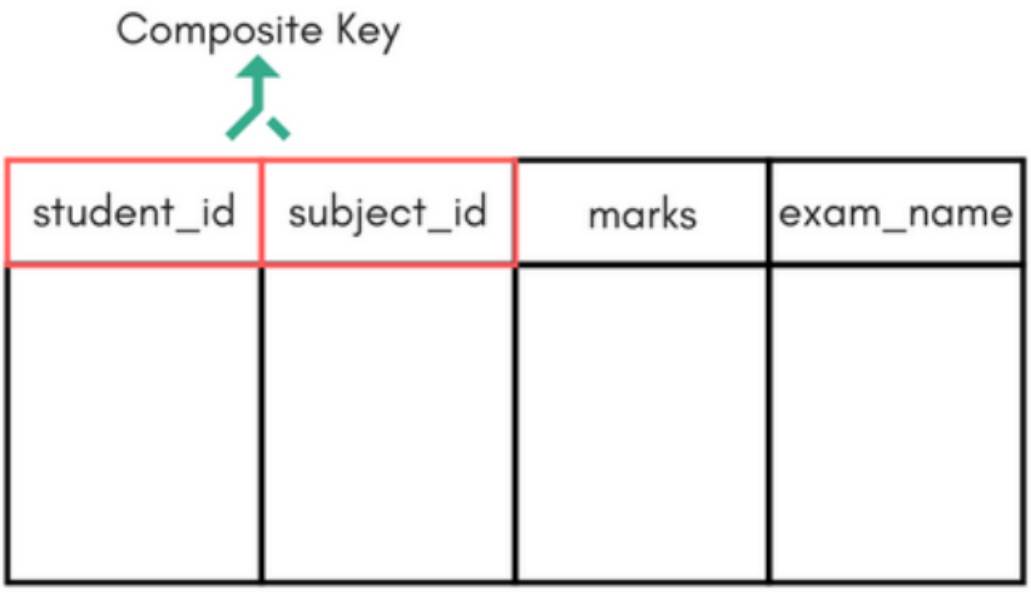
Candidate keys are defined as the minimal set of fields which can uniquely identify each record in a table. It is an attribute or a set of attributes that can act as a Primary Key for a table to uniquely identify each record in that table. There can be more than one candidate key.

In our example, student_id and phone both are candidate keys for table Student.

- A candiate key can never be NULL or empty. And its value should be unique.
- There can be more than one candidate keys for a table.
- A candidate key can be a combination of more than one columns (attributes).

Primary key is a candidate key that is most appropriate to become the main key for any table. It is a key that can uniquely identify each record in a table.

Key that consists of two or more attributes that uniquely identify any record in a table is called **Composite key**. But the attributes which together form the Composite key are not a key independently or individually.



Score Table - To save scores of the student for various subjects.

In the above picture we have a Score table which stores the marks scored by a student in a particular subject.

In this table student_id and subject_id together will form the primary key, hence it is a composite key.

The candidate key which are not selected as primary key are known as **secondary keys** or **alternative keys**.

Workbench Overview:

MySQL Workbench is a designing or a graphical tool, which is used for working with MySQL servers and databases. This tool compatible with the older server 5.x versions and does not support the 4.x server versions.

The functionalities of MySQL Workbench are as follows:

• **SQL Development:** This functionality provides the capability to execute SQL queries, create and manage connections to database servers using the built-in SQL Editor.

• **Data Modeling (Design):** This functionality enables you to create models of your database schema graphically, perform reverse and forward engineer between a schema and a live database, and edit all aspects of your database using the comprehensive Table Editor.

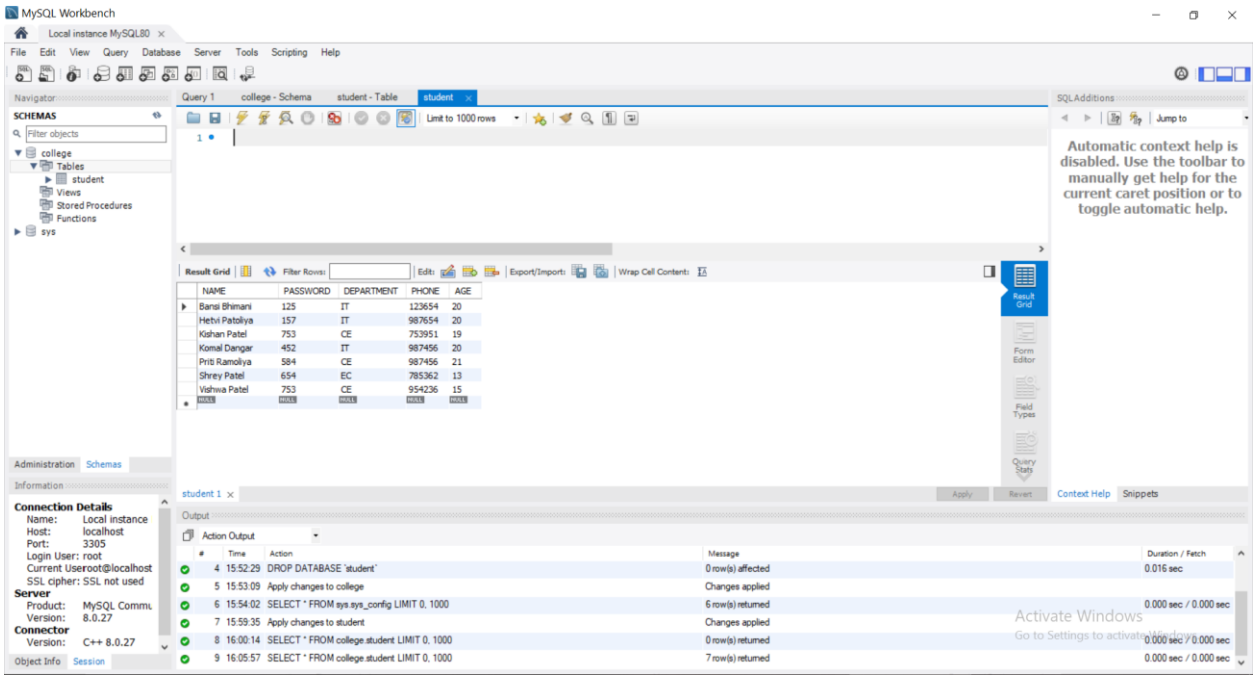
• **Server Administration:** This functionality enables you to administer MySQL server instances by administering users, performing backup and recovery, inspecting audit data, viewing database health, and monitoring the MySQL server performance.

• **Data Migration:** This functionality allows you to migrate from Microsoft SQL Server, Microsoft Access, and other RDBMS tables, objects, and data to MySQL.

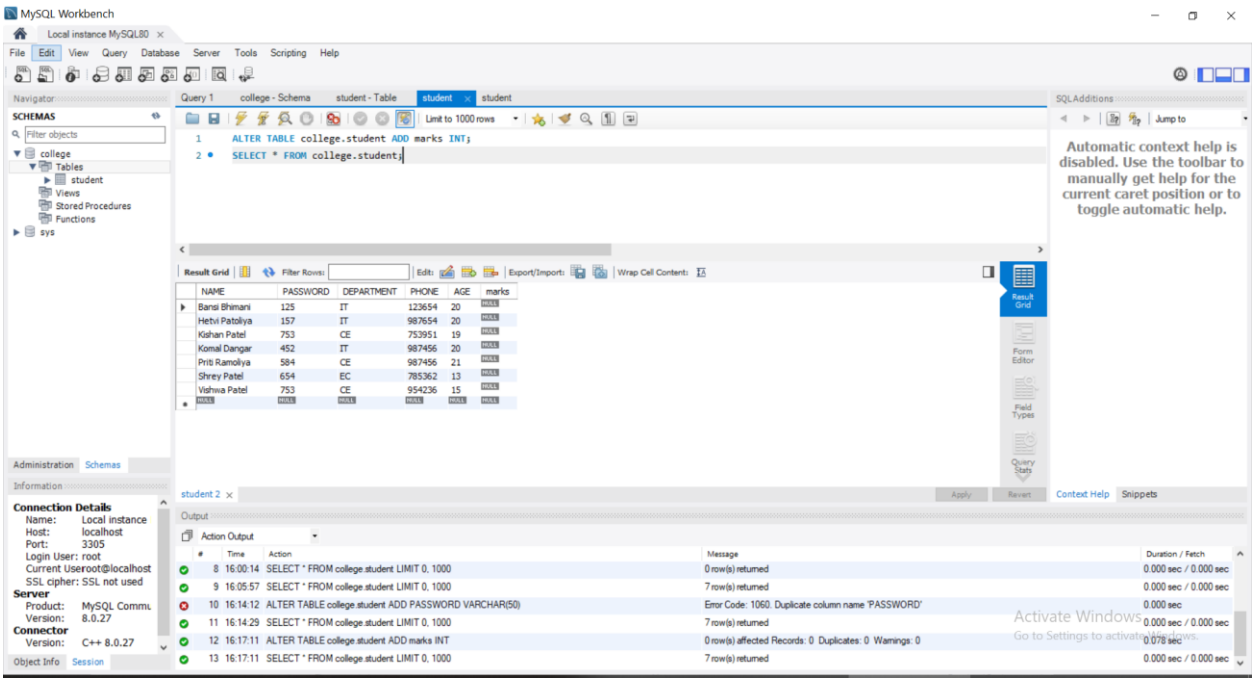
• **MySQL Enterprise Support:** This functionality provides support for Enterprise products such as MySQL Enterprise Backup, MySQL Firewall, and MySQL Audit.

In sql development I have fired various queries on student table of college database which is as shown below.

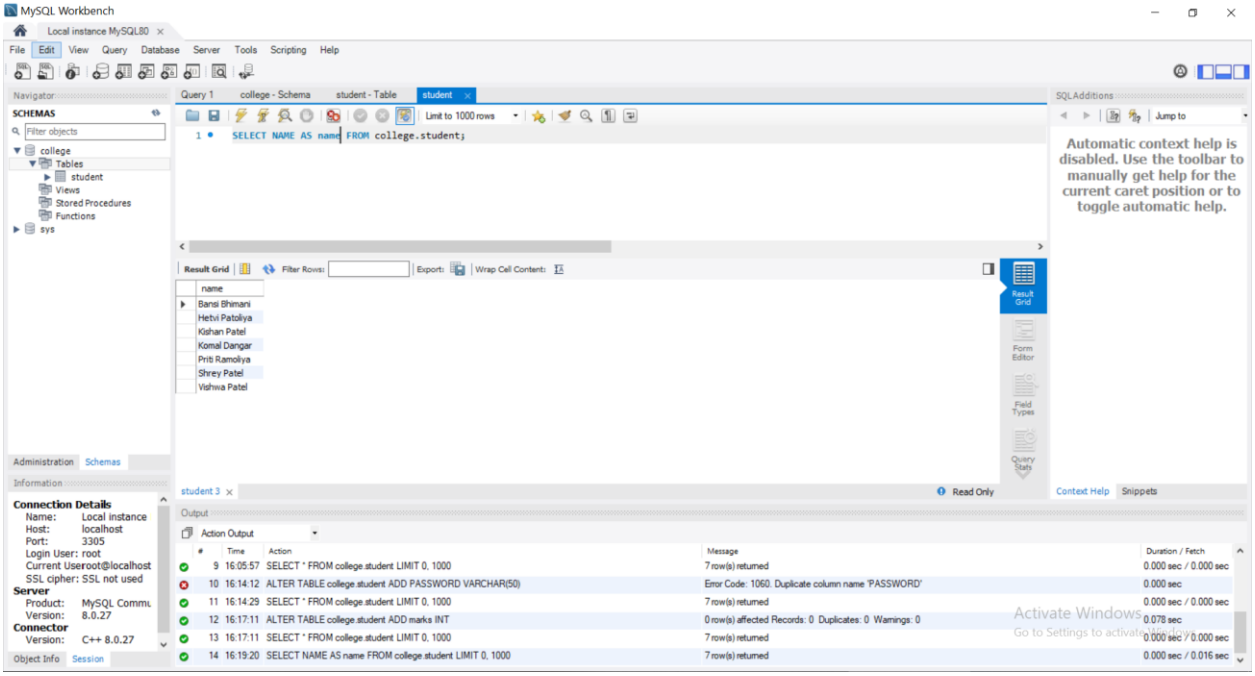
Table data:



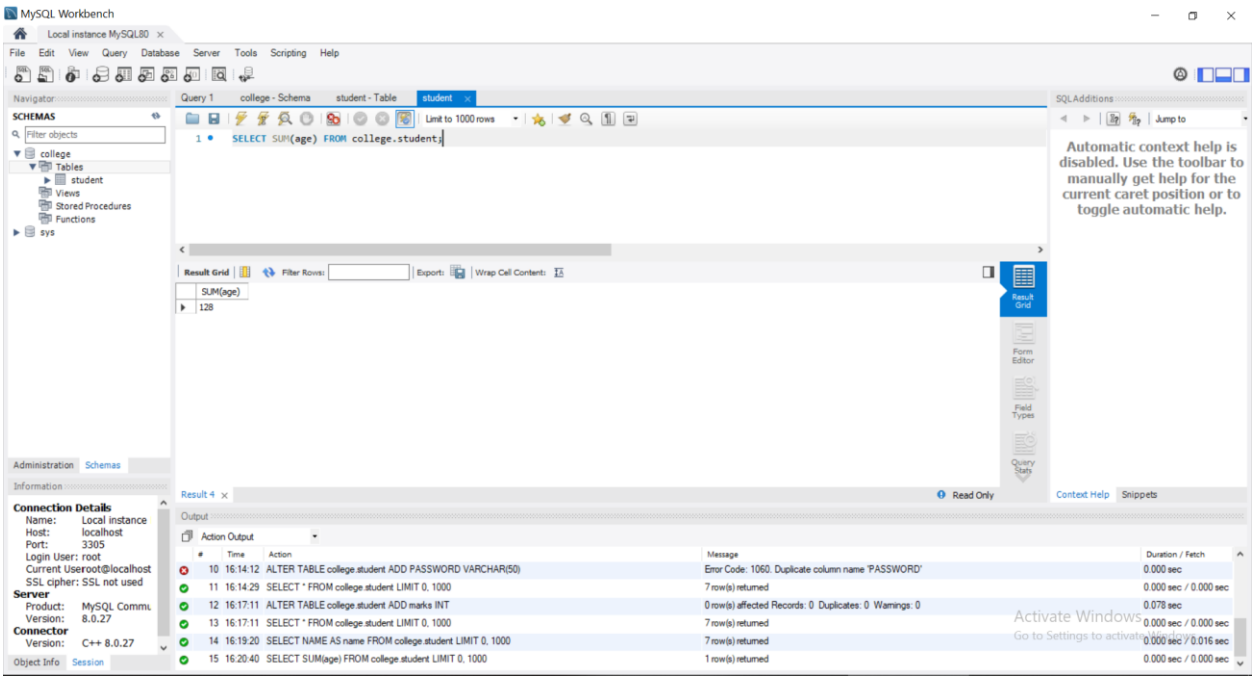
1. ALTER keyword



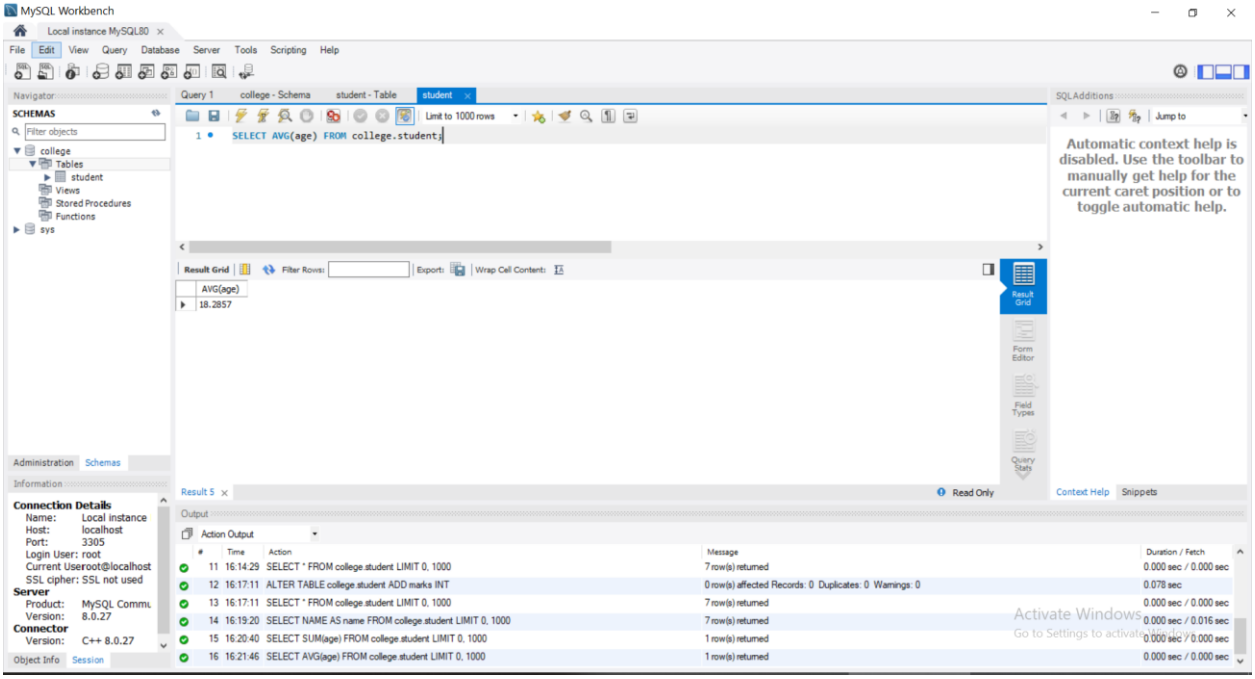
2. AS keyword



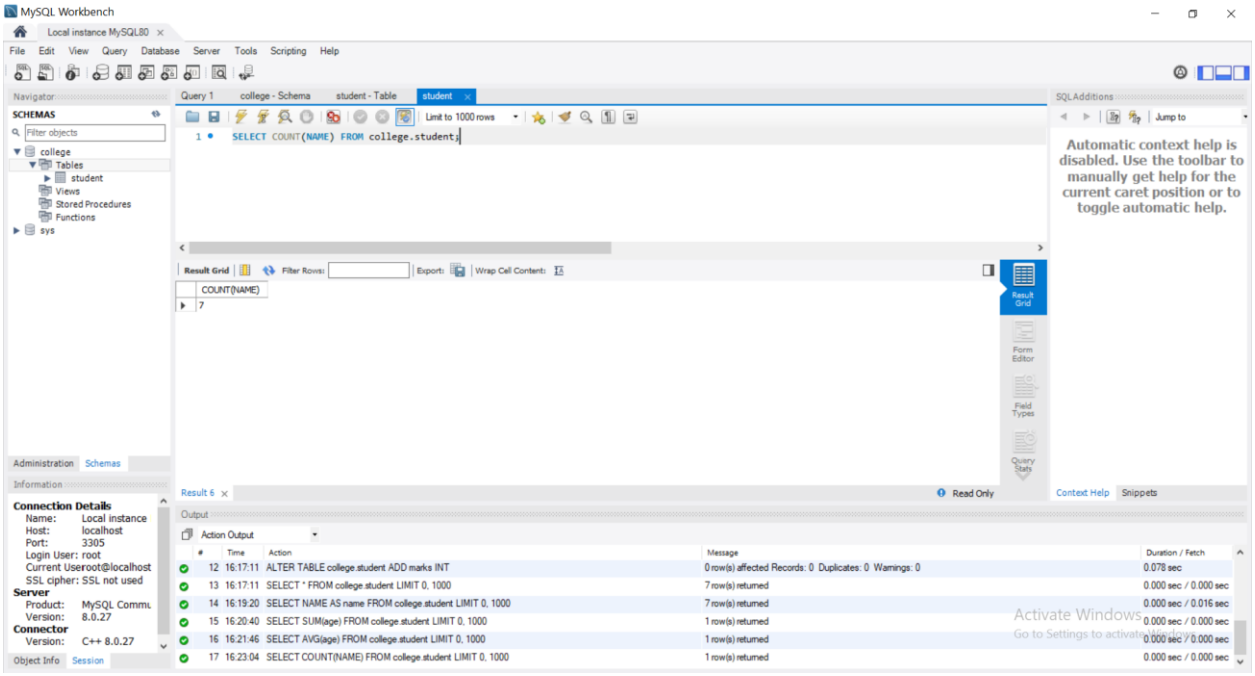
3. SUM keyword



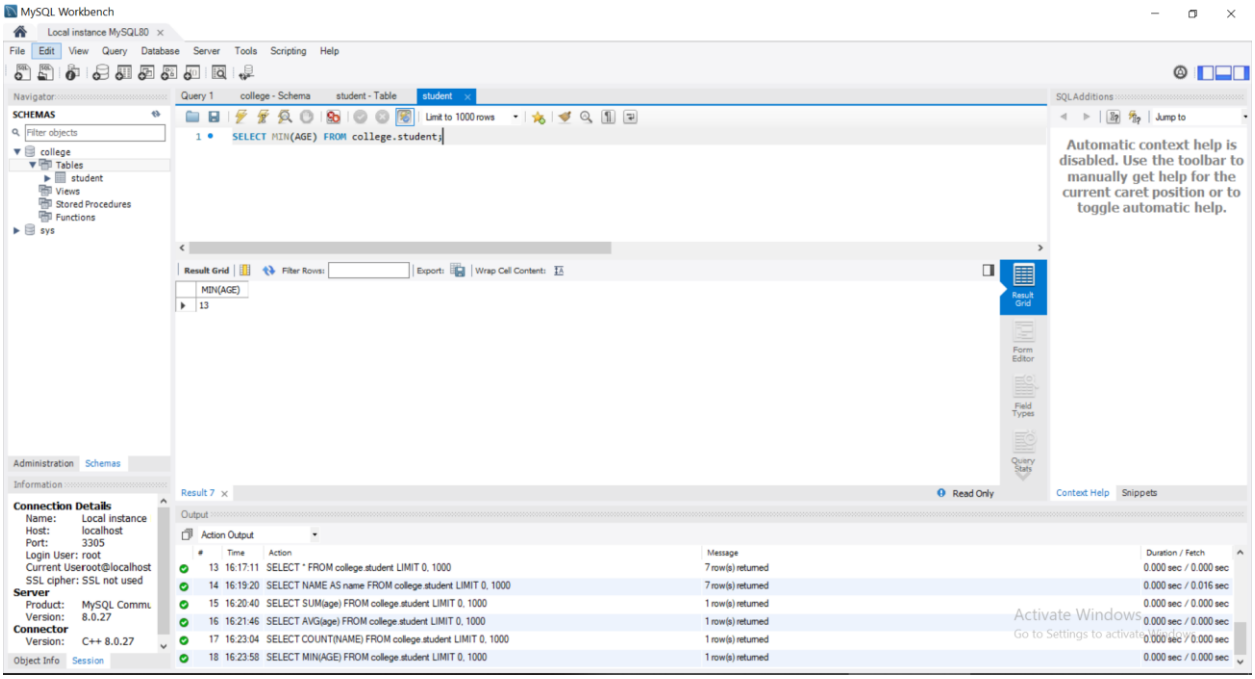
4. AVG keyword



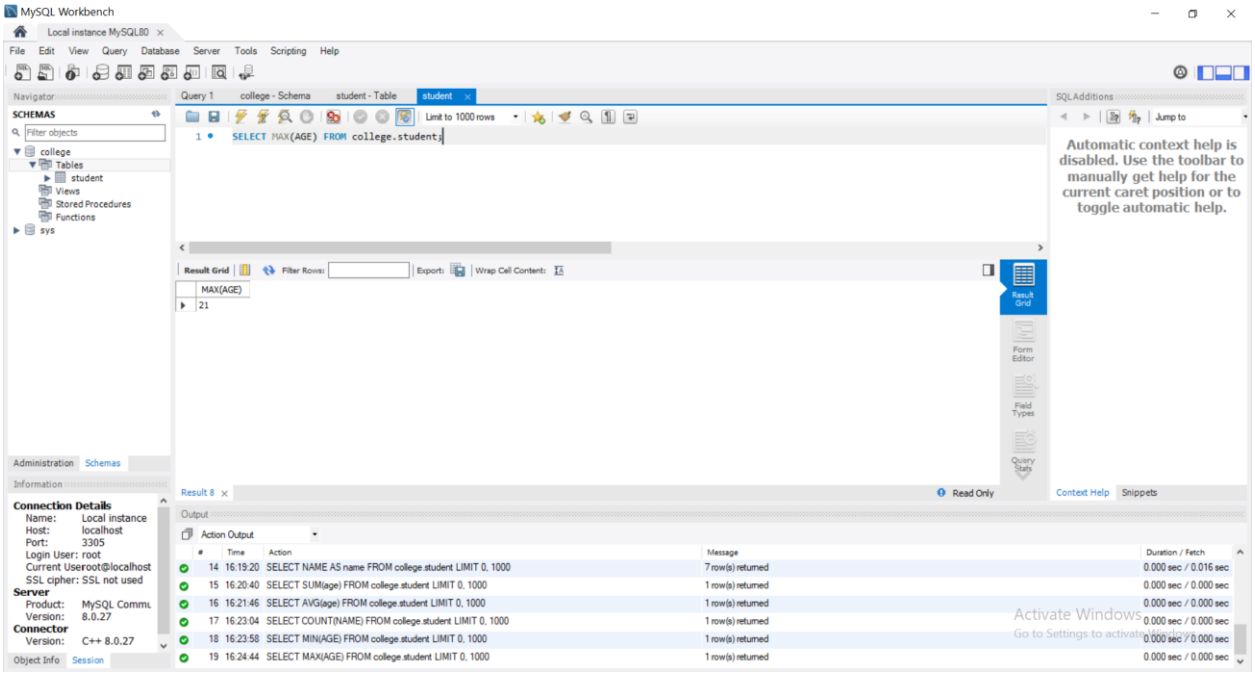
5. COUNT keyword



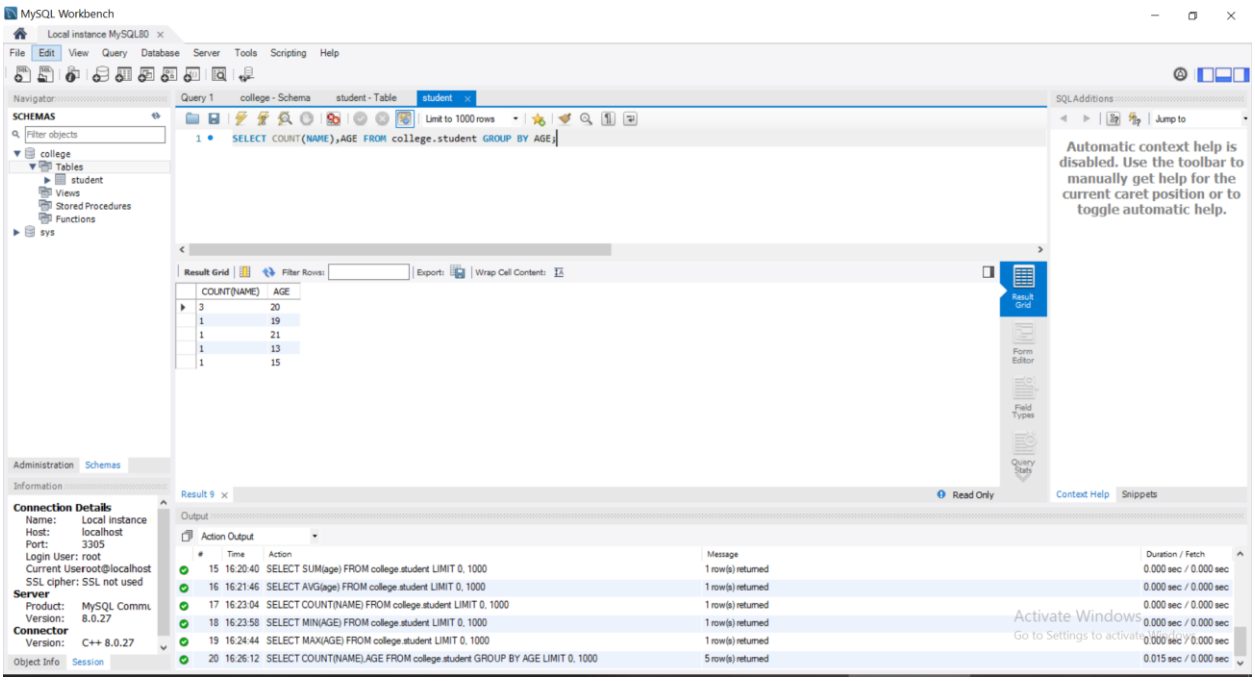
6. MIN keyword



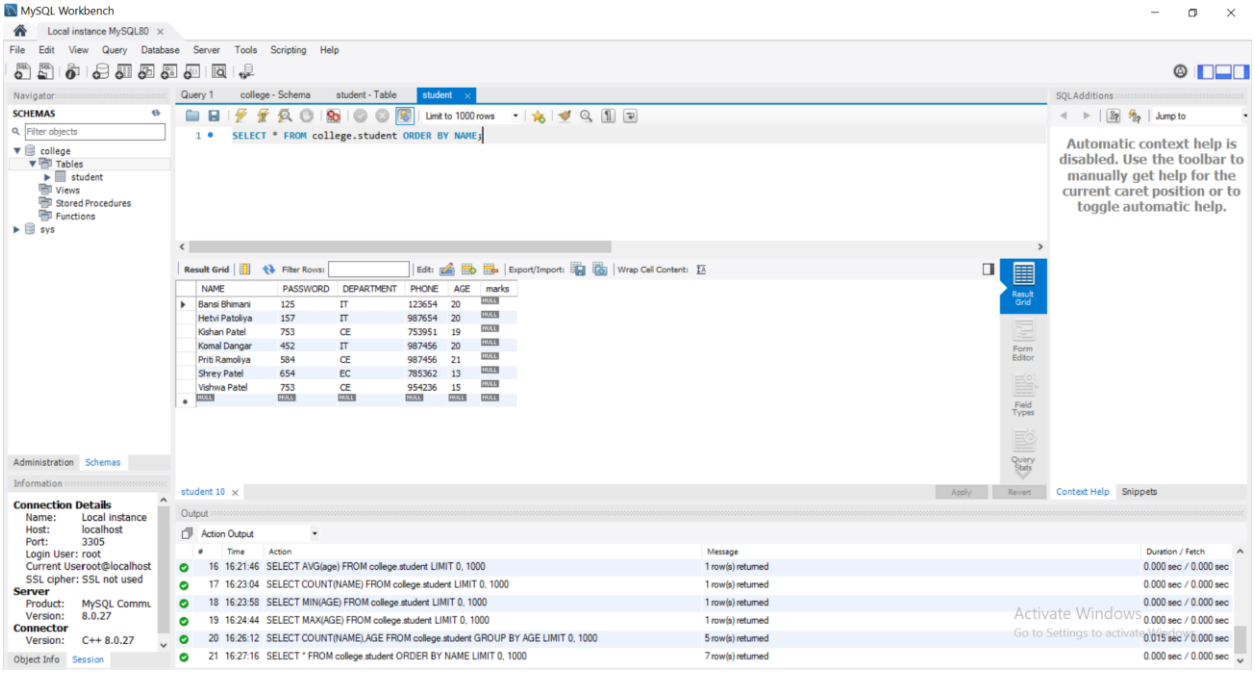
7. MAX keyword



8. GROUP BY keyword



9. ORDER BY keyword



Crude Operation:

I have made a very simple table in SQL Server Management Studio. Then, I have connected Sql with web form and performed crud operations. Its demonstration is shown here.

Form.aspx.cs:

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace CrudDemo
{
```

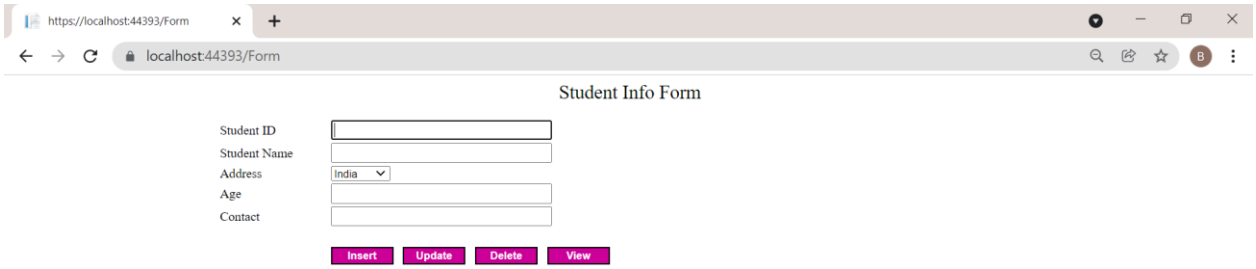
```
    public partial class Form : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }
        SqlConnection con = new SqlConnection("Data Source=DESKTOP-CNMD86D;Initial Catalog=DBDemo;Integrated
Security=True");
        protected void Button1_Click(object sender, EventArgs e)
        {
            con.Open();
            SqlCommand comm = new SqlCommand("Insert into StudentInfo
values('"+int.Parse(TextBox1.Text)+"','"+TextBox2.Text+"','"+DropDownList1.SelectedValue+"','"+double.Parse(TextBox3.Te
xt)+"','"+TextBox4.Text+"')",con);
            comm.ExecuteNonQuery();
            con.Close();
            ScriptManager.RegisterStartupScript(this, this.GetType(), "script", "alert('Successfully Inserted');", true);
        }
        protected void Button2_Click(object sender, EventArgs e)
        {
            con.Open();
            SqlCommand comm = new SqlCommand("Update StudentInfo set StudentName = '" + TextBox2.Text + "',Address = '"
+ DropDownList1.SelectedValue + "',Age = '" + double.Parse(TextBox3.Text) + "',Contact = '" + TextBox4.Text + "'Where
StudentID = '" + int.Parse(TextBox1.Text) + "'",con);
            comm.ExecuteNonQuery();
            con.Close();
            ScriptManager.RegisterStartupScript(this, this.GetType(), "script", "alert('Successfully Updated');", true);
        }

        protected void Button3_Click(object sender, EventArgs e)
        {
            con.Open();
            SqlCommand comm = new SqlCommand("Delete StudentInfo Where StudentID = '" + int.Parse(TextBox1.Text) + "'",
con);
            comm.ExecuteNonQuery();
            con.Close();
            ScriptManager.RegisterStartupScript(this, this.GetType(), "script", "alert('Successfully Deleted');", true);
        }

        protected void Button4_Click(object sender, EventArgs e)
        {
            SqlCommand comm = new SqlCommand("select * from StudentInfo", con);
            SqlDataAdapter d = new SqlDataAdapter(comm);
            DataTable dt = new DataTable();
            d.Fill(dt);
            GridView1.DataSource = dt;
            GridView1.DataBind();
        }
    }
}
```

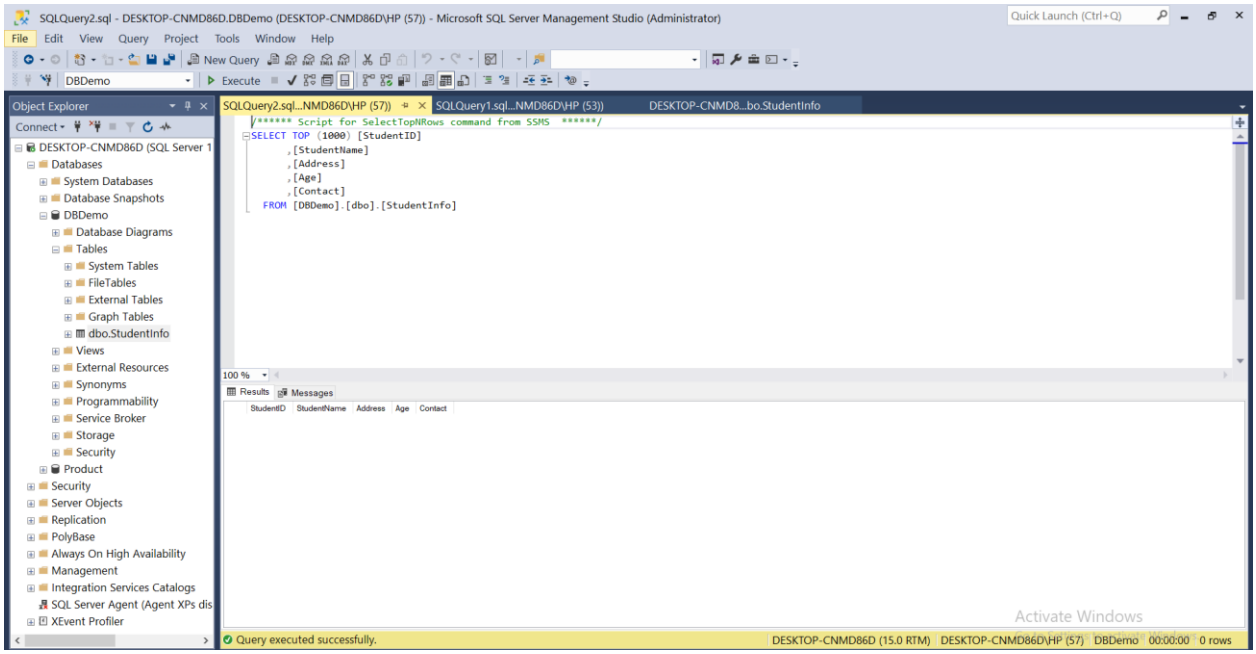
Form.aspx:(This is an auto generated code while I designed the form by drag and drop).

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Form.aspx.cs" Inherits="CrudDemo.Form" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
```

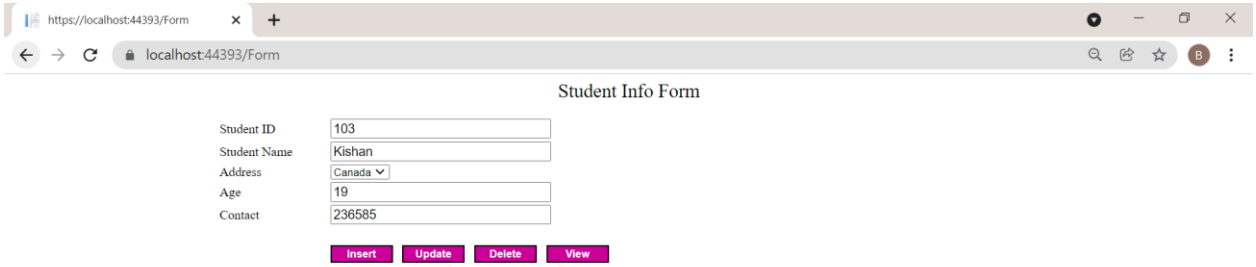
```
    <title></title>
    <style type="text/css">
        .auto-style1 {
            width: 100%;
        }
        .auto-style2 {
            width: 255px;
        }
        .auto-style3 {
            width: 135px;
        }
    </style>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <div style="font-size:x-large" align="center">Student Info Form</div>
            <br />

            <table class="auto-style1">
                <tr>
                    <td class="auto-style2"> </td>
                    <td class="auto-style3">Student ID</td>
                    <td>
                        <asp:TextBox ID="TextBox1" runat="server" Font-Size="Medium"  Width="268px"></asp:TextBox>
                    </td>
                </tr>
                <tr>
                    <td class="auto-style2"> </td>
                    <td class="auto-style3">Student Name</td>
                    <td>
                        <asp:TextBox ID="TextBox2" runat="server" Font-Size="Medium"  Width="268px"></asp:TextBox>
                    </td>
                </tr>
                <tr>
                    <td class="auto-style2"> </td>
                    <td class="auto-style3">Address</td>
                    <td>
                        <asp:DropDownList ID="DropDownList1" runat="server">
                            <asp:ListItem>India</asp:ListItem>
                            <asp:ListItem>USA</asp:ListItem>
                            <asp:ListItem>Canada</asp:ListItem>
                        </asp:DropDownList>
                    </td>
                </tr>
                <tr>
                    <td class="auto-style2"> </td>
                    <td class="auto-style3">Age</td>
                    <td>
                        <asp:TextBox ID="TextBox3" runat="server" Font-Size="Medium"  Width="268px"></asp:TextBox>
                    </td>
                </tr>
                <tr>
                    <td class="auto-style2"> </td>
                    <td class="auto-style3">Contact</td>
                    <td>
                        <asp:TextBox ID="TextBox4" runat="server" Font-Size="Medium"  Width="268px"></asp:TextBox>
                    </td>
                </tr>
```

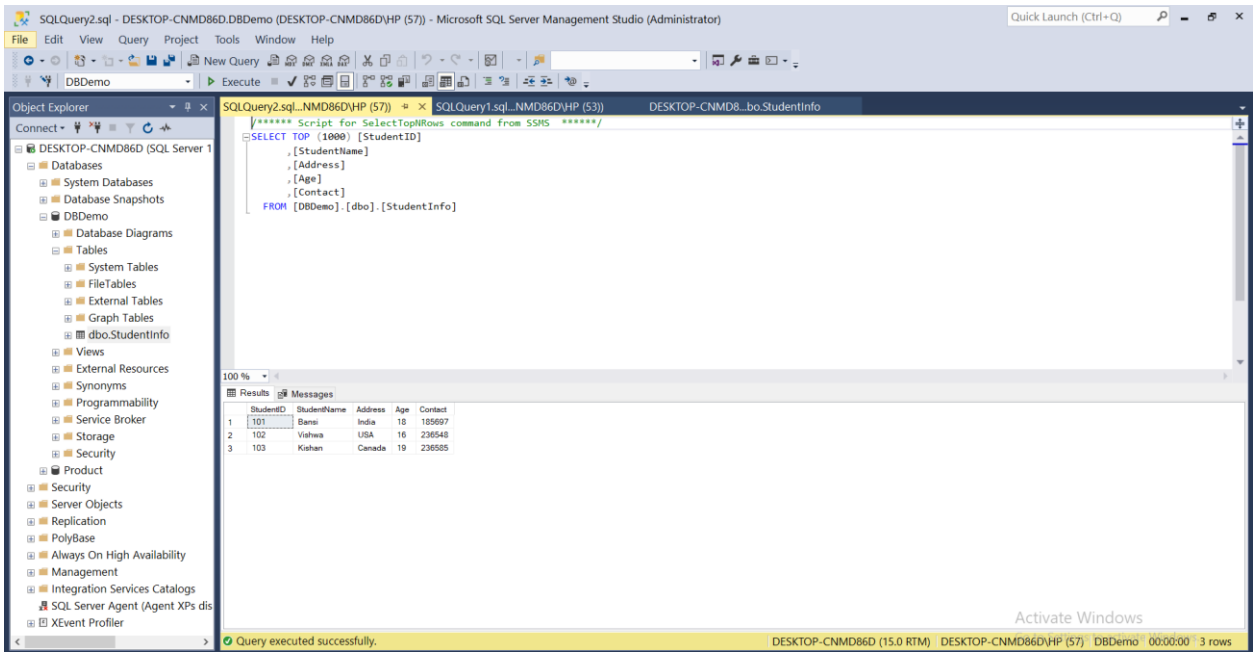
```
            <tr>
                <td class="auto-style2"> </td>
                <td class="auto-style3"> </td>
                <td> </td>
            </tr>
            <tr>
                <td class="auto-style2"> </td>
                <td class="auto-style3"> </td>
                <td>
                    <asp:Button ID="Button1" runat="server" BackColor="#CC0099" BorderColor="Black" Font-Bold="True"
ForeColor="White" OnClick="Button1_Click" Text="Insert" Width="78px" />
                 
                    <asp:Button ID="Button2" runat="server" BackColor="#CC0099" BorderColor="Black" Font-Bold="True"
ForeColor="White" OnClick="Button2_Click" Text="Update" Width="78px" />
                 
                    <asp:Button ID="Button3" runat="server" BackColor="#CC0099" BorderColor="Black" Font-Bold="True"
ForeColor="White" OnClick="Button3_Click" Text="Delete" Width="78px" />
                 
                    <asp:Button ID="Button4" runat="server" BackColor="#CC0099" BorderColor="Black" Font-Bold="True"
ForeColor="White" OnClick="Button4_Click" Text="View" Width="78px" />
                 </td>
            </tr>
            <tr>
                <td class="auto-style2"> </td>
                <td class="auto-style3"> </td>
                <td> </td>
            </tr>
            <tr>
                <td class="auto-style2"> </td>
                <td class="auto-style3"> </td>
                <td>
                    <asp:GridView ID="GridView1" runat="server" Width="683px">
                    </asp:GridView>
                </td>
            </tr>
        </table>

    </div>
  </form>
</body>
</html>
```



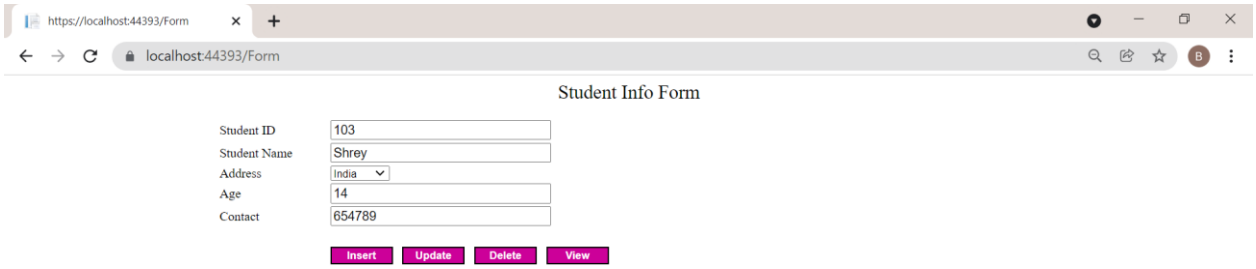

SQLQuery2.sql - DESKTOP-CNMD86D.DBDemo (DESKTOP-CNMD86D\HP (57)) - Microsoft SQL Server Management Studio (Administrator)
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
Execute
DBDemo
Object Explorer
Connect
DESKTOP-CNMD86D (SQL Server 1
Databases
System Databases
Database Snapshots
DBDemo
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.StudentInfo
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
Product
Security
Server Objects
Replication
PolyBase
Always On High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs dis
XEvent Profiler
SQLQuery2.sql...NMD86D\HP (57))
SQLQuery1.sql...NMD86D\HP (53))
DESKTOP-CNMD8...bo.StudentInfo
/****** Script for SelectTopNRows command from SSMS ******/
SELECT TOP (1000) [StudentID]
,[StudentName]
,[Address]
,[Age]
,[Contact]
FROM [DBDemo].[dbo].[StudentInfo]
100 %
Results
Messages
StudentID StudentName Address Age Contact
Activate Windows
Query executed successfully.
DESKTOP-CNMD86D (15.0 RTM) DESKTOP-CNMD86D\HP (57) DBDemo 00:00:00 0 rows

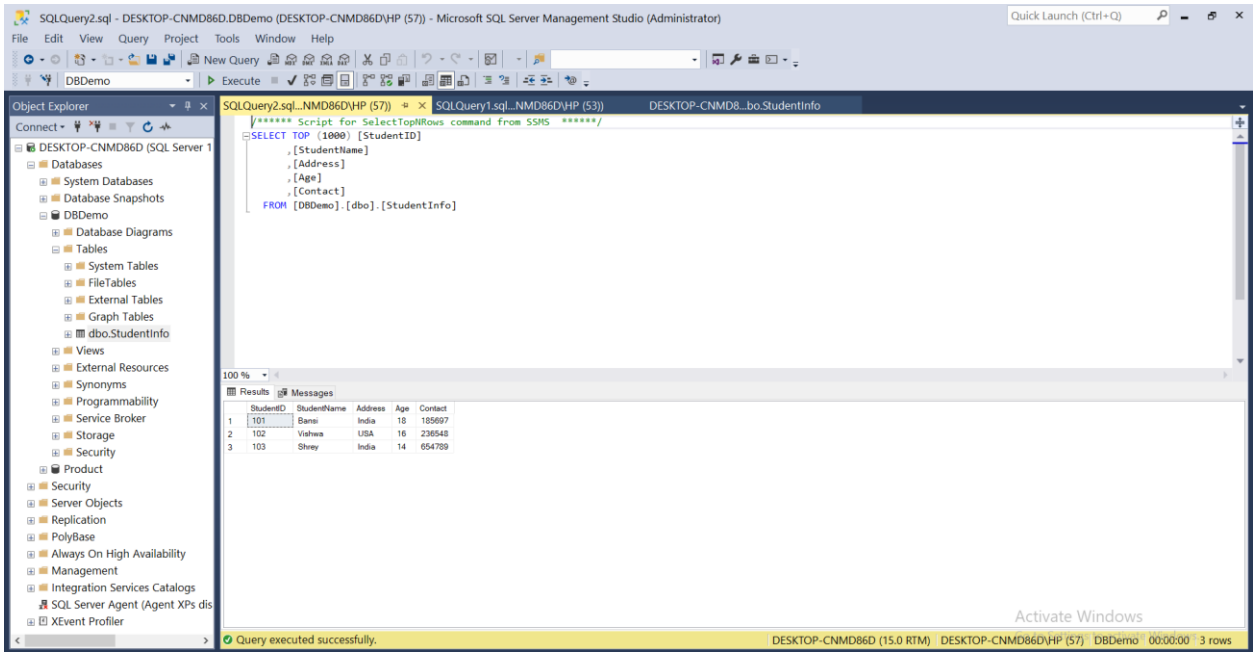
INSERT:


https://localhost:44393/Form
localhost:44393/Form
Student Info Form
Student ID 103
Student Name Kishan
Address Canada
Age 19
Contact 236585
Insert
Update
Delete
View

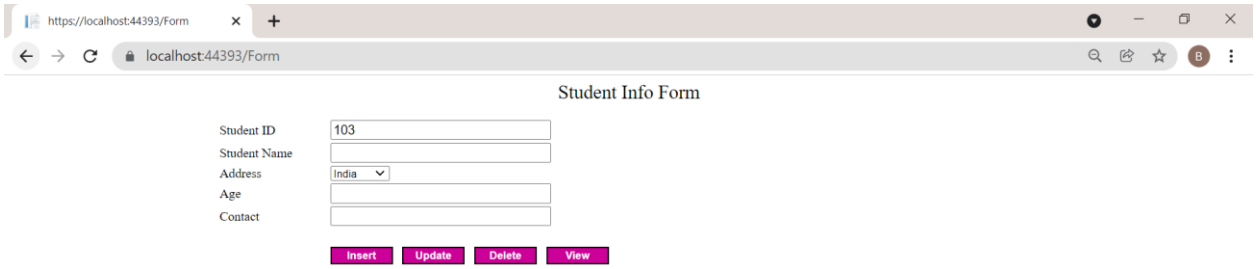

SQLQuery2.sql - DESKTOP-CNMD86D.DBDemo (DESKTOP-CNMD86D\HP (57)) - Microsoft SQL Server Management Studio (Administrator)
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
Execute
DBDemo
Object Explorer
Connect
DESKTOP-CNMD86D (SQL Server 1
Databases
System Databases
Database Snapshots
DBDemo
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.StudentInfo
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
Product
Security
Server Objects
Replication
PolyBase
Always On High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs dis
XEvent Profiler
SQLQuery2.sql...NMD86D\HP (57))
SQLQuery1.sql...NMD86D\HP (53))
DESKTOP-CNMD8...bo.StudentInfo
/****** Script for SelectTopNRows command from SSMS ******/
SELECT TOP (1000) [StudentID]
,[StudentName]
,[Address]
,[Age]
,[Contact]
FROM [DBDemo].[dbo].[StudentInfo]
100 %
Results
Messages
StudentID StudentName Address Age Contact
1 101 Bansi India 18 185697
2 102 Vishwa USA 16 236548
3 103 Kishan Canada 19 236585
Activate Windows
Query executed successfully.
DESKTOP-CNMD86D (15.0 RTM) DESKTOP-CNMD86D\HP (57) DBDemo 00:00:00 3 rows

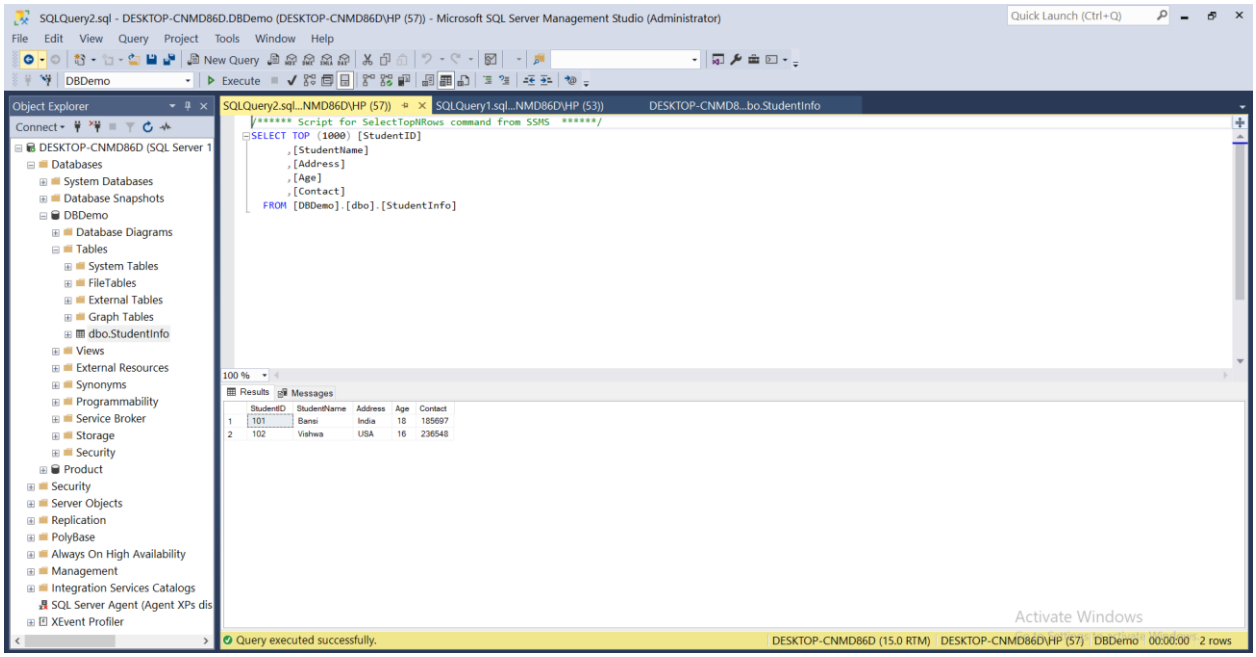
UPDATE:



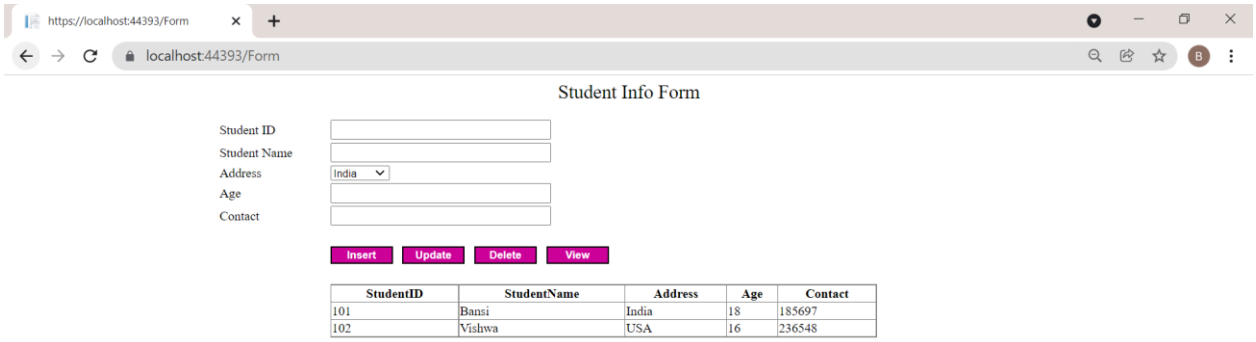


DELETE:




SQLQuery2.sql - DESKTOP-CNMD86D.DBDemo (DESKTOP-CNMD86D\HP (57)) - Microsoft SQL Server Management Studio (Administrator)
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
DBDemo
Execute
Object Explorer
Connect
DESKTOP-CNMD86D (SQL Server 1
Databases
System Databases
Database Snapshots
DBDemo
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.StudentInfo
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
Product
Security
Server Objects
Replication
PolyBase
Always On High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs dis
XEvent Profiler
SQLQuery2.sql...NMD86D\HP (57))
SQLQuery1.sql...NMD86D\HP (53))
DESKTOP-CNMD8...bo.StudentInfo
/****** Script for SelectTopNRows command from SSMS ******/
SELECT TOP (1000) [StudentID]
,[StudentName]
,[Address]
,[Age]
,[Contact]
FROM [DBDemo].[dbo].[StudentInfo]
100 %
Results
Messages
StudentID StudentName Address Age Contact
1 101 Bansi India 18 185697
2 102 Vishwa USA 16 236548
Activate Windows
Query executed successfully.
DESKTOP-CNMD86D (15.0 RTM)
DESKTOP-CNMD86D\HP (57)
DBDemo
00:00:00
2 rows

VIEW:


https://localhost:44393/Form
localhost:44393/Form
Student Info Form
Student ID
Student Name
Address
India
Age
Contact
Insert
Update
Delete
View
StudentID StudentName Address Age Contact
101 Bansi India 18 185697
102 Vishwa USA 16 236548