

Console Application – example

```
create database Music_DB
go
use Music_DB
go

create table RecordCompany(
RCid int primary key identity(1,1),
Name varchar(50)
)

Alter table RecordCompany
add budget float

select * from RecordCompany

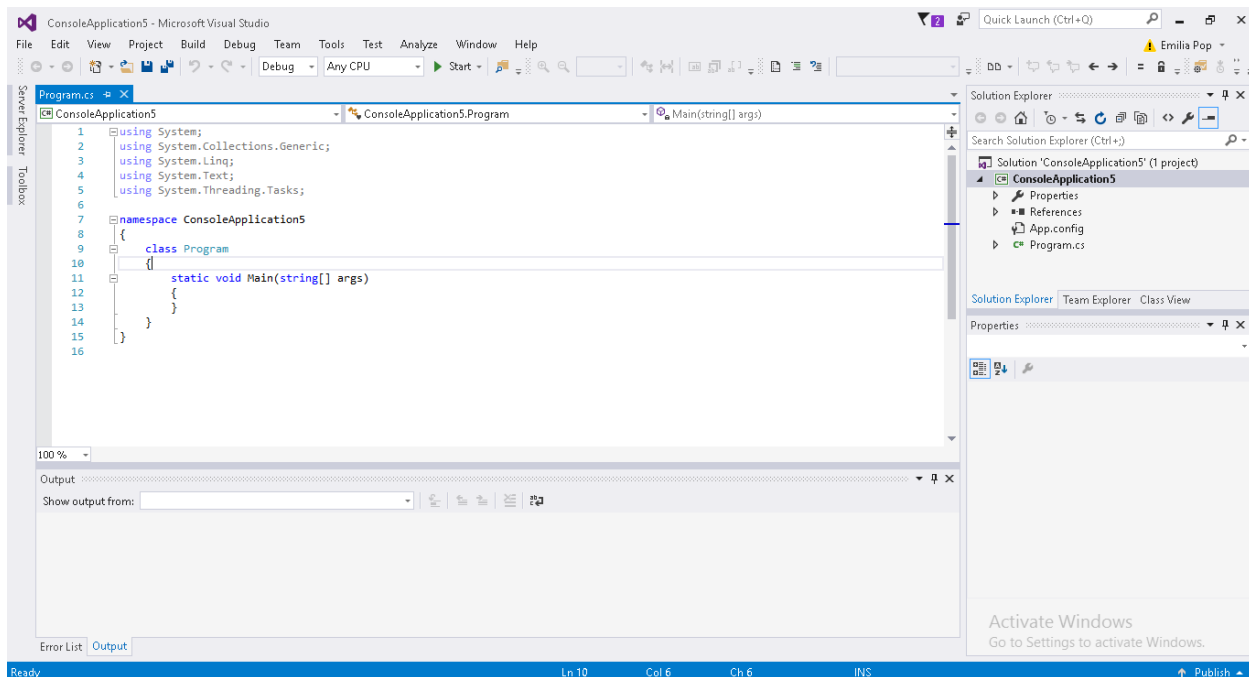
insert into RecordCompany values ('Record Company 1'),
('Record Company 2')

update RecordCompany
set budget=15.8

select * from RecordCompany
```

Results		Messages	
	RCid	Name	budget
1	1	Record Company 1	15.8
2	2	Record Company 2	15.8

Console Application (from Visual Studio)



Program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
```

```

using System.Threading.Tasks;
using System.Data.SqlClient;

namespace ConsoleApplication5
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.BackgroundColor = ConsoleColor.Green;
            Console.ForegroundColor = ConsoleColor.Blue;

            string connString = @"server=DESKTOP-ATJN5FL\SQLEXPRESS; Database=Music_DB;
Integrated Security=true";

            try
            {
                using (SqlConnection connection = new SqlConnection(connString))
                {
                    connection.Open();
                    Console.WriteLine($"The state of the connection: {connection.State}");
                    Console.WriteLine();

                    // SELECT
                    Console.WriteLine("SELECT: RecordCompany table: Name, budget");
                    SqlCommand selectCommand = new SqlCommand("select Name, budget from
RecordCompany", connection);
                    SqlDataReader reader = selectCommand.ExecuteReader();
                    //create SqlDataReader!!
                    //SqlDataReader block the connection as long as it is open => so, it must be
closed

                    while (reader.Read())
                    {
                        for (int i = 0; i < reader.FieldCount; i++)
                        {
                            Console.Write(reader[i] + ",");
                        }
                        Console.WriteLine();
                    }
                    reader.Close();
                    connection.Close();
                    Console.WriteLine();

                    // INSERT
                    connection.Open();
                    String Name = "Zone Studio";
                    float budget = 400.85F;
                    SqlCommand insertCommand = new SqlCommand("insert into RecordCompany(Name,
budget) values (@Name,@budget)", connection);
                    insertCommand.Parameters.AddWithValue("@Name", Name);
                    insertCommand.Parameters.AddWithValue("@budget", budget);
                    int rows = insertCommand.ExecuteNonQuery();
                    Console.WriteLine($"INSERT: The number of rows inserted is: {rows}");
                    connection.Close();
                    // Select again after insert :)
                    connection.Open();
                    reader = selectCommand.ExecuteReader();

```

```

while (reader.Read())
{
    for (int i = 0; i < reader.FieldCount; i++)
    {
        Console.Write(reader[i] + ",");
    }
    Console.WriteLine();
}
reader.Close();
connection.Close();
Console.WriteLine();

// UPDATE
connection.Open();
SqlCommand updateCommand = new SqlCommand("update RecordCompany set
Name=@Name_New, budget=@budget where Name=@Name", connection);
String Name_New = "Studio UPDATED";
float budget_New = 550.0F;
updateCommand.Parameters.AddWithValue("@Name_New", Name_New);
updateCommand.Parameters.AddWithValue("@budget", budget_New);
updateCommand.Parameters.AddWithValue("@Name", Name);
int rows2 = updateCommand.ExecuteNonQuery();
Console.WriteLine($"UPDATE: The number of affected rows is: {rows2}");
connection.Close();
// Select again after Update :)
connection.Open();
reader = selectCommand.ExecuteReader();
while (reader.Read())
{
    for (int i = 0; i < reader.FieldCount; i++)
    {
        Console.Write(reader[i] + ",");
    }
    Console.WriteLine();
}
reader.Close();
connection.Close();
Console.WriteLine();

//DELETE
connection.Open();
SqlCommand deleteCommand = new SqlCommand("delete from RecordCompany where
Name=@Name", connection);
String Name_Deleted = "Studio Updated";
deleteCommand.Parameters.AddWithValue("@Name", Name_Deleted);
int rows3 = deleteCommand.ExecuteNonQuery();
Console.WriteLine($"DELETE: The number of deleted rows is: {rows3}");
connection.Close();
// Select again after Delete :)
connection.Open();
reader = selectCommand.ExecuteReader();
while (reader.Read())
{
    for (int i = 0; i < reader.FieldCount; i++)
    {
        Console.Write(reader[i] + ",");
    }
}

```

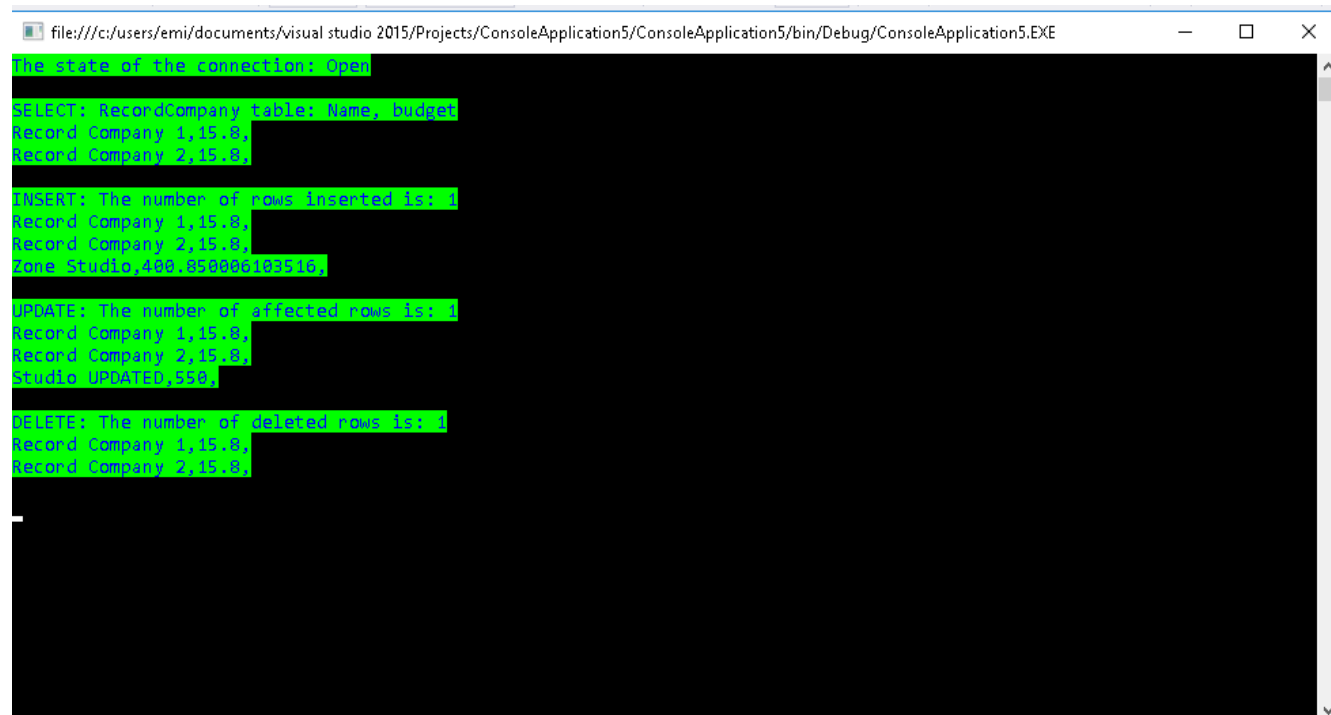
```

        }
        Console.WriteLine();
    }
    reader.Close();
    connection.Close();
    Console.WriteLine();

    Console.ReadKey();
}

}
catch (Exception ex)
{
    Console.ForegroundColor = ConsoleColor.Red;
    Console.WriteLine($"The error message is: \n {ex.Message}");
    Console.ReadKey();
}
}
}
}

```



The screenshot shows a console window titled "file:///c:/users/emi/documents/visual studio 2015/Projects/ConsoleApplication5/ConsoleApplication5/bin/Debug/ConsoleApplication5.EXE". The output is as follows:

```

The state of the connection: Open
SELECT: RecordCompany table: Name, budget
Record Company 1,15.8,
Record Company 2,15.8,

INSERT: The number of rows inserted is: 1
Record Company 1,15.8,
Record Company 2,15.8,
Zone Studio,400.850000103516,

UPDATE: The number of affected rows is: 1
Record Company 1,15.8,
Record Company 2,15.8,
Studio UPDATED,550,

DELETE: The number of deleted rows is: 1
Record Company 1,15.8,
Record Company 2,15.8,

```