# BÀI THỰC HÀNH SỐ 7 (4 tiết)

## Form - Thao tác với cơ sở dữ liệu SQL

#### I. Mục tiêu:

- Hiểu được cách kết nối cở sở dữ liệu
- Sử dụng các lớp kết nối và lớp không kết nối

### II. Hướng dẫn thực hành:

- 1. Sinh viên xem bài demo: Customers trong thư mục bài lab.
- 2. Các bước thực hiện

Bước 1: Sử dụng Connection để kết nối cơ sở dữ liệu

Bước 2: Thiết lập câu lệnh thực thi: Insert, Select, Update

Bước 3: Thực hiện lệnh:

- Mở kết nối
- Thực thi câu lệnh, xử lý dữ liệu trả về
- Đóng kết nối

#### 3. Mô tả chi tiết

begin

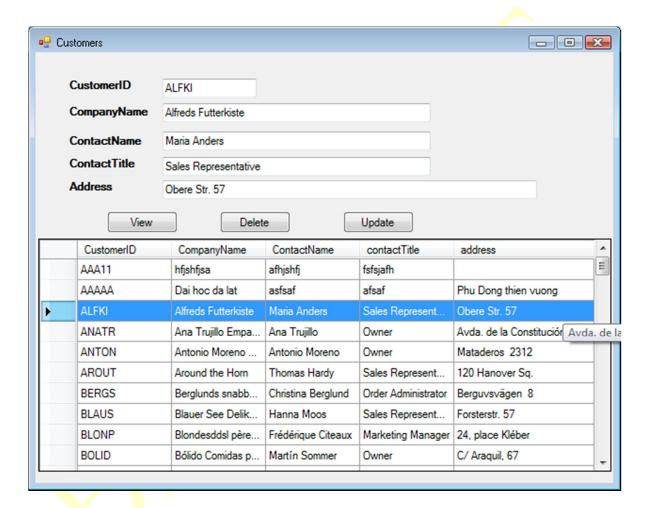
STT	Loại control	Tên thuộc t <mark>ín</mark> h	Giá trị
1	TextBox	Name	txtCustomerID
2	TextBox	Name	txtCompanyName
3	TextBox	Name	txtContactName
4	TextBox	Name	txtContactTitle
5	TextBox	Name	txtAddress
6	DataGridView	Name	dataGridViewCustomer
7	Button	Name	btnUpdate
8	Button	Name	btnDelete
9	Button	Name	btnView

Sử dụng store procedure (Sinh viên tạo store procedure như sau) CREATE PROCEDURE UpdateCustomers

```
( @customerID nchar(5),
@companyName nvarchar(40),
@contactName nvarchar(30),
@contactTitle nvarchar(30),
@address nvarchar(60),
@kq int output
)
AS
```

```
 \begin{tabular}{ll} if ( \ not \ exists \ ( \ select \ customerID \ from \ Customers \ where \ CustomerID= @ customerID)) \\ set \ @kq = 0 \\ else \\ begin \end{tabular}
```

**Update** Customers



## Hướng dẫn viết code:

```
using System.Data;
using System.Data.SqlClient;
using System.Data.Common;
namespace Customers
{
    public partial class frmCustomers : Form
    {
        SqlConnection myConnection;
}
```

```
SqlCommand;
        public frmCustomers()
            InitializeComponent();
    private void Form1_Load(object sender, EventArgs e)
        string strconnect = "server=(local); database=northwind;
        integrated security=true";
        myConnection = new SqlConnection(strconnect);
        myCommand = myConnection.CreateCommand();
        btnView.PerformClick();
//Khi click lên 1 dòng DataGridView, dòng dữ liệu sẽ hiển thị lên textbox
private void dataGridViewCustomer_Click(object sender, EventArgs e)
    if (this.dataGridViewCustomer.SelectedRows.Count > 0)
    {
        DataGridViewRow drow = this.dataGridViewCustomer.SelectedRows[0];
        DataRowView row = drow.DataBoundItem as DataRowView;
        this.txtCustomerID.Text = row[0].ToString();
        this.txtCompanyName.Text = row[1].ToString();
        this.txtContactName.Text = row[2].ToString();
        this.txtContactTitle.Text = row[3].ToString();
        this.txtAddress.Text = row[4].ToString();
    }
//Khi update dữ liệu sẽ update dưới cở sở dữ liệu và được load lại lên
DataGridView
private void btnUpdate_Click(object sender, EventArgs e)
{
    try
    {
        myCommand.CommandText = "UpdateCustomers";
        myCommand.CommandType = CommandType.StoredProcedure;
        SqlParameter paraCustomerID = new SqlParameter("@CustomerID",
        SqlDbType.NChar, 5);
        paraCustomerID.Value = this.txtCustomerID.Text;
        myCommand.Parameters.Add(paraCustomerID);
        SqlParameter paraCompanyName = new SqlParameter("@CompanyName",
        SqlDbType.NVarChar, 40);
        paraCompanyName.Value = this.txtCompanyName.Text;
        myCommand.Parameters.Add(paraCompanyName);
```

```
SqlParameter paraContactName = new SqlParameter("@ContactName",
       SqlDbType.NVarChar, 30);
       paraContactName.Value = this.txtContactName.Text;
       myCommand.Parameters.Add(paraContactName);
        SqlParameter paraContactTitle = new SqlParameter("@ContactTitle",
       SqlDbType.NVarChar, 30);
       paraContactTitle.Value = this.txtContactTitle.Text;
       myCommand.Parameters.Add(paraContactTitle);
        SqlDbType.NVarChar, 60);
       paraAddress.Value = this.txtAddress.Text;
       myCommand.Parameters.Add(paraAddress);
        SqlParameter parakq = new SqlParameter("@kq", SqlDbType.Int, 4);
       parakq.Direction = ParameterDirection.Output;
       myCommand.Parameters.Add(parakq);
        this.myConnection.Open();
       myCommand.ExecuteNonQuery();
        if ((int)parakq.Value == 0)
           MessageBox.Show("Khong Update duoc");
        else
           MessageBox.Show("Update thanh cong");
        this.myConnection.Close();
       btnView.PerformClick();
    }
   catch
    {
       MessageBox.Show("Loi ket noi");
//Khi click vào View load dữ liệu lên DataGridView
private void btnView_Click(object sender, EventArgs e)
   myCommand.CommandText = "select CustomerID, CompanyName,"
      + " ContactName, contactTitle, address"
      + " From customers";
   myCommand.CommandType = CommandType.Text;
    try
    {
        if (this.myConnection.State == ConnectionState.Closed)
           this.myConnection.Open();
        SqlDataAdapter myDataAdapter = new SqlDataAdapter(myCommand);
        DataSet ds = new DataSet();
       myDataAdapter.Fill(ds, "Customers");
        this.dataGridViewCustomer.DataSource = ds.Tables[0];
```

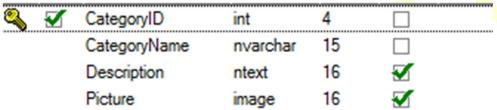
```
//ds.Tables["Customers"]
    this.myConnection.Close();

}
catch
{
    MessageBox.Show("Loi noi ket");
}
}
```

III. Bài tập:

1. Thiết kế form hiển thị, thêm, xóa, sửa cho bảng Categories.

Yêu cầu: thêm, xóa, sửa viết bằng store procedure.



2. Thiết kế form hiển thị, thêm, xóa, sửa cho bảng Products.

Yêu cầu: thêm, xóa, sửa viết bằng store procedure. Khi thêm cho chọn CategoryID từ bảng Categories, SupplierID từ bảng Suppliers

Design Table 'Products' in 'Northwind' on '(LOCAL)'						
	Column Name	Data Type	Length	Allow Nulls		
å	ProductID	int	4			
	ProductName	nvarchar	40			
	SupplierID	int	4	<b>V</b>		
	CategoryID	int	4	<b>V</b>		
	QuantityPerUnit	nvarchar	20	<b>V</b>		
	UnitPrice	money	8	<b>V</b>		
	UnitsInStock	smallint	2	V		
	UnitsOnOrder	smallint	2	<b>V</b>		
	ReorderLevel	smallint	2	V		
	Discontinued	bit	1			