
Chương 10

XÂY DỰNG ỨNG DỤNG DÙNG C# KẾT HỢP SQL SERVER

Mục tiêu

- Sau khi học xong chương này, SV có thể:
 - Biết mô hình ADO.Net và các đối tượng trong ADO.Net
 - Nắm được quy trình kết nối đến CSDL
 - Sử dụng được đối tượng Connection, Command và các đối tượng khác.
 - Xây dựng được một ứng dụng đơn giản dùng ngôn ngữ C# kết nối với CSDL trong SQL Server.

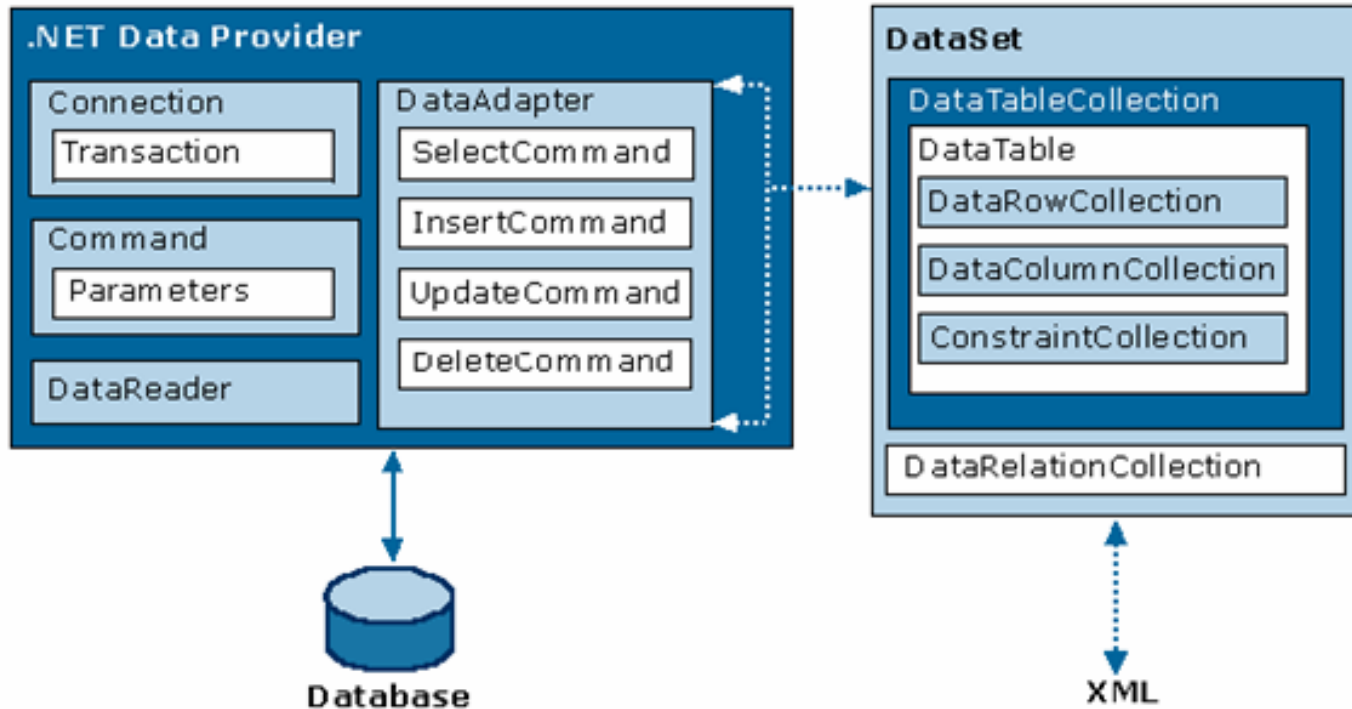
Nội dung chi tiết

- Giới thiệu ADO.Net
- Các đối tượng trong ADO.NET
- Quy trình kết nối đến CSDL
- Đối tượng Connection
- Đối tượng Command
- Ví dụ minh họa

Giới thiệu ADO.NET

- Là mô hình cho phép người lập trình thao tác với nhiều loại CSDL (SQL Server, MS Access, Oracle, ...) theo cùng một cách.
- Ứng với mỗi loại CSDL, sẽ có các thư viện .NET Framework Data Provider thích hợp, chứa các đối tượng để thao tác với CSDL:
 - Connection
 - Command
 - DataReader
 - Data Table
 - DataAdapter => DataSet.

Các đối tượng trong ADO.NET



Các đối tượng trong ADO.NET

- **Connection**: dùng để kết nối với CSDL.
- **Command**: dùng để thực hiện các câu lệnh truy vấn, gọi thực hiện các Stored Procedure.
- **DataTable**: chứa dữ liệu dạng bảng trên ứng dụng.
- **DataSet**: chứa nhiều DataTable và các Constraint trên ứng dụng.
- **DataAdapter**: đối tượng trung gian lấy dữ liệu trả về từ Command đưa vào DataTable và lấy dữ liệu từ DataTable đưa xuống cho đối tượng Command thực hiện thao tác cập nhật dữ liệu.

Quy trình kết nối đến CSDL

- Để thao tác với CSDL trong SQL Server, ta sử dụng thư viện **System.Data.SqlClient**.
- **Quy trình kết nối ứng dụng đến CSDL:**
 1. Tạo đối tượng Connection và xác định chuỗi kết nối (Connection String).
 2. Tạo đối tượng Command và xác định câu lệnh SQL.
 3. Mở đối tượng Connection (Open).
 4. Thực hiện câu lệnh SQL và xử lý kết quả (Execute).
 5. Đóng đối tượng Connection (Close).

Đối tượng Connection

- Khai báo và khởi tạo đối tượng SqlConnection:
 - Sử dụng user login của Windows:

//Khai báo sử dụng thư viện chứa các đối tượng ADO.net
using System.Data.SqlClient;

//Tạo chuỗi thông tin kết nối đến CSDL với phương thức login bằng
user login của Windows

**string ChuoiKetNoi = "Server=Sever_name; Database=Database_name;
Trusted_Connection=true ";**

//Khai báo và khởi tạo đối tượng kết nối

SqlConnection KetNoi = new SqlConnection(ChuoiKetNoi);

Đối tượng Connection

■ Khai báo và khởi tạo đối tượng SqlConnection:

- Sử dụng user login của Windows:

- VD:

//Khai báo thư viện chứa các đối tượng ADO.net

using System.Data.SqlClient;

//Tạo chuỗi thông tin kết nối đến CSDL QLSinhVien với phương thức login bằng user login của Windows

string ChuoiKetNoi = "Server=.\sqlexpress; Database=QLSinhVien;

Trusted_Connection=true";

//Khai báo và khởi tạo đối tượng kết nối

SqlConnection KetNoi = new SqlConnection(ChuoiKetNoi);

Đối tượng Connection

- Khai báo và khởi tạo đối tượng SqlConnection:
 - Sử dụng user login của SQL Sever:

//Khai báo thư viện chứa các đối tượng ADO.net

using System.Data.SqlClient;

//Tạo chuỗi thông tin kết nối đến CSDL với phương thức login bằng user login của SQL Sever

string ChuoiKetNoi = "Server=Sever_name; Database=Database_name;

User ID=YourUserID; Password=YourPassword";

//Khai báo và khởi tạo đối tượng kết nối

SqlConnection KetNoi = new SqlConnection(ChuoiKetNoi);

Đối tượng Connection

■ Khai báo và khởi tạo đối tượng SqlConnection:

- Sử dụng user login của SQL Sever:

- VD:

//Khai báo thư viện chứa các đối tượng ADO.net

using System.Data.SqlClient;

//Tạo chuỗi thông tin kết nối đến CSDL QLSinhVien với phương thức login bằng user login của SQL Sever

string ChuoiKetNoi = "Server=.\sqlexpress; Database=QLSinhVien;

User ID=giaovu1; Password=Abc12345";

//Khai báo và khởi tạo đối tượng kết nối

SqlConnection KetNoi = new SqlConnection(ChuoiKetNoi);

Đối tượng Command

- Khai báo và khởi tạo đối tượng SqlCommand:

//Khai báo thư viện chứa các đối tượng ADO.net

using System.Data.SqlClient;

//Khai báo và khởi tạo đối tượng Command

SqlCommand BoLenh = new SqlCommand();

//Gán tên thủ tục cho thuộc tính CommandText của Command

BoLenh.CommandText = "Tên_Stored_Procedure";

//Gán loại Command là thực hiện thủ tục

BoLenh.CommandType = CommandType.StoredProcedure;

//Gán đối tượng kết nối cho thuộc tính Connection của Command

BoLenh.Connection = KetNoi;

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa** như sau:

Danh Mục Khoa

Mã Khoa: TH

Tên Khoa: Tin học

Thêm Khoa Xóa Khoa

Ghi Không Ghi

	Mã Khoa	Tên Khoa
	AV	Anh Văn
	LS	Lịch sử
	SH	Sinh học
▶	TH	Tin học
	TR	Triết
	VL	Vật lý
*		

	Mã Sinh Viên	Họ Sinh Viên	Tên Sinh Viên
▶	A02	Trần Văn	Chính
	B01	Trần Thanh	Mai
	B02	Trần Thị Thu	Thủy
	B03	Trần Thị	Thanh
*			

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa** với các yêu cầu xử lý như sau:
 - Hiển thị danh sách các khoa trên lưới Khoa.
 - Khi chọn một khoa trên lưới Khoa thì:
 - Hiển thị thông tin khoa này ra các TextBox.
 - Đồng thời hiển thị danh sách các sinh viên của khoa này trên lưới Sinh viên.

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa** với các yêu cầu xử lý như sau:
 - Khi nhấn nút **Thêm Khoa** thì xóa dữ liệu trong các TextBox.
 - Khi nhấn nút **Xóa Khoa**:
 - Kiểm tra nếu khoa đang chọn đã có sinh viên thì không cho xóa khoa này.
 - Ngược lại, thực hiện xóa khoa đang chọn.
 - Khi nhấn nút **Ghi**:
 - Nếu trước đó đã nhấn nút **Thêm Khoa** thì thực hiện thêm Khoa mới, trong đó phải kiểm tra các ràng buộc: mã khoa duy nhất và tên khoa duy nhất.
 - Ngược lại, thì thực hiện cập nhật thông tin khoa đang chọn, trong đó phải kiểm tra ràng buộc tên khoa duy nhất.
 - Khi nhấn nút **Không Ghi** thì bỏ qua thao tác cập nhật.

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

1. Xây dựng các thủ tục trong SQL Server:

```
--Xây dựng thủ tục hiển thị danh sách các Khoa
create proc sp_HienThiKhoa
as
select * from KHOA
go
```

```
--Xây dựng thủ tục hiển thị danh sách SV theo Khoa
create proc sp_HienThiSinhVienTheoKhoa(@makh char(2))
as
select * from SINHVIEN where MAKH=@makh
go
```


Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

1. Xây dựng các thủ tục trong SQL Server:

--Xây dựng thủ tục thêm Khoa mới

```
create proc sp_ThemKhoa(@makh char(2),@tenkh nvarchar(30),@kq nvarchar(100) output)
as
```

```
set @kq= ''
```

--Kiểm tra Mã Khoa duy nhất

```
if exists(select MAKH from KHOA where MAKH=@makh)
    set @kq=N'Mã khoa ' + @makh + N' đã có trong bảng KHOA'
```

--Kiểm tra Tên Khoa duy nhất

```
if exists(select MAKH from KHOA where TENKH=@tenkh)
    set @kq=N'Tên khoa ' + @tenkh + N' đã có trong bảng KHOA'
```

--Nếu không có lỗi

```
if @kq= ''
    insert into KHOA values(@makh,@tenkh)
go
```

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

1. Xây dựng các thủ tục trong SQL Server:

```
--Xây dựng thủ tục sửa thông tin Khoa
create proc sp_SuaKhoa(@makh char(2), @tenkh nvarchar(30),@kq nvarchar(100) output)
as

set @kq=''

--Kiểm tra Mã Khoa hợp lệ
if not exists(select MAKH from KHOA where MAKH=@makh)
    set @kq=N'Mã khoa ' + @tenkh + N' không tồn tại'

--Kiểm tra Tên Khoa duy nhất
if exists(select MAKH from KHOA where TENKH=@tenkh and MAKH<>@makh)
    set @kq=N'Tên khoa ' + @tenkh + N' đã có trong bảng KHOA'

--Nếu không có lỗi
if @kq=''
    update KHOA set TENKH=@tenkh where MAKH=@makh

go
```

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

1. Xây dựng các thủ tục trong SQL Server:

```
--Xây dựng thủ tục xóa Khoa
create proc sp_XoaKhoa(@makh char(2),@kq nvarchar(100) output)
as

set @kq=''

--Kiểm tra nếu Khoa đã có SV thì không cho xóa
if exists(select MAKH from SINHVIEN where MAKH=@makh)
    set @kq=N'Khoa ' + @makh + N' đã có sinh viên nên không thể xóa'

--Nếu không có lỗi
if @kq=''
    delete KHOA where MAKH=@makh

go
```

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

2. Khai báo thư viện và các biến trong Visual Studio:

```
//Khai báo thư viện chứa các đối tượng ADO.net  
using System.Data.SqlClient;
```

```
//Khai báo các biến  
SqlConnection KetNoi;  
SqlCommand BoLenh;  
SqlDataAdapter BoDocGhiKhoa, BoDocGhiSV;  
DataTable BangKhoa;  
BindingSource BSKhoa;  
DataTable BangSinhVien;  
bool ThemMoi;
```

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

3. Xây dựng các thủ tục xử lý trong Visual Studio dùng C#:

```
//Xây dựng thủ tục tạo kết nối với CSDL QLSinhVien trong SQL Server
void Tao_Ket_Noi()
{
    string ChuoiKetNoi = "Server=.\sqlexpress; Database=QLSinhVien; Trusted_Connection=true";
    KetNoi = new SqlConnection(ChuoiKetNoi);
}
```

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

3. Xây dựng các thủ tục xử lý trong Visual Studio dùng C#:

```
//Xây dựng thủ tục hiển thị Khoa
void Hien_Thi_Khoa()
{
    if (KetNoi == null)
        Tao_Ket_Noi();
    BoLenh = new SqlCommand();
    BoLenh.CommandText = "sp_HienThiKhoa";
    BoLenh.CommandType = CommandType.StoredProcedure;
    BoLenh.Connection = KetNoi;
    BoDocGhiKhoa = new SqlDataAdapter(BoLenh);
    BangKhoa = new DataTable("KHOA");
    BoDocGhiKhoa.Fill(BangKhoa);
    //Tạo đối tượng quản lý dòng hiện hành
    BSKhoa = new BindingSource();
    BSKhoa.DataSource = BangKhoa;
    //Tạo sự kiện thay đổi Khoa hiện hành trong lưới Khoa
    BSKhoa.PositionChanged += BS_Khoa_PositionChanged;
}
```

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

3. Xây dựng các thủ tục xử lý trong Visual Studio dùng C#:

```
//Xây dựng sự kiện thay đổi Khoa hiện hành trong lưới Khoa
private void BS_Khoa_PositionChanged(object sender, EventArgs e)
{
    if (BSKhoa.Position >= 0)
    {
        //Lấy thông tin khoa hiện hành
        DataRow khoa = ((DataRowView)BSKhoa.Current).Row;
        //Xuất thông tin Khoa hiện hành ra các TextBox
        txtMaKhoa.Text = khoa["MAKH"].ToString();
        txtTenKhoa.Text = khoa["TENKH"].ToString();
        //Gọi thủ tục hiển thị danh sách SV của Khoa hiện hành
        string makhoa = khoa["MAKH"].ToString();
        Hien_Thi_Sinh_Vien_Theo_Khoa(makhoa);
        //Hiển thị danh sách SV của Khoa hiện hành ra lưới Sinh viên
        luoisinhvien.AutoGenerateColumns = false;
        luoisinhvien.DataSource = BangSinhVien;
        ThemMoi = false;
    }
}
```

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

3. Xây dựng các thủ tục xử lý trong Visual Studio dùng C#:

```
//Xây dựng thủ tục hiển thị danh sách SV theo Khoa
void Hien_Thi_Sinh_Vien_Theo_Khoa(string Makhoa)
{
    if (KetNoi==null)
        Tao_Ket_No();
    BoLenh = new SqlCommand();
    BoLenh.CommandText = "sp_HienThiSinhVienTheoKhoa";
    BoLenh.CommandType = CommandType.StoredProcedure;
    BoLenh.Connection = KetNoi;
    //Gán giá trị cho tham số đầu vào @makh của thủ tục sp_HienThiSinhVienTheoKhoa
    BoLenh.Parameters.AddWithValue("@makh", Makhoa);
    BoDocGhiSV = new SqlDataAdapter(BoLenh);
    BangSinhVien = new DataTable("SINHVIEN");
    BoDocGhiSV.Fill(BangSinhVien);
}
```


Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

3. Xây dựng các thủ tục xử lý trong Visual Studio dùng C#:

```
//Xây dựng hàm thêm 1 Khoa mới
string Them_Khoa(string Makhoa,string Tenkhoa)
{
    BoLenh = new SqlCommand();
    BoLenh.CommandText = "sp_ThemKhoa";
    BoLenh.CommandType = CommandType.StoredProcedure;
    BoLenh.Connection = KetNoi;
    //Gán giá trị cho tham số đầu vào @makh và @tenkh của thủ tục sp_ThemKhoa
    BoLenh.Parameters.AddWithValue("@makh", Makhoa);
    BoLenh.Parameters.AddWithValue("@tenkh", Tenkhoa);
    //Gán tham số đầu ra
    BoLenh.Parameters.Add("@kq", SqlDbType.NVarChar,100).Direction = ParameterDirection.Output;
    KetNoi.Open();
    BoLenh.ExecuteNonQuery();
    KetNoi.Close();
    string kq = BoLenh.Parameters["@kq"].Value.ToString();
    return kq;
}
```

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

3. Xây dựng các thủ tục xử lý trong Visual Studio dùng C#:

```
//Xây dựng hàm sửa thông tin 1 Khoa
string Sua_Khoa(string Makhoa, string Tenkhoa)
{
    BoLenh = new SqlCommand();
    BoLenh.CommandText = "sp_SuaKhoa";
    BoLenh.CommandType = CommandType.StoredProcedure;
    BoLenh.Connection = KetNoi;
    //Gán giá trị cho tham số đầu vào @makh và @tenkh của thủ tục Sp_SuaKhoa
    BoLenh.Parameters.AddWithValue("@makh", Makhoa);
    BoLenh.Parameters.AddWithValue("@tenkh", Tenkhoa);
    //Gán tham số ra
    BoLenh.Parameters.Add("@kq", SqlDbType.NVarChar,100).Direction = ParameterDirection.Output;
    KetNoi.Open();
    BoLenh.ExecuteNonQuery();
    KetNoi.Close();
    string kq = BoLenh.Parameters["@kq"].Value.ToString();
    return kq;
}
```

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

3. Xây dựng các thủ tục xử lý trong Visual Studio dùng C#:

```
//Xây dựng hàm xóa 1 Khoa
string Xoa_Khoa(string Makhoa)
{
    BoLenh = new SqlCommand();
    BoLenh.CommandText = "Sp_XoaKhoa";
    BoLenh.CommandType = CommandType.StoredProcedure;
    BoLenh.Connection = KetNoi;
    //Gán giá trị cho tham số đầu vào @makh của thủ tục sp_XoaKhoa
    BoLenh.Parameters.AddWithValue("@makh", Makhoa);
    //Gán tham số ra
    BoLenh.Parameters.Add("@kq", SqlDbType.NVarChar,100).Direction = ParameterDirection.Output;
    KetNoi.Open();
    BoLenh.ExecuteNonQuery();
    KetNoi.Close();
    string kq = BoLenh.Parameters["@kq"].Value.ToString();
    return kq;
}
```

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

3. Xây dựng các thủ tục xử lý trong Visual Studio dùng C#:

```
//Xây dựng sự kiện FormLoad
private void frmDanhMucKhoa_Load(object sender, EventArgs e)
{
    Tao_Ket_Noi();
    Hien_Thi_Khoa();
    //Gán dữ liệu nguồn cho lưới Khoa
    LwoiKhoa.DataSource = BSKhoa;
    //Gọi sự kiện thay đổi Khoa hiện hành để hiển thị d/s SV của Khoa đầu tiên
    BS_Khoa_PositionChanged(sender, e);
}
```

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

3. Xây dựng các thủ tục xử lý trong Visual Studio dùng C#:

```
//Xử lý sự kiện khi nhấn nút Thêm khoa
private void btnThemKhoa_Click(object sender, EventArgs e)
{
    txtMaKhoa.Text = "";
    txtTenKhoa.Text = "";
    ThemMoi = true;
}
```

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

3. Xây dựng các thủ tục xử lý trong Visual Studio dùng C#:

```
//Xử lý sự kiện khi nhấn nút Xóa khoa
private void btnXoaKhoa_Click(object sender, EventArgs e)
{
    if (MessageBox.Show("Bạn có muốn xóa Khoa hiện hành không?", "Thông báo",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)
    {
        string kq = Xoa_Khoa(txtMaKhoa.Text);
        if (kq == "")
        {
            MessageBox.Show("Đã xóa Khoa thành công");
            frmDanhMucKhoa_Load(sender, e);
        }
        else
            MessageBox.Show(kq);
    }
}
```

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

3. Xây dựng các thủ tục xử lý trong Visual Studio dùng C#:

```
//Xử lý sự kiện khi nhấn nút Ghi
private void btnGhi_Click(object sender, EventArgs e)
{
    string kq = "";
    if (ThemMoi == true)
        kq = Them_Khoa(txtMaKhoa.Text,txtTenKhoa.Text);
    else
        kq = Sua_Khoa(txtMaKhoa.Text, txtTenKhoa.Text);
    if (kq == "")
    {
        if (ThemMoi == true)
            MessageBox.Show("Đã thêm Khoa mới thành công ");
        else
            MessageBox.Show("Đã cập nhật Khoa thành công");
        //Load lại Khoa
        frmDanhMucKhoa_Load(sender, e);
    }
    else
        MessageBox.Show(kq);
}
```

Ví dụ minh họa

- Xây dựng màn hình (form) **Danh mục Khoa:**

3. Xây dựng các thủ tục xử lý trong Visual Studio dùng C#:

```
//Xử lý sự kiện khi nhấn nút Không Ghi  
private void btnKhongGhi_Click(object sender, EventArgs e)  
{  
    BS_Khoa_PositionChanged(sender, e);  
    ThemMoi = false;  
}
```

Bài tập ứng dụng

1. Sử dụng CSDL Quản lý sinh viên:

- Xây dựng màn hình Danh mục Khoa và danh sách SV tương ứng.
- Xây dựng màn hình Danh mục Sinh viên và kết quả học tập tương ứng.

Bài tập ứng dụng

2. Sử dụng CSDL Quản lý bán hoa:

- Xây dựng màn hình Danh mục Loại hoa và danh sách hoa tương ứng.
- Xây dựng màn hình Danh mục Chủ đề hoa và danh sách hoa tương ứng.
- Xây dựng màn hình Danh mục Hóa đơn và chi tiết hóa đơn tương ứng.

Câu hỏi trắc nghiệm

1. Chọn phát biểu SAI về ADO.Net?

- a. ADO.Net là mô hình cho phép người lập trình thao tác với nhiều loại CSDL theo cùng một cách.
- b. ADO.Net là mô hình chỉ cho phép người lập trình thao tác với CSDL SQL Server.
- c. ADO.Net tích hợp sẵn các thư viện chứa các đối tượng để thao tác với CSDL.
- d. Các đối tượng thao tác với CSDL trong ADO.Net gồm: Connection, Command, DataReader, Data Table, DataAdapter, DataSet, ...

Câu hỏi trắc nghiệm

2. Đối tượng Command trong ADO.Net dùng để:

- a. Kết nối với CSDL.
- b. Chứa dữ liệu dạng bảng trên ứng dụng.
- c. Lấy dữ liệu trả về đưa vào DataTable và lấy dữ liệu từ DataTable lưu xuống CSDL.
- d. Thực hiện các câu lệnh truy vấn, gọi thực hiện các Stored Procedure.

Câu hỏi trắc nghiệm

3. Chọn trình tự thực hiện quy trình kết nối ứng dụng đến CSDL:

1. Mở đối tượng Connection (Open).
 2. Đóng đối tượng Connection (Close).
 3. Tạo đối tượng Command và xác định câu lệnh SQL.
 4. Thực hiện câu lệnh SQL và xử lý kết quả (Execute).
 5. Tạo đối tượng Connection và xác định chuỗi kết nối (Connection String).
- a. 1, 2, 3, 4, 5
 - b. 5, 4, 3, 2, 1
 - c. 5, 3, 1, 4, 2
 - d. 1, 5, 3, 4, 2

Câu hỏi trắc nghiệm

4. Có thể gọi thực hiện các thủ tục (Stored Procedure) trong SQL Server từ ứng dụng dùng ngôn ngữ C#?

- a. Đúng
- b. Sai

Câu hỏi trắc nghiệm

5. Đây là thủ tục gì trong SQL Server?

```
create proc sp_??? (@makh char(2), @tenkh nvarchar(30),@kq nvarchar(100) output)
as

set @kq=''

--Kiểm tra Mã Khoa hợp lệ
if not exists(select MAKH from KHOA where MAKH=@makh)
    set @kq=N'Mã khoa ' + @tenkh + N' không tồn tại'

--Kiểm tra Tên Khoa duy nhất
if exists(select MAKH from KHOA where TENKH=@tenkh and MAKH<>@makh)
    set @kq=N'Tên khoa ' + @tenkh + N' đã có trong bảng KHOA'

--Nếu không có lỗi
if @kq=''
    update KHOA set TENKH=@tenkh where MAKH=@makh

go
```

- a. Xóa khoa
- b. Thêm Khoa mới
- c. Sửa thông tin Khoa
- d. Xem thông tin Khoa