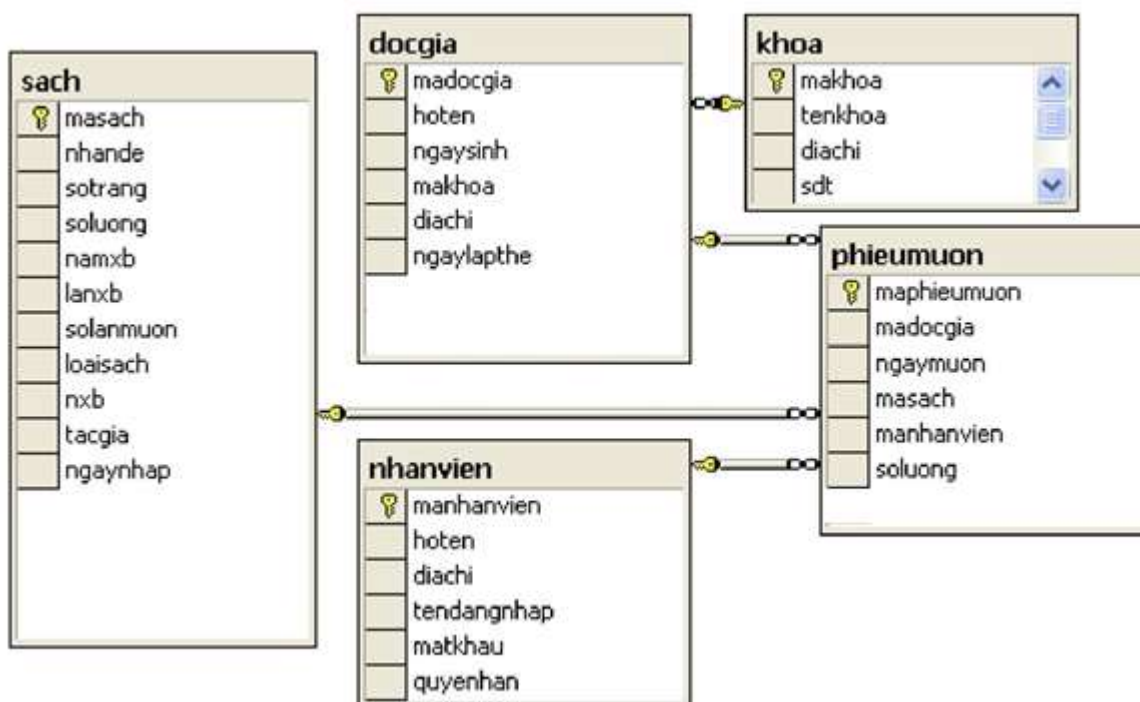


Bài 10: Làm việc với mô hình kết nối

Sinh viên tạo cơ sở dữ liệu **quanlythuvien** trong SQL Server có quan hệ như sau:



Thiết kế Form để tạo mới 1 tài khoản như sau (làm việc trên **bảng nhanvien**):

The form is titled "Tạo mới một tài khoản" and contains a table with the following columns: **Manv**, **Họ tên**, **Địa chỉ**, **Tên đăng nhập**, and **Quyền hạn**. Below the table is a section for "Thông tin cơ bản" with input fields for:

- Mã nhân viên
- Họ tên
- Địa chỉ
- Tên đăng nhập
- Quyền hạn (with a dropdown arrow)

On the right side of the form, there are five buttons: **Tạo mới**, **Tìm kiếm**, **Xoá bỏ**, **Thoát**, and a set of navigation buttons: **◀**, **<**, **>**, and **>|**.

Frmtaomoitk sử dụng các trường, phương thức và sự kiện sau:

```

namespace baimau
{
    public partial class Frmtaomoitk : Form
    {
        SqlConnection cn;
        SqlCommand cmdSelect;
        SqlCommand cmdInsert;
        SqlCommand cmdXoa;
        int i = 0;
        public Frmtaomoitk()...
        private void Mokeanoi()...
        private void LoadListView()...
        private void LoadItem(int i)...)...
        private void LoadCombox()...
        private void XoaTextBox()...
        private int KiemTraMa(string ma)...)...
        private void Frmtaomoitk_Load(object sender, EventArgs e)...)...
        private void butdau_Click(object sender, EventArgs e)...)...
        private void buttien_Click(object sender, EventArgs e)...)...
        private void butlui_Click(object sender, EventArgs e)...)...
        private void butTaomoi_Click(object sender, EventArgs e)...)...
        private void butXoabo_Click(object sender, EventArgs e)...)...
        private void buttimkiem_Click(object sender, EventArgs e)...)...
    }
}

```

Tên điều khiển	Thuộc tính
Form	Name: Frmtaomoitk Text: Tạo mới một tài khoản sử dụng chương trình
ListView	Name: listView1 Columns: Add thêm 4 cột: Họ tên, Địa chỉ, Tên đăng nhập và Quyền hạn View: Details GridLines: True
GroupBox	Name: groupBox1 Text: Thông tin cơ bản
Label	Tạo ra 5 label để hiển thị: Mã nhân viên, Họ tên, Địa chỉ, Tên đăng nhập và quyền hạn.
TextBox	Tạo ra 4 TextBox lần lượt với các tên: txtmanv, txthoten, txtdiachi, txttendangnhap
Button	Tạo 8 button lần lượt với các tên butdau, butlui, buttien, butcuoi, buttaomoi, buttimkiem, butxoabo, butthoat

	Ý nghĩa
cn	Dùng để kết nối đến cơ sở dữ liệu quanlythuvien
cmdSelect	sqlCommand sử dụng câu lệnh select để hiển thị và tìm kiếm
cmdInsert	sqlCommand sử dụng câu lệnh Insert để tạo thêm 1 tài khoản
cmdXoa	sqlCommand sử dụng câu lệnh Delete để xóa 1 tài khoản
i	Tài khoản thứ i

Các phương thức

+ Hàm dựng **Frmtaomoitk** để tạo giao diện

```
public Frmtaomoitk()
{
    InitializeComponent();
}
```

+ Phương thức **Moketnoi()**: Kiểm tra đường kết nối, nếu đang mở thì đóng lại, sau đó mở lại đường kết nối

```
private void Moketnoi()
{
    if (cn.State == ConnectionState.Open)
        cn.Close();
    cn.Open();
}
```

+ Phương thức **LoadListView**: Lấy dữ liệu của bảng nhanvien | nạp dữ liệu lên listView1. Phương thức này được gọi khi thay đổi dữ liệu trong bảng nhân viên như nhập thêm hoặc xóa đi 1 nhân viên. Sử dụng 2 đối tượng SqlCommand, SqlDataReader

```
private void LoadListView()
{
    Moketnoi();
    cmdSelect = new SqlCommand("select * from nhanvien", cn);
    SqlDataReader r = cmdSelect.ExecuteReader();
    listView1.Items.Clear(); // Xóa tất cả dữ liệu trong listView1
    while (r.Read())
    {
        string[] st = new string[5];
        st[0] = r[0].ToString();
        st[1] = r[1].ToString();
        st[2] = r[2].ToString(); // Không hiển thị mật khẩu, nên không có r[3]
        st[3] = r[4].ToString();
        st[4] = r[5].ToString();
        ListViewItem lv = new ListViewItem(st);
        listView1.Items.Add(lv);
    }
    cmdSelect.Dispose();
}
```

+ Phương thức **LoadItem**: Lấy dữ liệu từ dòng thứ i của listView1 đưa vào txtmanv, txthoten, txtdiachi, txttendangnhap và comboBox1. Phương thức này được gọi khi di chuyển qua lại thông tin của các nhân viên.

```
private void LoadItem(int i)
{
    txtmanv.Text = listView1.Items[i].Text;
    txthoten.Text = listView1.Items[i].SubItems[1].Text;
    txtdiachi.Text = listView1.Items[i].SubItems[2].Text;
    txttendangnhap.Text = listView1.Items[i].SubItems[3].Text;
    comboBox1.Text = listView1.Items[i].SubItems[4].Text;
}
```

+ Phương thức **LoadCombox**: Đưa dữ liệu vào cho comboBox1. Giả sử chỉ có 3 quyền hạn: admin, sinhvien và Thuthu. Phương thức này được gọi khi vừa nạp Form lên

```
private void LoadCombox()
{
    comboBox1.Items.Add("Admin");
    comboBox1.Items.Add("Sinhvien");
    comboBox1.Items.Add("ThuThu");
    comboBox1.Text = "Admin";
}
```

+ Phương thức **XoaTextBox**: Xóa hết dữ liệu trong các textBox, phương thức này được gọi khi nhập thêm 1 tài khoản.

```
private void XoaTextBox()
{
    txtmanv.Clear();
    txthoten.Clear();
    txtdiachi.Clear();
    txttendangnhap.Clear();
    txtmanv.Focus();
}
```

+ Phương thức **KiemTraMa**: Kiểm tra xem có mã nhân viên nào bằng với ma hay không. Phương thức này được gọi khi nhập thêm 1 tài khoản

```
private int KiemTraMa(string ma)
{
    Mocketnoi();
    cmdSelect = new SqlCommand("select count(*) from nhanvien
                                where manhanvien='"+ma.Trim()+"'");
    cmdSelect.Connection = cn;
    return (int)cmdSelect.ExecuteScalar();
}
```

+ Sự kiện **Frmtaomoitk_Load**: Tạo và mở ra đường kết nối đến cơ sở dữ liệu quanlythuvien, tên máy chủ, sử dụng cơ chế xác thực kiểu Windows, tạo dữ liệu vào cho các điều khiển.

```
private void Frmtaomoitk_Load(object sender, EventArgs e)
{
    try
    {
        cn = new SqlConnection("Data Source=.....;Initial Catalog=quanlythuvien;
                                Trusted_Connection=yes");
        cn.Open();
    }
    catch (Exception loi) { MessageBox.Show("Không thể kết nối được"); }
    LoadListView(); //Nạp dữ liệu vào cho listView1
    i = 0;
    LoadItem(i); // Nạp dữ liệu vào cho các textBox và comboBox1
    LoadCombox();
}
```

+ Sự kiện **butdau_Click**: Nạp dữ liệu của dòng đầu tiên từ listView1 vào cho các textBox và comboBox

```
private void butdau_Click(object sender, EventArgs e)
{
    i = 0;
    LoadItem(i);
}
```

+ Sự kiện **buttien_Click**: Nạp dữ liệu của dòng tiếp theo từ listView1 vào cho các textBox và comboBox

```
private void buttien_Click(object sender, EventArgs e)
{
    i++;
    if (i == listView1.Items.Count)
        i = listView1.Items.Count - 1; LoadItem(i);
}
```

+ Sự kiện **butlui_Click**:

```
private void butlui_Click(object sender, EventArgs e)
{
    i--;
    if (i < 0)
        i = 0;
    LoadItem(i);
}
```

+ Sự kiện **butcuoi_Click**:

```
private void butcuoi_Click(object sender, EventArgs e)
{
    i = listView1.Items.Count - 1;
    LoadItem(i);
}
```

+ Sự kiện **butTaomoi_Click**: Được sử dụng để thêm 1 tài khoản (1 nhân viên), nút butTaomoi có 2 trạng thái tạo mới và lưu. Nếu người sử dụng kích vào nút Tạo mới sẽ chuyển sang trạng thái là lưu và ngược lại.

```
private void butTaomoi_Click(object sender, EventArgs e)
{
    if (butTaomoi.Text.Equals("Tạo mới"))
    {
        XoaTextBox();
        butTaomoi.Text = "Luu";
    }
    else
    {
        // Kiểm tra xem mã nhân viên này có hay chưa ?
        if (KiemTraMa(txtmanv.Text)==1) {
            MessageBox.Show("Mã này đã có");
            txtmanv.Clear();
            txtmanv.Focus();
        }
        else
        {
            string ma = txtmanv.Text;
            string hoten = txthoten.Text;
            string diachi = txtdiachi.Text;
            string tendangnhap = txttendangnhap.Text;
            string matkhau = ""; // Khi tạo 1 tài khoản thì mật khẩu ban đầu là rỗng
            string quyenhan = comboBox1.Text;
            Mocketnoi();
            string sql="insert into nhanvien values('"+ma+"','"+hoten+"','"+diachi+"','"+tendangnhap+"','"+matkhau+"','"+quyenhan+"')";
            cmdInsert = new SqlCommand(sql,cn);
            cmdInsert.ExecuteNonQuery();
            MessageBox.Show("Đã lưu thành công");
            LoadListView(); //Nạp lại dữ liệu mới vào listView1
            butTaomoi.Text = "Tạo mới";
            cmdInsert.Dispose();
        }
    }
}
```


+ Sự kiện **buttimkiem_Click**: Khi người sử dụng gõ 1 mã nhân viên vào txtmanv và kích vào nút buttimkiem, nếu tìm thấy mã nhân viên này sẽ hiển thị kết quả lên các textBox và comboBox

```
private void buttimkiem_Click(object sender, EventArgs e)
{
    Mocketnoi();
    string sql = "select * from nhanvien where manhanvien='" + txtmanv.Text + "'";
    cmdSelect = new SqlCommand(sql, cn);
    SqlDataReader dr = cmdSelect.ExecuteReader();
    if (dr.Read()) // Đã tìm thấy
    {
        txtmanv.Text = dr[0].ToString();
        txthoten.Text = dr[1].ToString();
        txtdiachi.Text = dr[2].ToString();
        txttendangnhap.Text = dr[4].ToString();
        comboBox1.Text = dr[5].ToString(); }
    else
        MessageBox.Show("Không tìm thấy");
}
```

+ Sự kiện **butXoabo_Click**: Xóa nhân viên có mã nhân viên ở txtmanv

```
private void butXoabo_Click(object sender, EventArgs e)
{
    DialogResult dr = MessageBox.Show("Chắc chắn xóa hay không ?",
                                        "Thông báo", MessageBoxButtons.YesNo);
    if (dr == DialogResult.Yes) // Nếu người sử dụng chọn nút yes
    {
        Mocketnoi();
        string sql = "delete from nhanvien where manhanvien='" + txtmanv.Text + "'";
        cmdXoa = new SqlCommand(sql, cn);
        if (cmdXoa.ExecuteNonQuery() == 1)
        {
            MessageBox.Show("Xóa thành công");
            LoadListView();
            LoadItem(0);
        }
        else
            MessageBox.Show("Không tồn tại mã nhân viên " + txtmanv.Text);
        cmdXoa.Dispose();
    }
}
```