

Hướng dẫn tuần 06

+ SQL Server

- Viết store thêm sinh viên

```
create procedure sp_ThemSV @masv varchar(15) , @hoten nvarchar(30), @ngaysinh datetime, @phai  
nvarchar, @lop varchar(15), @dtb float  
AS  
BEGIN  
  
    insert into SINHVIEN (MaSV, HoTen, NgaySinh, Phai , Lop, DTB) Values (@masv, @hoten,  
@ngaysinh, @phai , @lop, @dtb)  
END
```

- Viết store cập nhật sinh viên

```
create procedure sp_capNhatSV @masv varchar(15) , @hoten nvarchar(30), @ngaysinh datetime, @phai  
nvarchar, @lop varchar(15), @dtb float  
AS  
BEGIN  
  
    update SINHVIEN set HoTen =@hoten, NgaySinh = @ngaysinh, Phai = @phai, Lop = @lop, DTB  
= @dtb where masv = @masv  
END
```

- Viết store xóa sinh viên

```
create procedure sp_xoaSV @masv varchar(15)  
AS  
BEGIN  
  
    delete from SINHVIEN where masv = @masv  
END
```

+ Solution QuanLySinhVien

Add → New Project → Windows Form Application → GUI

Sửa lại tên form là frmSINHVIEN

+ Thiết kế form sinh viên

Sử dụng bộ công cụ “Common Controls” trong tools box , các control sử dụng bao gồm:
Lable, TextBox, DateTimePicker, ComboBox và Button

+ Lớp DAL

Viết các phương thức:

- đọc danh sách lớp

```
public DataTable loadDSLop()
{
    DataTable dt = new DataTable();
    SqlCommand cmd = new SqlCommand();
    SqlDataAdapter da = new SqlDataAdapter();

    DTO_SINHVIEN sv = new DTO_SINHVIEN();

    cmd.Connection = _Conn;
    cmd.CommandType = CommandType.Text;
    cmd.CommandText = "select MaLop from LOP order by malop";

    openConnection();

    da.SelectCommand = cmd;
    da.Fill(dt);
    closeConnection();

    return dt;
}
```

- Thêm sinh viên vào Database

```
public int insertIntoSV(DTO_SINHVIEN SV)
{
    SqlCommand cmd = new SqlCommand();

    cmd.Connection = _Conn;
    cmd.CommandType = CommandType.StoredProcedure;
    cmd.CommandText = "sp_ThemSV";

    cmd.Parameters.Add("@masv", SqlDbType.Text).Value = SV.docghiMASV;
    cmd.Parameters.Add("@hoten", SqlDbType.NVarChar).Value = SV.docghiHoTen;
    cmd.Parameters.Add("@ngaysinh", SqlDbType.DateTime).Value = SV.docghiNgaySinh;
    cmd.Parameters.Add("@phai", SqlDbType.NVarChar).Value = SV.docghiPhai;
    cmd.Parameters.Add("@lop", SqlDbType.Text).Value = SV.docghiLop;
    cmd.Parameters.Add("@dtb", SqlDbType.Float).Value = SV.docghiDTB;

    try
    {
        openConnection();
        cmd.ExecuteNonQuery();
    }
```

```

        closeConnection();
        return 1;
    } catch
    {
        return 0;
    }
}

```

- Cập nhật thông tin sinh viên

```

public int updateSV(string masv, string hoten, DateTime ngaysinh, string phai, string lop, float dtb)
{
    SqlCommand cmd = new SqlCommand();

    cmd.Connection = _Conn;
    cmd.CommandType = CommandType.StoredProcedure;
    cmd.CommandText = "sp_capNhatSV";

    cmd.Parameters.Add("@masv", SqlDbType.Text).Value = masv;
    cmd.Parameters.Add("@hoten", SqlDbType.NVarChar).Value = hoten;
    cmd.Parameters.Add("@ngaysinh", SqlDbType.DateTime).Value = ngaysinh;
    cmd.Parameters.Add("@phai", SqlDbType.NVarChar).Value = phai;
    cmd.Parameters.Add("@lop", SqlDbType.Text).Value = lop;
    cmd.Parameters.Add("@dtb", SqlDbType.Float).Value = dtb;

    try
    {
        openConnection();

        cmd.ExecuteNonQuery();

        closeConnection();
        return 1;
    }
    catch
    {
        return 0;
    }
}

```

- Xóa một sinh viên

```

public int deleteSV(string masv)
{
    SqlCommand cmd = new SqlCommand();

    cmd.Connection = _Conn;

```

```

cmd.CommandType = CommandType.StoredProcedure;
cmd.CommandText = "sp_xoaSV";

cmd.Parameters.Add("@masv", SqlDbType.Text).Value = masv;

try
{
    openConnection();

    cmd.ExecuteNonQuery();

    closeConnection();
    return 1;
}
catch
{
    return 0;
}
}

```

- Đọc danh sách sinh viên

```

public DataTable loadDSSinhVien()
{
    DataTable dt = new DataTable();
    SqlCommand cmd = new SqlCommand();
    SqlDataAdapter da = new SqlDataAdapter();

    DTO_SINHVIEN sv = new DTO_SINHVIEN();

    cmd.Connection = _Conn;
    cmd.CommandType = CommandType.Text;
    cmd.CommandText = "select * from SINHVIEN order by masv";

    openConnection();

    da.SelectCommand = cmd;
    da.Fill(dt);
    closeConnection();

    return dt;
}

```

+ Lớp BUS

- Phương thức lấy danh sách lớp

```
public DataTable getDSLop()
{
    DataTable dt = new DataTable();

    dt = dal_sv.loadDSLop();

    return dt;
}
```

- Phương Thức thêm sinh viên

```
public int NhapSV(string masv, string hoten, DateTime ngaysinh, string phai, string lop, float dtb )
{
    DTO_SINHVIEN SV = new DTO_SINHVIEN();

    SV.docghiMASV = masv;
    SV.docghiHoTen = hoten;
    SV.docghiNgaySinh = ngaysinh;
    SV.docghiPhai = phai;
    SV.docghiLop = lop;
    SV.docghiDTB = dtb;

    return dal_sv.insertIntoSV(SV);
}
```

- Phương thức xóa sinh viên

```
public int xoaSV(string masv)
{
    return dal_sv.deleteSV(masv);
}
```

- Phương thức cập nhật thông tin sinh viên

```
public int suaSV(string masv, string hoten, DateTime ngaysinh, string phai, string lop, float dtb)
{
    return dal_sv.updateSV(masv, hoten, ngaysinh, phai, lop, dtb);
}
```

- Phương thức lấy danh sách sinh viên

```
public DataTable getDSSinhVien()
{
    DataTable dt = new DataTable();
}
```

```
dt = dal_sv.loadDSSinhVien();
```

```
return dt;
```

```
}
```

+ Form Sinh Viên

Khai báo 3 biến toàn cục

```
BUS_SINHVIEN bsv = new BUS_SINHVIEN();
```

```
DataTable dtSV = new DataTable();
```

```
int index = 0;
```

- sự kiện frmSINHVIEN_Load

```
private void frmSINHVIEN_Load(object sender, EventArgs e)
```

```
{
```

```
    DataTable dt = new DataTable();
```

```
    dt = bsv.getDSLop();
```

```
    cboLop.DataSource = dt;
```

```
    cboLop.DisplayMember = "malop";
```

```
    cboLop.ValueMember = "malop";
```

```
    refreshDataSV();
```

```
    showTTTSV(0);
```

```
}
```

- Sự kiện Click của nút Thêm

```
private void cmdThem_Click(object sender, EventArgs e)
```

```
{
```

```
    int i;
```

```
    i = bsv.NhapSV(txtMaSV.Text, txtHoTen.Text, DateTime.Parse(dtpNgaySinh.Text), cboPhai.Text, cboLop.SelectedValue.ToString(), float.Parse(txtDTB.Text));
```

```
    if (i == 1)
```

```
    {
```

```
        MessageBox.Show("Đã thêm thành công");
```

```
        refreshDataSV();
```

```
    }
```

```
    else MessageBox.Show("không thêm được sinh viên này");
```

```
}
```

- Phương thức làm mới dữ liệu

```
private void refreshDataSV()
{
    dtSV = bsv.getDSSinhVien();
    index = 0;
    showTTTSV(index);
}
```

- Phương thức hiển thị thông tin sinh viên lên các control

```
private void showTTTSV(int index)
{
    if (index >= 0 && index < dtSV.Rows.Count)
    {
        txtMaSV.Text = dtSV.Rows[index]["masv"].ToString();
        txtHoTen.Text = dtSV.Rows[index]["hoten"].ToString();
        dtpNgaySinh.Text = dtSV.Rows[index]["ngaysinh"].ToString();
        cboPhai.Text = dtSV.Rows[index]["phai"].ToString();
        cboLop.Text = dtSV.Rows[index]["lop"].ToString();
        txtDTB.Text = dtSV.Rows[index]["dtb"].ToString();
    }
}
```

- Sự kiện Click của nút Lui

```
private void cmdLui_Click(object sender, EventArgs e)
{
    if(index > 0 )
    {
        index = index - 1;
        showTTTSV(index);
    }
}
```

- Sự kiện Click của nút Tới

```
private void cmdToi_Click(object sender, EventArgs e)
{
    if(index < dtSV.Rows.Count -1)
    {
        index = index + 1;
        showTTTSV(index);
    }
}
```

- Sự kiện Click của nút Xóa

```
private void cmdXoa_Click(object sender, EventArgs e)
{
    int i;

    i = bsv.xoaSV(txtMaSV.Text);

    if (i == 1)
    {
        MessageBox.Show("Đã xóa thành công");
        refreshDataSV();
    }
    else MessageBox.Show("không xóa được sinh viên này");
}
```

- Sự kiện Click của nút Sửa

```
private void cmdSua_Click(object sender, EventArgs e)
{
    int i;

    i = bsv.suaSV(txtMaSV.Text, txtHoTen.Text, DateTime.Parse(dtpNgaySinh.Text), cboPhai.Text,
cboLop.SelectedValue.ToString(), float.Parse(txtDTB.Text));

    if (i == 1)
    {
        MessageBox.Show("Đã sửa thông tin thành công");
        refreshDataSV();
    }
    else MessageBox.Show("không sửa được thông tin của sinh viên này");
}
```