

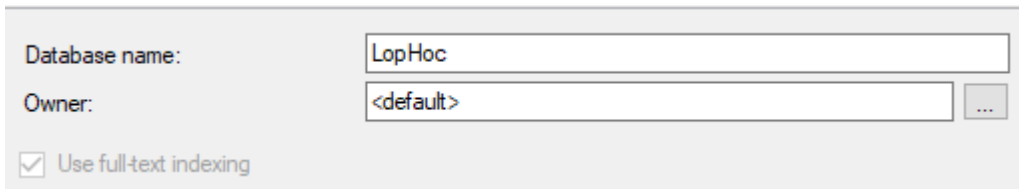
## THỰC HÀNH BUỔI 07

### Mục tiêu

- Vận dụng OOP tạo ứng dụng quản lý mức 2 (quan hệ một – nhiều)
- Sử dụng Entity Framework Database First để sinh model
- Tạo giao diện tương tác
- Vận dụng Linq để truy vấn

### THAM KHẢO LẠI CODE BUỔI 06

#### Bước 1: Tạo CSDL



Database name: LopHoc

Owner: <default> ...

☒ Use full-text indexing

```
CREATE TABLE LOPHOC
(
    MALOP VARCHAR(50) NOT NULL,
    TENLOP NVARCHAR(50) NOT NULL,
    CONSTRAINT PK_LOPHOC PRIMARY KEY(MALOP)
)

CREATE TABLE SINHVIEN
(
    MASV VARCHAR(50) NOT NULL,
    TENSX NVARCHAR(50) NOT NULL,
    DIACHI NVARCHAR(100) NOT NULL,
    TUOI INT NOT NULL,
    MALOP VARCHAR(50) NOT NULL,
    CONSTRAINT PK_SINHVIEN PRIMARY KEY(MASV),
    CONSTRAINT FK_SINHVIEN_LOPHOC FOREIGN KEY(MALOP) REFERENCES
LOPHOC(MALOP)
)
```

```
INSERT INTO LOPHOC VALUES('LH001', 'LAP TRINH WEB')
```

```
INSERT INTO LOPHOC VALUES('LH002', 'LAP TRINH WEB NANG CAO')
```

```
INSERT INTO LOPHOC VALUES('LH003', 'LAP TRINH OOP')
```

```
INSERT INTO SINHVIEN VALUES('SV001', N'NGUYỄN VĂN A', 'TPHCM', 18,  
'LH001')
```

```
INSERT INTO SINHVIEN VALUES('SV002', N'NGUYỄN VĂN B', 'TPHCM', 18,  
'LH001')
```

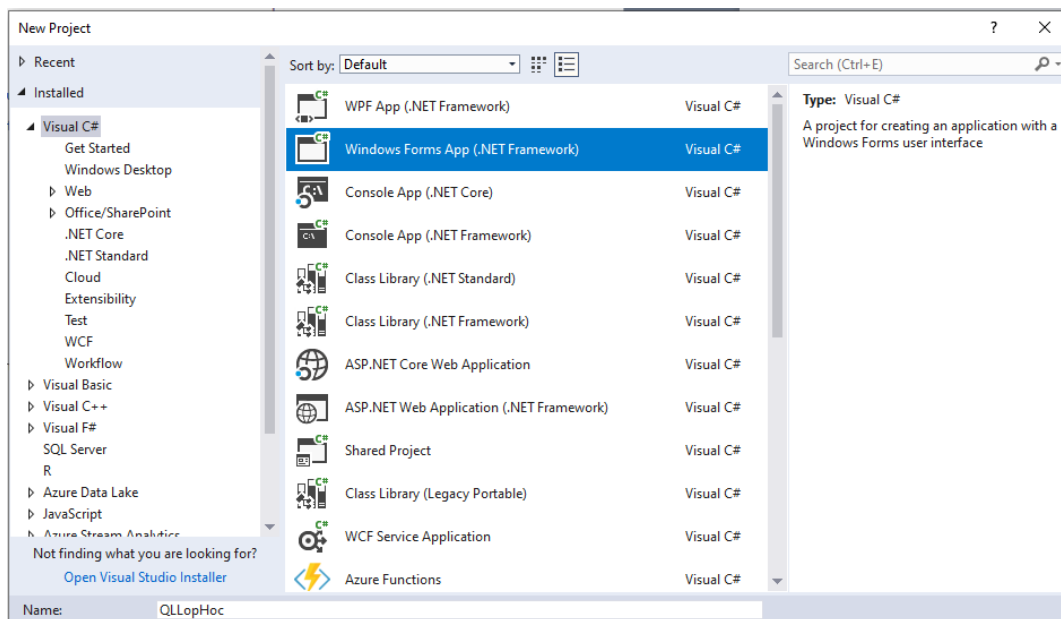
```
INSERT INTO SINHVIEN VALUES('SV003', N'NGUYỄN VĂN C', 'HANOI', 18,  
'LH002')
```

```
INSERT INTO SINHVIEN VALUES('SV004', N'NGUYỄN VĂN D', 'HANOI', 18,  
'LH002')
```

```
INSERT INTO SINHVIEN VALUES('SV005', N'NGUYỄN VĂN E', 'TPHCM', 18,  
'LH003')
```

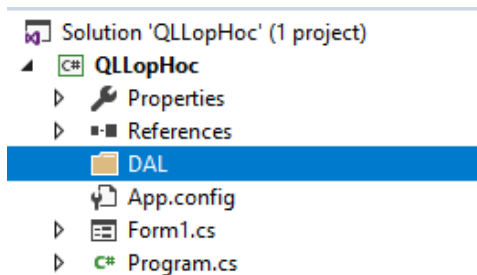
```
INSERT INTO SINHVIEN VALUES('SV006', N'NGUYỄN VĂN F', 'TPHCM', 18,  
'LH003')
```

## Bước 2: Tạo ứng dụng Windows

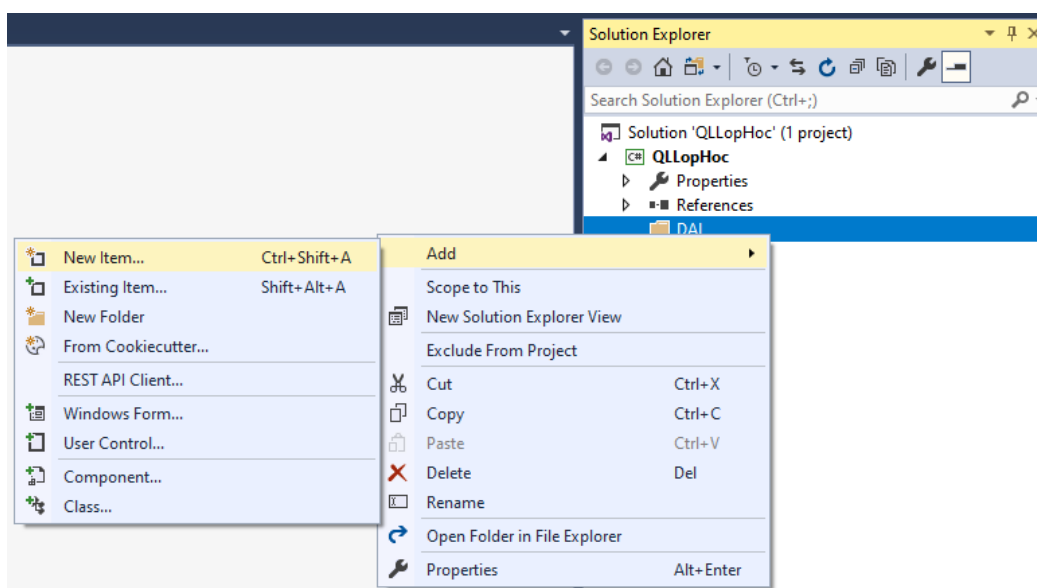


Đặt tên QLLopHoc

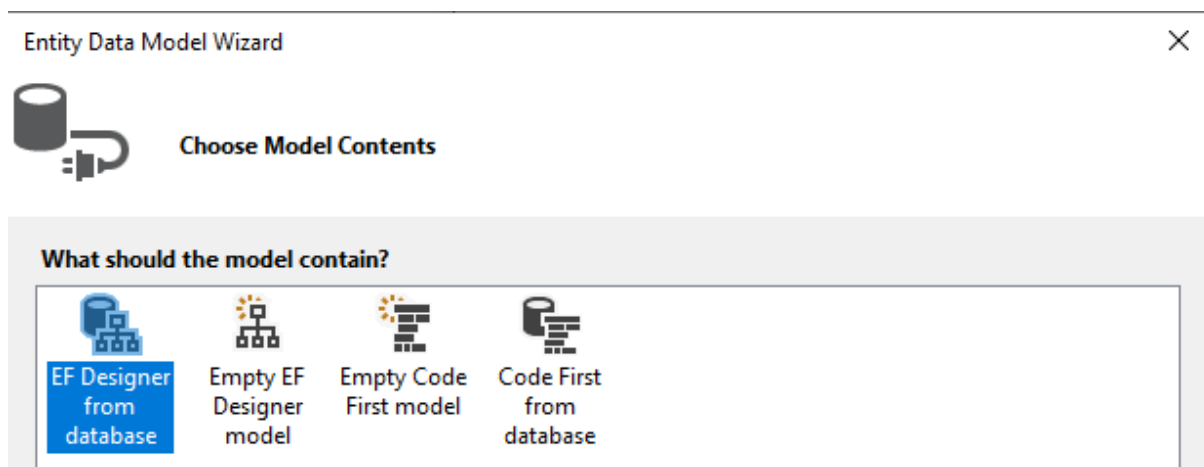
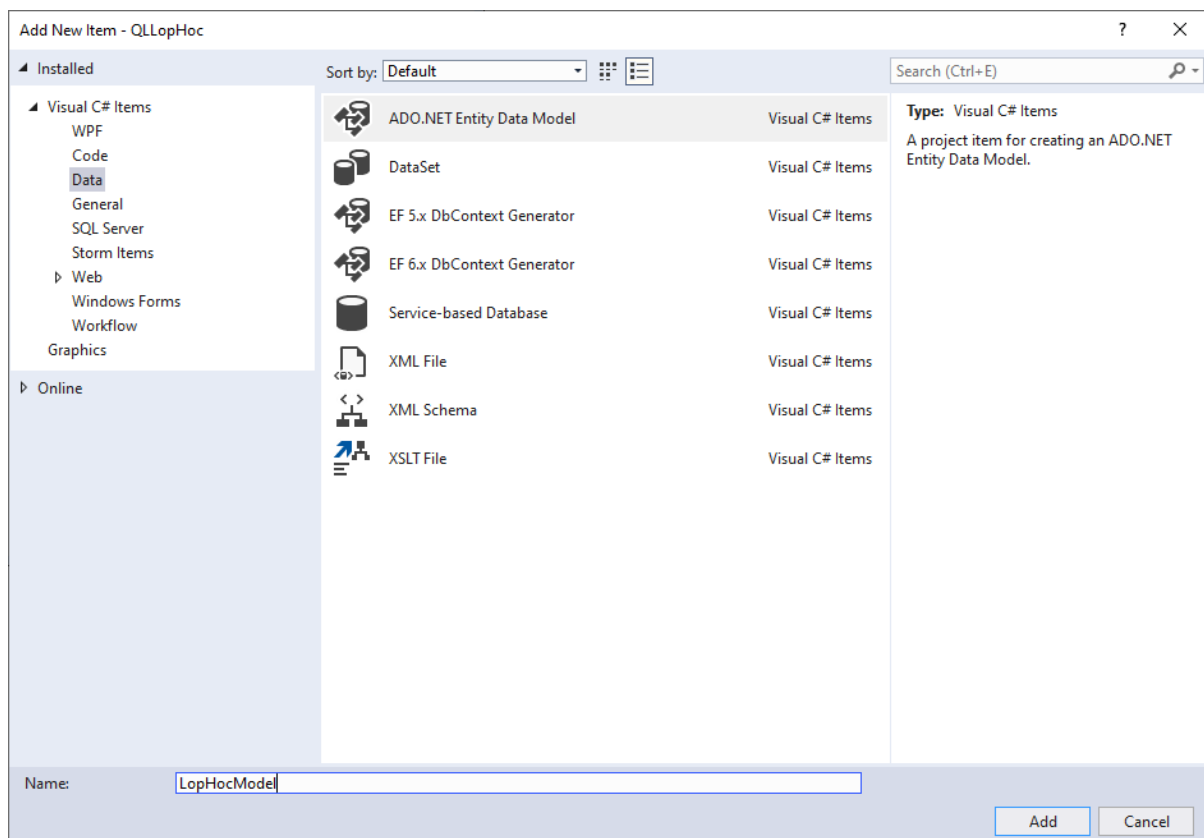
Tạo thư mục DAL (là nơi lưu model tạo từ Database First)



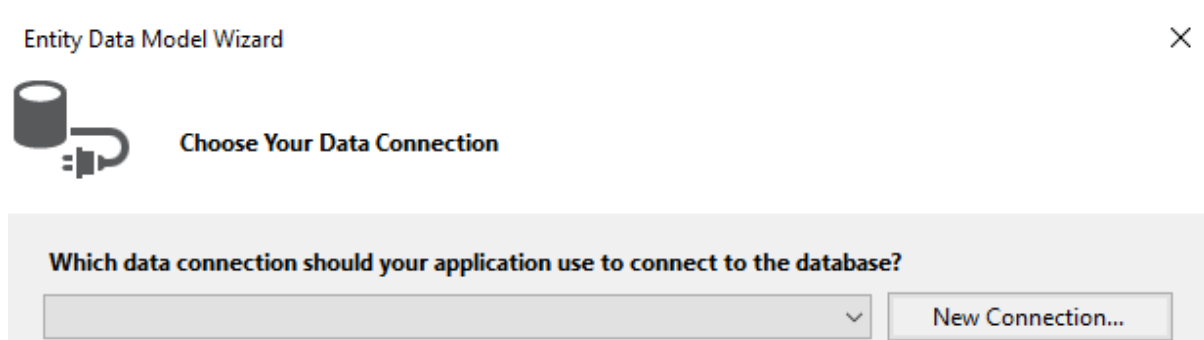
Nhấp chuột phải lên thư mục DAL → Add → New Item

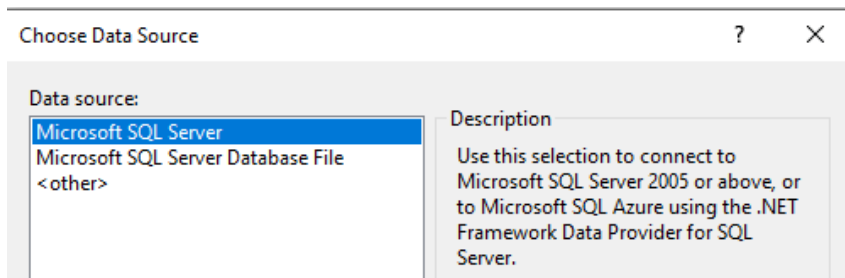


Chọn Data → ADO.NET Entity Data Model → Đặt tên **LopHocModel**

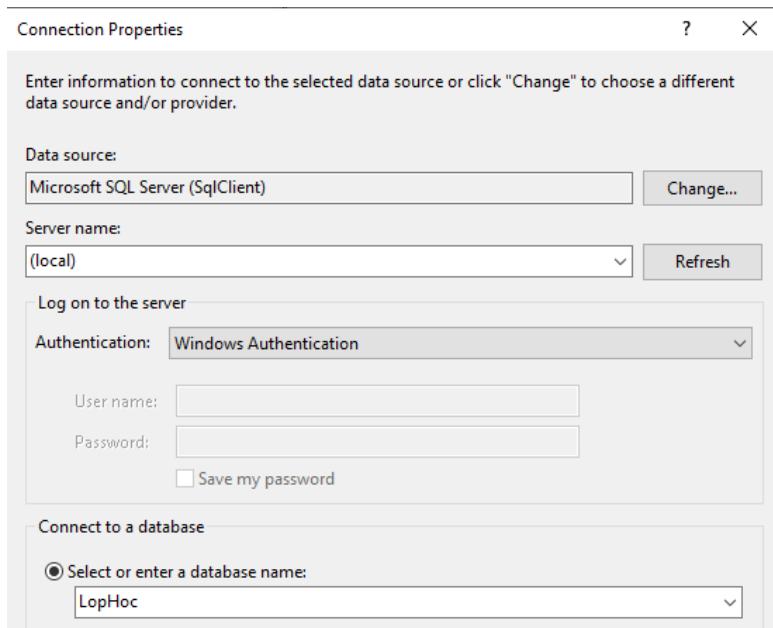


Chọn **New Connection**

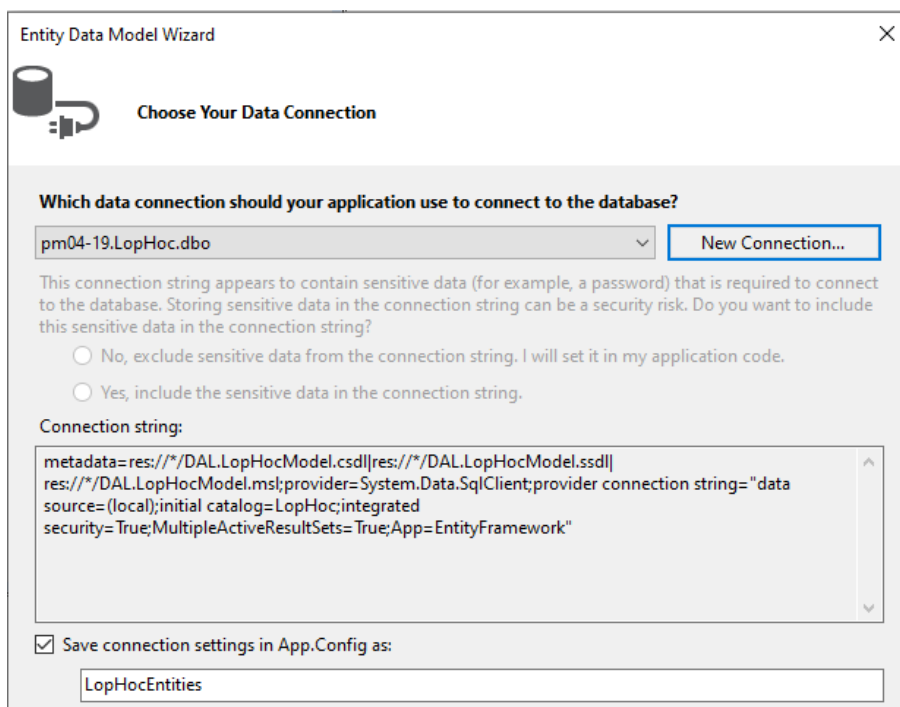





Tại Server name chọn (local) và chọn database “LopHoc” như hình sau:



Kết quả:



Entity Data Model Wizard



**Choose Your Version**

**Which version of Entity Framework do you want to use?**

☒ Entity Framework 6.x

☐ Entity Framework 5.0

Chọn 2 bảng cần:

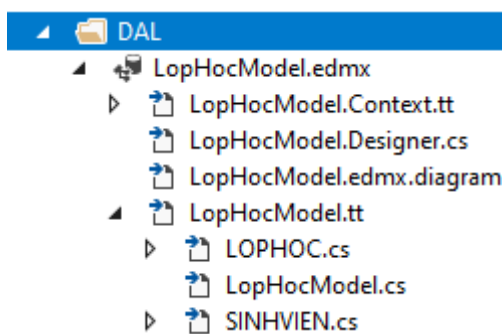
**Which database objects do you want to include in your model?**

- ☒ **Tables**
  - ☒ **dbo**
    - ☒ LOPHOC
    - ☒ SINHVIEN
  - ☐ Views
  - ☐ Stored Procedures and Functions

☒ Pluralize or singularize generated object names

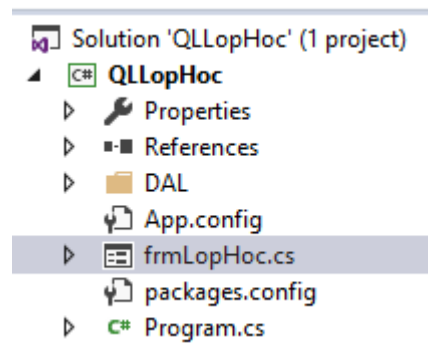
☒ Include foreign key columns in the model

Kết quả:

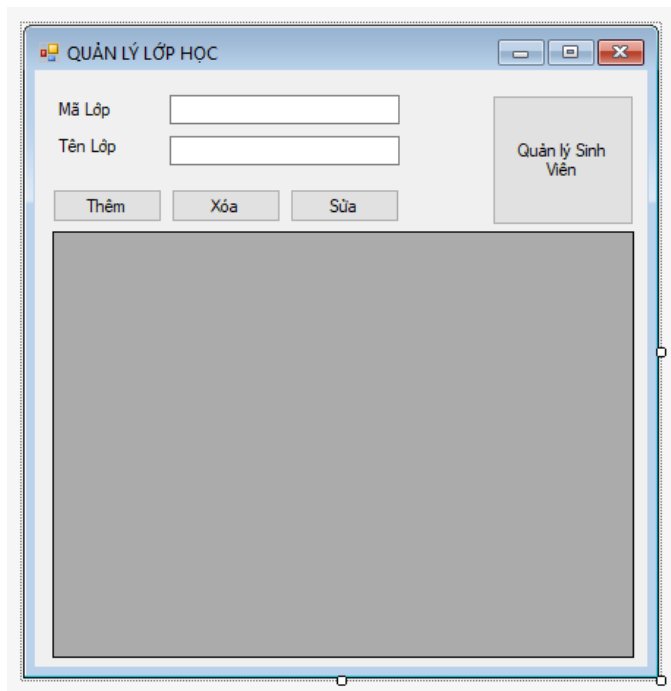


### Bước 3: Tạo form quản lý lớp học

Đổi tên file thành **frmLopHoc.cs**



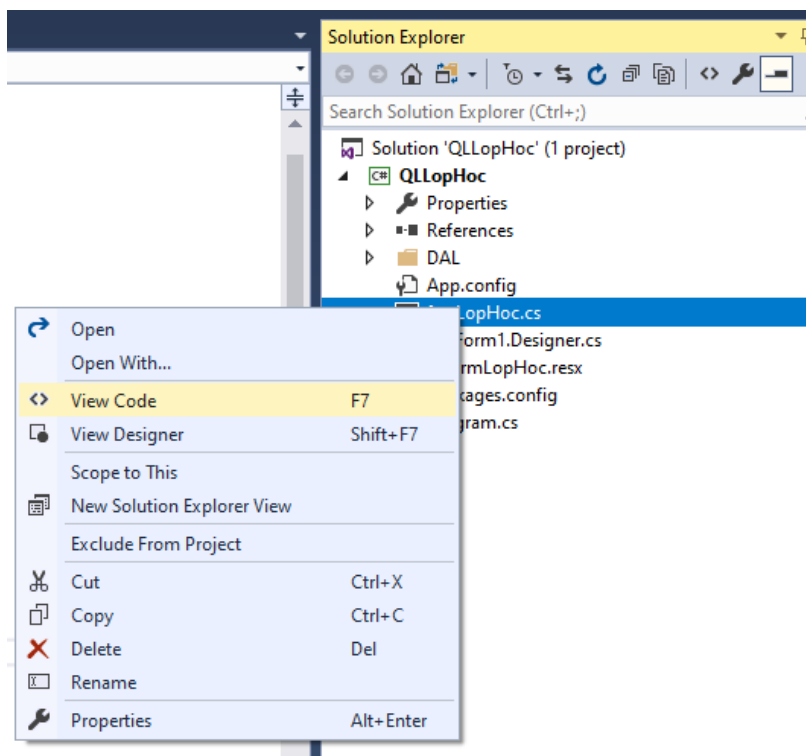
Tạo giao diện như sau:



Đặt tên cho các control:

- txtMaLop
- txtTenLop
- btnThemLop (cho button Thêm)
- btnXoaLop (cho button Xóa)
- btnSuaLop (cho button Sửa)
- dgvLopHoc (cho DataGridView)
- btnQLSV (cho button Quản lý Sinh Viên)

Nhấp chuột phải lên frmLopHoc → ViewCode



Trong code behind ta bổ sung

**using QLLopHoc.DAL;**



#### Bước 4: Tạo các chức năng quản lý cơ bản cho lớp học

Code:

```
private LopHocEntities database = new LopHocEntities();
public frmLopHoc()
{
    InitializeComponent();
    LoadThongTinLop();
    ChangeGridViewHeaderName();
}

private void ChangeGridViewHeaderName()
{
    dgvLopHoc.Columns[0].HeaderText = "Mã Lớp";
    dgvLopHoc.Columns[1].HeaderText = "Tên Lớp";
}

private void AddLopHocBinding()
{
    //Refresh lại
    txtMaLop.DataBindings.Clear();
    txtTenLop.DataBindings.Clear();

    //Add lại binding
    txtMaLop.DataBindings.Add("Text", dgvLopHoc.DataSource, "MaLop");
    txtTenLop.DataBindings.Add("Text", dgvLopHoc.DataSource, "TenLop");
}

private void LoadThongTinLop()
{
    var dsLopHoc = from lop in database.LOPHOCs
                   select new { MaLop = lop.MALOP, TenLop = lop.TENLOP };

    dgvLopHoc.DataSource = dsLopHoc.ToList();

    //Add binding
    AddLopHocBinding();
}

private void btnThemLop_Click(object sender, EventArgs e)
{
    string MaLop = txtMaLop.Text;
    string TenLop = txtTenLop.Text;
    //Da xuất hiện trong CSDL
    LOPHOC lop = database.LOPHOCs.Where(l => l.MALOP ==
MaLop).SingleOrDefault();
    if (lop != null)
    {
        MessageBox.Show("Mã lớp học đã tồn tại");
        return;
    }
    else if (String.IsNullOrEmpty(MaLop) || String.IsNullOrEmpty(TenLop))
    {
        MessageBox.Show("Mã lớp hoặc Tên lớp không được để trống");
        return;
    }
    else
    {
        lop = new LOPHOC();
        lop.MALOP = MaLop;
    }
}
```

```

        lop.TENLOP = TenLop;
        database.LOPHOCs.Add(lop);
        database.SaveChanges();
        LoadThongTinLop();
        MessageBox.Show("Thêm mới lớp học thành công");
    }
}

private void btnXoaLop_Click(object sender, EventArgs e)
{
    string MaLop = txtMaLop.Text;
    string TenLop = txtTenLop.Text;
    //Đã xuất hiện trong CSDL
    LOPHOC lop = database.LOPHOCs.Where(l => l.MALOP ==
MaLop).SingleOrDefault();
    if (lop == null)
    {
        MessageBox.Show("Mã lớp học không tồn tại");
        return;
    }
    else if (String.IsNullOrEmpty(MaLop))
    {
        MessageBox.Show("Mã lớp cần xóa không được để trống");
        return;
    }
    else
    {
        if (lop.SINHVIENs.Count > 0)
        {
            MessageBox.Show("Hãy xóa sinh viên trong lớp trước");
            return;
        }
        database.LOPHOCs.Remove(lop);
        database.SaveChanges();
        LoadThongTinLop();
        MessageBox.Show("Xóa lớp học thành công");
    }
}

private void btnSuaLop_Click(object sender, EventArgs e)
{
    string MaLop = txtMaLop.Text;
    string TenLop = txtTenLop.Text;
    //Đã xuất hiện trong CSDL
    LOPHOC lop = database.LOPHOCs.Where(l => l.MALOP ==
MaLop).SingleOrDefault();
    if (lop == null)
    {
        MessageBox.Show("Mã lớp học không tồn tại");
        return;
    }
    else if (String.IsNullOrEmpty(MaLop))
    {
        MessageBox.Show("Mã lớp cần sửa không được để trống");
        return;
    }
    else
    {
        lop.TENLOP = TenLop;
        database.SaveChanges();
        LoadThongTinLop();
        MessageBox.Show("Cập nhật thông tin lớp học thành công");
    }
}

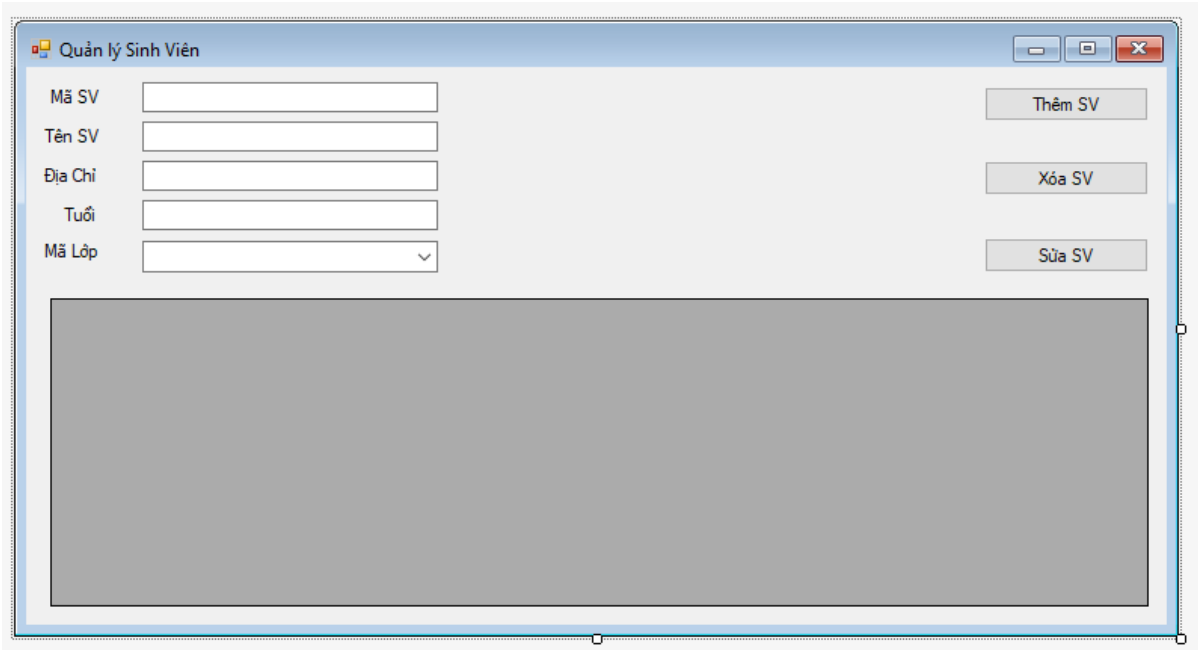
```

```
    }  
}  
  
private void btnQLSV_Click(object sender, EventArgs e)  
{  
    this.Hide();  
    frmSinhVien formSV = new frmSinhVien();  
    formSV.ShowDialog();  
    formSV = null;  
    this.Show();  
}
```

Nhớ đăng kí sự kiện cho từng nút trước khi qua bước 5

Bước 5: Xử lý code cho quản lý sinh viên từng lớp (quan hệ 1 – n)

Tạo giao diện quản lý Sinh viên như sau:



Đặt tên cho các control:

- txtMaSV
- txtTenSV
- txtDiaChiSV
- txtTuoi
- cmbLopHoc (combobox cho Mã Lớp)
- dgvSinhVien (data grid view để hiển thị thông tin sinh viên)
- btnThemSV
- btnXoaSV
- btnSuaSV

Chuẩn bị:

```
public partial class frmSinhVien : Form
{
    private LopHocEntities database = new LopHocEntities();
    public frmSinhVien()
    {
        InitializeComponent();
        LoadThongTinSinhVien();
    }
}
```

Bổ sung các hàm để data grid view thể hiện thông tin sinh viên

```
private void AddSinhVienBinding()
{
    //Refresh lại
    txtMaSV.DataBindings.Clear();
    txtTenSV.DataBindings.Clear();
    txtTuoi.DataBindings.Clear();
    txtDiaChiSV.DataBindings.Clear();
    cmbLopHoc.DataBindings.Clear();

    //Add lại binding
    txtMaSV.DataBindings.Add("Text", dgvSinhVien.DataSource, "MaSV");
    txtTenSV.DataBindings.Add("Text", dgvSinhVien.DataSource, "TenSV");
    txtTuoi.DataBindings.Add("Text", dgvSinhVien.DataSource, "Tuoi");
    txtDiaChiSV.DataBindings.Add("Text", dgvSinhVien.DataSource, "DiaChi");
    cmbLopHoc.DataBindings.Add("Text", dgvSinhVien.DataSource, "TenLop");
}

private void LoadThongTinSinhVien()
{
    var dsLopHoc = from sv in database.SINHVIENS
                   select new { MaSV = sv.MASV, TenSV = sv.TENSV, DiaChi =
sv.DIACHI, Tuoi = sv.TUOI, TenLop = sv.LOPHOC.TENLOP };

    dgvSinhVien.DataSource = dsLopHoc.ToList();
    cmbLopHoc.DataSource = database.LOPHOCs.ToList();
    cmbLopHoc.DisplayMember = "TenLop";

    //Add binding
    AddSinhVienBinding();
}
```

Hàm xử lý thêm mới SV:

```
private void btnThemSV_Click(object sender, EventArgs e)
{
    string MaSV = txtMaSV.Text;
    string TenSV = txtTenSV.Text;
    string TuoiSV = txtTuoi.Text;
    string DiaChi = txtDiaChiSV.Text;
    LOPHOC lop = cmbLopHoc.SelectedValue as LOPHOC;
```

```

        //Da xuất hiện trong CSDL
        SINHVIEN sv = database.SINHVIENs.Where(s => s.MASV ==
MaSV).SingleOrDefault();
        if (sv != null)
        {
            MessageBox.Show("Mã sinh viên đã tồn tại");
            return;
        }
        else if (String.IsNullOrEmpty(MaSV) || String.IsNullOrEmpty(TenSV))
        {
            MessageBox.Show("Mã sinh viên hoặc Tên sinh viên không được để
trống");
            return;
        }
        else
        {
            SINHVIEN sinhvien = new SINHVIEN();
            sinhvien.MASV = MaSV;
            sinhvien.TENSV = TenSV;
            sinhvien.DIACHI = DiaChi;
            sinhvien.TUOI = Convert.ToInt32(TuoiSV);
            sinhvien.LOPHOC = lop;

            database.SINHVIENs.Add(sinhvien);
            database.SaveChanges();

            LoadThongTinSinhVien();
            MessageBox.Show("Thêm mới sinh viên thành công");
        }
    }
}

```

## Hàm xử lý xóa thông tin sinh viên

```

private void btnXoaSV_Click(object sender, EventArgs e)
{
    string MaSV = txtMaSV.Text;
    //Da xuất hiện trong CSDL
    SINHVIEN sv = database.SINHVIENs.Where(s => s.MASV ==
MaSV).SingleOrDefault();
    if (sv == null)
    {
        MessageBox.Show("Mã sinh viên không tồn tại");
        return;
    }
    else if (String.IsNullOrEmpty(MaSV))
    {
        MessageBox.Show("Mã sinh viên cần xóa không được để trống");
        return;
    }
    else
    {
        database.SINHVIENs.Remove(sv);
        database.SaveChanges();
        LoadThongTinSinhVien();
        MessageBox.Show("Xóa sinh viên thành công");
    }
}
}

```

Bài tập:

- Hãy bổ sung hàm edit thông tin Sinh viên
- Vận dụng cho bài tập tại lớp lý thuyết