

Lab 07:

# LẬP TRÌNH WINDOWS FORM KẾT NỐI CSDL CƠ BẢN

## A. MỤC TIÊU:

- ✓ Hướng dẫn sinh viên làm quen với việc xây dựng ứng dụng Windows App có kết nối với CSDL SQL Server.
- ✓ Sử dụng chức năng Data Source Configuration Wizard để tạo Data Source.
- ✓ Thiết kế các Form nhập liệu cho các bảng trong cơ sở dữ liệu.
- ✓ Thực thi các câu truy vấn dữ liệu cơ bản trong ứng dụng

## B. NỘI DUNG:

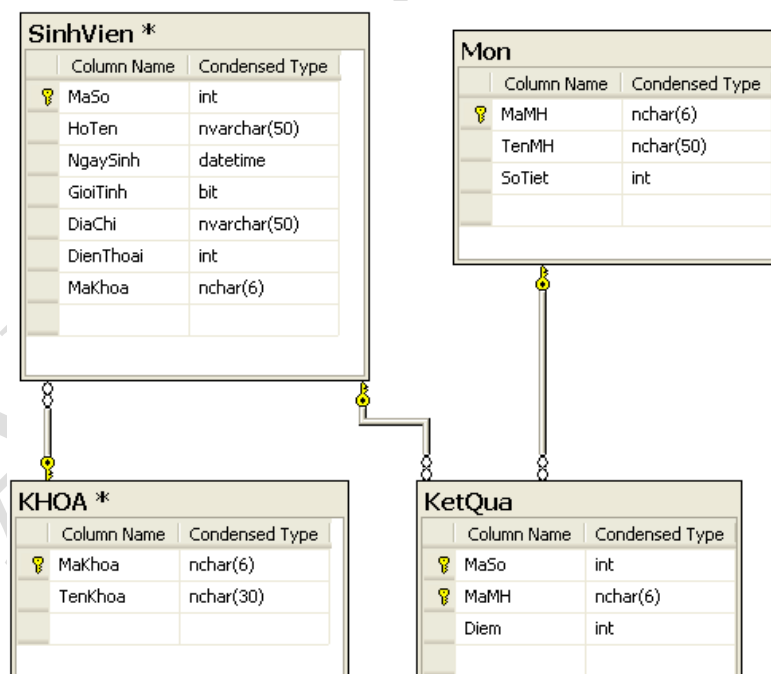
**Bài tập 1:** Xây dựng ứng dụng Windows Form minh họa **quản lý thông tin sinh viên cơ bản**, các thông tin cần quản lý bao gồm:

- ✓ Thông tin sinh viên: MaSo, HoTen, NgaySinh, GioiTinh, DiaChi, DienThoai, Khoa
- ✓ Thông tin các môn học: MaMH, TenMH, SoTiet
- ✓ Thông tin các khoa: MaKhoa, TenKhoa

Tra cứu điểm các môn học của sinh: mỗi sinh viên sẽ có các điểm với các môn học tương ứng.

Tra cứu danh sách sinh viên theo khoa

Cơ sở dữ liệu được mô tả qua Diagram như sau:

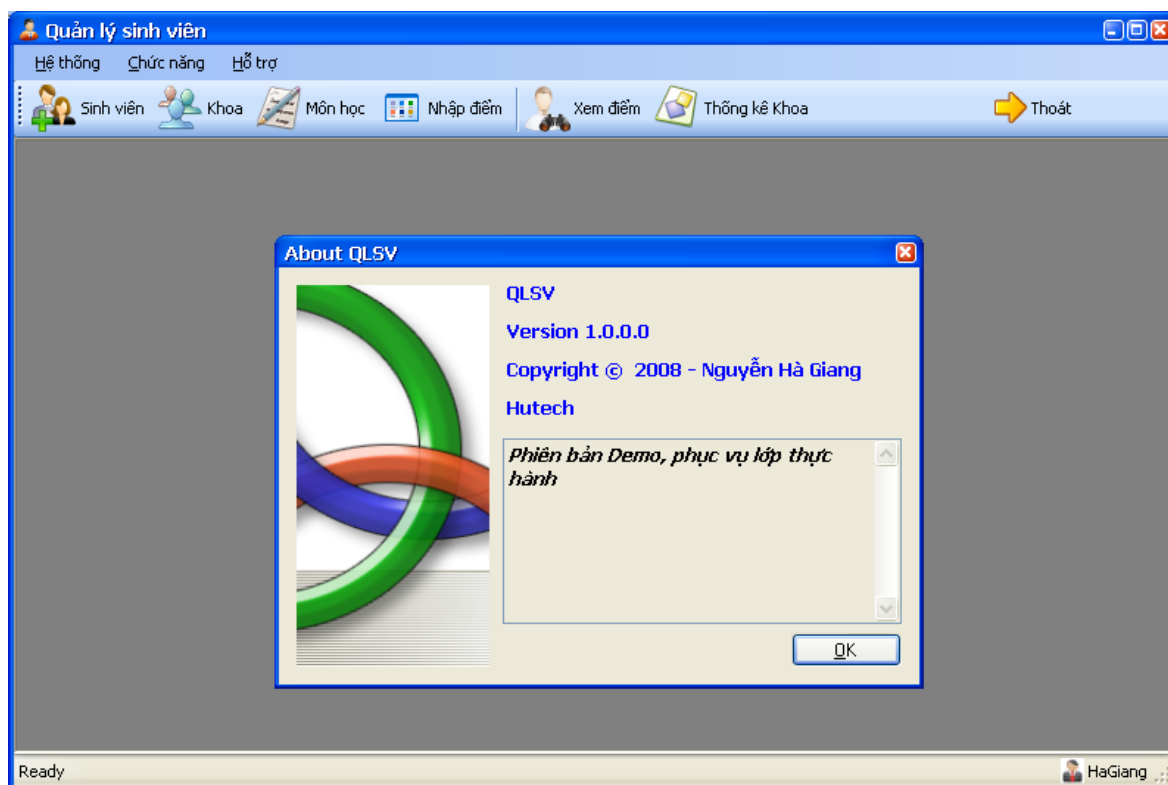


Bảng 1: Mô tả cơ sở dữ liệu cho ứng dụng

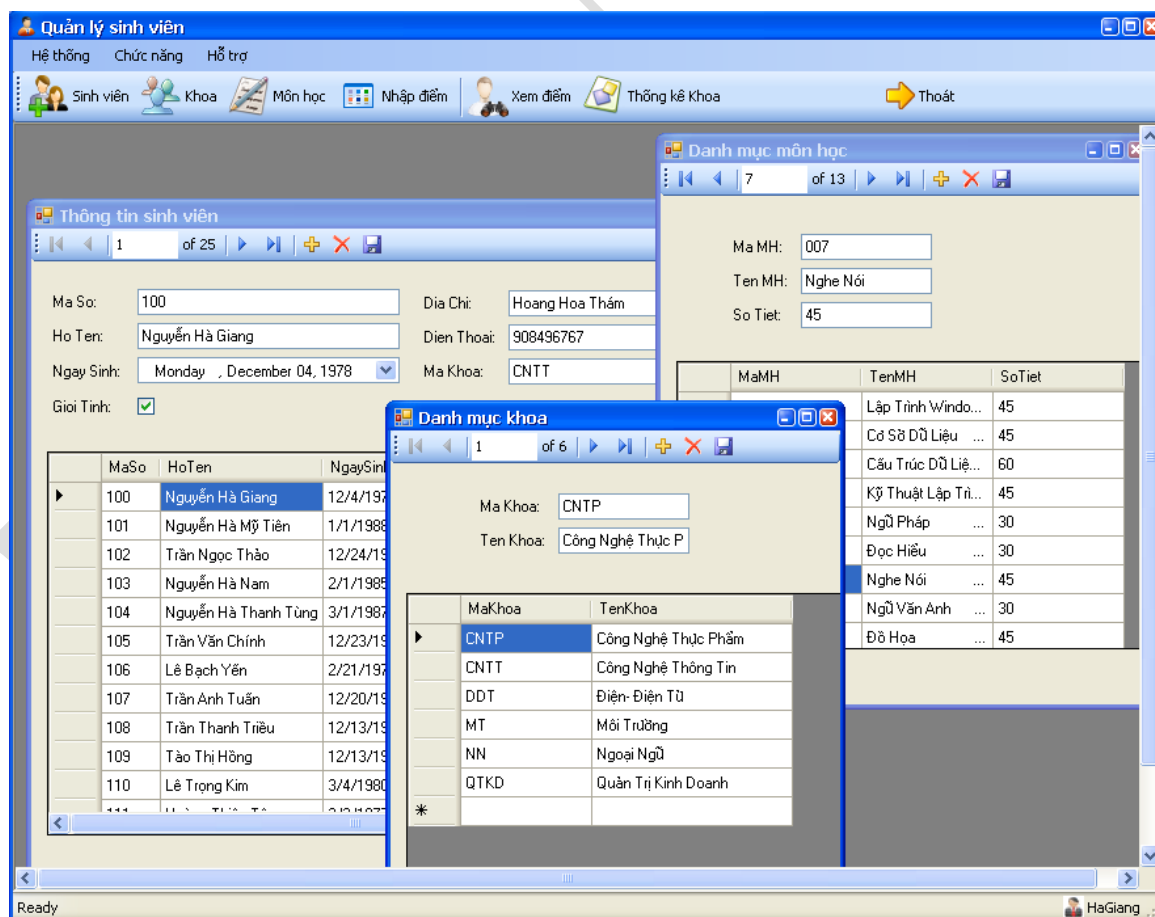
**Yêu cầu:** Viết chương trình Windows Application dạng MDI có các chức năng chính như sau:

- ✓ Chức năng nhập liệu: Thông tin sinh viên, Danh mục các khoa, Danh mục các môn học, Điểm của các sinh viên
- ✓ Chức năng thống kê/báo cáo: Xem danh sách sinh viên với điểm tương ứng, Xem sinh viên trong từng khoa tương ứng

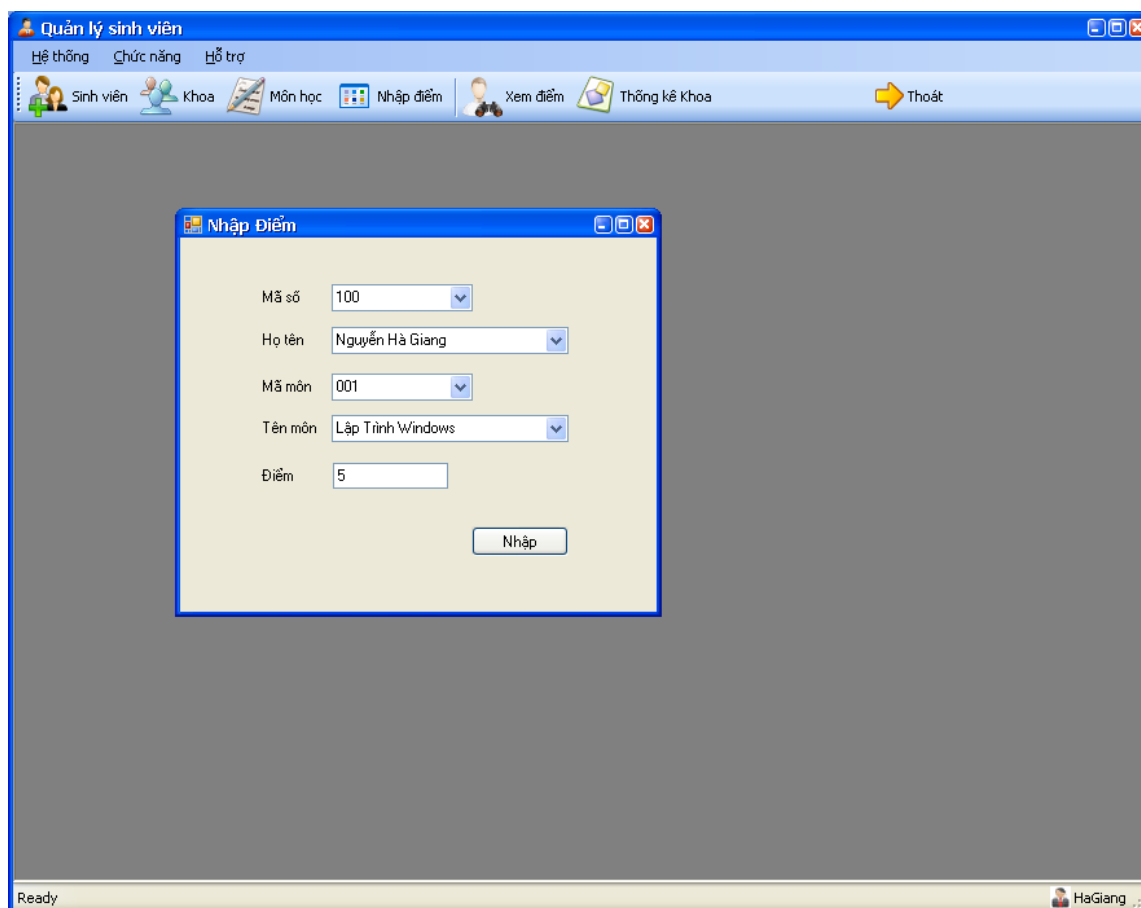
Màn hình chính của chương trình minh họa như sau



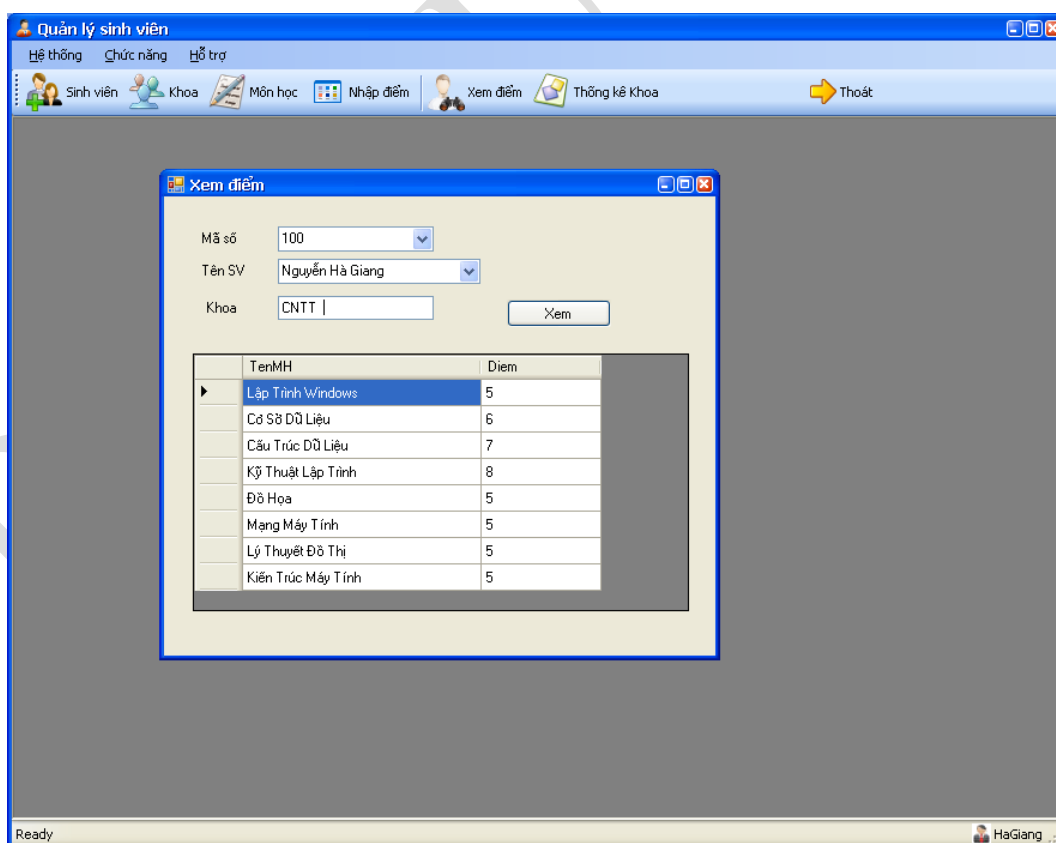
Hình 2: Màn hình chính của chương trình



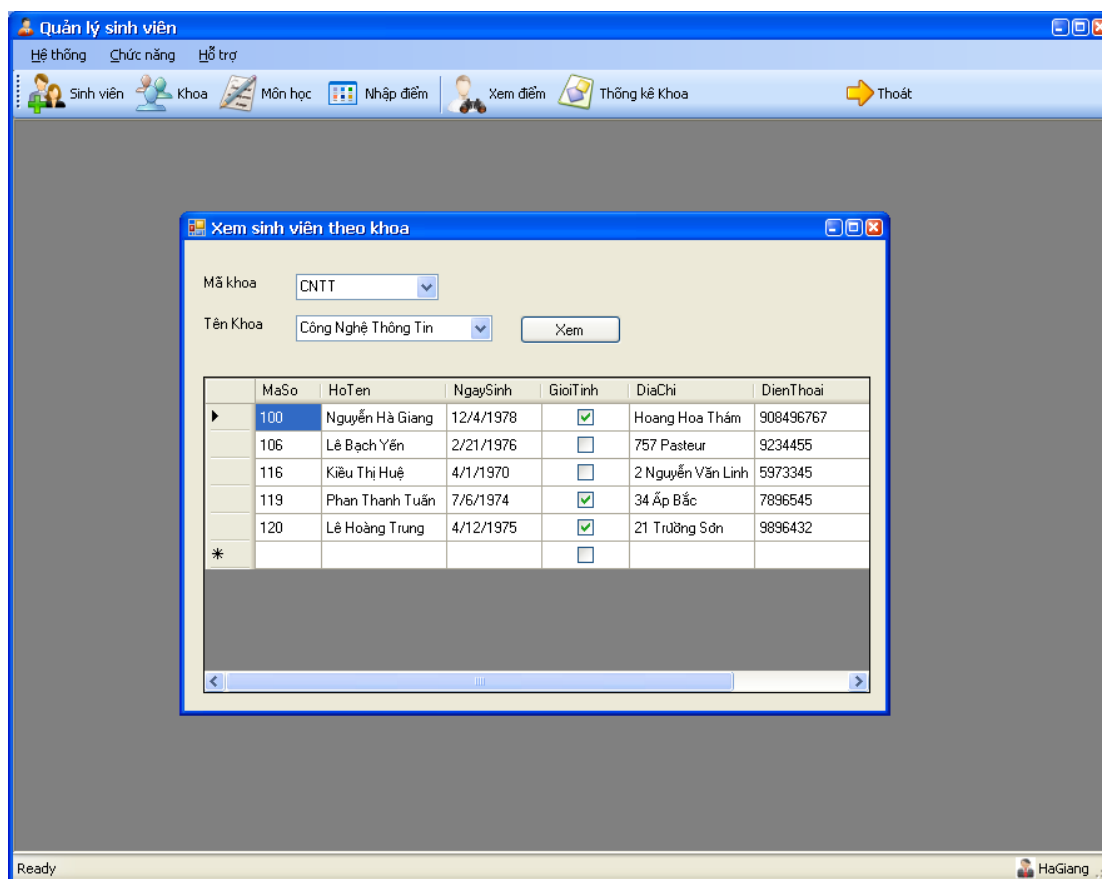
Hình 3: Màn hình các chức năng nhập liệu



Hình 4: Màn hình nhập điểm



Hình 5: Màn hình tra cứu điểm

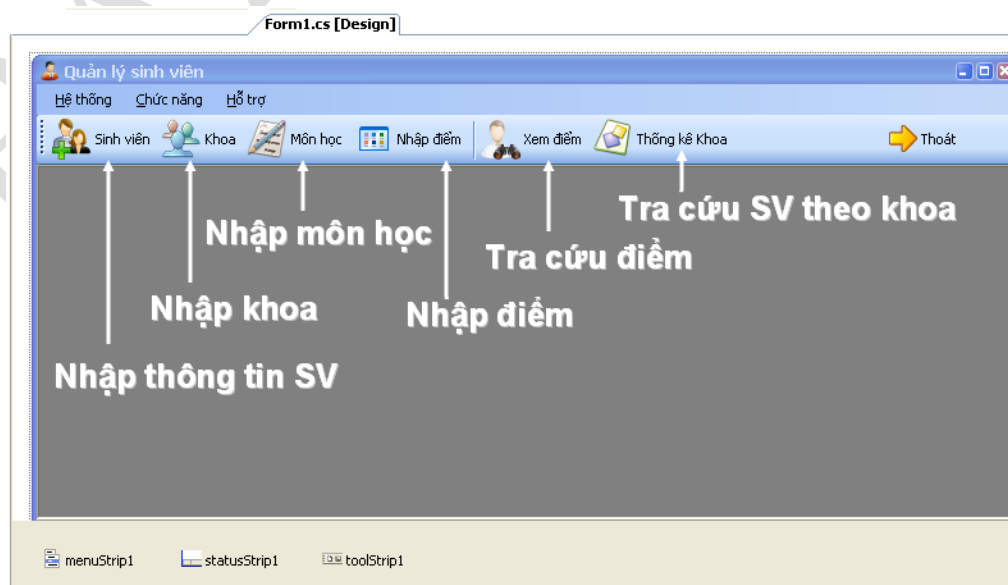


Hình 6: Màn hình xem sinh viên thuộc khoa

## Hướng dẫn cơ bản:

### 1. Tạo ứng dụng MDI

- ✓ Thiết kế giao diện Form chính (đây là parent Form của ứng dụng) như hình 7 minh họa bên dưới.
- ✓ Các chức năng chính của ứng dụng được thể hiện trong MenuStrip và trên thanh ToolStrip.
- ✓ Mỗi chức năng ví dụ như nhập danh sách sinh viên sẽ có hai cách thao tác: và trên ToolStrip.

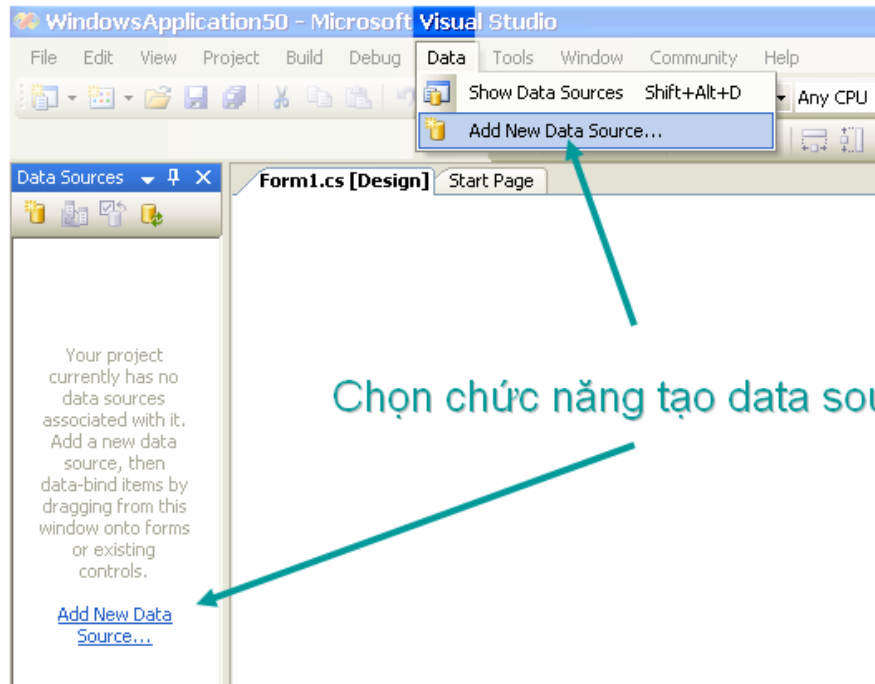


Hình 7: Thiết kế Form chính

## 2. Tạo Data Source cho ứng dụng: Sử dụng chức năng Data Source Configuration Wizard

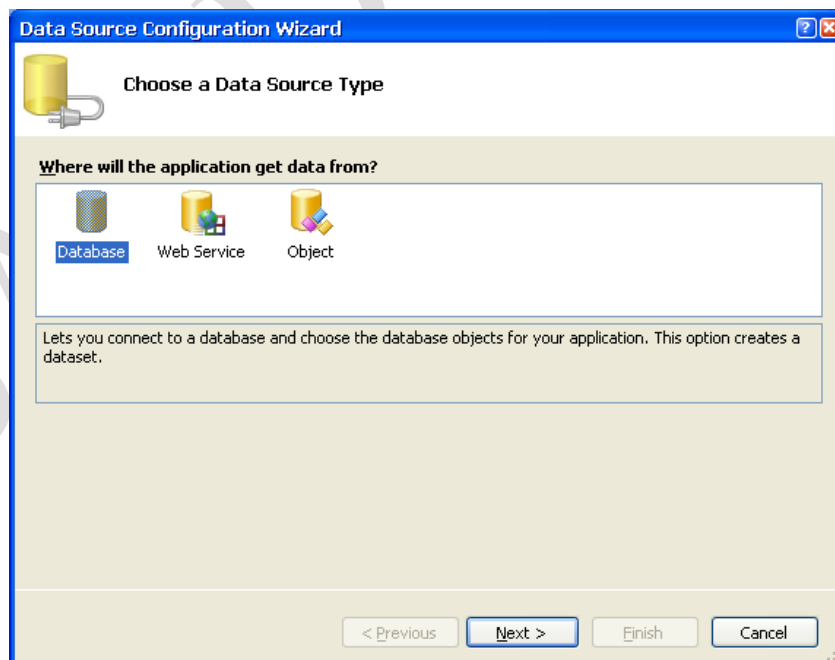
**Lưu ý:** Đã có cơ sở dữ liệu cho ứng dụng có tên là StudentDB trong SQL Server.

- ✓ Trong menu **Data** | **Add New Data Source...** hoặc open cửa sổ Data Source (Shift + Alt+D), lúc này Data Source sẽ rỗng và có item **Add New Data Source...**, Click chọn item này để tiến hành cấu hình Data Source



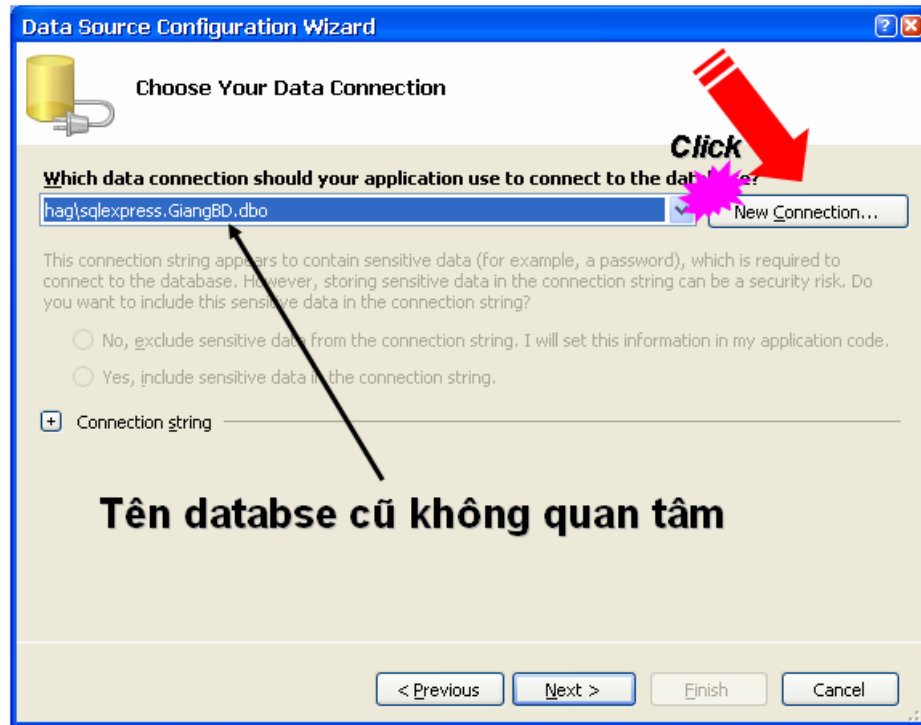
Hình 8: Chức năng Add New Data Source...

- ✓ Chọn Database trong cửa sổ Data Source Configuration Wizard



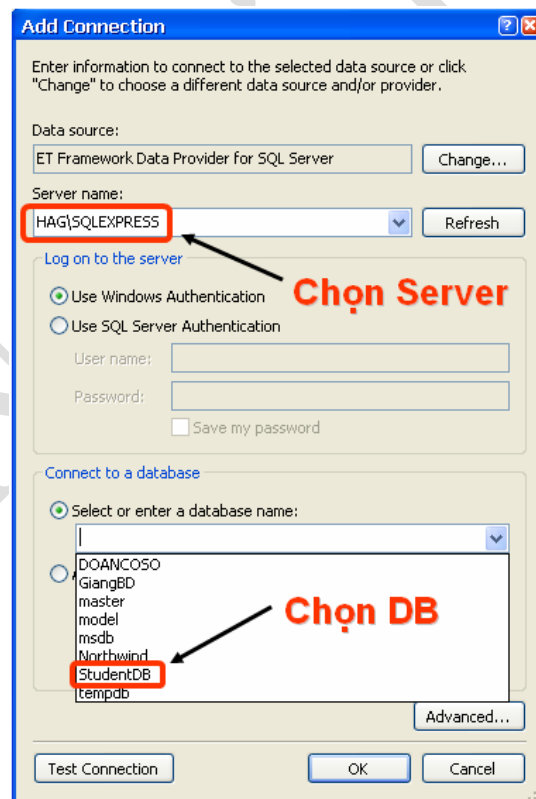
Hình 9: Chọn Database

- ✓ Bước tiếp theo chọn **New Connection**



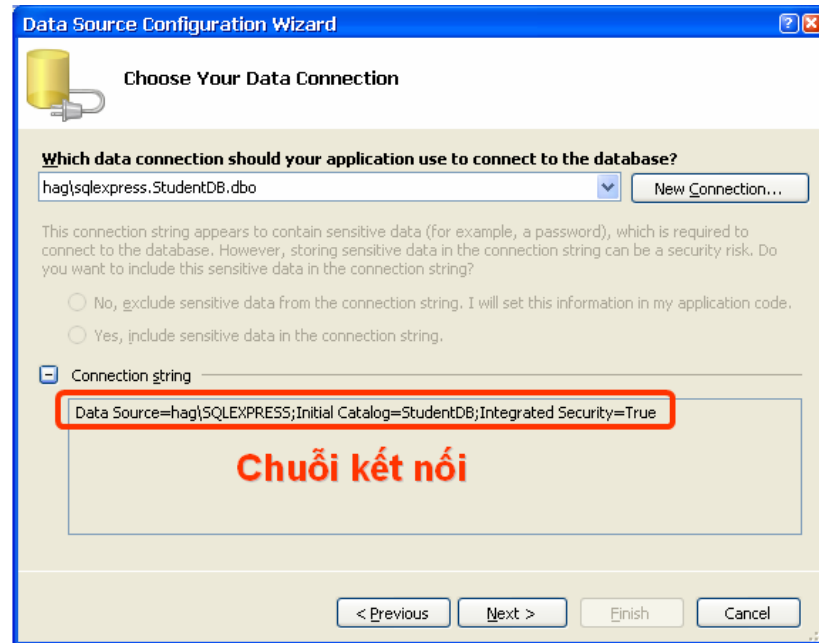
Hình 10: Chọn chức năng New Connection

- ✓ Cấu hình trong cửa sổ **Add Connection** tiếp theo



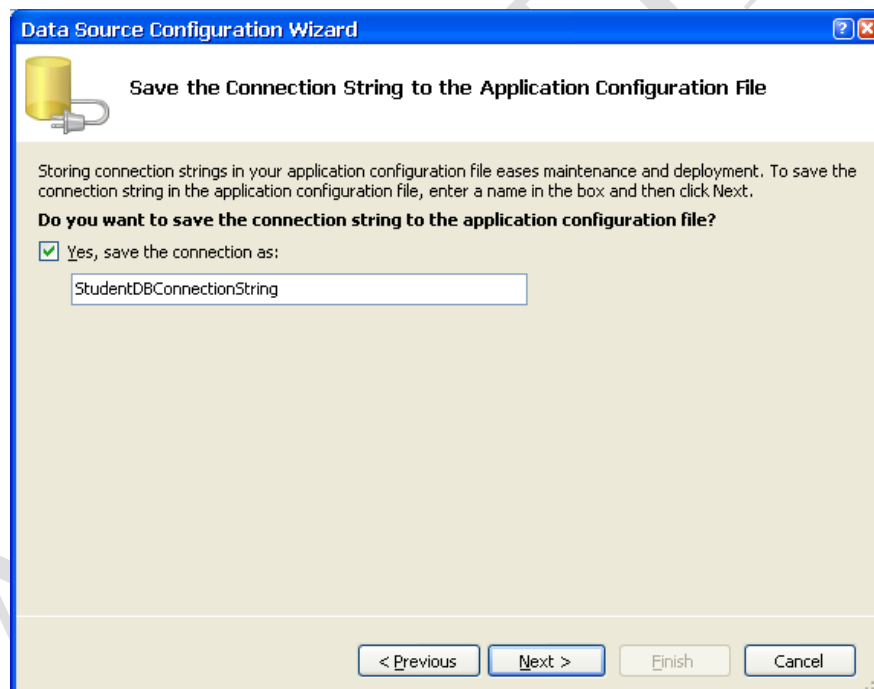
Hình 11: Cấu hình connection

- ✓ Kết quả được các thông số kết nối DB:



Hình 12: Chuỗi kết nối

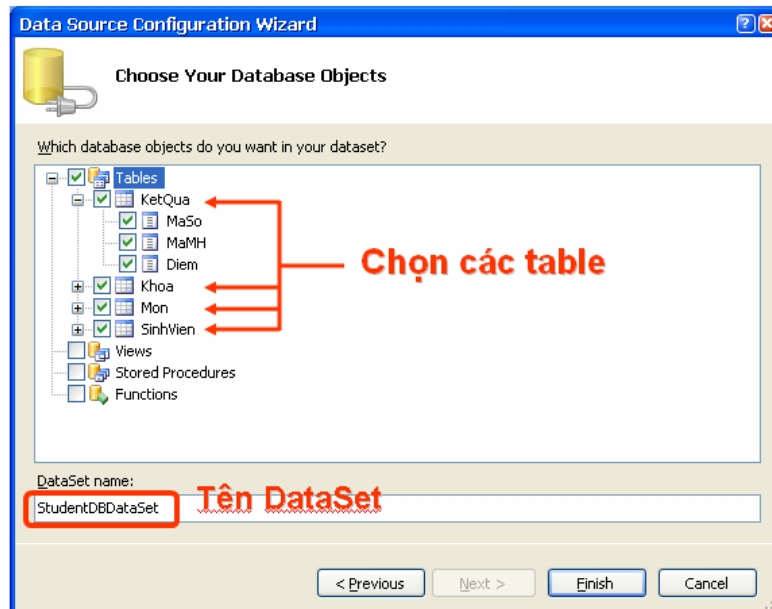
- ✓ Lưu chuỗi kết nối trong file cấu hình



Hình 13: Lưu chuỗi kết nối

- ✓ Chọn bảng dữ liệu cần xử lý trong project:

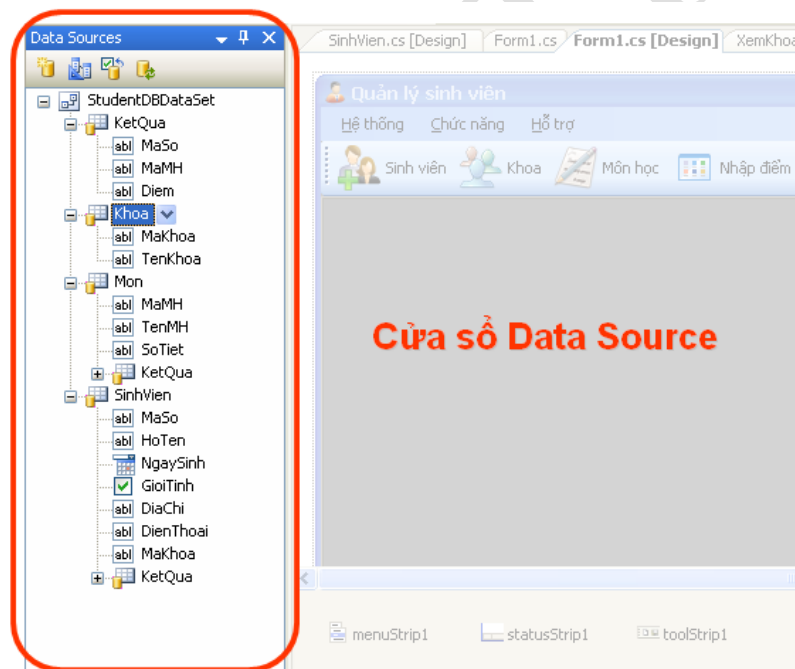
Trong cửa sổ này chúng ta có thể hiệu chỉnh lại tên DataSet sẽ được phát sinh cho project và chọn các bảng dữ liệu cần thiết để làm việc trong StudentDB



Hình 14: Chọn các Table – Field...

Cuối cùng nhấn **Finish** để hoàn thành bước cấu hình

- ✓ Sau các bước cấu hình như trên thì Data Source View có các thành phần như sau:



Hình 15: Cửa sổ Data Source chứa các bảng - field

- ✓ Trong cửa sổ Data Source chứa các table và các field cho phép kéo thả các bảng vào form, theo hai dạng: DataGridView và Details
- ✓ Khi kéo thả một bảng vào Form, VS sẽ phát sinh các cơ chế tự động binding, cho phép user thêm, xóa, sửa các thông tin trong các bảng và sẽ tự động Update vào cơ sở dữ liệu thông qua DataAdapter...

**Tóm lại:** Project đã có một Data Source View cho phép thao tác trên CSDL của Data Source đó.

### 3. Tạo các Form con chứa các chức năng tương ứng

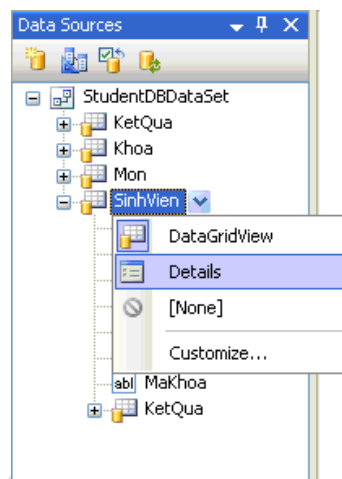
- ✓ SinhVien: Form nhập liệu thông tin sinh viên



- ✓ Khoa: Form nhập thông tin khoa
- ✓ MonHoc: Form nhập danh sách các môn học
- ✓ Diem: Form nhập điểm cho từng sinh viên

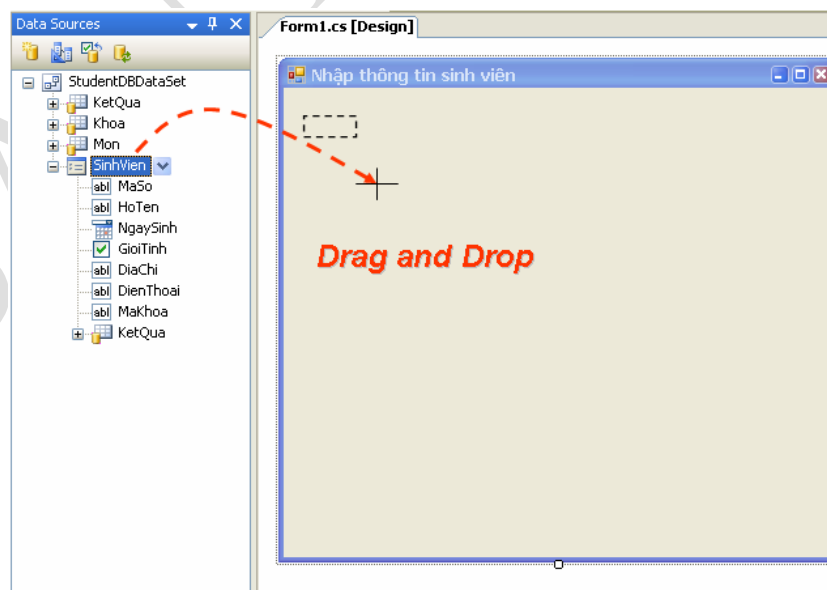
### 3.1. Minh họa tạo Form con SinhVien:

- ✓ Bước 1: Tạo Form con SinhVien
- ✓ Bước 2: Mục đích là xây dựng Form SinhVien vừa nhập thông tin dạng Details và GridView nên:
  - Trong Data Source View click chuột vào nút “**Button Down**” của bảng SinhVien. Sau đó chọn Details.



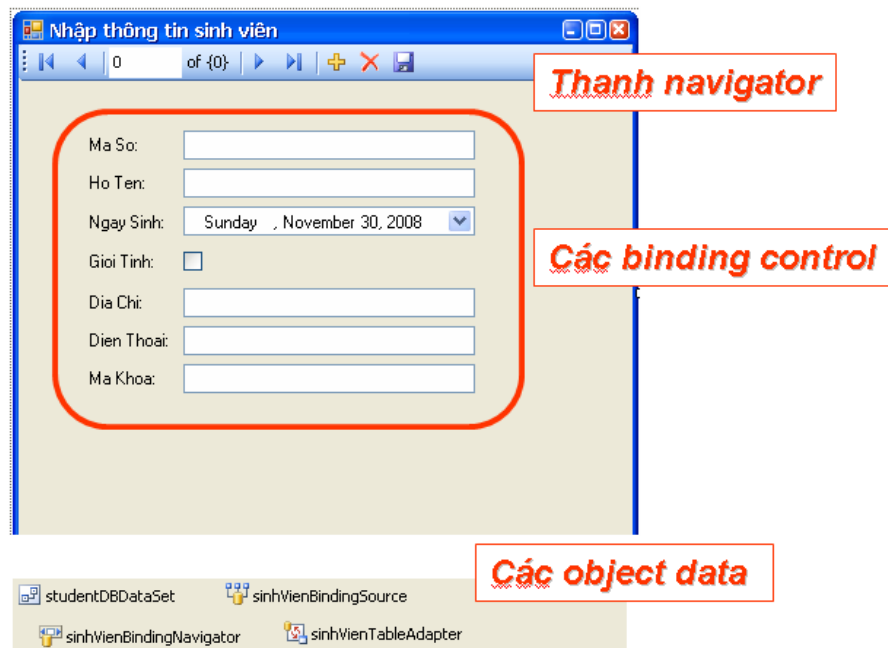
Hình 17: Chọn bảng SinhVien dạng Detail

- Kéo bảng SinhVien thả vào SinhVien: khi đó VS sẽ phát sinh tự động các label, textbox và các đối tượng để thực hiện việc kết dữ liệu bảng SinhVien với các control trên Form



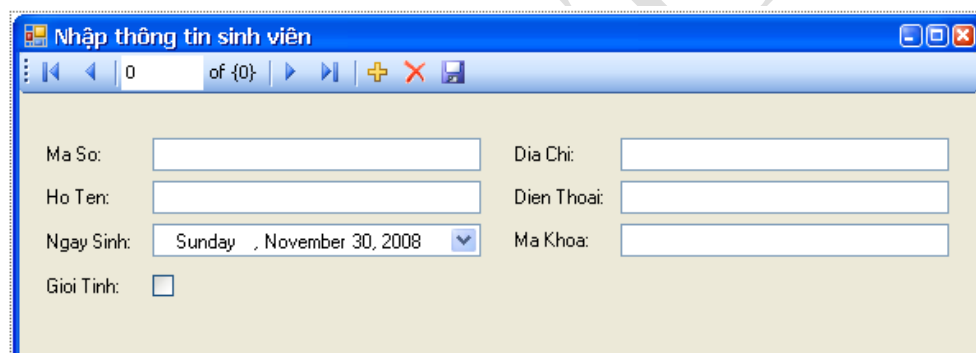
Hình 18: Kéo bảng SinhVien từ Data Source thả vào Form

VS tự động phát sinh các control và code liên quan như sau



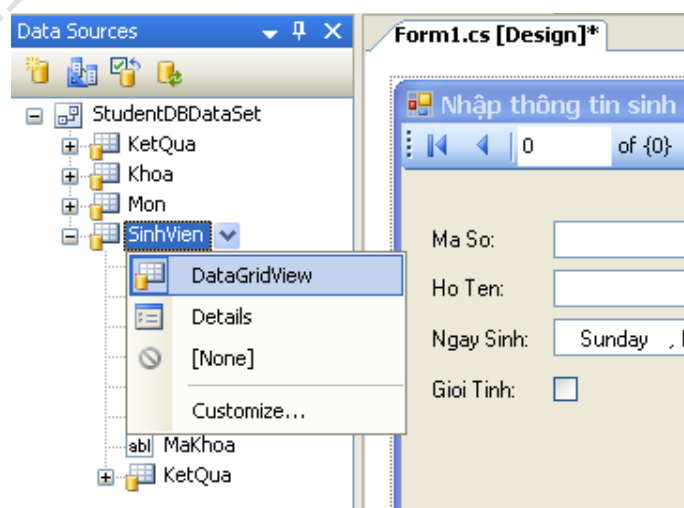
Hình 19: Kết quả sao khi kéo thả table SinhVien vào Form.

- Ta thiết kế lại form SinhVien cho thích hợp để dành không gian cho DataGridView sẽ thêm bên dưới các control này



Hình 20: Bố trí lại các control trên Form

- Trong Data Source View kích chuột vào nút “**Button Down**” của bảng SinhVien. Sau đó chọn DataGridView

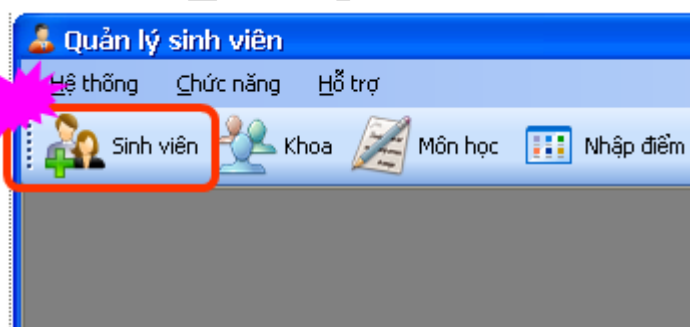


Hình 21: Chọn bảng SinhVien dạng DataGridView trong Data Source

- Bước 3.4: Kéo bảng SinhVien thả vào Form và được kết quả như hình sau

Hình 22: Form SinhVien sau khi kéo bảng SinhVien dạng DataGridView

- ✓ Bước 3: Tạo chức năng gọi Form SinhVien trong Form chính của ứng dụng. Trong menu item Sinh Viên và trong thanh ToolStrip có button “Sinh Viên”, hai chức năng này cùng gọi Form Sinh Vien ra màn hình. Trình xử lý cho cả hai chức năng này được minh họa như sau



Hình 23: Chức năng nhập ds sinh viên trên ToolStrip

Code minh họa, có tính chất gợi ý, sinh viên hiểu và tự viết lại cho đúng

```
foreach (Form f in this.MdiChildren)
    // Kiểm tra xem đã có Form sinh vien chưa
    if (f.Name == "SinhVien")
    {
        f.Activate(); // nếu có form sinh vien thì active
        return;
    }
    // trường hợp không có thì tạo mới
    SinhVien sv = new SinhVien();
    sv.MdiParent = this; // thiết lập form cha
    sv.Show(); // hiển thị
```

Ý nghĩa của phần code trên như sau: Khi gọi chức năng “Sinh viên”, form cha sẽ kiểm tra trong danh sách các form con của nó, có chứa form SinhVien hay chưa, nếu đã có rồi nó chỉ cần

activate form SinhVien con là đủ, còn trường hợp ngược lại sẽ tạo thể hiện của Form SinhVien con, thiết lập quan hệ MdiParent cho form SinhVien con và cuối cùng là Show ra màn hình.

Tương tự: Sinh viên tạo các form: Khoa, Mon và NhapDiem một cách tương tự, sau đó định nghĩa các trình xử lý cho các button trên Form chính để gọi các form con...

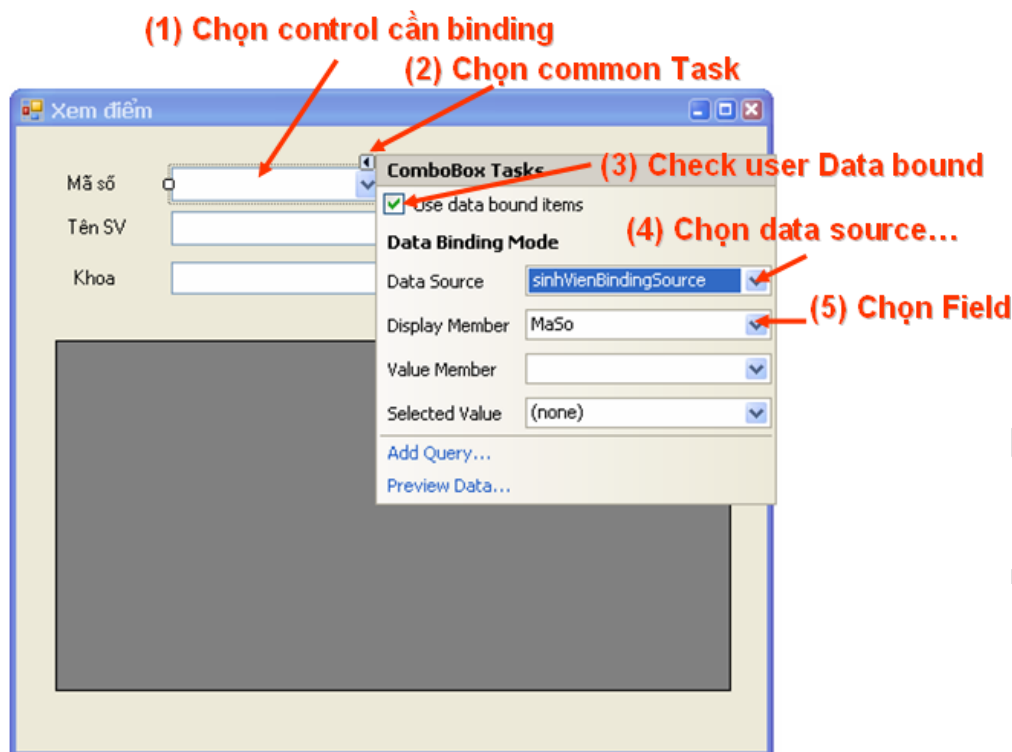
**3.2. Tạo Form xem điểm:** Form sẽ lấy mã số sinh viên A được chọn và truy vấn với bảng Kết Quả để lấy các điểm đã có của sinh viên A. Các bước thực hiện minh họa như sau:

**Bước 1:** Tạo Form mới có tên XemDiem và thiết kế form như hình dưới



Hình 24: Màn hình thiết kế của Form XemDiem.

- ✓ Các control trên form được binding với bảng dữ liệu SinhVien trong DataSource. Trong đó ComboBox thứ 1 gắn với Field MaSo, ComboBox thứ 2 gắn với Field HoTen, và Text Box cuối cùng gắn với Field Khoa.
- ✓ Để thiết lập cơ chế binding này ta sử dụng thuộc tính: Data Source và DisplayMember của hai ComboBox. Trong đó Data Source chính là bảng SinhVien và DisplayMember là tên cột trong bảng SinhVien cần hiển thị.
- ✓ Màn hình bên dưới minh họa các binding cột dữ liệu tương ứng cho ComboBox



Hình 25: Sử dụng Common Task để binding dữ liệu cho ComboBox

**Bước 2:** Viết phần xử lý cho button Xem, phần xử lý này chính là trình xử lý sự kiện cho button Xem của Form XemDiem.

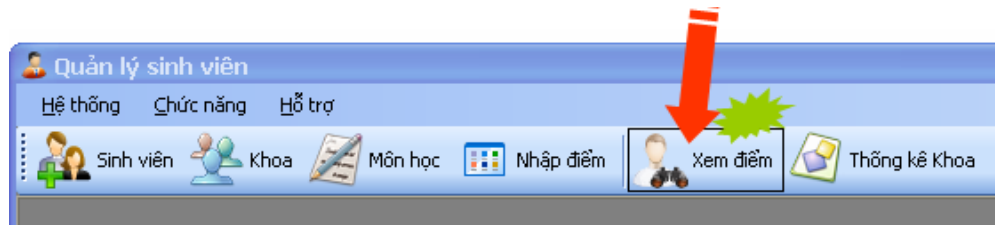
```
// lấy chuỗi connectionstring từ app.config
string conn =
global::StudentManaging2.Properties.Settings.Default.StudentDBConnectionString;
string str;
// truy vấn lấy điểm các môn học của SV có mã số được chọn
str = string.Format("Select TenMH, Diem From KetQua,Mon where [KetQua].MaMH =
[Mon].MaMH AND MaSo = {0}",comboBox1.Text);
SqlDataAdapter da = new SqlDataAdapter(str,conn);
DataSet ds = new DataSet();
// đổ vào dataset
da.Fill(ds);
// hiển thị ra DataGridView
dataGridView1.DataSource = ds.Tables[0];
```

**Lưu ý:** Chuỗi **conn** được lấy phụ thuộc vào tên của Project và tên của connectionString mà sinh viên đặt cho project của mình, nên **conn** bên dưới chỉ minh họa, sinh viên phải sửa lại cho đúng với project riêng của mình.

**StudentManaging2:** Tên của project

**StudentDBConnectionString:** Tên của connection string khai báo trong quá trình Data Source Configuration Wizard.

**Bước 3:** Gọi Form XemDiem từ Form chính của chương trình: Hàm xử lý này nằm trong Form chính (Parent Form) của ứng dụng, được gọi khi user chọn chức năng “Xem điểm” trên Form chính.



Hình 26: Chức năng Xem Điểm trên ToolStrip của Form chính

Sinh viên tự cài đặt chức năng này...

**Tương tự:** Tạo Form xem danh sách sinh viên thuộc khoa.

-----**Hết Lab 07**-----