



Chương 4: ADO.NET

Mục tiêu

- Xây dựng cơ sở dữ liệu trên Hệ quản trị CSDL SQL Server, viết các Stored Procedures trên CSDL đó.
- -Xây dựng ứng dụng đa form, tạo menu.
- -Sử dụng các đối tượng ADO.NET để kết nối CSDL
- -Thao tác CRUD trong .NET

Nội dung

➤ XÂY DỰNG ỨNG DỤNG ADO.NET

- Tạo kết nối sử dụng the Data Source Configuration Wizard
- Tạo kết nối đến CSDL MS Access và SQL Server
- Đối tượng DataSet
- Tạo và hiển thị dữ liệu từ DataSet
- Đọc dữ liệu vào các điều khiển cơ bản

➤ TRUY CẬP ADO.NET

- DataSet và DataAdapter
- Sử dụng GridView để cập nhật dữ liệu trực tiếp trên DataSet
- Cập nhật dữ liệu từ Text Box vào CSDL

Tổng quan ADO.NET

Data

Stored
into

	A	B	C	D	E	F
1	Name	Age	Location			
2	Julie	24	London			
3	Michael	32	Paris			
4	Rochelle	18	Seattle			
5	Rudolf	56	Brisbane			
6	Candida	43	Durban			

Database (Oracle, SQL Server)



Client

ADO.NET

Data access technology

Tổng quan ADO.NET

- Lịch sử phát triển công nghệ kết nối cơ sở dữ liệu
↳ ODBC → DAO → RDO → OLE DB → ADO → ADO.NET
- So sánh ADO v.s ADO.NET

ADO	ADO.NET
Online	Online và Offline
Một bảng	Nhiều bảng
Binary	XML

Tổng quan ADO.NET

➤ Tổng quan các lớp

↳ Library

- System.Data.dll
- System.Data.OracleClient.dll

↳ Namespace

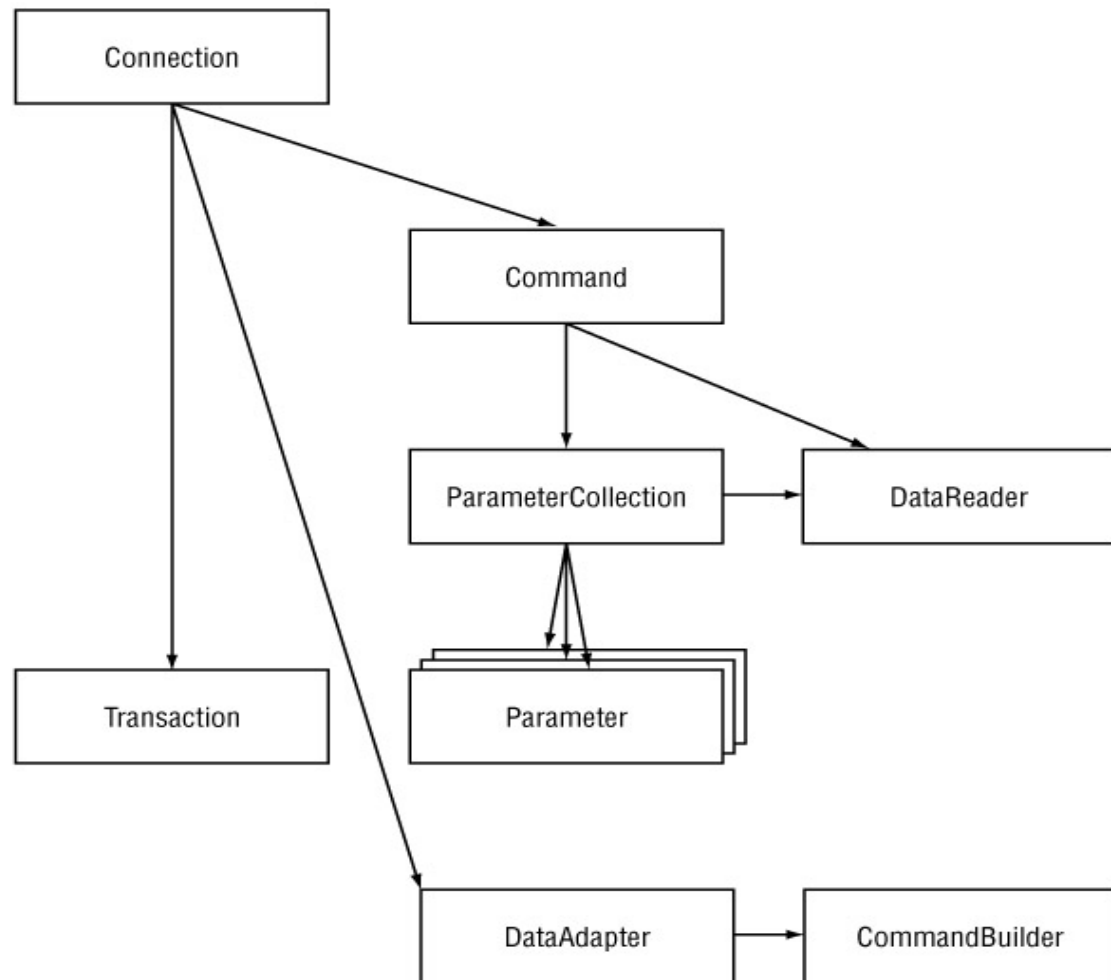
- System.Data
- System.Data.SqlTypes
- System.Data.Odbc
- System.Data.OleDb
- System.Data.SqlClient
- System.Data.OracleClient
- System.Data.SqlServerCe
- System.Data.Common

Tổng quan ADO.NET

- Mô hình đối tượng ADO.NET
 - ↳ Có 2 cách làm việc với ADO.NET
 - Tương tác trực tiếp với cơ sở dữ liệu: *managed provider classes*
 - Làm việc theo cách *disconnected*: *data set classes*

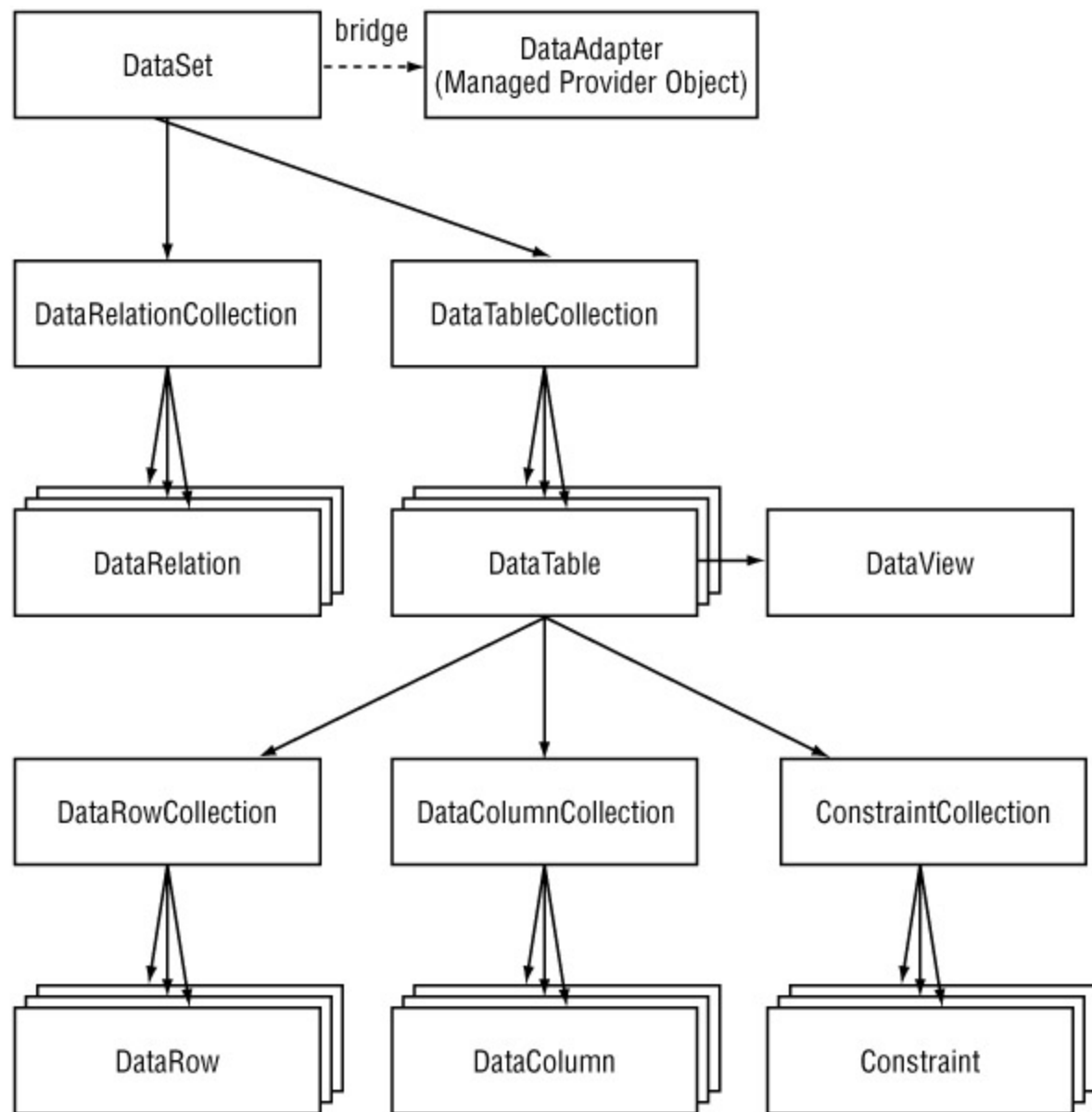
Tổng quan ADO.NET

➤ Managed Provider Classes



Tổng quan ADO.NET

➤ Generic Data Set Classes

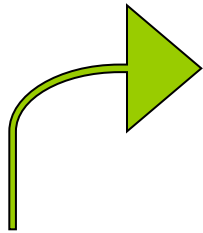


.NET Data Provider

➤ .NET Data Provider?

- ✚ Là một tập các lớp được thiết kế để làm việc với một data store cụ thể
- ✚ Mỗi .NET Data Provider cùng thực hiện một tập lớp cơ sở:
 - Connection, Command, Parameter, ParameterCollection, DataReader, DataAdapter, CommandBuilder và Transaction
- ✚ Tên các class chỉ khác nhau phần prefix

.NET Data Provider



SQL Server .NET Framework Data Provider
OLE DB .NET Framework Data Provider
ODBC .NET Framework Data Provider
Oracle .NET Framework Data Provider
SQL Server CE .NET Framework Data Provider

.NET DATA PROVIDERS



Connection

Establish connection with the database

Command

Retrieve and manipulate data in the database

.NET Data Provider

- Lớp *Connection*
 - ↳ SqlConnection, OleDbConnection, OdbcConnection, OracleConnection
- Lớp *Command*
 - ↳ SqlCommand, OleDbCommand, OdbcCommand, OracleCommand
- Lớp *Parameter*
 - ↳ SqlParameter, OleDbParameter, OdbcParameter, OracleParameter
- Lớp *ParameterCollection*
 - ↳ SqlParameterCollection, OleDbParameterCollection, OdbcParameterCollection, OracleParameterCollection

.NET Data Provider

- Lớp *DataReader*
 - ↳ SqlDataReader, OleDbDataReader, OdbcDataReader, OracleDataReader
- Lớp *DataAdapter*
 - ↳ SqlDataAdapter, OleDbDataAdapter, OdbcDataAdapter, OracleDataAdapter
- Lớp *CommandBuilder*
 - ↳ SqlCommandBuilder, OleDbCommandBuilder, OdbcCommandBuilder, OracleCommandBuilder
- Lớp *Transaction*
 - ↳ SqlTransaction, OleDbTransaction, OdbcTransaction, OracleTransaction



Lập trình .NET

Đối tượng Connection

- Thành phần đầu tiên để truy cập dữ liệu trong ADO.NET
- Phải chỉ ra một vài mảnh thông tin để kết nối
 - ↪ Server
 - ↪ Database
 - ↪ Xác nhận / Ủy quyền



Đối tượng Connection

➤ Server

➤ Dùng tham số “**server**” chỉ server của SQL Server

- Truyền vào chuỗi kết nối
- Dùng “.” hay “localhost” để kết nối cơ sở dữ liệu cục bộ

Server →

```
string connectionString = "Server=localhost;...";
```


Đối tượng Connection

➤ Database

↪ Dùng tham số “**database**” chỉ database của SQL Server

- Truyền vào chuỗi kết nối

Database → `string connectionString = "Database=pubs;...";`

Đối tượng Connection

➤ Xác nhận

↪ Có hai cách xác nhận kết nối client đến server

- Windows authentication
 - `Integrated Security = SSPI` trong chuỗi kết nối
- SQL Server authentication
 - `User ID/uid` và `Password/pwd` trong chuỗi kết nối

Thông tin
Windows →

```
string connectionString = "Data Source=  
DESKTOP-H68FGFI\SQLSERVER;Initial Catalog=QLSV;  
Integrated Security=True"
```

Thông tin
SQL →

```
string connectionString = "User ID=Hoangtn1204; Password=123;  
Data Source= DESKTOP-H68FGFI\SQLSERVER; Initial Catalog=QLSV";
```

Đối tượng Connection

➤ Kết nối

➤ Dùng **SqlConnection** để kết nối đến SQL Server

- Tạo object
- Thiết lập chuỗi kết nối
 - Truyền vào constructor
 - Thiết lập sau khi tạo đối tượng thông qua property "ConnectionString"
- Gọi phương thức **Open**



Đối tượng Connection

➤ Kết nối

parameters →

```
string cs = "server=.;Integrated Security=SSPI;database=pubs";
```

create →

```
SqlConnection connection = new SqlConnection(cs);
```

open →

```
connection.Open();  
...
```

Đối tượng Connection

➤ Ngắt kết nối

➤ Đóng SqlConnection khi đã hoàn tất

- Gọi `Close()` hay `Dispose()`
- Đặt lời gọi trong khối finally hay câu lệnh using

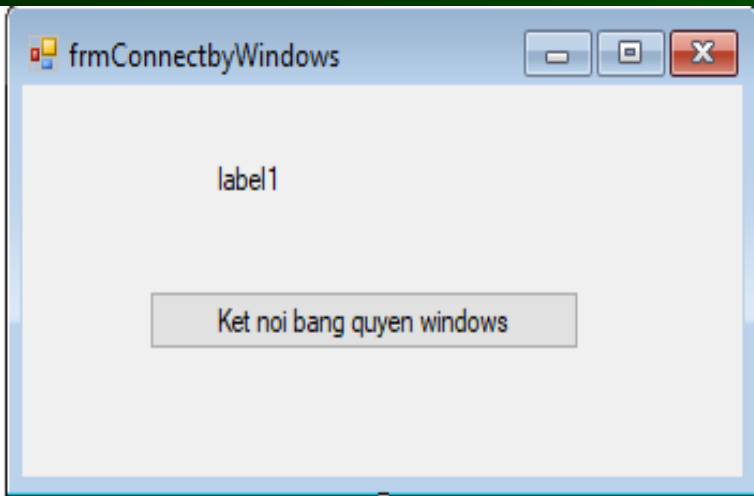
```
static void Main()
{
    SqlConnection connection = null;

    try
    {
        ...
        connection.Open();
        ...
    }
    finally
    {
        connection.Dispose();
    }
}
```

Mở →

Đóng →

Kết nối bằng quyền Windows



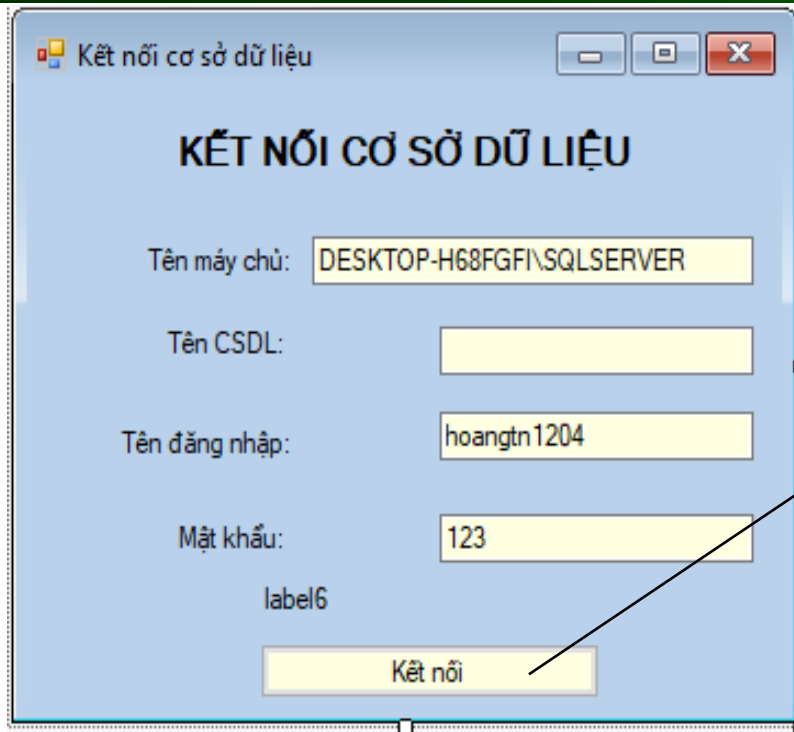
STT	Tên	Kiểu	Text	Ý nghĩa
1	frmConnectbyWindows	Form	frmConnectbyWindows	Kết nối CSDL
2	btnKetnoibangquyenwindows	Button	Ket noi bang quyen windows	Kích hoạt
3	lblTB	Label		Đưa thông báo

```

private void btnKetnoibangquyenwindows_Click(object sender, EventArgs e)
{
    SqlConnection Conn = new SqlConnection();
    Conn.ConnectionString =
        @"Data Source=DESKTOP-H68FGFI\SQLSERVER;Initial Catalog=QLSV;Integrated Security=True";
    try
    {
        Conn.Open();
        lblTB.Text = "Ket noi bang windows thanh cong!";
    }
    catch (Exception ex)
    {
        lblTB.Text = "Ket noi that bai"+ex.ToString();
    }
}

```

Kết nối dùng quyền truy cập SQL Server



```
private void btnKetNoi_Click(object sender, EventArgs e)
{
    DAL.clsConnect connect = new DAL.clsConnect(
        txtTenDangNhap.Text, txtMatKhai.Text,
        txtTenCSDL.Text, txtTenMayChu.Text);
    if (connect.Connect() != null)
    {
        lblThongbaoTrangThai.Text = "Kết nối thành công";
        //Gán giá trị cho thuộc tính vừa thêm
        StrConectionString = connect.GetConnectionString();
    }
    else lblThongbaoTrangThai.Text = "Kết nối thất bại";
}
```

- Tham khảo lớp clsConnect ở cuối file slide này hoặc truy cập: <https://www.tinhoccoban.net/2018/10/lap-trinh-windows-ket-noi-co-so-du-lieu.html>

➤ using System.Data.SqlClient;

```
private void btnLoginSQL_Click(object sender, EventArgs e)
{
    string sqlConnectionString = "";
    sqlConnectionString += "Data Source =" + txtServer.Text.Trim();
    sqlConnectionString += ";Initial Catalog=" + txtDatabase.Text.Trim();
    sqlConnectionString += ";User ID=" + txtUserID.Text.Trim();
    sqlConnectionString += ";Password=" + txtPassword.Text.Trim();
    SqlConnection connection = new SqlConnection(sqlConnectionString);
    try
    {
        connection.Open();
        lblThongBao.Text = "Kết nối thành công đến server " + txtServer.Text;
    }
    catch (SqlException ex)
    {
        lblThongBao.Text = "Kết nối thất bại đến " + txtServer.Text;
    }
    finally
    {
        connection.Dispose();
    }
}
```


Đối tượng Connection – Các tham số

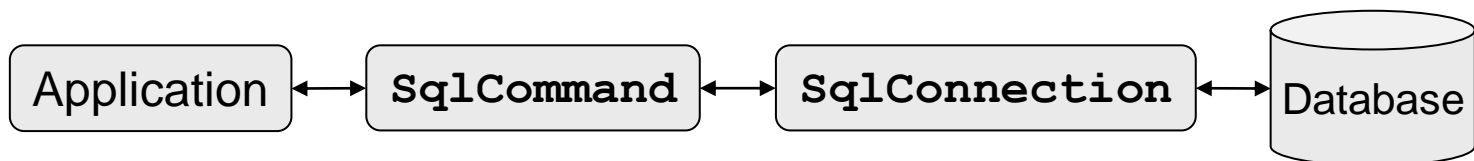
Property	Kiểu
ConnectionString	string
ConnectionTimeout	int
Database	string
DataSource	string
ServerVersion	string
State	ConnectionState

Đối tượng Connection – Phương thức

Method	Kiểu trả về
Open()	void
Close()	void
CreateCommand()	SqlCommand
BeginTransaction()	SqlTransaction
ChangeDatabase()	void

Đối tượng Command

- Dùng sau khi connection đã được thiết lập
- Dùng SqlCommand để thực thi câu lệnh SQL
 - Phải chỉ ra command text
 - Truyền cho constructor
 - Dùng thuộc tính **CommandText**
 - Phải chỉ ra connection
 - Truyền cho constructor
 - Dùng thuộc tính **Connection**
 - Dùng phương thức **CreateCommand** của connection



Đối tượng Command

- Dùng SqlCommand để thực thi câu lệnh SQL

Tạo
command



```
SqlConnection connection = new SqlConnection(...);  
...  
string text = "select * from tblSinhVien";  
  
SqlCommand command = new SqlCommand(text, connection);  
...
```

Show dữ liệu – sử dụng quyền Windows



Show dữ liệu – nút xem – sử dụng quyền windows

```
SqlConnection Conn = new SqlConnection();  
1 reference  
private void btnXem_Click(object sender, EventArgs e)  
{  
    Conn.ConnectionString =  
    @"Data Source=DESKTOP-H68FGFI\SQLSERVER;Initial Catalog=QLSV;Integrated Security=True";  
    try  
    {  
        //if (Conn.State== ConnectionState.Closed)  
        Conn.Open();  
        SqlCommand Comm = new SqlCommand();  
        Comm.CommandText = "Select * from tblSinhvien";  
        Comm.CommandType = CommandType.Text;  
        Comm.Connection = Conn;  
        Comm.ExecuteNonQuery();  
        DataTable dt = new DataTable();  
        SqlDataAdapter da = new SqlDataAdapter(Comm);  
        da.Fill(dt);  
        dgvTTSV.DataSource = dt;  
    }  
    catch (Exception ex)  
    {  
    }  
}
```

Đối tượng Command

➤ ExecuteReader

- ↪ Trả về tập dữ liệu lưu trong SqlDataReader
- ↪ Forward-only và Read-only
- ↪ Ném ngoại lệ Exception nếu command lỗi



```

string      text      = "select * from tblSinhVien";
SqlConnection connection = new SqlConnection(...);
SqlCommand  command    = new SqlCommand(text, connection);
...
SqlDataReader reader = command.ExecuteReader();
  
```

↑
Nhận kết quả dữ liệu trả về

↑
Thực thi command

DataReader

- Thuộc tính **HasRows** kiểm tra dữ liệu có không
- Hai cách truy cập các dòng của tập kết quả
 - Dùng foreach duyệt qua các dòng của đối tượng **IDataRecord**
 - Dùng while với **Read()**

DataReader

- Ba cách truy cập các cột
 - ↪ Truy cập thông qua tên
 - ↪ Truy cập thông qua chỉ mục `getXXX()`
 - ↪ Dùng `for` với `FieldCount`

loop →

Truy cập data trong dòng
dùng index →

Truy cập data trong dòng
dùng phương thức get →

```
static void Display(SqlDataReader reader)
{
    while (reader.Read())
    {
        string last  = (string)reader["au_lname"];
        string first = (string)reader[2];

        string zip   = reader.GetString(7);
        ...
    }
}
```

DataReader

- Gọi Close() khi sử dụng xong SqlDataReader
 - ↪ Đóng connection
 - ↪ Không thể truy cập dữ liệu

```
string      text      = "select * from tblSinhVien";  
SqlConnection connection = new SqlConnection(...);  
SqlCommand  command   = new SqlCommand(text, connection);  
...  
SqlDataReader reader = command.ExecuteReader();  
...  
reader.Close();
```



đóng reader
khi hoàn tất

Đối tượng Command

➤ ExecuteNonQuery

➤ Dùng ExecuteNonQuery khi không có dữ liệu trả về

- Trả về 1 số int báo số dòng bị ảnh hưởng



```

string text = "insert into authors "
    + "(au_id, au_lname, au_fname, contract) values "
    + "('111-11-1111', 'Adams', 'Mark', 1)";

SqlCommand command = new SqlCommand(text, connection);

int rowsAffected = command.ExecuteNonQuery();
...
  
```

Thực thi
command →

Đối tượng Command

➤ ExecuteScalar

➤ Dùng **ExecuteScalar** để thực thi select trả về 1 giá trị đơn (object)



Đối tượng Command

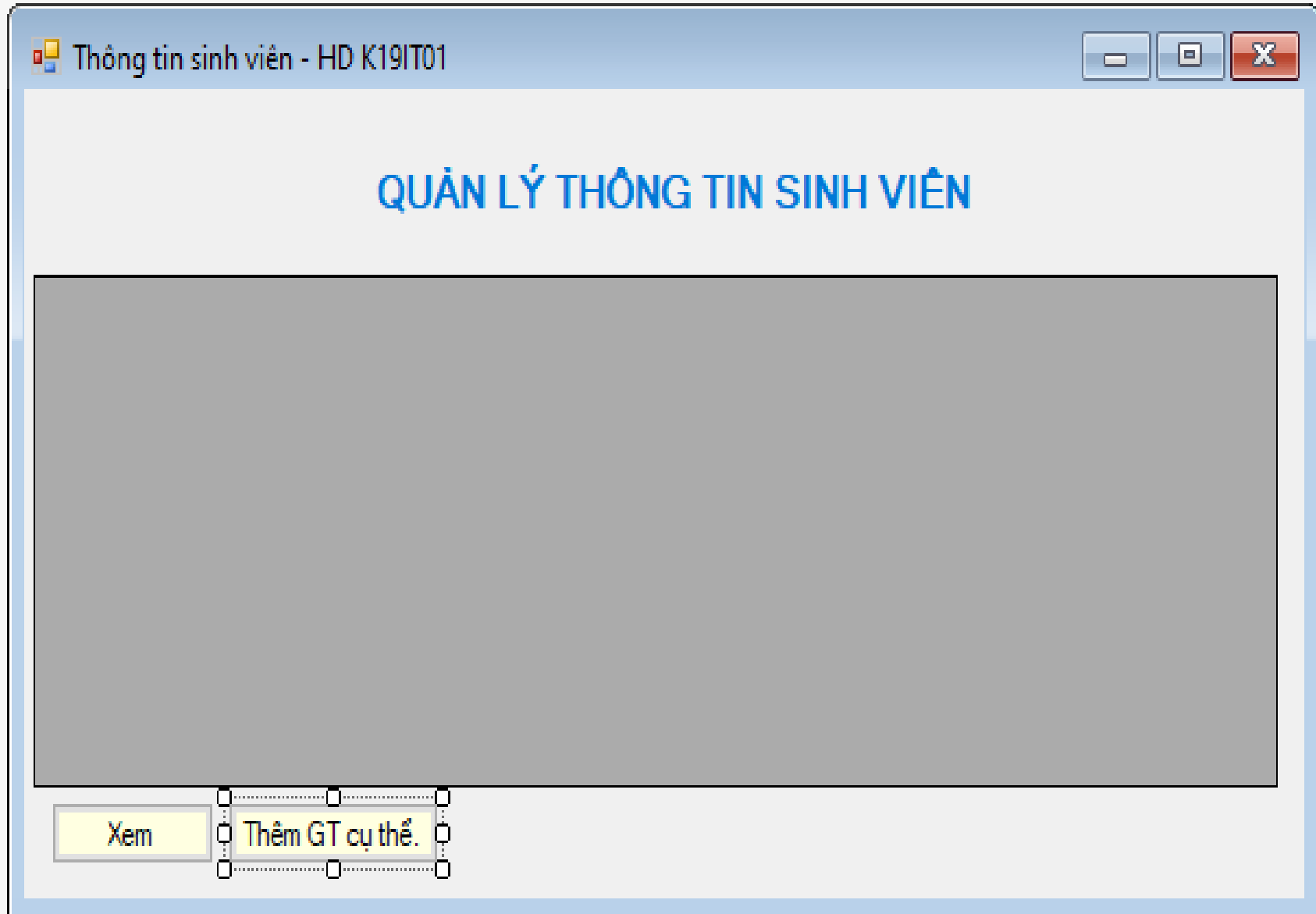
➤ Insert dữ liệu

↳ Dùng ExecuteNonQuery

```
mySqlCommand.CommandText =  
    "INSERT INTO Customers (CustomerID, CompanyName)  
    VALUES ( 'J2COM', 'Jason Price Corporation')";
```

```
int numberOfRows = mySqlCommand.ExecuteNonQuery();
```

Thêm giá trị cho bảng tblSinhVien



Thêm giá trị cho bảng tblSinhVien - btnThemcuthe

```
private void btnThemcuthe_Click(object sender, EventArgs e)
{
    Conn.ConnectionString =
@"Data Source=DESKTOP-H68FGFI\SQLSERVER;Initial Catalog=QLSV;Integrated Security=True";
    try
    {
        //if (Conn.State== ConnectionState.Closed)
        Conn.Open();
        SqlCommand Comm = new SqlCommand();
        Comm.CommandText = "Insert into tblSinhVien(MaSV,TenSV,SDT,QueQuan,GhiChu)" +
            "values('20011930055','Nguyễn Thị Nhật Vy','0933434356','Ninh Bình','Ko')";
        Comm.CommandType = CommandType.Text;
        Comm.Connection = Conn;
        Comm.ExecuteNonQuery();
        DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter(Comm);
        da.Fill(dt);
        dgvTTSV.DataSource = dt;
    }
    catch (Exception ex)
    {
    }
}
```

Thêm giá trị cho bảng tblSinhVien - btnThemcuthe

```
private void btnThemcuthe_Click(object sender, EventArgs e)
{
    Conn.ConnectionString =
@"Data Source=DESKTOP-H68FGFI\SQLSERVER;Initial Catalog=QLSV;Integrated Security=True";
    try
    {
        if (Conn.State == ConnectionState.Closed)
            Conn.Open();
        SqlCommand Comm = new SqlCommand();
        Comm.CommandText = "Insert into tblSinhVien(MaSV,TenSV,SDT,QueQuan,GhiChu)" +
            "values('20011930055',N'Nguyễn Thị Nhật Vy','0933434356',N'Ninh Bình','Ko')";
        Comm.CommandType = CommandType.Text;
        Comm.Connection = Conn;
        Comm.ExecuteNonQuery();
        DataTable dt = new DataTable();
        SqlDataAdapter da = new SqlDataAdapter(Comm);
        da.Fill(dt);
        dgvTTSV.DataSource = dt;
    }
    catch (Exception ex)
    {
    }
}
```


Thêm giá trị cho bảng tblSinhVien

Thông tin sinh viên - HD K19IT01

QUẢN LÝ THÔNG TIN SINH VIÊN

Mã SV: SDT: Ghi chú:

Tên SV: Quê quán:

Thêm giá trị cho bảng tblSinhVien

Quản lý thông tin sinh viên HD K19IT02

QUẢN LÝ THÔNG TIN

Mã SV: SDT: Ghi chú:

Tên SV: Quê quán:

Thêm giá trị cho bảng tblSinhVien btnAdd

```
public partial class frmSinhVien : Form
{
    /// <summary>
    /// Tạo thêm biến chứa chuỗi kết nối
    /// </summary>
    string StrConectionString;
    /// <summary>
    /// Phương thức khởi dựng mặc định với tham số là chuỗi kết nối
    /// </summary>
    /// <param name="_StrConnectionString">Chuỗi kết nối CSDL</param>
    1 reference
    public frmSinhVien(string _StrConnectionString)
    {
        InitializeComponent();
        StrConectionString = _StrConnectionString;
    }

    private void btnAdd_Click(object sender, EventArgs e)
    {
        dt.Clear();
        SqlConnection connection = new SqlConnection(StrConectionString);
        string text = "Insert into tblSinhvien(MaSV,TenSV,SDT,QueQuan,GhiChu)"
            + "values(@MaSV,@TenSV,@SDT,@QueQuan,@GhiChu)";
        if (connection.State == ConnectionState.Closed) connection.Open();
        SqlCommand command = new SqlCommand(text, connection);
        command.Parameters.AddWithValue("@MaSV", txtMaSV.Text.ToString());
        command.Parameters.AddWithValue("@TenSV", txtTenSV.Text.ToString());
        command.Parameters.AddWithValue("@SDT", txtSDT.Text.ToString());
        command.Parameters.AddWithValue("@QueQuan", txtQueQuan.Text.ToString());
        command.Parameters.AddWithValue("@GhiChu", txtGhiChu.Text.ToString());
        command.ExecuteNonQuery();
        command.Dispose();
        connection.Dispose();
        Display();
    }
}
```

Hàm Display()

```
void Display()
{
    dt.Clear();
    SqlConnection connection = new SqlConnection(StrConectionString);
    string text = "select * from tblSinhVien";
    SqlCommand command = new SqlCommand(text, connection);
    SqlDataAdapter da = new SqlDataAdapter(command);
    da.Fill(dt);
    dgvTTSV.DataSource = dt;
    DataBinding(dt);
}

1 reference
void DataBinding(DataTable dt)
{
    txtMaSV.DataBindings.Add("Text", dt, "MaSV");
    txtTenSV.DataBindings.Add("Text", dt, "TenSV");
    txtSDT.DataBindings.Add("Text", dt, "QueQuan");
    txtGhiChu.DataBindings.Add("Text", dt, "Ghichu");
}
```

Cập nhật dữ liệu

Quản lý thông tin sinh viên HD K19IT02

QUẢN LÝ THÔNG TIN

Mã SV: SDT: Ghi chú:

Tên SV: Quê quán:

Cập nhật dữ liệu

```
private void btnUpdate_Click(object sender, EventArgs e)
{
    //dt.Clear();
    SqlConnection connection = new SqlConnection(StrConectionString);
    string text = "Update tblSinhVien " +
        "set TenSV='" + txtTenSV.Text + "', SDT='" + txtSDT.Text + "'," +
        "QueQuan='" + txtQueQuan.Text + "', GhiChu='" + txtGhiChu.Text +
        "' where MaSV='" + txtMaSV.Text + "'";
    if (connection.State == ConnectionState.Closed) connection.Open();
    SqlCommand command = new SqlCommand(text, connection);
    command.ExecuteNonQuery();
    command.Dispose();
    connection.Dispose();
    capnhat
}
```

Đối tượng Command

- Update dữ liệu
 - ↪ Dùng ExecuteNonQuery

```
mySqlCommand.CommandText =  
    "UPDATE Customers " +  
    "SET CompanyName = 'New Company' " +  
    "WHERE CustomerID = 'J2COM'";  
  
numberOfRows = mySqlCommand.ExecuteNonQuery();
```

Đối tượng Command

➤ Delete dữ liệu

↪ Dùng ExecuteNonQuery

```
mySqlCommand.CommandText =  
    "DELETE FROM Customers " +  
    "WHERE CustomerID = 'J2COM'";
```

```
numberOfRows = mySqlCommand.ExecuteNonQuery();
```


Đối tượng Command

- Câu lệnh DDL: Create table, Alter table, Drop table, ...

↳ Dùng ExecuteNonQuery

```
mySqlCommand.CommandText =  
    "ALTER TABLE MyPersons " +  
    "ADD EmployerID nchar(5) CONSTRAINT FK_Persons_Customers " +  
    "REFERENCES Customers(CustomerID)";
```

```
result = mySqlCommand.ExecuteNonQuery();
```

Đối tượng Command

- Câu lệnh DDL: Create table, Alter table, Drop table, ...
 - ↪ Dùng ExecuteNonQuery

```
mySqlCommand.CommandText = "DROP TABLE MyPersons";
```

```
result = mySqlCommand.ExecuteNonQuery();
```

Đối tượng Command

Property	Kiểu
Connection	string
CommandText	string
CommandType	CommandType
CommandTimeout	String
Parameters	SqlParameterCollection
Transaction	SqlTransaction

Đối tượng Command

Method	Kiểu trả về
ExecuteReader()	SqlDataReader
ExecuteNonQuery()	int
ExecuteScalar()	object

Đối tượng Command

➤ Stored procedure

↳ Dùng ExecuteReader

- Thiết lập property **CommandType** **= CommandType.StoredProcedure**
- Thiết lập property **CommandText** bằng tên procedure
- Truyền tham số vào **parameters**
- Gọi **ExecuteXXX()**

Đối tượng Command

➤ Stored procedure

Tên procedure



```
SqlCommand command = new SqlCommand("byroyalty", connection);  
  
command.CommandType = CommandType.StoredProcedure;  
  
command.Parameters.Add("@percentage", SqlDbType.Int);  
command.Parameters["@percentage"].Value = 50;  
  
reader = command.ExecuteReader();  
...
```

command type →

parameters →

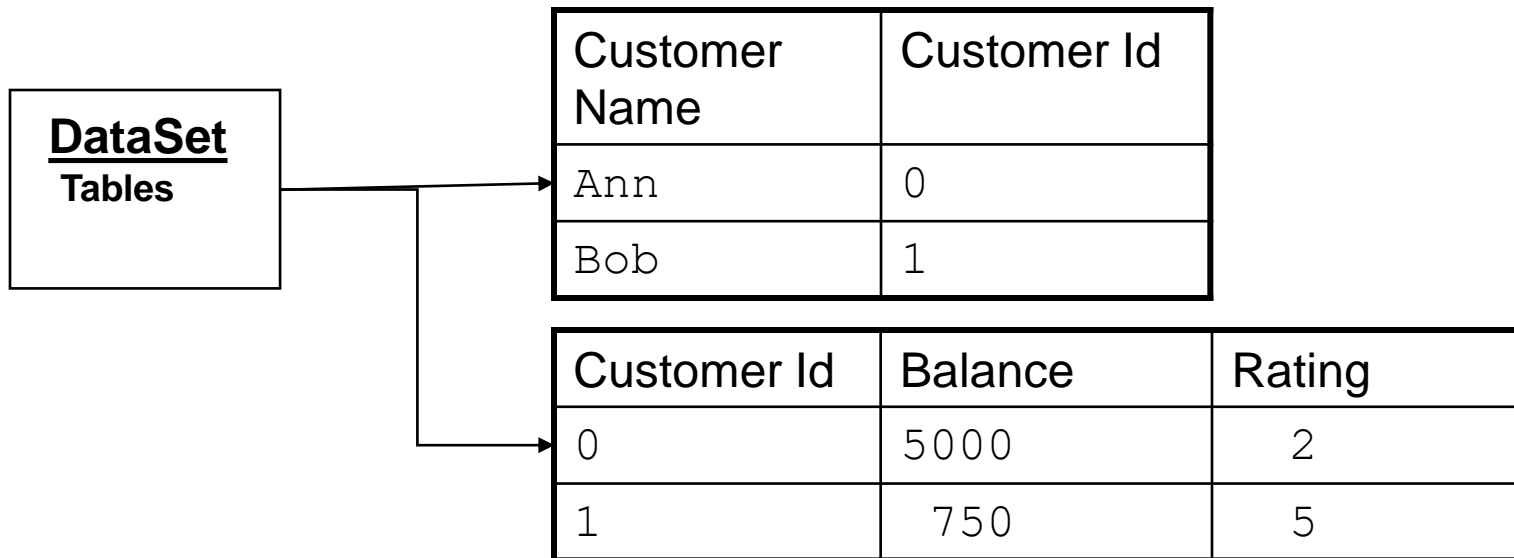
execute →

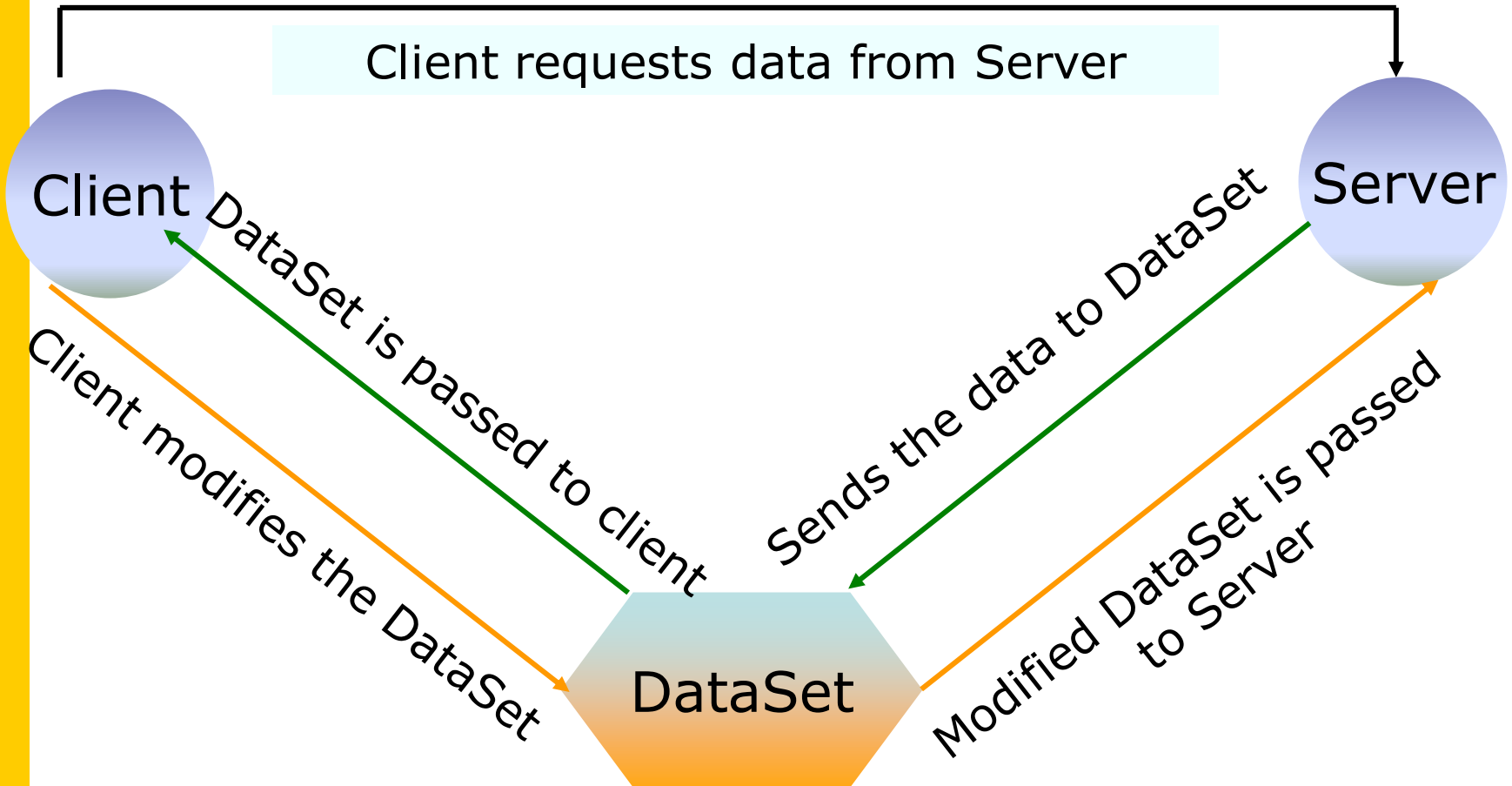


Lập trình .NET

Dữ liệu offline

- Dữ liệu offline (Disconnected data) là gì?
 - ↳ Là mô hình dùng bộ nhớ tạm để lưu dữ liệu
 - Các table, quan hệ, dòng, cột, ...
 - Không kết nối và độc lập với data source





DataSet

- Lớp DataSet mô hình các dữ liệu offline
 - ↪ DataSet có property dạng collection gồm tập các đối tượng DataTable

Bảng trong
data set



```
public class DataSet ...  
{  
    public DataTableCollection Tables { get; }  
    ...  
}
```

DataTable

- DataTable là mô hình trong bộ nhớ của table
 - ↪ Gồm: Dòng, cột, ...

rows và columns
hiện tại trong bảng →

```
public class DataTable ...  
{  
    public DataRowCollection Rows { get; }  
    public DataColumnCollection Columns { get; }  
    ...  
}
```

DataRow

- DataRow là mô hình bộ nhớ của dòng bên trong DataTable
 - ↪ Có một số cách truy cập column
 - ↪ Các dòng được sinh bởi table, không được tạo trực tiếp

Truy cập dữ liệu cột
bằng tên hay số



Tất cả cột
trong dòng



```
public class DataRow ...  
{  
    public object this[string] { get; set; }  
    public object this[int]    { get; set; }  
  
    public object[] ItemArray { get; set; }  
    ...  
}
```

DataColumn

- DataColumn mô hình cột của DataTable
 - ↪ Chỉ rõ tên và kiểu dữ liệu khi tạo

```
public class DataColumn ...  
{  
    public DataColumn(string name, Type type);  
    ...  
}
```

↑
Tên cột

↑
Đối tượng **Type**
cho kiểu của dữ liệu cột

Tạo DataSet

- Tạo DataSet: Có 2 cách tạo
 - ↪ `DataSet myDataSet = new DataSet();`
 - ↪ `DataSet myDataSet = new DataSet("myDataSet");`

Tạo DataSet

- Điền dữ liệu vào DataSet: Có 2 cách
 - ↪ Fill từ data source có trước
 - ↪ Định nghĩa cấu trúc “bằng tay” sau đó fill dữ liệu

Tạo DataSet

- Fill từ source như Database
 - ↪ Dùng DataAdapter và phương thức Fill

Tạo adapter →

Dùng adapter để
Fill vào DataSet →

```
string text = "select * from authors";  
SqlConnection conn;  
...  
SqlDataAdapter adapter = new SqlDataAdapter(text, conn);  
  
DataSet authors = new DataSet();  
  
adapter.Fill(authors);
```


Truy cập dữ liệu trong DataSet

- DataSet có 3 cấp dữ liệu bên trong
 - ↪ Tập các table
 - ↪ Các dòng trong mỗi table
 - ↪ Các giá trị cột trong mỗi dòng

Truy cập DataSet →

```
foreach (DataTable table in myDataSet.Tables)
{
    foreach (DataRow row in table.Rows)
    {
        foreach (object data in row.ItemArray)
        {
            // process column value
        }
    }
}
```

Tự tạo DataTable

- Có thể tự tạo DataTable
 - Định nghĩa cấu trúc bảng
 - Tạo các dòng, thêm các dòng vào bảng, fill dữ liệu

Tạo table → `DataTable customers = new DataTable("Customers");`

Định nghĩa các cột → `customers.Columns.Add("Name", typeof(string));`
`customers.Columns.Add("Id", typeof(Int32));`

Tạo row → `DataRow row = customers.NewRow();`

Fill dữ liệu → `row[0] = "Ann";`

`row[1] = 0;`

Thêm dòng vào table → `customers.Rows.Add(row);`

`...`

Tự tạo DataSet

- Có thể tự tạo DataSet
 - ↪ Tạo đối tượng DataSet
 - ↪ Tạo các table và thêm vào DataSet

Tạo table



```
DataTable customers = new DataTable("Customers");  
...
```

Tạo DataSet



```
DataSet data = new DataSet();
```

Thêm table vào DataSet



```
data.Tables.Add(customers);  
...
```

Cập nhật cơ sở dữ liệu thông qua DataAdapter

- Có thể cập nhật cơ sở dữ liệu sau khi thay đổi DataSet
 - Dùng **SqlCommandBuilder** để tạo các câu SQL cần thiết
 - Dùng phương thức **Update** của **DataAdapter** để gọi các thay đổi

```
string text = "select * from authors";

SqlDataAdapter adapter = new SqlDataAdapter(text, conn);

DataSet ds = new DataSet();

adapter.Fill(ds, "authors");

ds.Tables[0].Rows[4][2] = "Bob";

SqlCommandBuilder b = new SqlCommandBuilder(adapter);

adapter.Update(ds, "authors");
```

Chỉ bảng nguồn
khi fill →

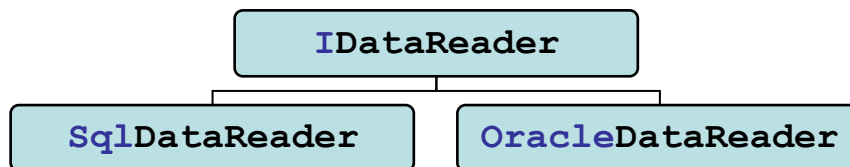
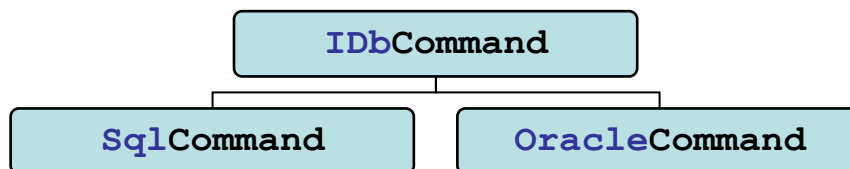
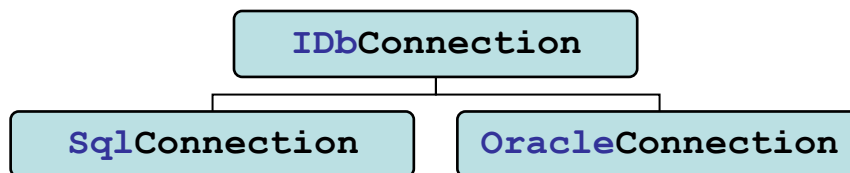
Chỉnh sửa DataSet →

Xây dựng SQL →

Cập nhật database →

Viết code độc lập cơ sở dữ liệu

- Mỗi cơ sở dữ liệu có một tập các lớp
 - ↪ Thực thi cùng interface



Dùng tham chiếu interface

- Dùng tham chiếu interface giúp viết code tổng quát hơn

```
string text = "select * from authors";  
  
IDbConnection connection = new SqlConnection(...);  
...  
IDataReader reader = command.ExecuteReader();
```

↑
Các tham chiếu interface
độc lập cơ sở dữ liệu

↑
Các lớp
cơ sở dữ liệu cụ thể

DataView

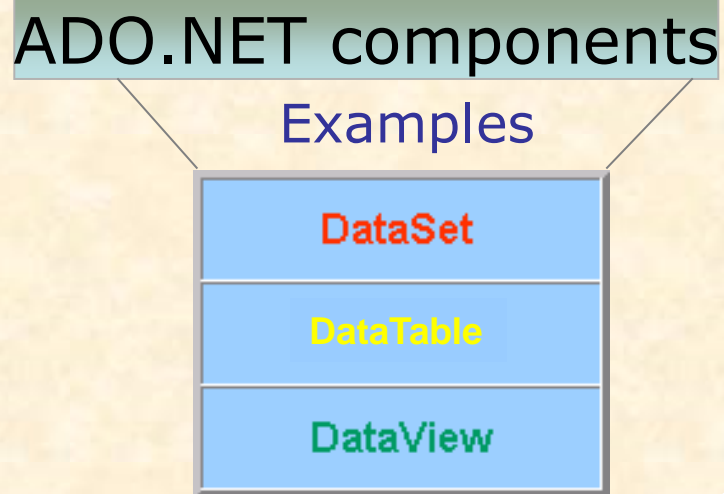
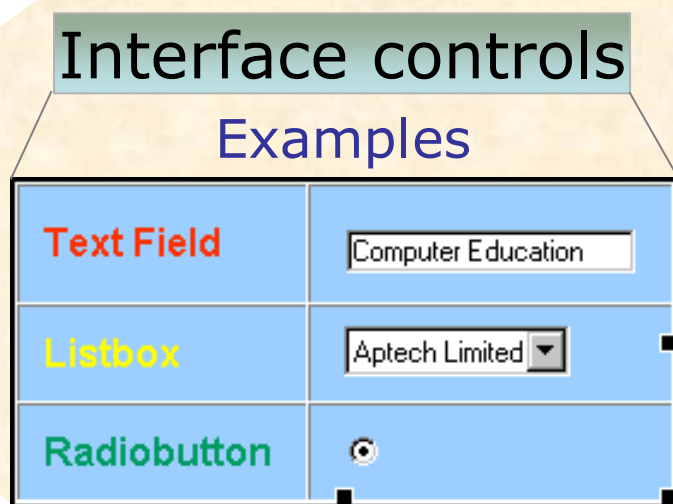
- Tầng lưu trữ dữ liệu trong DataTable
- Cung cấp hiển thị DataTable for sắp xếp, lọc và tìm kiếm
- Có thể sử dụng subset của dữ liệu lưu trữ in DataTable
- Có 2 cách sử dụng dữ liệu trong DataTable,



Lập trình .NET

Data Binding

- Data binding là gì?
 - ↪ Cách thức liên kết việc đọc/ghi giữa control và dữ liệu (database, array, collection)



Data Binding tích hợp 2 thành phần

Data Binding

➤ Các loại binding

- Data binding đơn giản: Kết nối **Một giá trị đơn** giữa dataset với một property bất kỳ của control
- Data binding phức tạp: Kết nối **Một tập giá trị** giữa dataset với control thể hiện nhiều giá trị hơn 1 thời điểm

Data Binding

➤ Data binding đơn giản:

↳ Data binding với property Text của **TextBox**

- `txtTextBox.DataBindings.Add("Text",ds,"Bang.TênCột");`
- `txtTextBox.DataBindings.Add("Text",ds.Tables[0],"TênCột");`

➤ Data binding phức tạp

↳ Data binding với **DataGridView**

- `grd.DataSource = ds; grd.DataMember = "Bảng";`
- `grd.DataSource = ds.Tables[0];`

Data Binding

- Data binding phức tạp
 - ↪ Data binding với **Combobox/ListBox**
 - ↪ `cb.DataSource = ds;`
`cb.DisplayMember = "Bảng.Cột1";`
`cb.ValueMember = "Bảng.Cột2";`
 - ↪ `cb.DataSource = ds.Tables[0];`
`cb.DisplayMember = "Cột1";`
`cb.ValueMember = "Cột2";`

Đồng bộ dữ liệu giữa các control

- Lớp CurrencyManager giúp việc đồng bộ dữ liệu giữa các control
 - ↳ Property quan trọng:
 - Position
 - Count
 - ↳ Phương thức
 - Position++
 - Position--

```
CurrencyManager cm = (CurrencyManager)this.BindingContext[ds,"Bảng"];  
if (cm.Position < cm.Count - 1)  
{  
    cm.Position++;  
}
```

Đồng bộ dữ liệu giữa các control

➤ Những chức năng cơ bản

↪ Update

- `cm.EndCurrentEdit();`
`adapter.Update(ds.Bảng);`

↪ Next

- `cm.Position++;`

↪ Back

- `cm.Position--;`

Đồng bộ dữ liệu giữa các control

➤ Những chức năng cơ bản

↳ First

- `cm.Position = 0;`

↳ Last

- `vt = this.BindingContext[ds, "Bảng"].Count - 1`
- `cm.Position = vt;`

↳ Delete

- `vt = this.BindingContext[ds, "Bảng"].Position`
- `cm.RemoveAt(vt);`

↳ Add new

- `cm.AddNew();`

Đồng bộ dữ liệu giữa các control

➤ Chú ý quan trọng:

➤ Data source của các control phải đồng nhất với nhau

- Hoặc cùng dataset
- Hoặc cùng datatable

➤ Tốt nhất: nên dùng datatable

Quan hệ dữ liệu dạng Master – detail

- Master – detail: dạng quan hệ 1-n trong cơ sở dữ liệu

De tai - sinh vien

Đề tài

	madt	tendt
▶	dt01	quản lý bán hàng...
	dt02	nhận dạng chữ vi...
*		

Sinh viên

	masv	hoten	namsinh
▶	sv01	trần văn tuần	1985
	sv02	lê thanh thanh	1986
	sv04	lê minh hằng	1986
*			

De tai - sinh vien

Đề tài

	madt	tendt
	dt01	quản lý bán hàng...
▶	dt02	nhận dạng chữ vi...
*		

Sinh viên

	masv	hoten	namsinh
▶	sv03	nguyễn xuân minh	1988
*			

Quan hệ dữ liệu dạng Master – detail

Bước 1: fill dữ liệu

- Fill 2 bảng vào dataset (bao gồm khóa chính và khóa ngoại)

Bước 2: tạo quan hệ

- `DataColumn colMaster = ds.Tables["tenMaster"].Columns["cộtPK"];`
- `DataColumn colDetail = ds.Tables["tenDetail"].Columns["cộtFK"];`
- `DataRelation relation = new DataRelation("tenquanhe", colMaster, colDetail);`
- `ds.Relations.Add(relation);`

Quan hệ dữ liệu dạng Master – detail

➤ Bước 3: (cách 1)

➤ dataGridViewMaster.DataSource = ds;

➤ dataGridViewMaster.DataMember =
"tenMaster";

➤ dataGridViewDetail.DataSource = ds;

➤ dataGridViewDetail.DataMember =
"tenMaster.tenquanhe";

Quan hệ dữ liệu dạng Master – detail

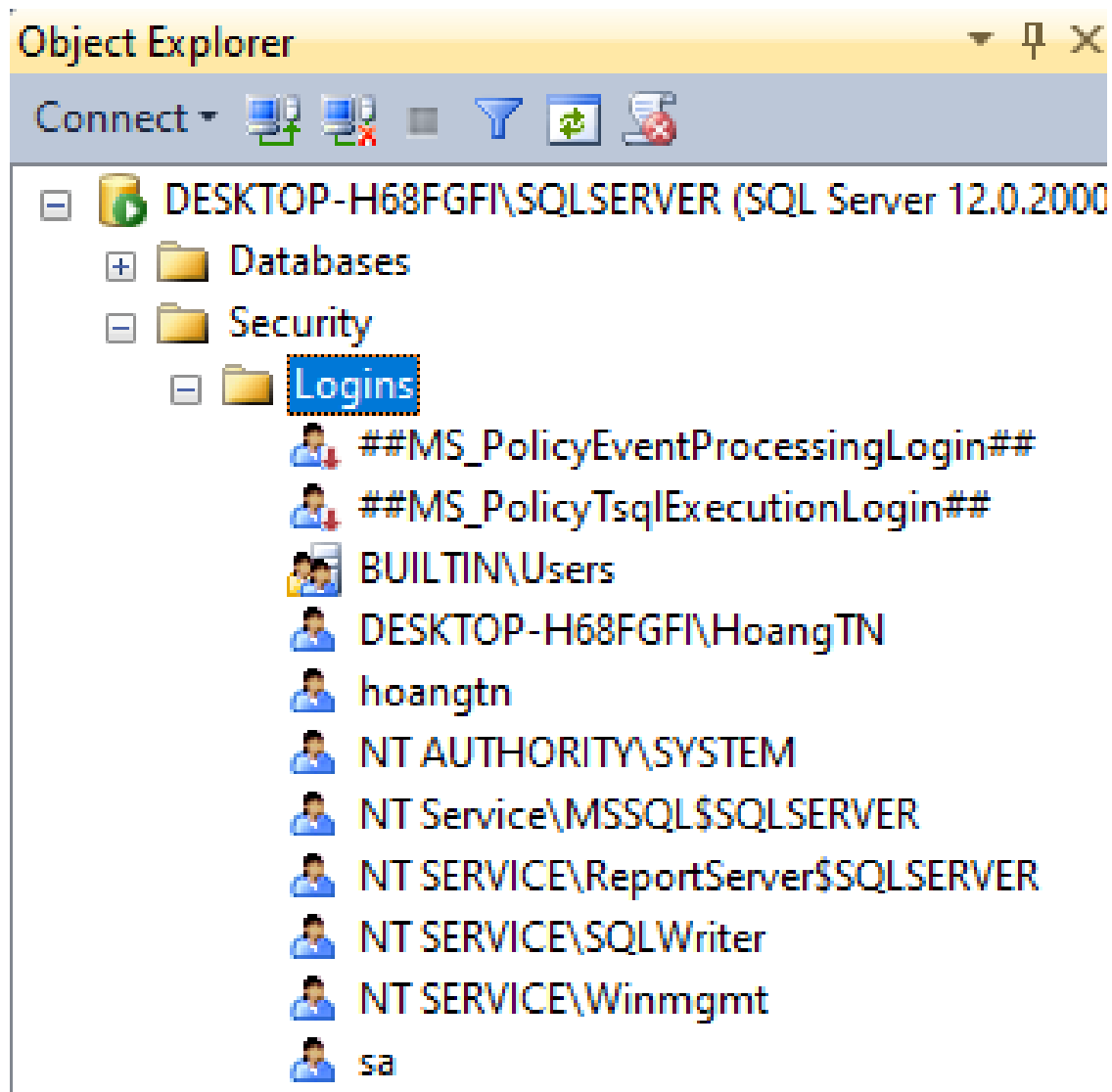
➤ Bước 3: (cách 2)

↪ `dataGridViewMaster.DataSource =
ds.Tables["tenMaster"];`

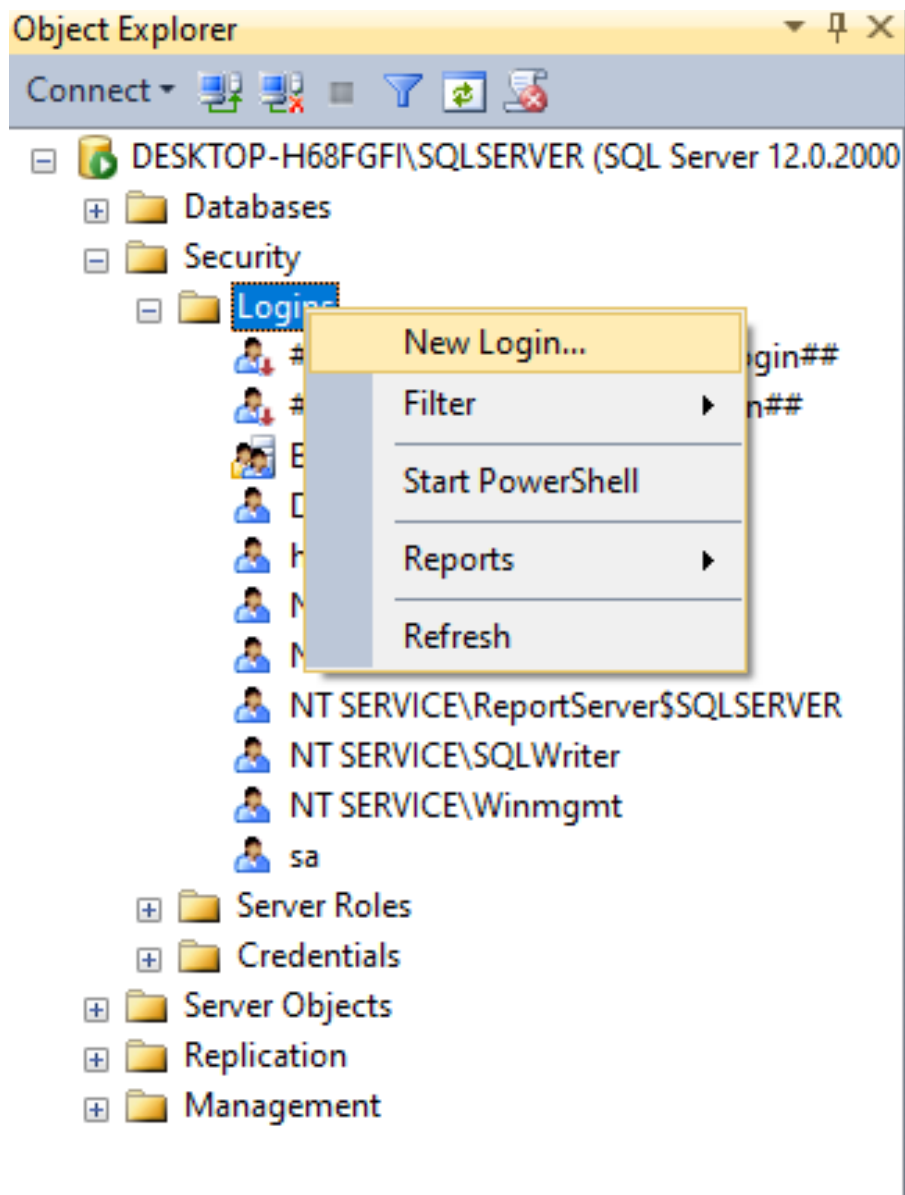
↪ `dataGridViewDetail.DataSource =
ds.Tables["tenMaster"];`

↪ `dataGridViewDetail.DataMember =
"tenquanhe";`

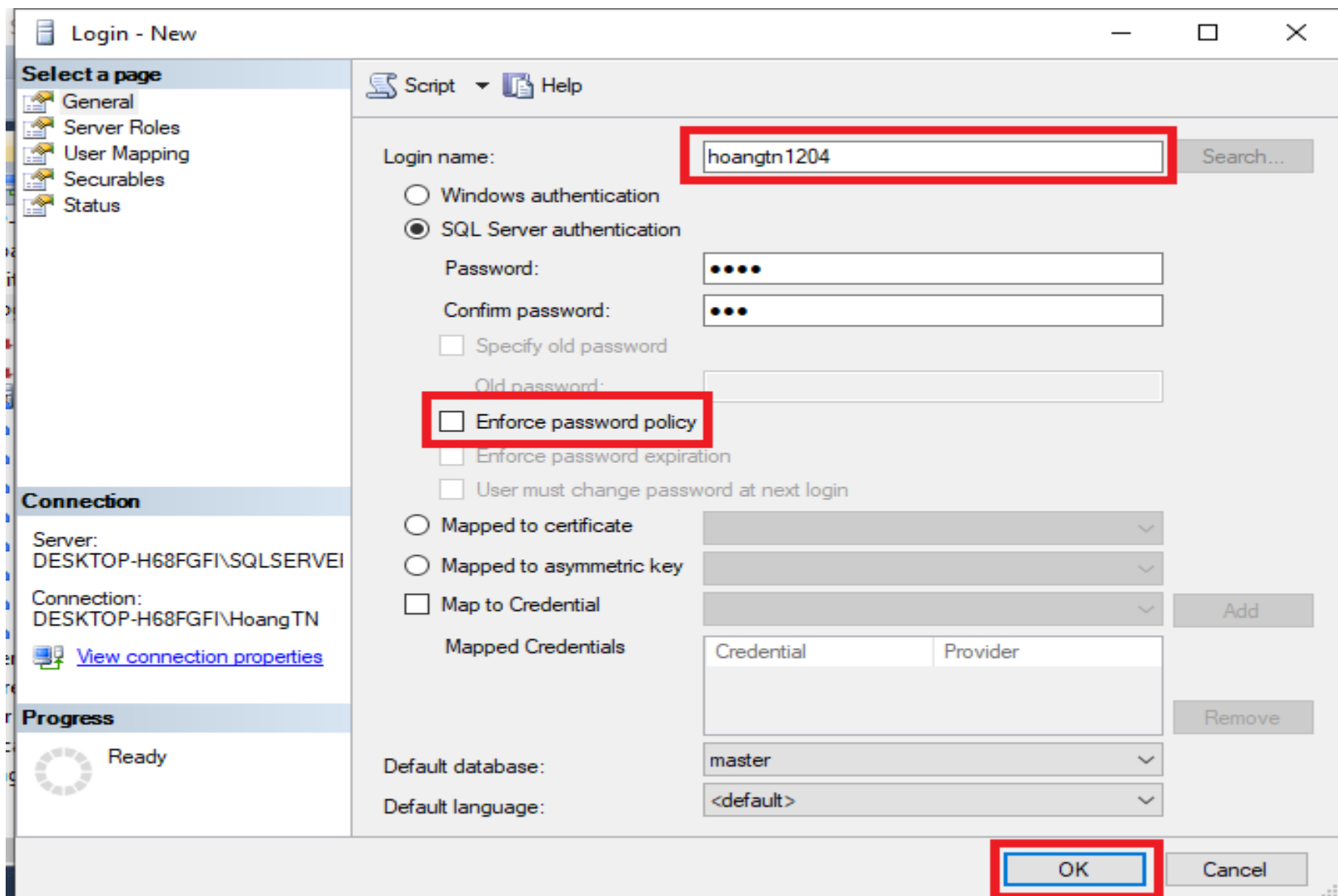
Hướng dẫn tạo csdl trên sql server



Hướng dẫn tạo csdl trên sql server



Hướng dẫn tạo csdl trên sql server



Login - New

Select a page

- General
- Server Roles
- User Mapping
- Securables
- Status

Script Help

Login name: Search...

☐ Windows authentication

☒ SQL Server authentication

Password:

Confirm password:

☐ Specify old password

Old password:

☐ Enforce password policy

☐ Enforce password expiration

☐ User must change password at next login

☐ Mapped to certificate

☐ Mapped to asymmetric key

☐ Map to Credential Add

Mapped Credentials

Credential	Provider

Remove

Default database:

Default language:

OK Cancel

Connection

Server: DESKTOP-H68FGFI\SQLSERVEI

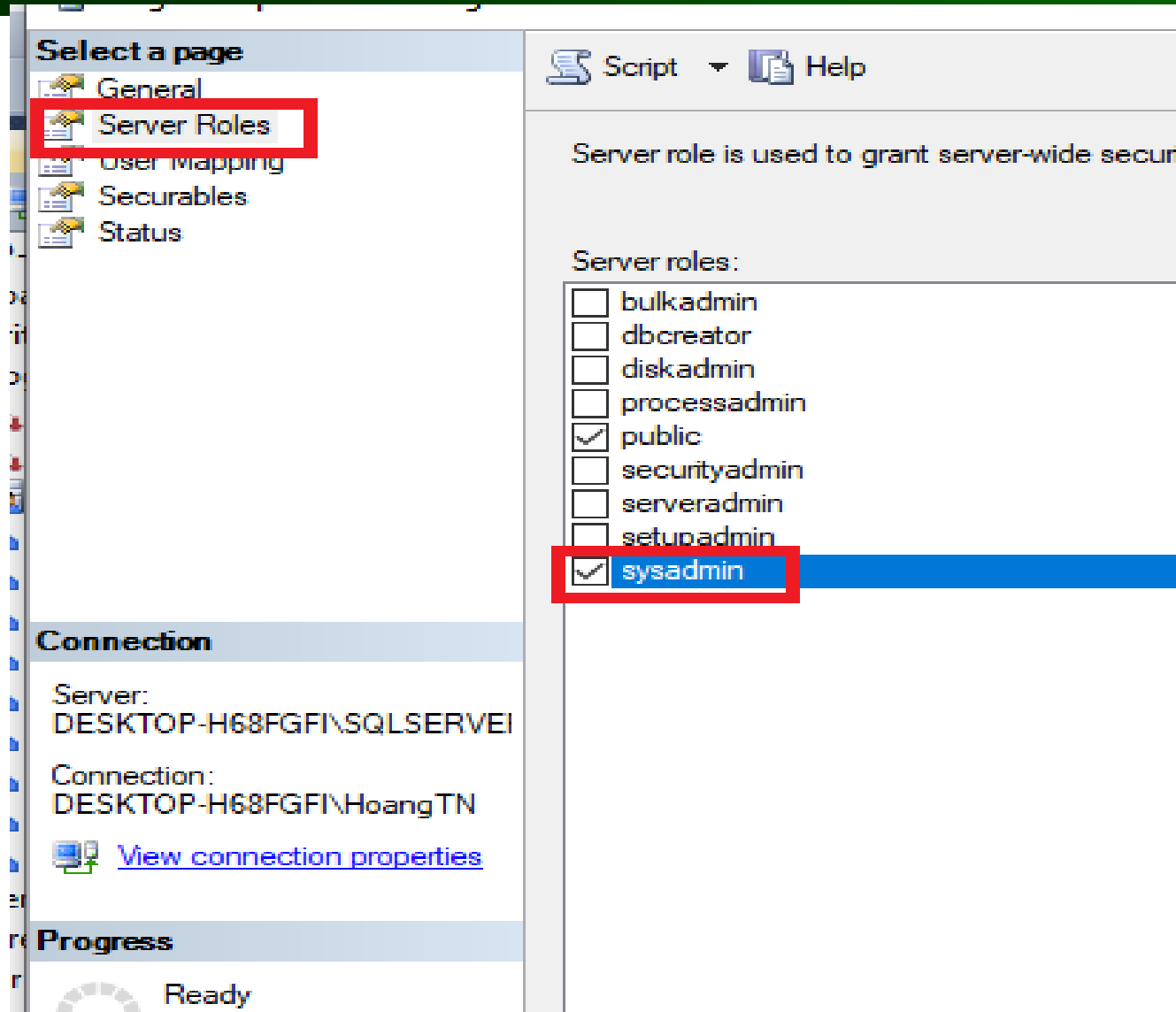
Connection: DESKTOP-H68FGFI\HoangTN

[View connection properties](#)

Progress

Ready

Hướng dẫn tạo csdl trên sql server



Hướng dẫn tạo csdl trên sql server

Connect to Server

Microsoft SQL Server 2014

Server type: Database Engine

Server name: DESKTOP-H68FGFI\SQLSERVER

Authentication: SQL Server Authentication

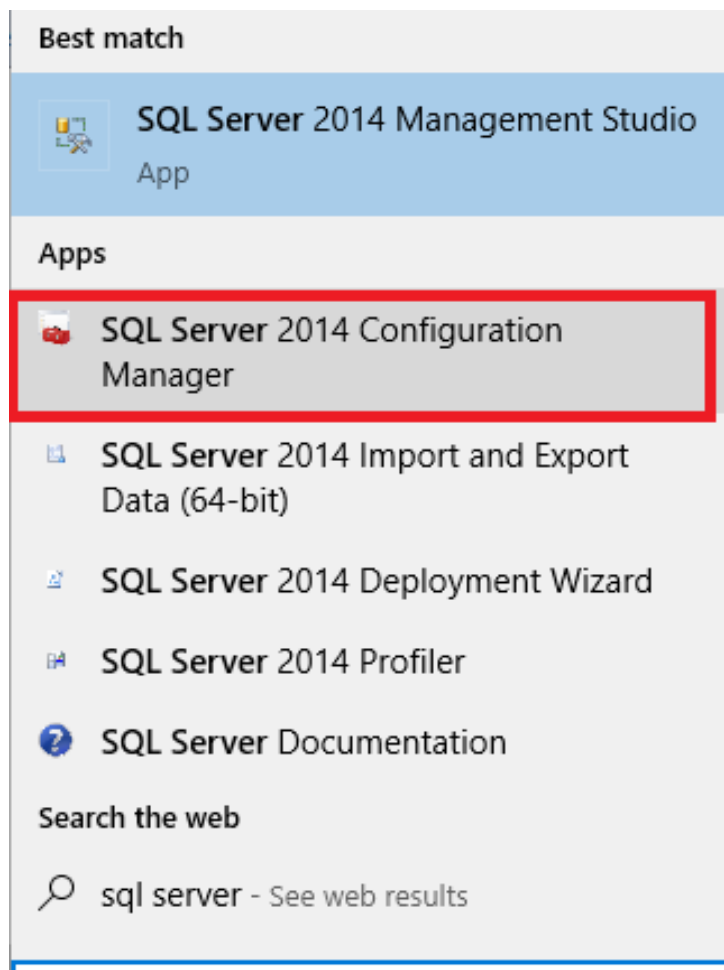
Login: hoangtn1204

Password: ****

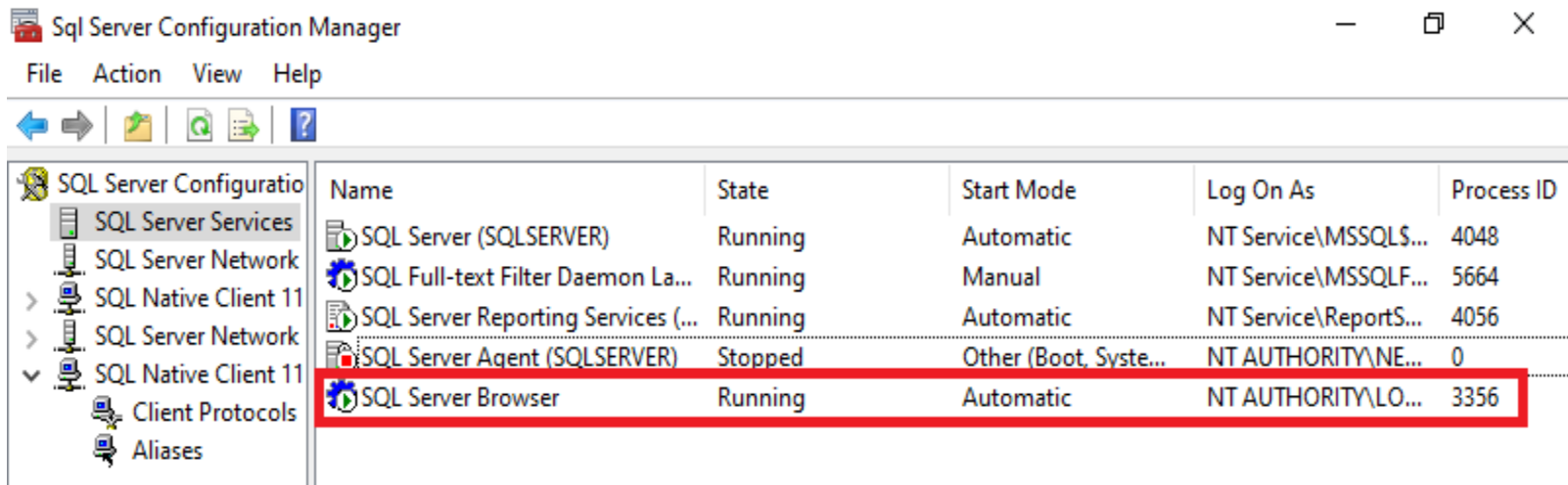
☐ Remember password

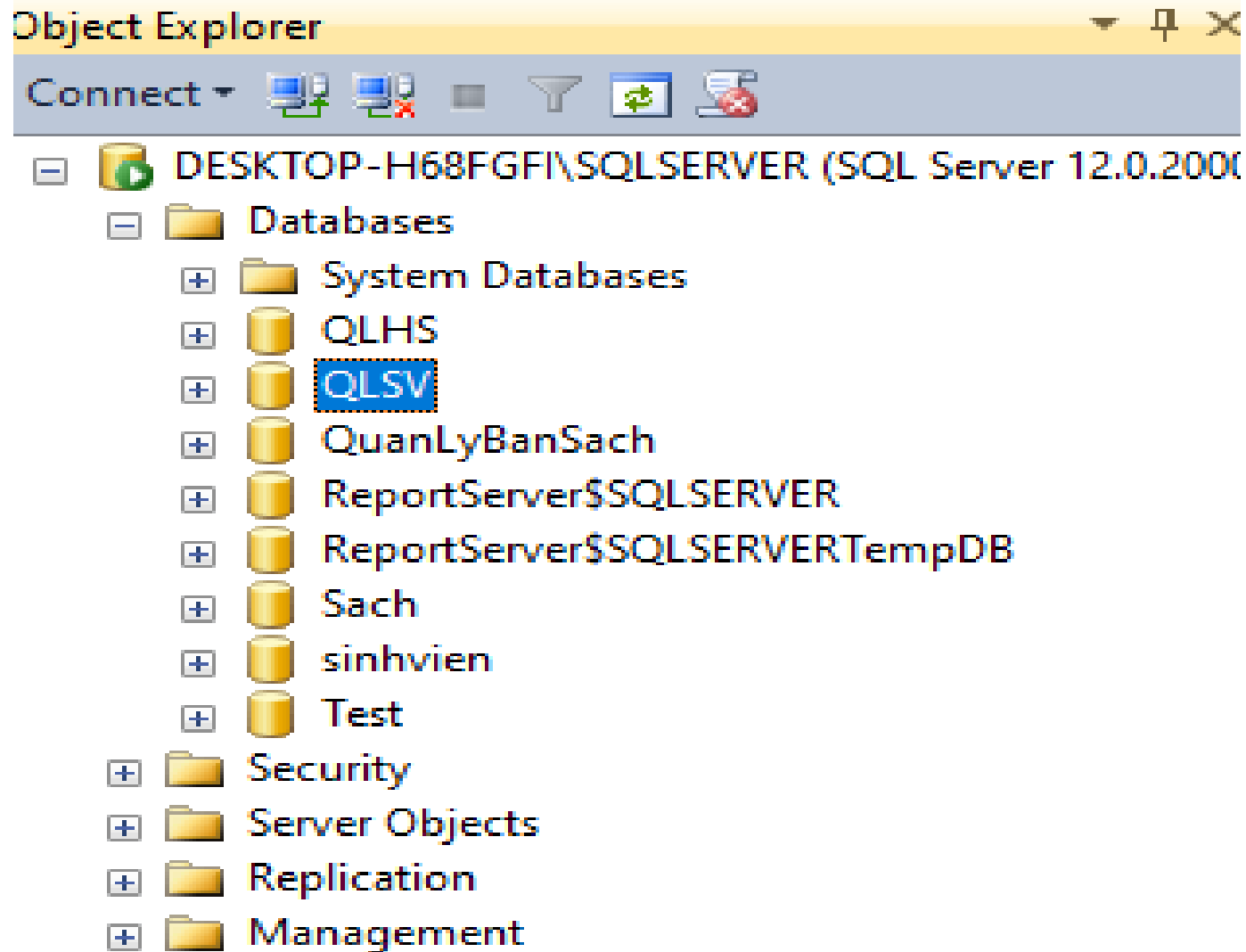
Connect Cancel Help Options >>

Fix không cho kết nối từ xa



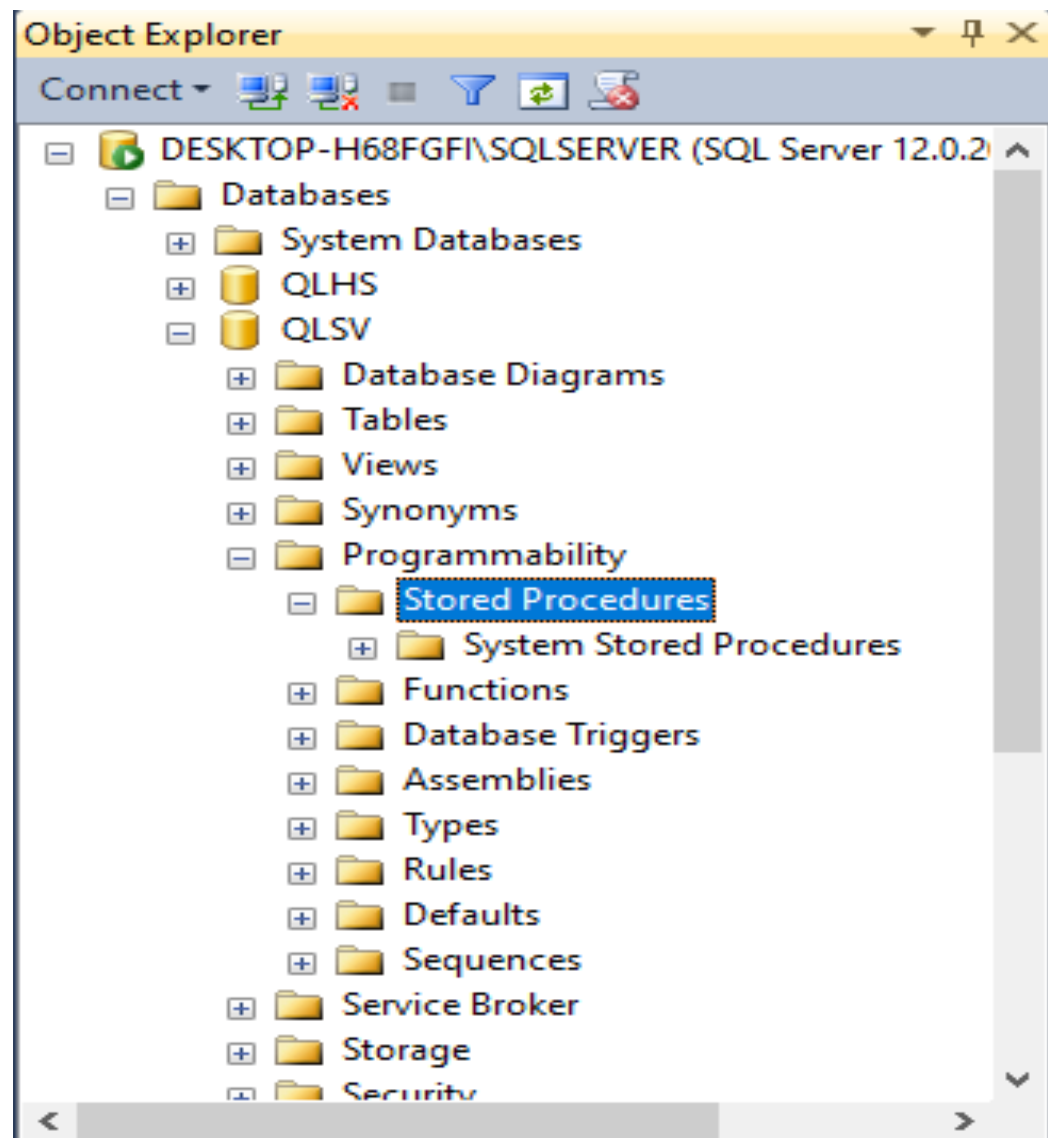
Fix không cho kết nối từ xa





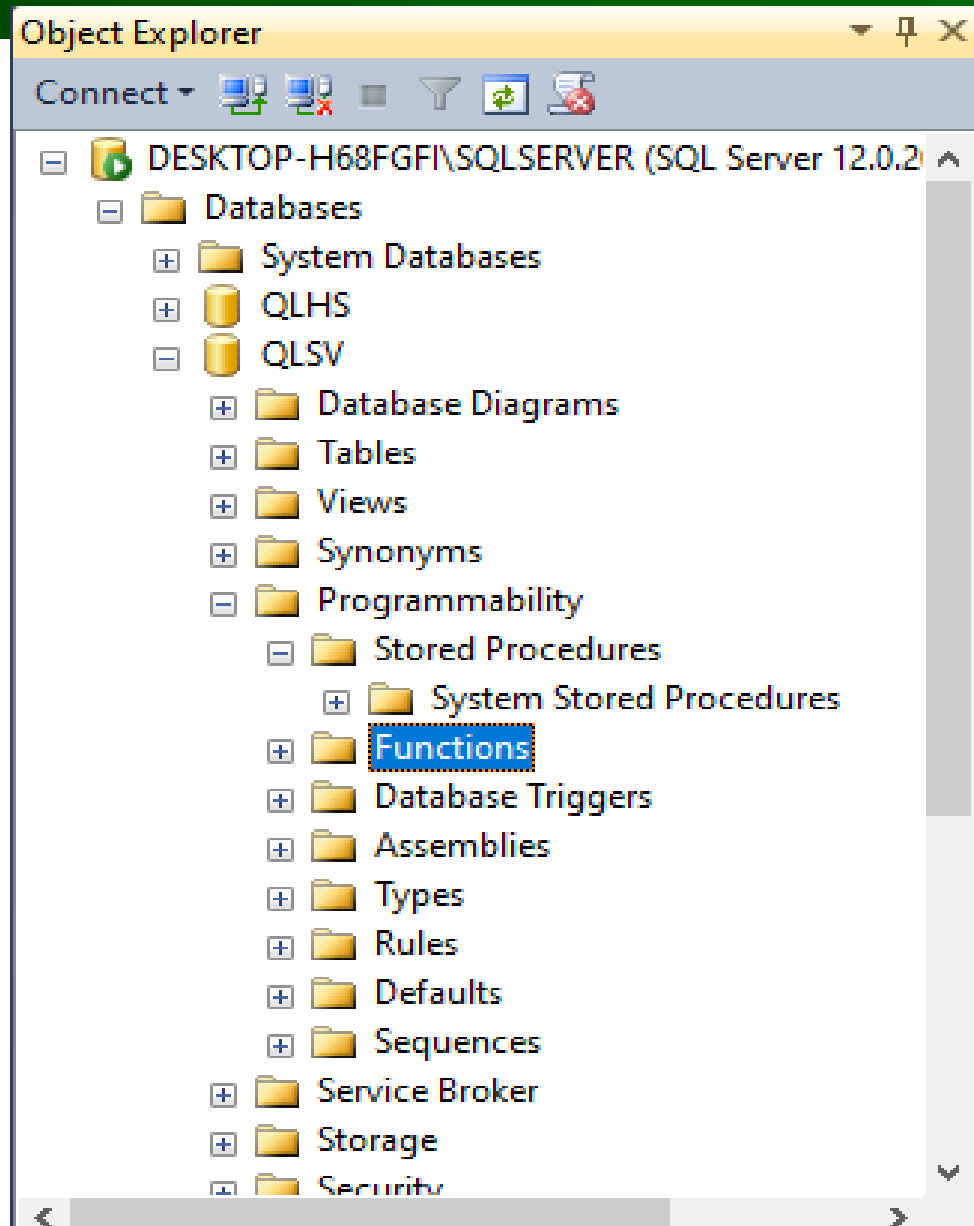
Tạo CSDL

- Vị trí tạo thủ tục




Tạo CSDL

➤ Vị trí tạo hàm



DESKTOP-H68FGFI\S...- dbo.tblSinhVien X

	Column Name	Data Type	Allow Nulls
	MaSV	char(11)	<input type="checkbox"/>
	TenSV	nvarchar(50)	<input checked="" type="checkbox"/>
	SDT	char(10)	<input checked="" type="checkbox"/>
	QueQuan	nvarchar(100)	<input checked="" type="checkbox"/>
	GhiChu	nchar(10)	<input checked="" type="checkbox"/>

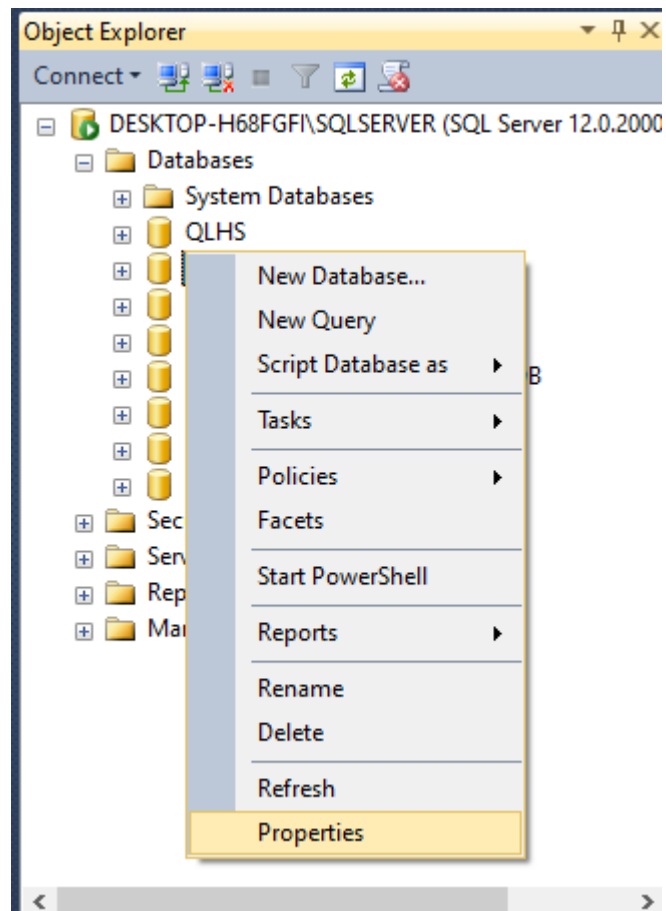
DESKTOP-H68FGFI\S...LSV - dbo.tblDiem × SQLQuery1.sql - DE...(Login)

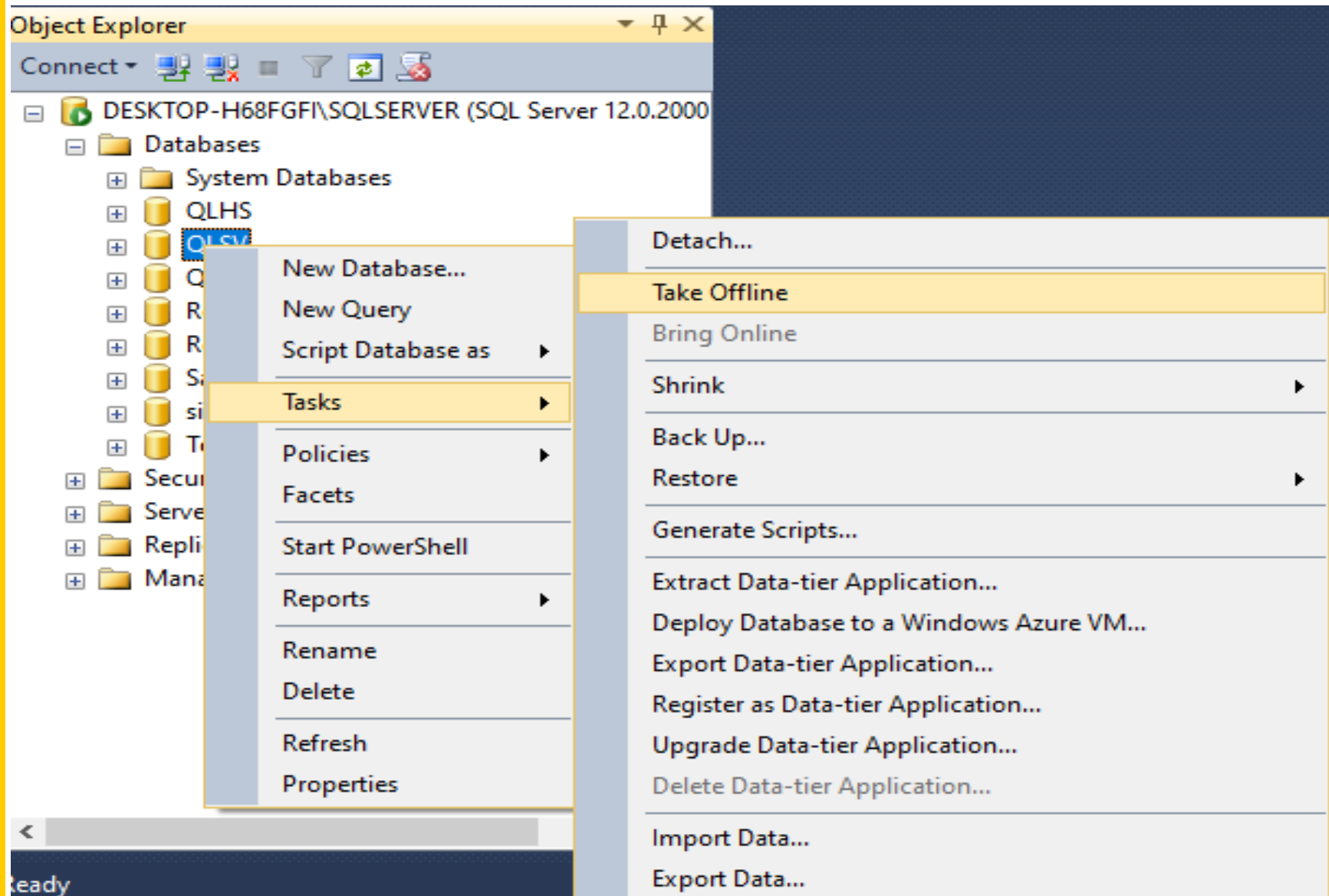
	Column Name	Data Type	Allow Nulls
▶ 🔑	MaSV	char(11)	<input type="checkbox"/>
▶ 🔑	MaMH	char(10)	<input type="checkbox"/>
	DiemLan1	float	<input checked="" type="checkbox"/>
	DiemLan2	float	<input checked="" type="checkbox"/>
	GhiChu	nchar(10)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

DESKTOP-H68FGFI\S...V - dbo.tblMonHoc × DESKTOP-H68FGFI\S...LSV -

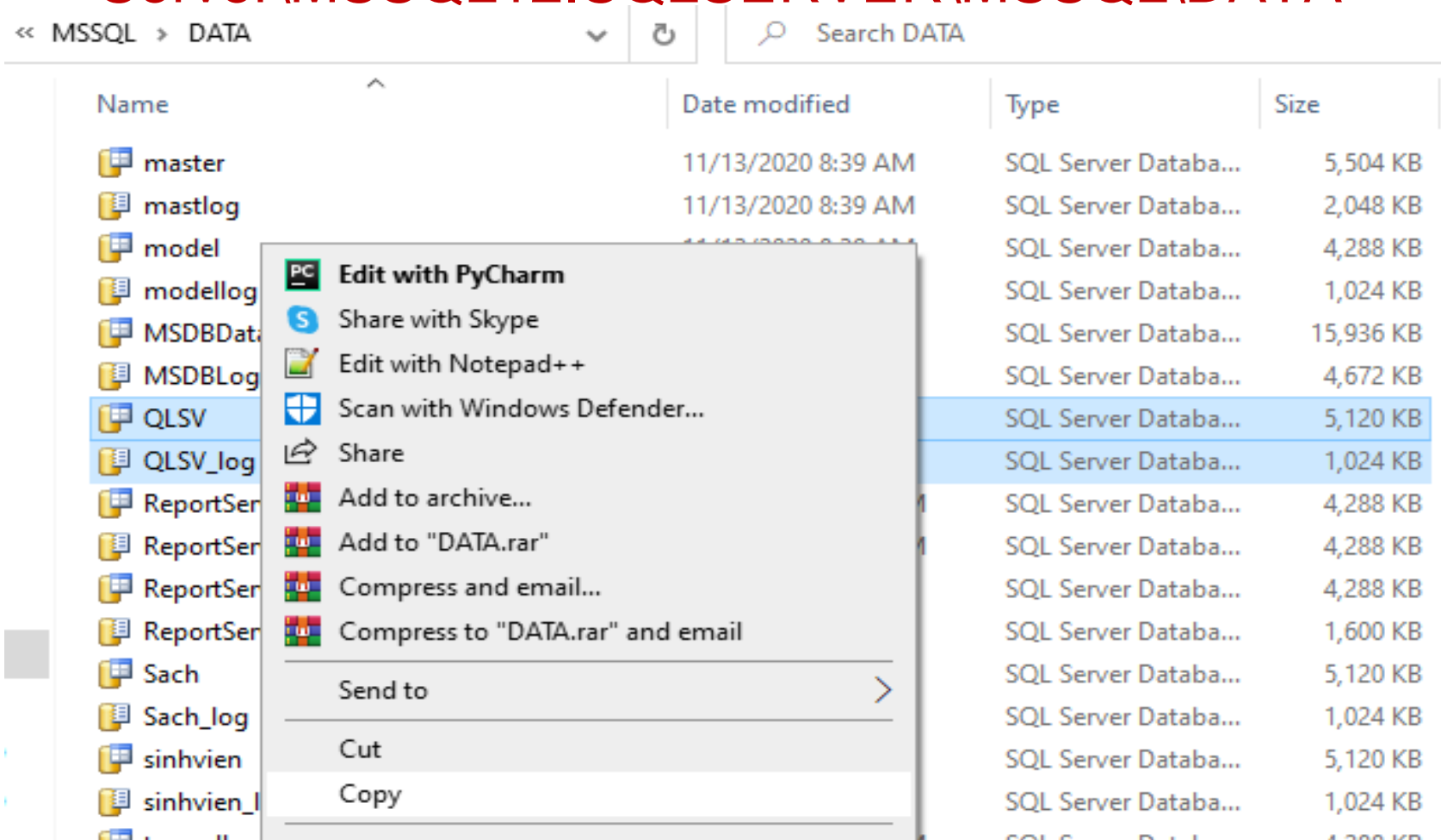
	Column Name	Data Type	Allow Nulls
▶ 🔑	MaMH	char(10)	<input type="checkbox"/>
	TenMH	nvarchar(50)	<input checked="" type="checkbox"/>
	SoTC	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

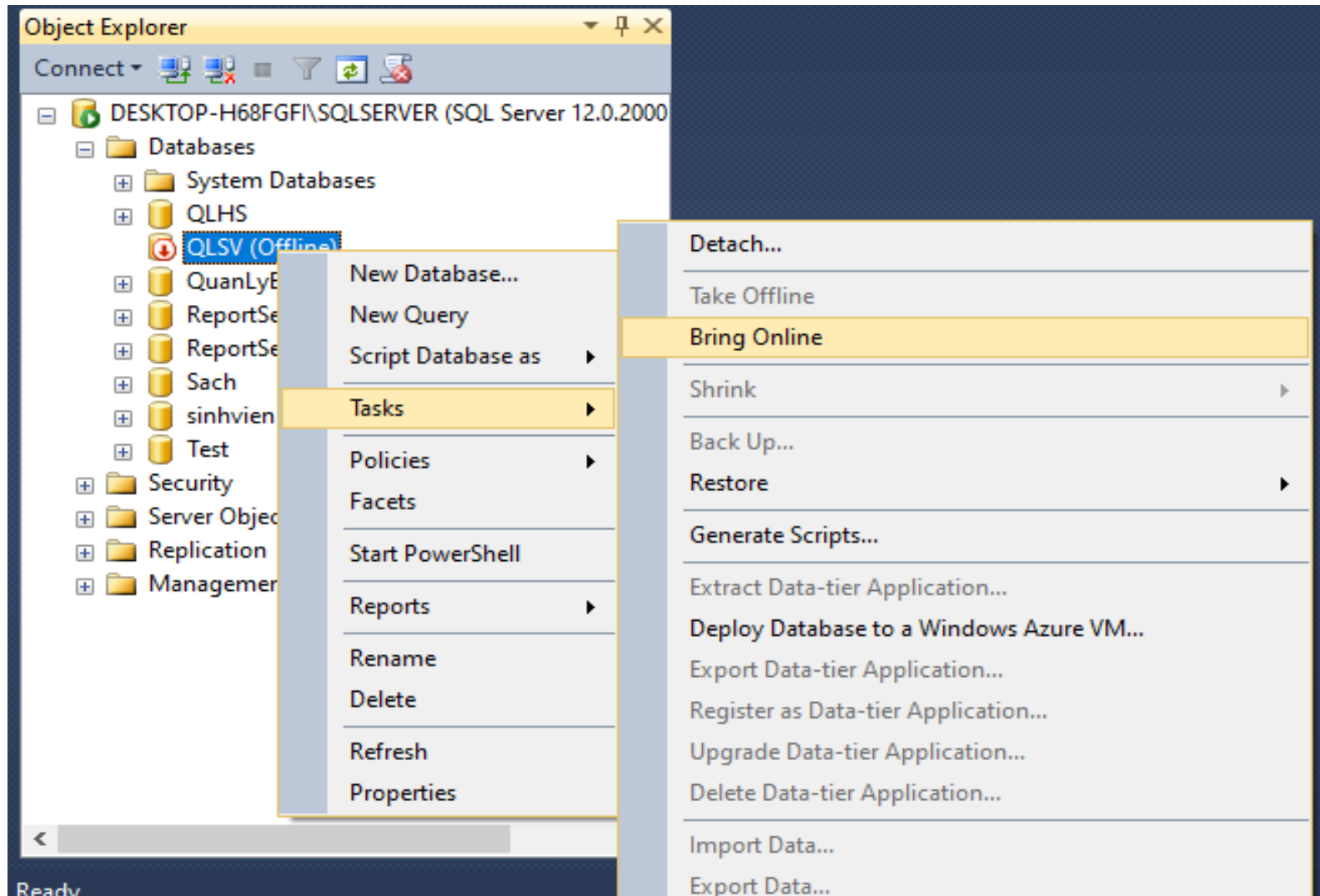
Tạo CSDL, vị trí vật lý của CSDL





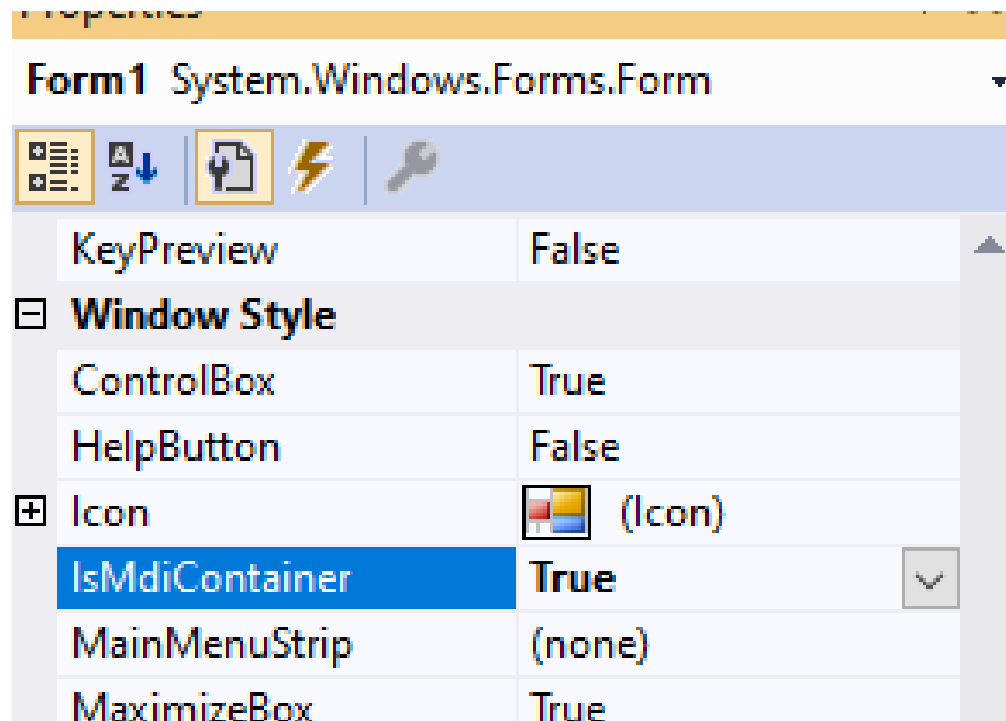
- C:\Program Files\Microsoft SQL Server\MSSQL12.SQLSERVER\MSSQL\DATA





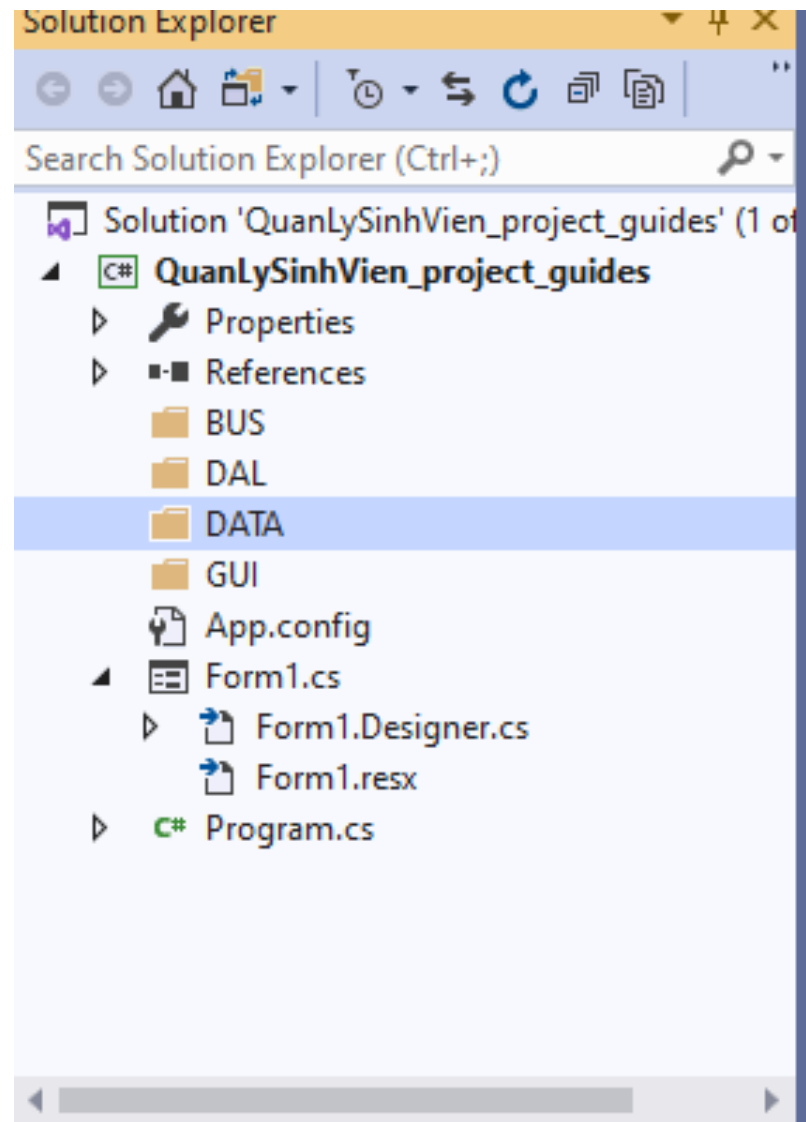
Tạo project

- Chú ý cho form main chuyển thành form chính



Tạo project

- Tạo project có kiến trúc



Tạo project

- IConnect, lưu ý thêm `using System.Data.SqlClient;`

```
0 references
interface IConnect
{
    0 references
    string UserID { get; set; }
    0 references
    string Password { get; set; }
    0 references
    string Database { get; set; }
    0 references
    string Server { get; set; }
    0 references
    string GetConnectionString();
    0 references
    SqlConnection Connect();
}
```

Tạo project

- Phương thức khởi dựng mặc định có tham số

```
public string Server { get; set; }  
/// <summary>  
/// Phương thức khởi dựng mặc định có tham số  
/// </summary>  
/// <param name="_UserID">Tên tài khoản đăng nhập vào SQL Server</param>  
/// <param name="_Password">Mat khau đăng nhập vào SQL server</param>  
/// <param name="_Database">Ten CSDL trong SQL Server</param>  
/// <param name="_Server">Ten máy chủ chứa SQL server</param>  
0 references  
public clsConnect(string _UserID, string _Password,  
    string _Database, string _Server)  
{  
    UserID = _UserID; Password = _Password;  
    Database = _Database; Server = _Server;  
}
```


Tạo project

➤ Phương thức GetConectionString

```
public string GetConnectionString()  
{  
    string strChuoiketNoiCoSoDuLieu;  
    strChuoiketNoiCoSoDuLieu = "";  
    strChuoiketNoiCoSoDuLieu += "Data Source =" + Server + ";";  
    strChuoiketNoiCoSoDuLieu += "Initial Catalog =" + Database + ";";  
    strChuoiketNoiCoSoDuLieu += "User ID =" + UserID + ";";  
    strChuoiketNoiCoSoDuLieu += "Password =" + Password + ";";  
    return strChuoiketNoiCoSoDuLieu;  
}
```

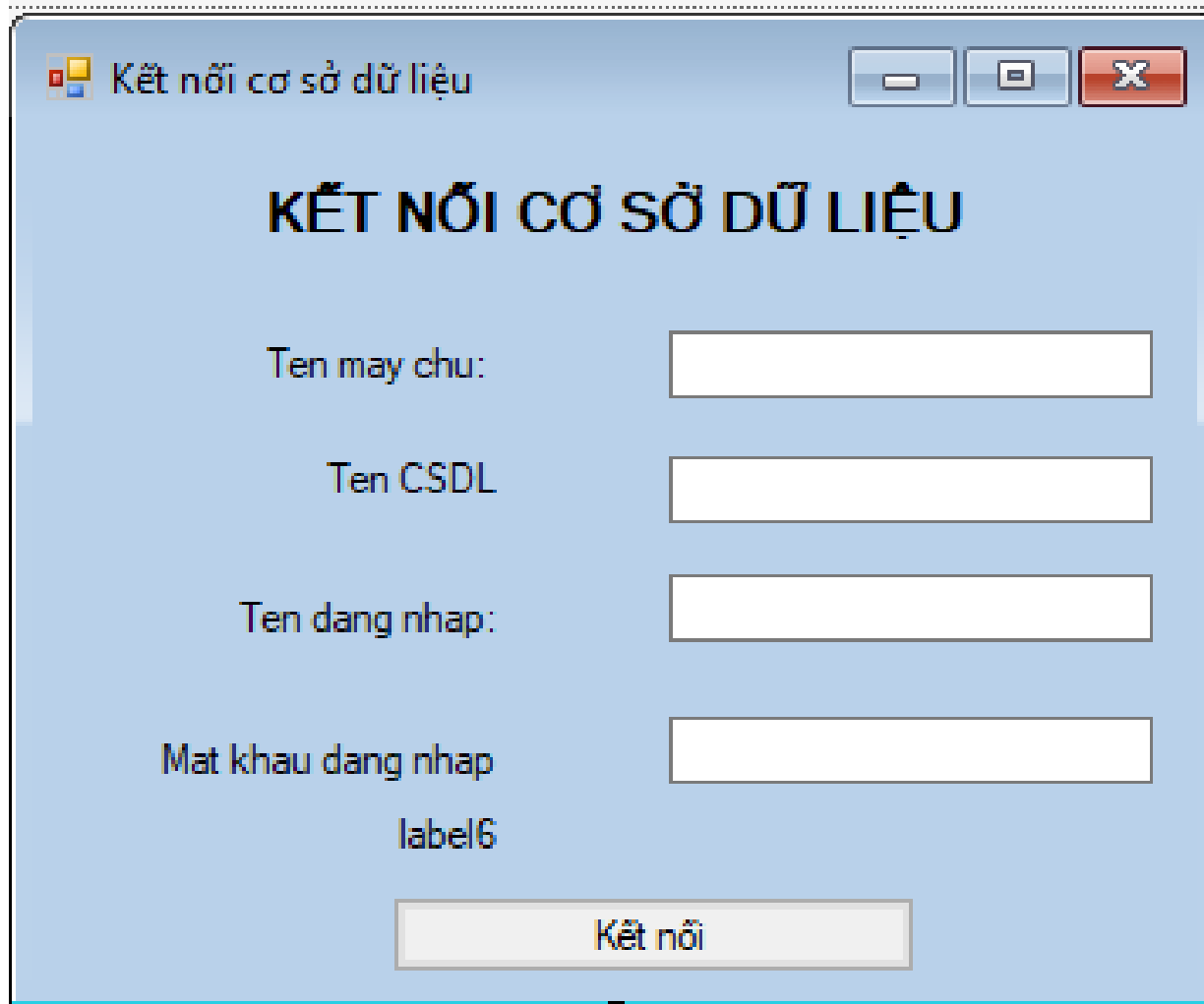
Tạo project

➤ Phương thức mở kết nối

```
public SqlConnection Connect()
{
    SqlConnection Conn = new SqlConnection();
    Conn.ConnectionString = GetConnectionString();
    try
    {
        Conn.Open();
    }
    catch (Exception)
    {
        throw;
    }
    return Conn;
}
```

Tạo project

➤ Form kết nối CSDL



Kết nối cơ sở dữ liệu

KẾT NỐI CƠ SỞ DỮ LIỆU

Ten máy chủ:

Ten CSDL

Ten đăng nhập:

Mat khau đăng nhập

label6

Kết nối

Xây dựng chức năng theo mô hình 3 lớp.

➤ Tạo ISinhVien

```
interface ISinhVien
{
    2 references
    string MaSV { get; set; }
    2 references
    string TenSV { get; set; }
    2 references
    string SDT { get; set; }
    2 references
    string QueQuan { get; set; }
    1 reference
    string GhiChu { get; set; }
    0 references
    int Them1SV(string sqlConnection);
    0 references
    int Sua1SV(string sqlConnection);
    0 references
    int Xoa1SV(string sqlConnection);
}
```

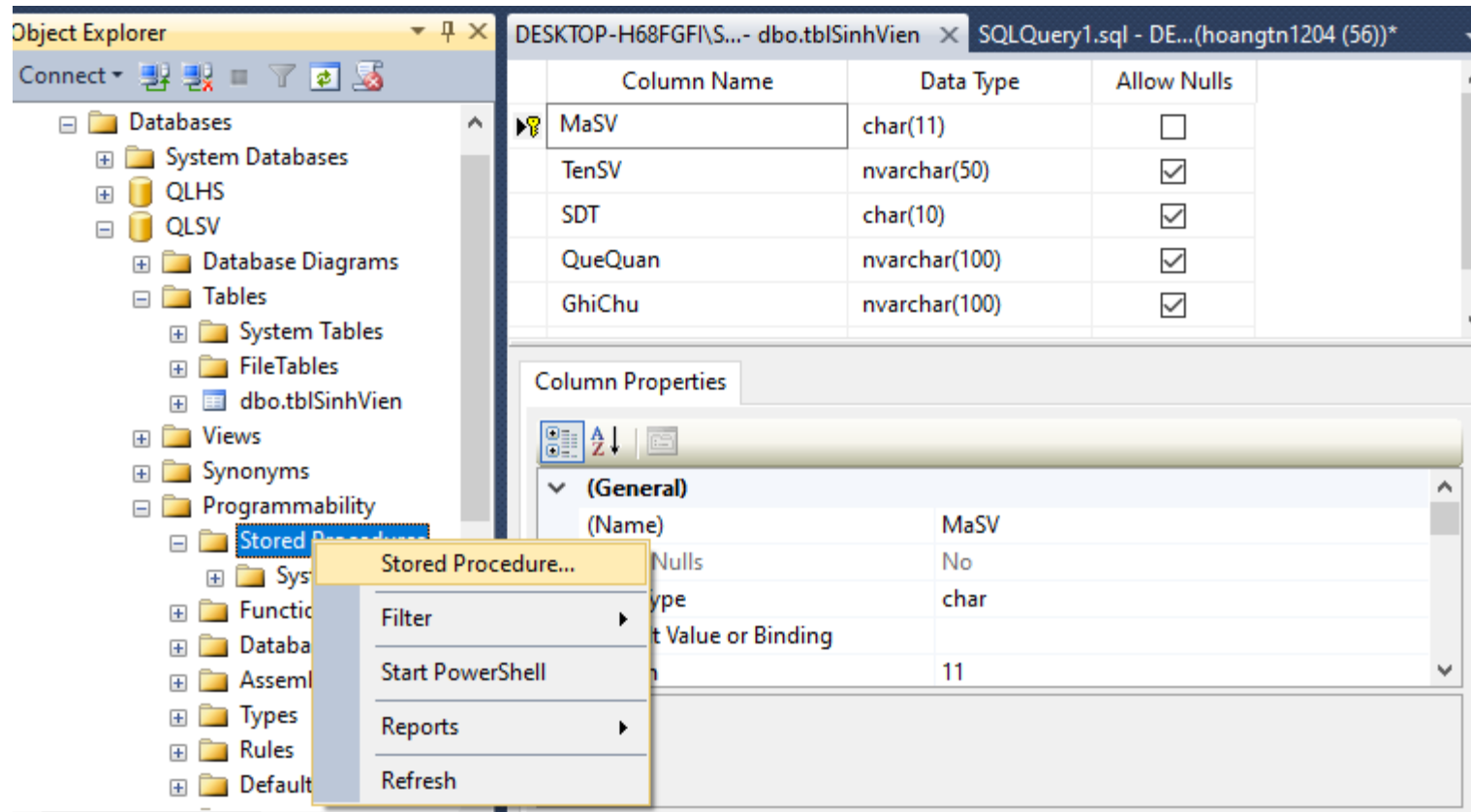
Xây dựng chức năng theo mô hình 3 lớp.

➤ clsSinhVien

```
class clsSinhVien:ISinhVien
{
    2 references
    public string MaSV { get; set; }
    2 references
    public string TenSV { get; set; }
    2 references
    public string SDT { get; set; }
    2 references
    public string QueQuan { get; set; }
    1 reference
    public string GhiChu { get; set; }
    0 references
    public clsSinhVien(string _masv,string _tensv,
        string _sdt,string _quequan)
    {
        MaSV = _masv;TenSV = _tensv;SDT = _sdt;QueQuan = _quequan;
    }
    1 reference
    public int Them1SV(string strConection) { return 0; }
    1 reference
    public int Sua1SV(string strConection) { return 0; }
    1 reference
    public int Xoa1SV(string strConection) { return 0; }
}
```

Xây dựng chức năng theo mô hình 3 lớp.

➤ Tạo thủ tục thêm mới



Xây dựng chức năng theo mô hình 3 lớp.

➤ Thủ tục thêm mới một sinh viên

```
create PROCEDURE sp_Them1SV
(
    @MaSV char(11),@TenSV nvarchar(50),@SDT char(10),@QueQuan nvarchar(100),
    @GhiChu nvarchar(100)
)
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    -- Insert statements for procedure here
insert into tblSinhVien(MaSV,TenSV,SDT,QueQuan,GhiChu)
values (@MaSV,@TenSV,@SDT,@QueQuan,@GhiChu)
END
GO
```

Xây dựng chức năng theo mô hình 3 lớp.

- Viết code cho phương thức thêm 1 sinh viên trong class

```
public int Them1SV(string strConection)
{
    SqlConnection connection = new SqlConnection(strConection);
    if (connection.State == ConnectionState.Closed) connection.Open();
    SqlCommand command = new SqlCommand();
    command.Connection = connection;
    command.CommandText = "sp_Them1SV";
    command.CommandType = CommandType.StoredProcedure;
    command.Parameters.AddWithValue("@MaSV", MaSV );
    command.Parameters.AddWithValue("@TenSV", TenSV);
    command.Parameters.AddWithValue("@SDT", SDT);
    command.Parameters.AddWithValue("@QueQuan", QueQuan);
    command.Parameters.AddWithValue("@GhiChu", GhiChu);
    command.ExecuteNonQuery();
    command.Dispose();
    connection.Dispose();
    return 0;
}
```


Xây dựng chức năng theo mô hình 3 lớp.

- Viết code cho phương thức sửa 1 sinh viên trong class

```
public int Sua1SV(string strConection) {  
    SqlConnection connection = new SqlConnection(strConection);  
    if (connection.State == ConnectionState.Closed) connection.Open();  
    SqlCommand command = new SqlCommand();  
    command.Connection = connection;  
    command.CommandText = "sp_Sua1SV";  
    command.CommandType = CommandType.StoredProcedure;  
    command.Parameters.AddWithValue("@MaSV", MaSV);  
    command.Parameters.AddWithValue("@TenSV", TenSV);  
    command.Parameters.AddWithValue("@SDT", SDT);  
    command.Parameters.AddWithValue("@QueQuan", QueQuan);  
    command.Parameters.AddWithValue("@NgaySinh", NgaySinh);  
    command.Parameters.AddWithValue("@GhiChu", GhiChu);  
    command.ExecuteNonQuery();  
    command.Dispose();  
    connection.Dispose();  
    return 0;  
}
```

Xây dựng chức năng theo mô hình 3 lớp.

- Viết code cho thủ tục sửa 1 sinh viên trong sql server

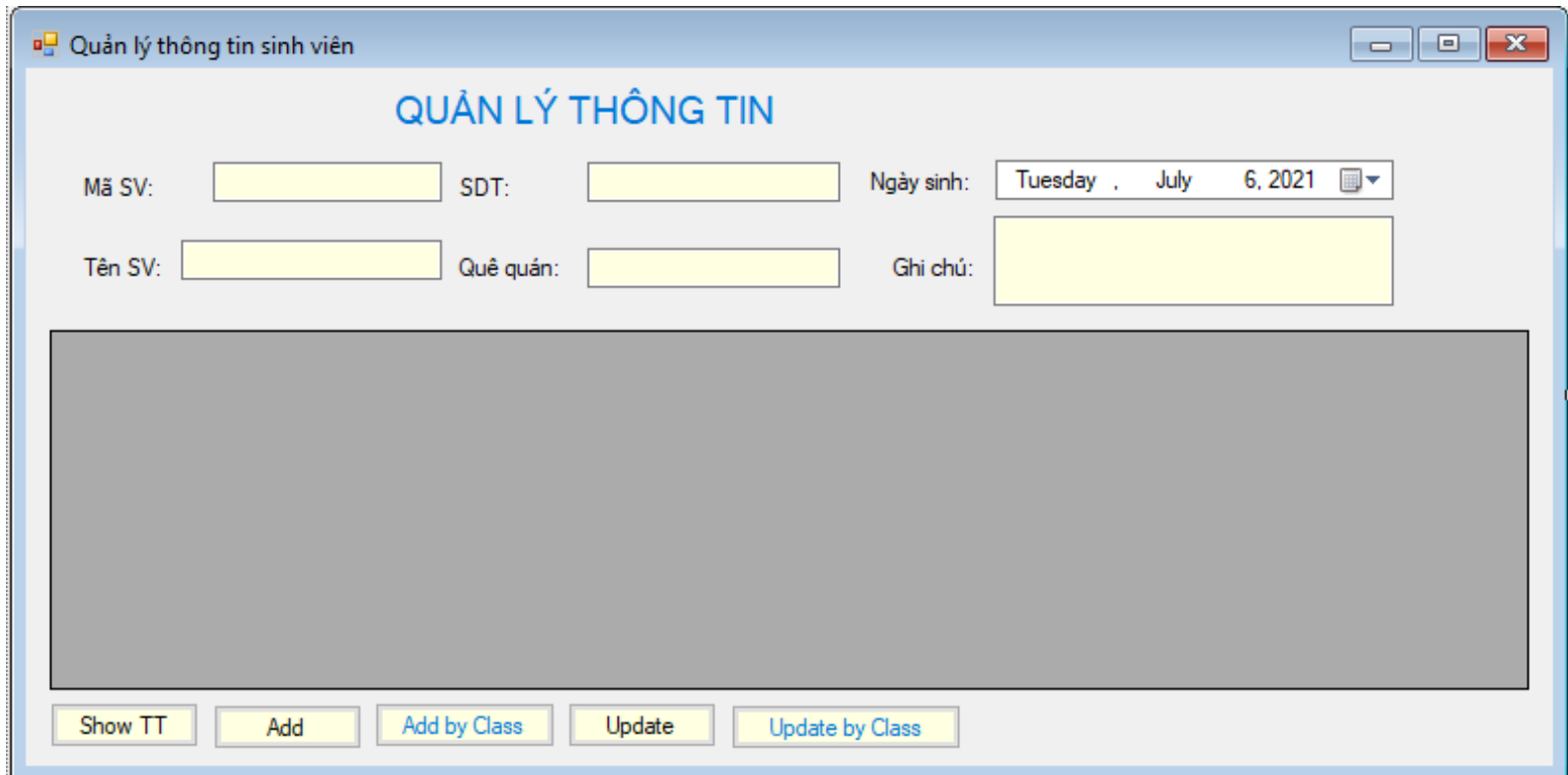
```
-- =====
-- Author:      <hoangtn, Hoang>
-- Create date: <06 thang 07 nam 2021>
-- Description: <Sua 1 sinh vien>
-- =====

ALTER PROCEDURE [dbo].[sp_Sua1SV]
(
    @MaSV char (11),
    @TenSV nvarchar(50),
    @NgaySinh date,
    @SDT char(10),
    @QueQuan nvarchar(100),
    @GhiChu nvarchar(100)
)
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    -- Insert statements for procedure here
    Update tblSinhVien
    SET
        TenSV=@TenSV, NgaySinh=@NgaySinh, SDT=@SDT, QueQuan=@QueQuan, GhiChu=@GhiChu
    Where MaSV=@MaSV
END
```

Xây dựng chức năng theo mô hình 3 lớp.

➤ Sự kiện click cho nút AddbyClass



```
private void btnAddbyClass_Click(object sender, EventArgs e)
{
    DAL.clsSinhVien sv = new DAL.clsSinhVien(txtMaSV.Text, txtTenSV.Text
        , txtSDT.Text, txtQueQuan.Text, txtGhiChu.Text);
    sv.Them1SV(StrConectionString);
}
```

Xây dựng chức năng theo mô hình 3 lớp.

➤ Sự kiện click cho nút UpdatebyClass

Quản lý thông tin sinh viên

QUẢN LÝ THÔNG TIN

Mã SV: SDT: Ngày sinh:

Tên SV: Quê quán: Ghi chú:

	MaSV	TenSV	NgaySinh	SDT	QueQuan	GhiChu
	20011930061	Mai Duy Chung		0911222333	Thanh Hóa	NEM CHUA
	20011930077	Phan Thị Mai Hoa		0933434356	Vinh Phu	Có Không
	20011930088	Hoàng Thị Luyến		0987222134	Nam Định	Ko
	20011930125	Mai Thúy Hằng	5/20/2001	0983222567	Hải Dương	Học khóa K19
▶	20011930131	Hoàng Thị Thu H...		0983255187	Thái Bình	Que lua
	20011930133	Nguyễn Thị Phúc...	5/20/2001	0981222555	Thạch Thất - Hà ...	Hôm nay nghỉ ốm
	20011930142	Trần Minh Nguyên		0975223456	Thái Bình	Thái Bình

Show TT Add Add by Class Update Update by Class

```
private void btnUpdatebyClass_Click(object sender, EventArgs e)
{
    DAL.clsSinhVien sv = new DAL.clsSinhVien
        (txtMaSV.Text.Trim(), txtTenSV.Text.Trim()
        ,txtSDT.Text.Trim(), txtQueQuan.Text.Trim(), txtGhiChu.Text.Trim(), dtpNgaySinh.Value)
    sv.Sua1SV(StrConectionString);
}
```

Crystal Report

Crystal Report

➤ Bản cài đặt:

<https://www.tektutorialshub.com/crystal-reports/download-crystal-reports-for-visual-studio-2019/>

Crystal Reports Version	Supported IDE	tải bản này cho win 64 bit	Developer Edition Download	Runtime Download
Service Pack 29	Visual Studio 2019, 2017,2015,2013,2012,2010		SP 29	32 Bit 64 Bit
Service Pack 28	Visual Studio 2019, 2017,2015,2013,2012,2010		SP 28	32 Bit 64 Bit
Service Pack 27	Visual Studio 2019, 2017,2015,2013,2012,2010		SP 27	32 Bit 64 Bit
Service Pack 26	Visual Studio 2019, 2017,2015,2013,2012,2010		SP 26	32 Bit 64 Bit
Service Pack 25	Visual Studio 2019, 2017,2015,2013,2012,2010		SP 25	32 Bit 64 Bit

➤ Tạo CSDL

DESKTOP-H68FGFI\S...LSV - dbo.tblDiem X

	Column Name	Data Type	Allow Nulls
▶	MaSV	char(11)	<input type="checkbox"/>
▶	MaMH	char(10)	<input type="checkbox"/>
	Diem	float	<input checked="" type="checkbox"/>
	GhiChu	nchar(10)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

DESKTOP-H68FGFI\...V - dbo.tblMonHoc X DESKTOP-H68FGFI\S...LSV -

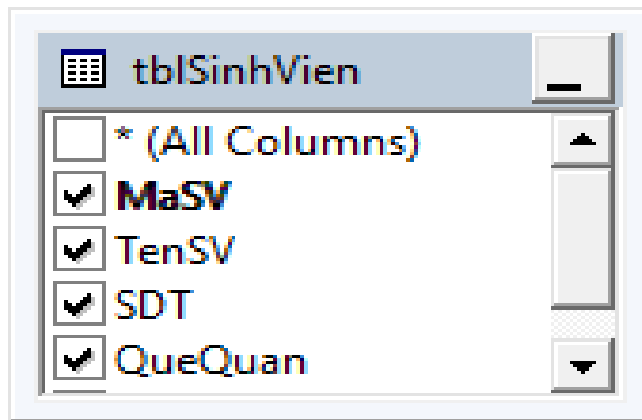
	Column Name	Data Type	Allow Nulls
▶	MaMH	char(10)	<input type="checkbox"/>
	TenMH	nvarchar(50)	<input checked="" type="checkbox"/>
	SoTC	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

DESKTOP-H68FGFI\S...- dbo.tblSinhVien X DESKTOP-H68FGFI\...V - db

	Column Name	Data Type	Allow Nulls
▶	MaSV	char(11)	<input type="checkbox"/>
	TenSV	nvarchar(50)	<input checked="" type="checkbox"/>
	SDT	char(10)	<input checked="" type="checkbox"/>
	QueQuan	nvarchar(100)	<input checked="" type="checkbox"/>
	GhiChu	nvarchar(100)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

➤ Tạo View

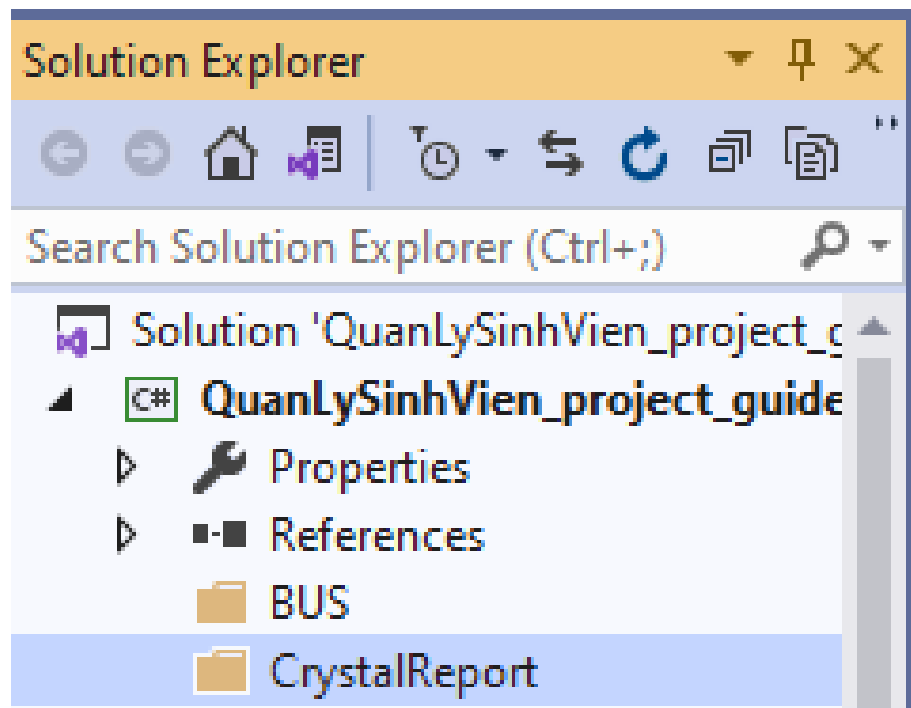
DESKTOP-H68FGFI\S...dbo.v_tblSinhVien* X DESKTOP-H68FGFI\S...- dbo.tblSinhVien

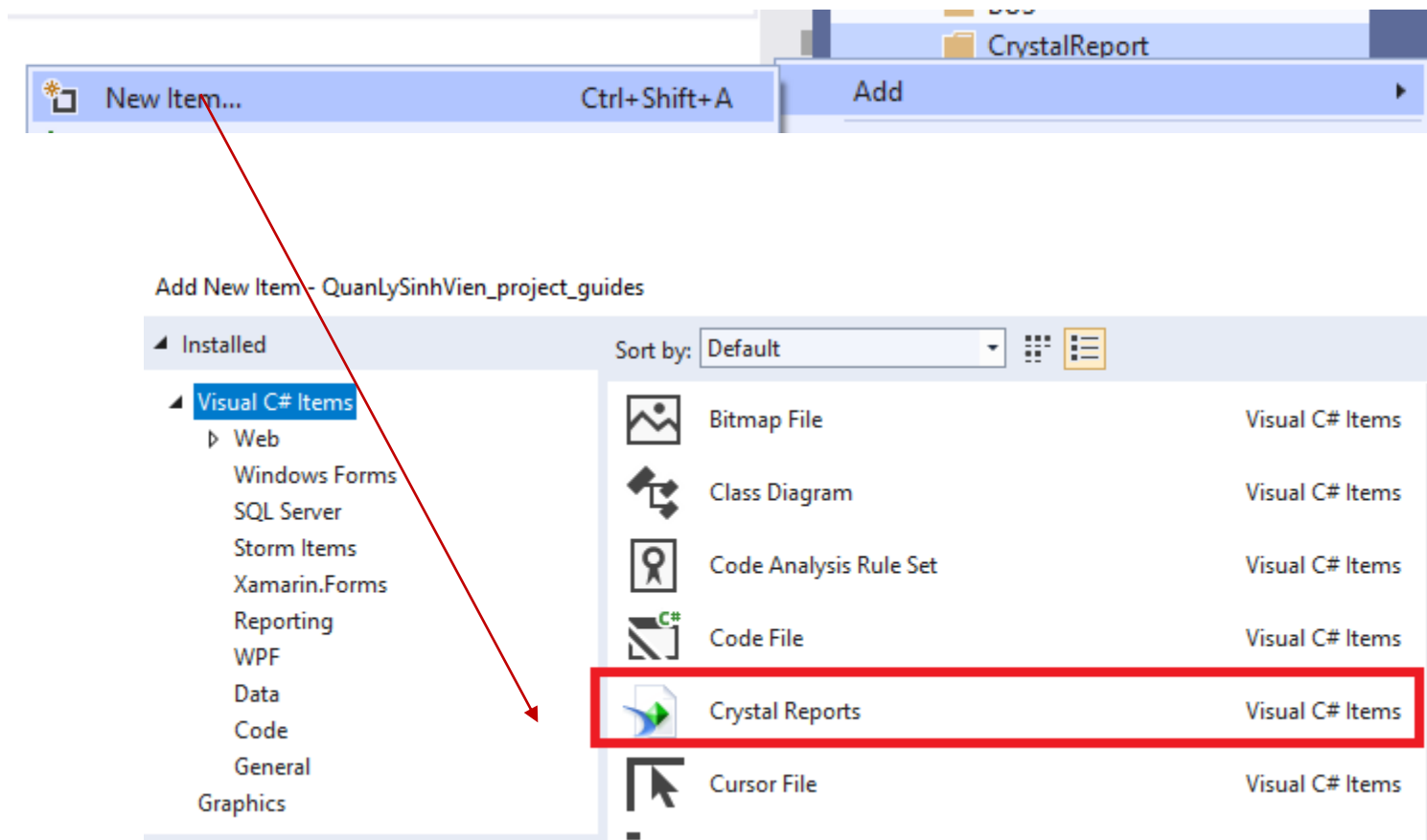


	Column	Alias	Table	Outp...	Sort Type
▶	MaSV		tblSinhVien	<input checked="" type="checkbox"/>	
	TenSV		tblSinhVien	<input checked="" type="checkbox"/>	

```
SELECT  MaSV, TenSV, SDT, QueQuan, GhiChu
FROM    dbo.tblSinhVien
```

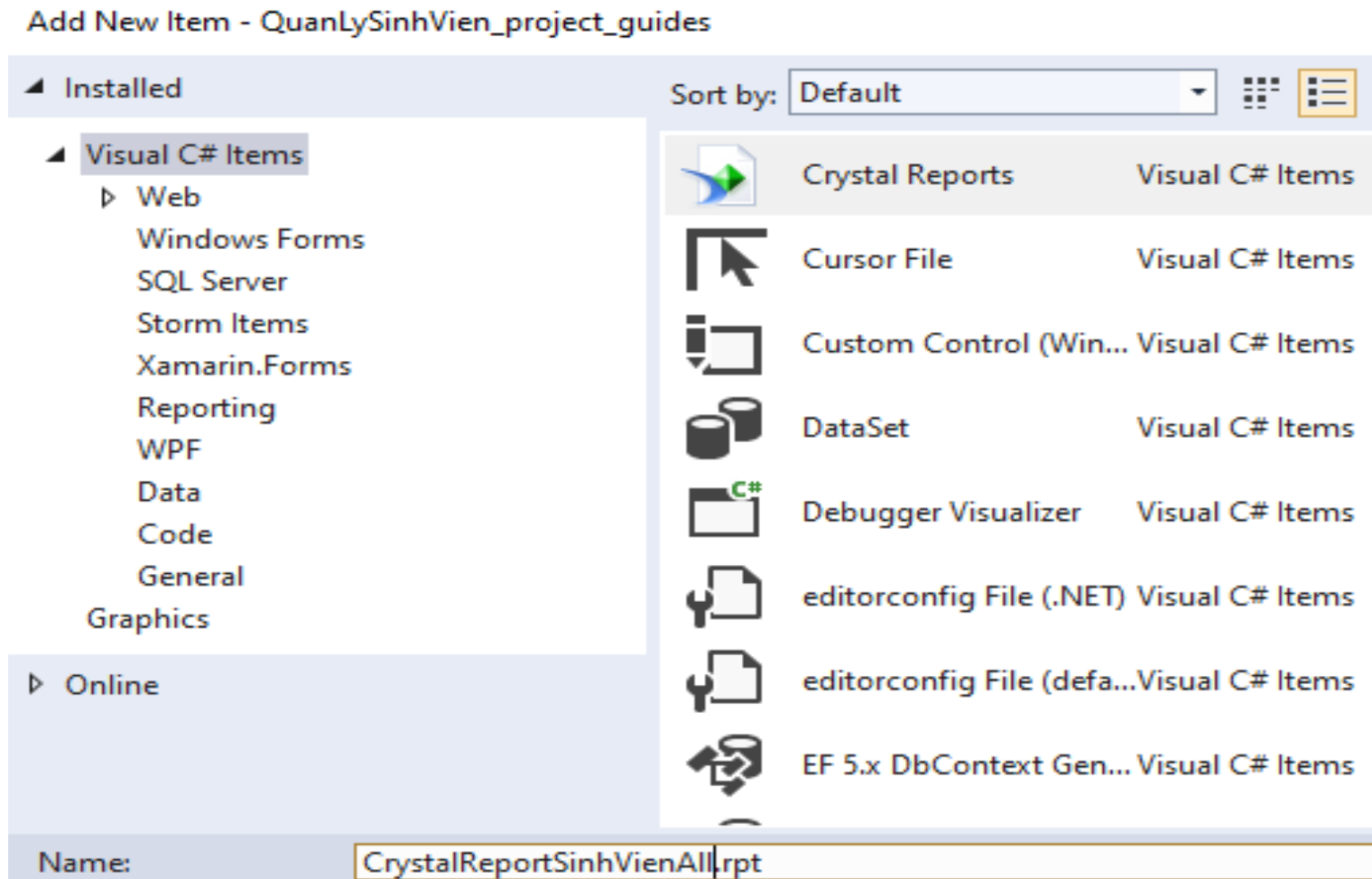

➤ Trong Project tạo thư mục CrystalReport



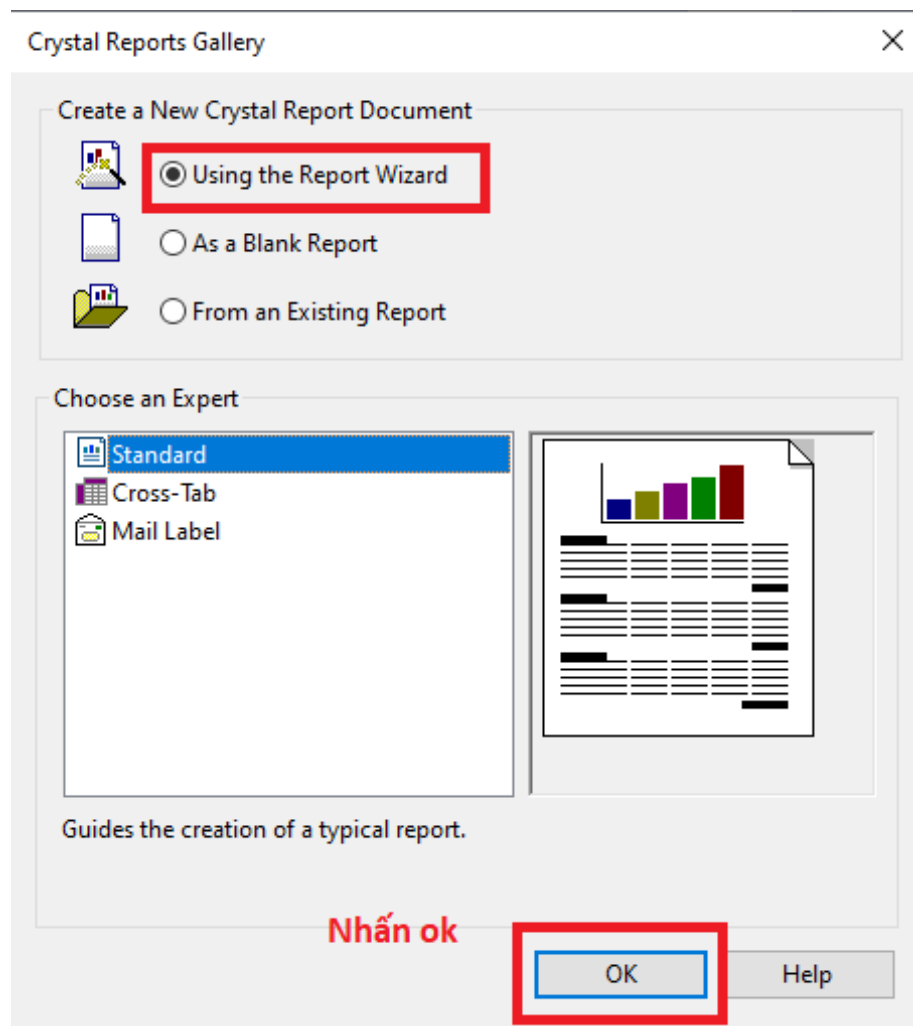


Crystal Report

- Đặt tên file là CrystalReportSinhVienAll



Crystal Report



Crystal Report

Standard Report Creation Wizard

Data

Choose the data you want to report on.

Available Data Sources:

- + Project Data
- + My Connections
- Create New Connection
 - + Access/Excel (DAO)
 - + ADO.NET (XML)
 - + Database Files
 - + ODBC (RDO)
 - + **OLE DB (ADO)**
 - + More Data Sources

Nhấn vào dấu +

Selected Tables:

< Back

Next >

Finish

Cancel

Help

Crystal Report

OLE DB (ADO)



OLE DB Provider

Select a provider from the list or select a data link file.

Provider:

- Microsoft Jet 4.0 OLE DB Provider
- Microsoft Office 12.0 Access Database Engine O
- Microsoft Office 16.0 Access Database Engine O
- Microsoft OLE DB Provider for Analysis Services
- Microsoft OLE DB Provider for ODBC Drivers
- Microsoft OLE DB Provider for Oracle
- Microsoft OLE DB Provider for Search
- Microsoft OLE DB Provider for SQL Server**
- Microsoft OLE DB Simple Provider

Use Data Link File: ☐

Microsoft Data Link File:

< Back **Next >** Finish Cancel Help

Crystal Report

OLE DB (ADO) ×

Connection Information
Provide necessary information to log on to the chosen data source.

Server: **Tên máy chủ** DESKTOP-H68FGFI\SQLSERVER ▼

User ID: **Tài khoản**
Đăng nhập SQL server hoangtn1204

Password: **Mật khẩu** ...

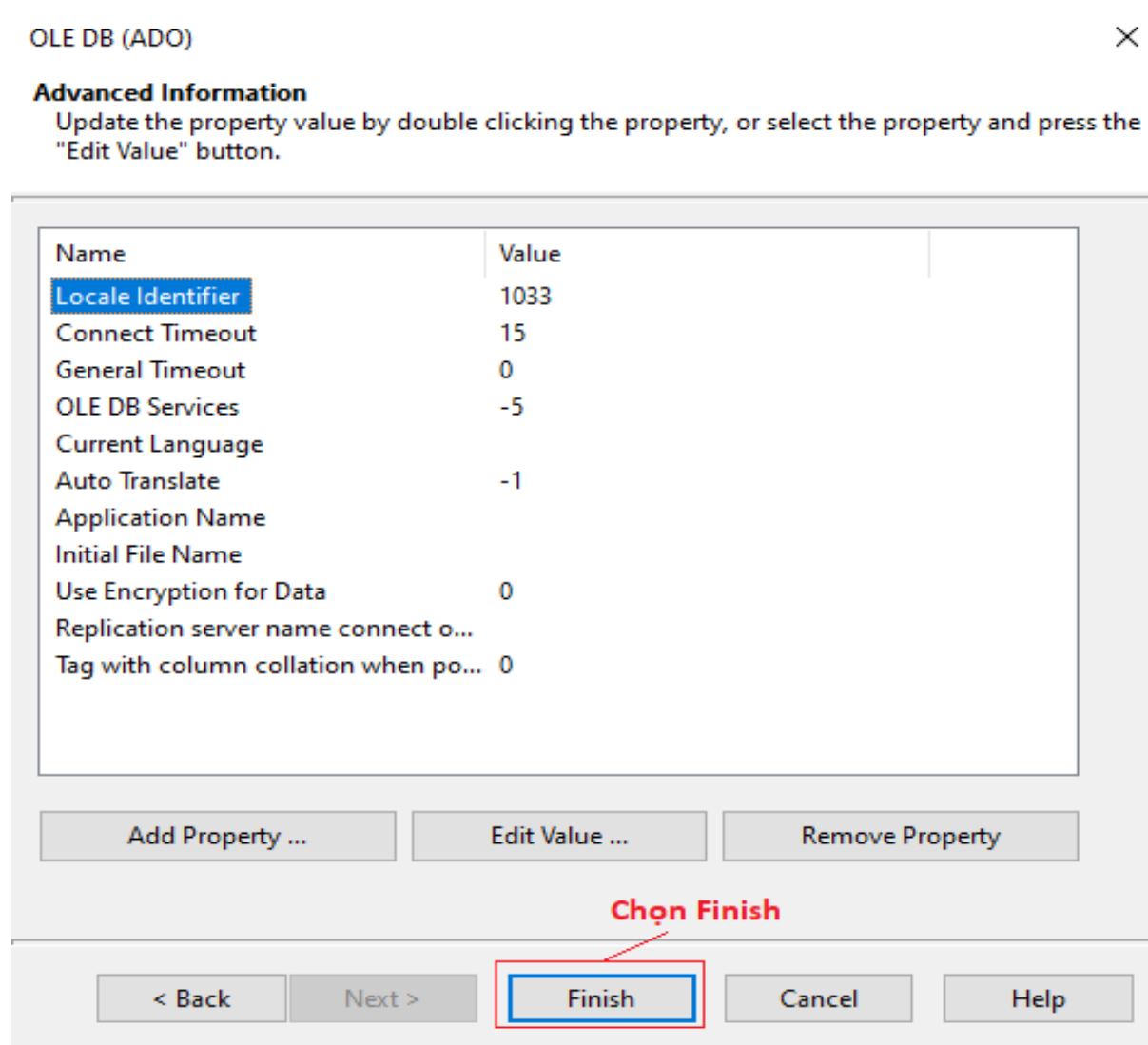
Database: **Cơ sở dữ liệu** QLSV ▼

Integrated Security: ☐

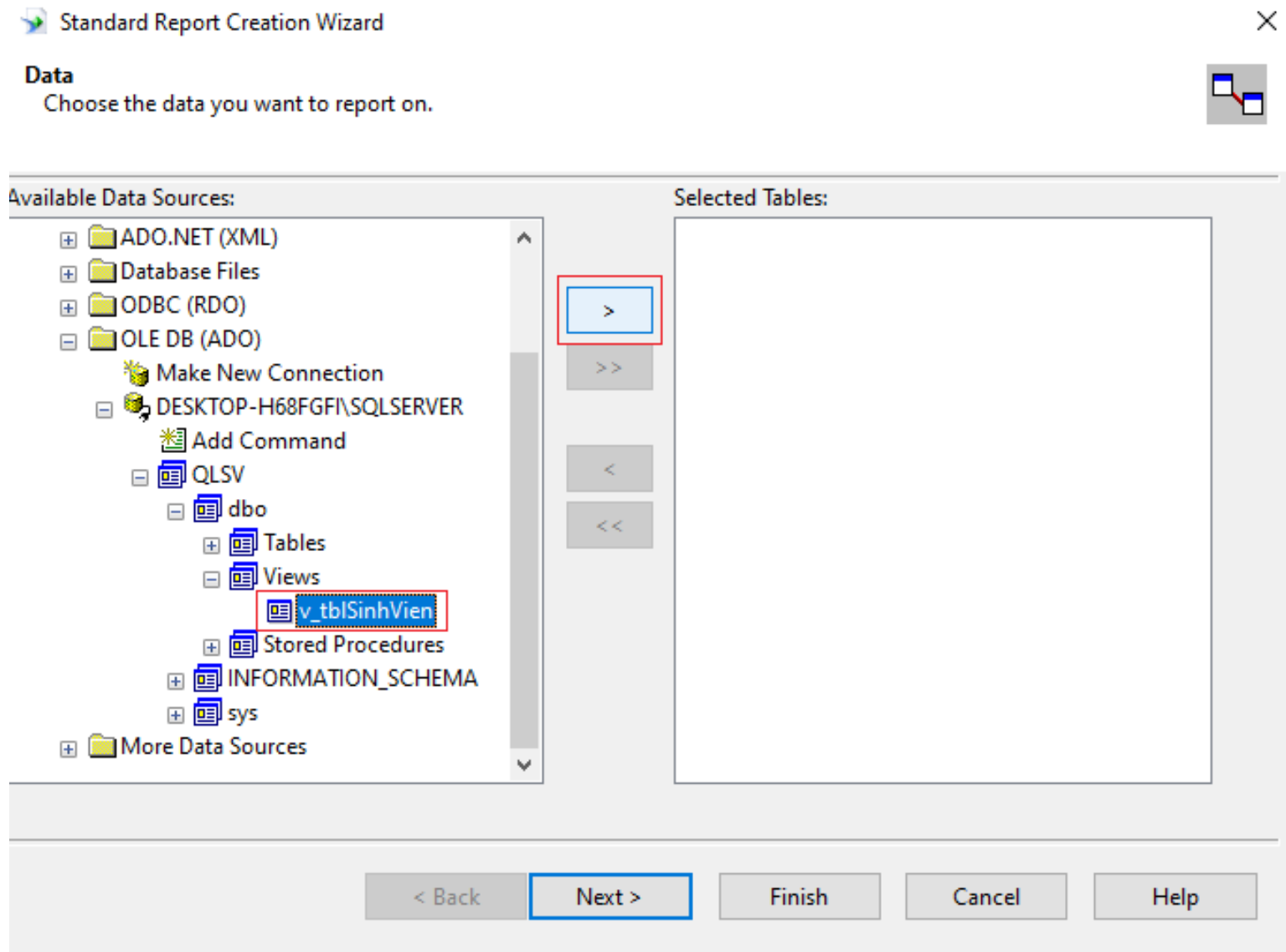
Chọn Next

< Back **Next >** Finish Cancel Help

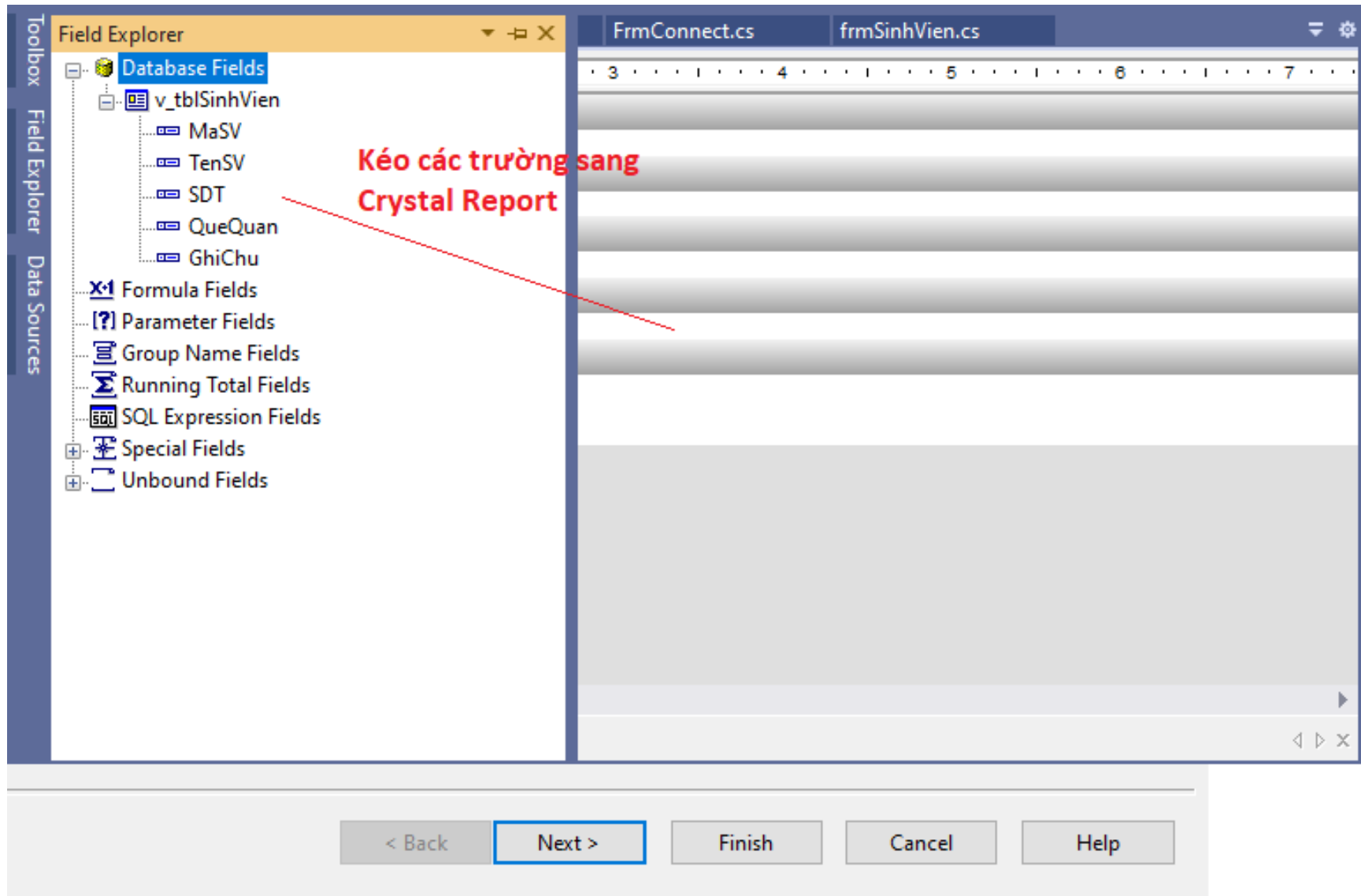
Crystal Report

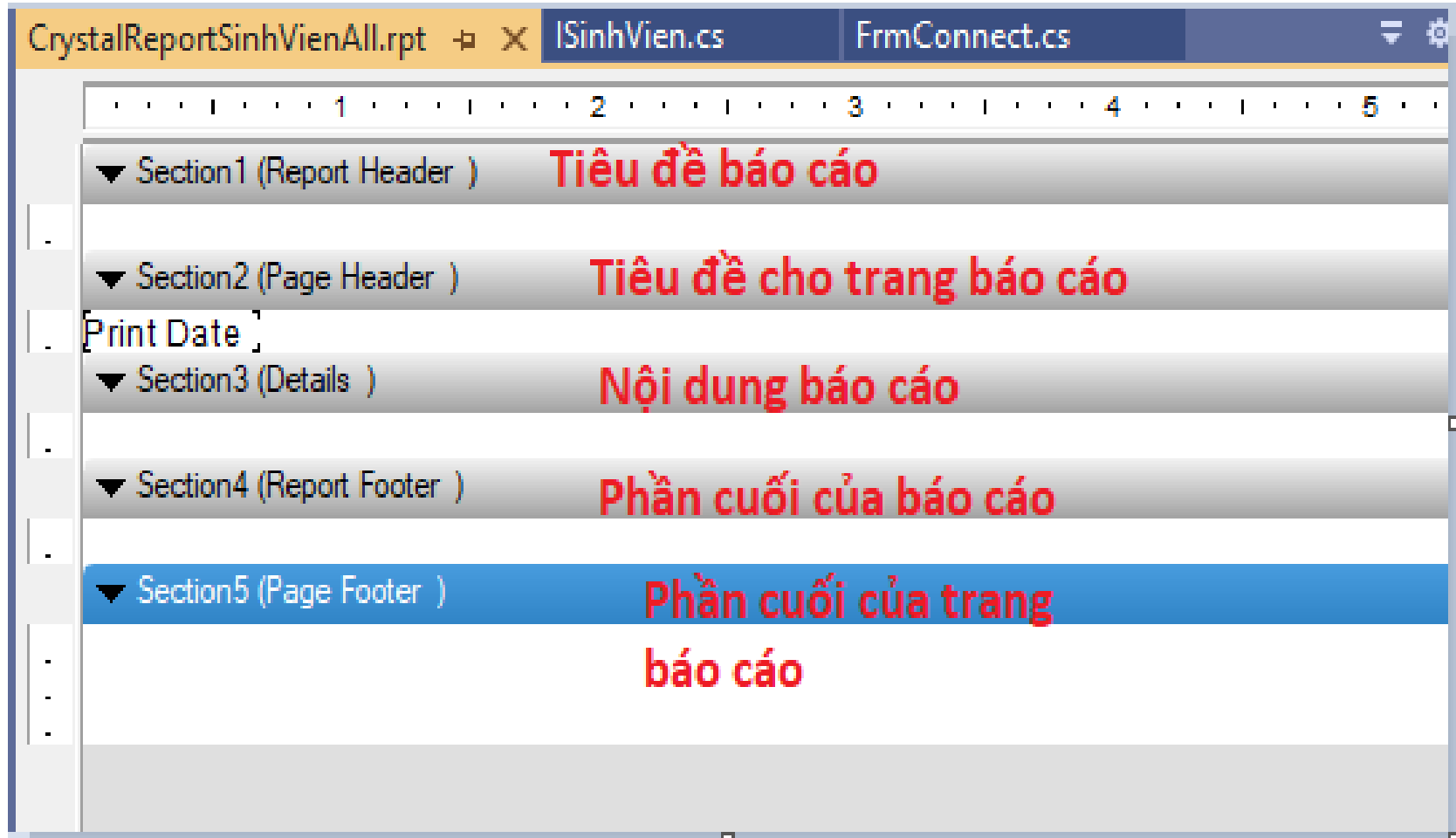


Crystal Report



Crystal Report





Crystal Report

CrystalReportSinhVienAll.rpt* x ISinhVien.cs FrmConnect.cs

Section1 (Report Header)

BÁO CÁO SINH VIÊN ĐANG HỌC

Section2 (Page Header)

Mã SV Tên SV SDT Quê quán Ghi

Section3 (Details)

MãSV TênSV SDT QueQuan Ghi

CrystalReportSinhVienAll.rpt* x ISinhVien.cs FrmConnect.cs

☒ Save data in the report

1 / 1 100%

SAP CRYSTAL REPORTS®

BÁO CÁO SINH VIÊN ĐANG HỌC

Mã SV	Tên SV	SDT	Quê quán
20011930055	Nguyễn Thị Nhật Vy	0933434356	Ninh Bình
20011930061	Mai Duy Chung	0911222333	Thanh Hóa
20011930077	Phan Th? Mai Hoa	0933434356	Vinh Phu
20011930088	Hoàng Thị Luyến	0987222134	Nam Định
20011930125	Mai Thúy Hằng		Hải Dương
20011930131	Hoàng Thị Thu Hòa	0983255187	Thái Bình
20011930133	Nguyễn Thị Phúc Hươ	0981222555	Thạch Thất - Hà Nội
20011930142	Trần Minh Nguyễn	0975223456	Thái Bình
20011930149	Hoàng Phương Thảo	0975455677	Bắc Ninh
20011930272	Hà Trọng		Thái Bình
20011930275	Đặng Linh Chi		Vĩnh Phúc
20011930321	Nguyễn Duy Tiến	0983466788	Hà Nội

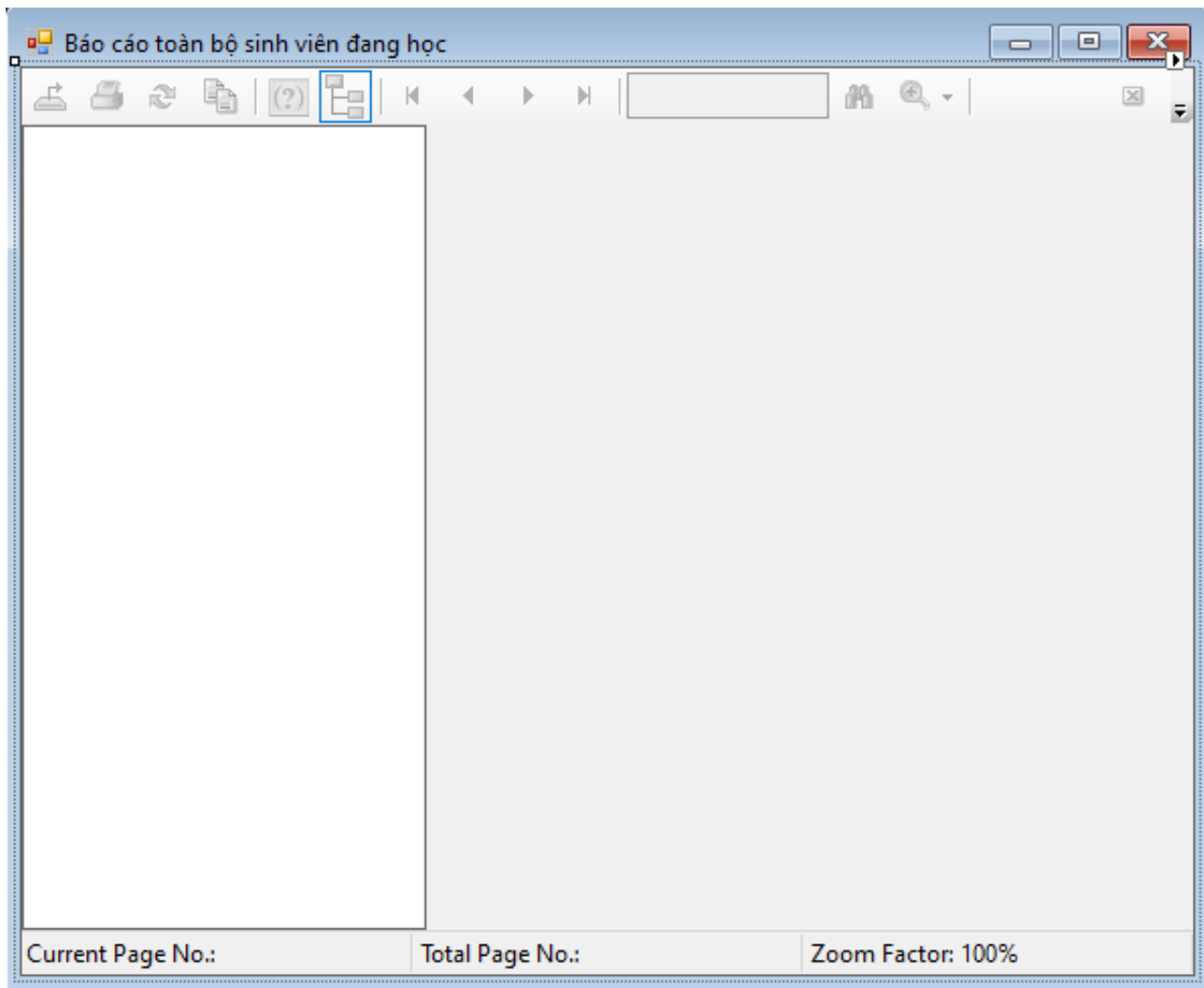
Crystal Report

➤ Trong C# tạo lớp:

```
namespace QuanLySinhVien_project_guides.BUS
{
    1 reference
    class ClsReportSinhVien
    {
        1 reference
        public void BaoCaoAllSV(CrystalReportViewer reporViewer,string CrystalReportFileName)
        {
            ReportDocument Report = new ReportDocument();
            Report.Load(CrystalReportFileName);
            Report.SetDatabaseLogon( "hoangtn1204", "123", @"DESKTOP-H68FGFI\SQLSERVER", "QLSV");
            reporViewer.ReportSource = Report;
        }
    }
}
```

Crystal Report

- Tạo form frmReportSV, add thêm CrystalReportViewer



Crystal Report

- Trong phương thức load form ghi đoạn code

```
private void frmReportSV_Load(object sender, EventArgs e)
{
    BUS.ClsReportSinhVien report = new BUS.ClsReportSinhVien();
    report.BaoCaoAllSV(rpvSinhvienAll1, @"..\..\CrystalReport\cRSinhVien.rpt");
}
```



The End