

BÀI THỰC HÀNH: SỬ DỤNG STORED PROCEDURE

1. Stored Procedure

Stored Procedure là tập hợp của các câu lệnh T-SQL được biên dịch trước (pre_compiled). Stored Procedure được đặt tên và được xử lý như một khối lệnh thống nhất.

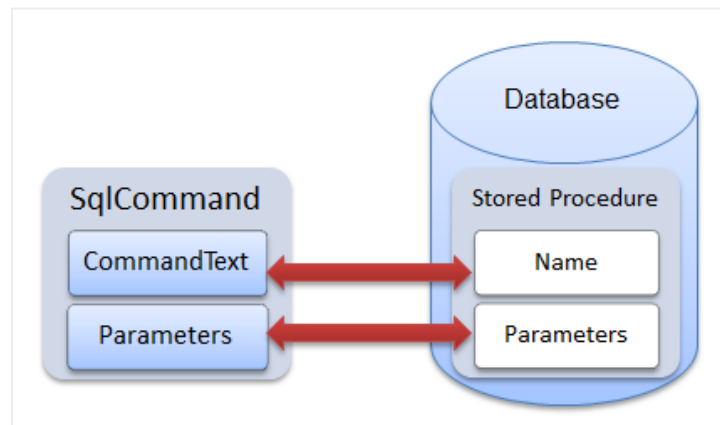
Lợi ích khi quản lý dữ liệu bằng Stored Procedure

- Tốc độ truy cập dữ liệu nhanh
- Chương trình được module hoá
- Đảm bảo tính nhất quán về dữ liệu
- Nâng cao khả năng bảo mật dữ liệu

Cú pháp đơn giản:

```
CREATE PROCEDURE procedure_name
@parameter1 data_type [ output] /*các tham số*/,
@parameter2 data_type [ output]
AS
BEGIN
[khai báo các biến cho xử lý]
{ Các câu lệnh transact-sql}
END
GO
```

2. Sử dụng Stored Procedure Trong C#



Để gọi một Stored Procedure trong C# ta làm như sau:

Bước 1: Tạo kết nối với CSDL qua SqlConnection

Bước 2: Thiết lập SqlCommand để thực thi Stored Procedure. Tạo đối tượng SqlCommand cho biết Stored Procedure nào sẽ được thực thi và thiết lập chế độ thực thi Stored Procedure cho SqlCommand.

```
SqlCommand cmd = new SqlCommand("Stored Procedure Name", connection);
```

```
cmd.CommandType = CommandType.StoredProcedure;
```

Bước 3: Bổ sung tham số cho Stored Procedure

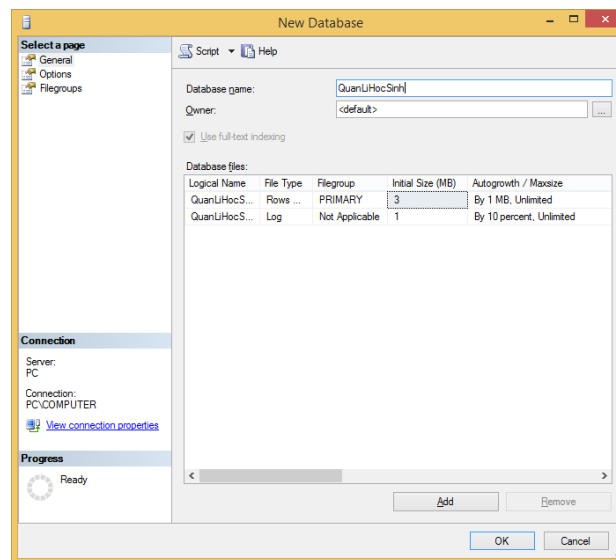
```
cmd.Parameters.Add(new SqlParameter("@MaHS", maHS));
```

Lưu ý: Tên của tham số được truyền vào trong SqlParameter phải giống với tên tham số của Stored Procedure đã tạo. Sau đó thực thi command giống như với các đối tượng SqlCommand khác.

3. Thực hành

3.1. Tạo Stored Procedure

- Mở Microsoft SQL Server Management Studio và kết nối vào server.
- Thêm một cơ dữ liệu mới tên là QuanLiHocSinh



- Tạo một bảng có tên là HOCSINH với các thuộc tính sau

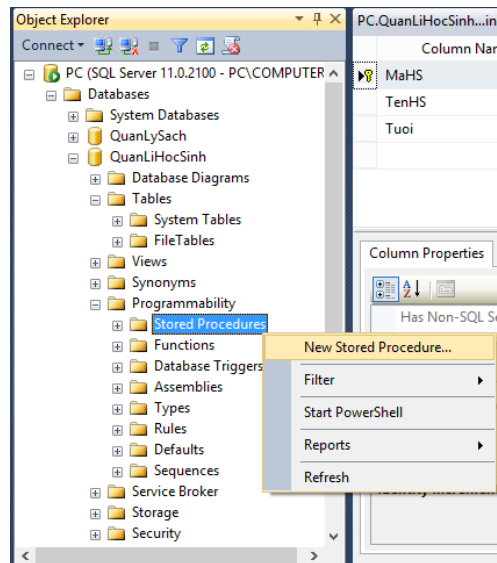
PC.QuanLiHocSinh...inh - dbo.HOCSINH			
	Column Name	Data Type	Allow Nulls
	MaHS	int	<input type="checkbox"/>
	TenHS	nvarchar(50)	<input type="checkbox"/>
	Tuoi	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>



Lưu ý: Trường **MaHS** ở đây thiết lập là khóa chính, tự động tăng

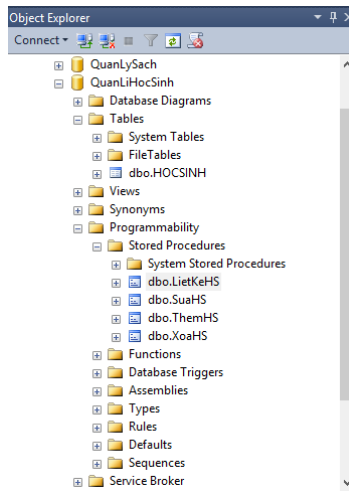
Has Non-SQL Server Subscriber	No
Identity Specification	Yes
(Is Identity)	Yes
Identity Increment	1
Identity Seed	1

– Trong Object Explorer, xỏ Databases, xỏ CSDL ta muốn tạo Stored Procedure, xỏ Programmability

- Click phải vào Stored Procedures, chọn New Stored Procedure



- Ở đây ta có thể định nghĩa các Stored Procedure như sau:
- Để kiểm tra lỗi cú pháp của các câu lệnh, ta vào menu Query, click Parse, hoặc click vào biểu tượng  trên thanh toolbar
- Để tạo Stored Procedure, ta vào menu Query, click Execute, hoặc click vào biểu tượng  Execute trên thanh toolbar
- Để xem Stored Procedure đã được tạo ra chưa, ta click phải vào Stored Procedures và chọn Refresh



Mã code cho các Stored Procedure:

```
CREATE PROCEDURE LietKeHS
```

```
AS
BEGIN
    SELECT * FROM HOCSINH;
END
GO
```

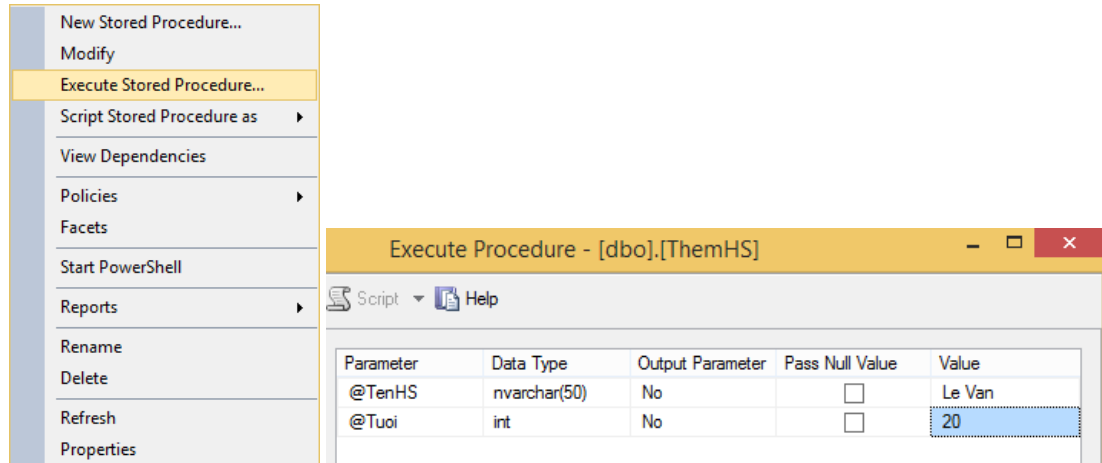
```
CREATE PROCEDURE SuaHS
    @MaHS int,
    @TenHS nvarchar(50),
    @Tuoi int = 0
AS
BEGIN
    UPDATE HOCSINH
    SET TenHS = @TenHS,
        Tuoi = @Tuoi
    WHERE MaHS = @MaHS;
END
GO
```

```
CREATE PROCEDURE ThemHS
    @TenHS nvarchar(50),
    @Tuoi int = 0
AS
BEGIN
    INSERT INTO HOCSINH (TenHS, Tuoi)
    VALUES(@TenHS, @Tuoi);
END
GO
```

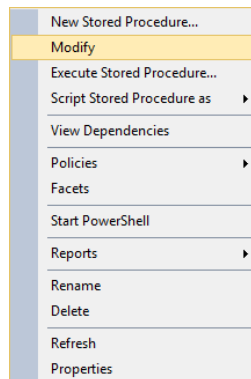
```
CREATE PROCEDURE XoaHS
    @MaHS int
AS
BEGIN
    DELETE HOCSINH
    WHERE MaHS = @MaHS;
END
GO
```

- Một số thao tác làm việc trên Stored Procedures:

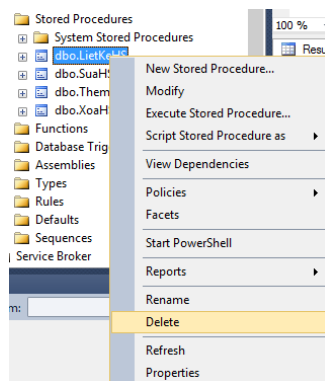
Để thực thi Stored Procedure nào thì ta click phải vào Stored Procedure đó và chọn **Execute Stored Procedure**. Nếu thủ tục có tham số đầu vào thì sẽ xuất hiện hộp thoại cho ta nhập tham số đầu vào và nhấn OK.



Sửa Stored Procedure: Nếu cảm thấy không hài lòng về đoạn code của Stored Procedure ta có thể sửa lại. Click phải vào Stored Procedure muốn sửa và chọn Modify, sau đó ta sửa lại nội dung Stored Procedure trong Query Editor và click **Execute** để cập nhật nội dung thủ tục.

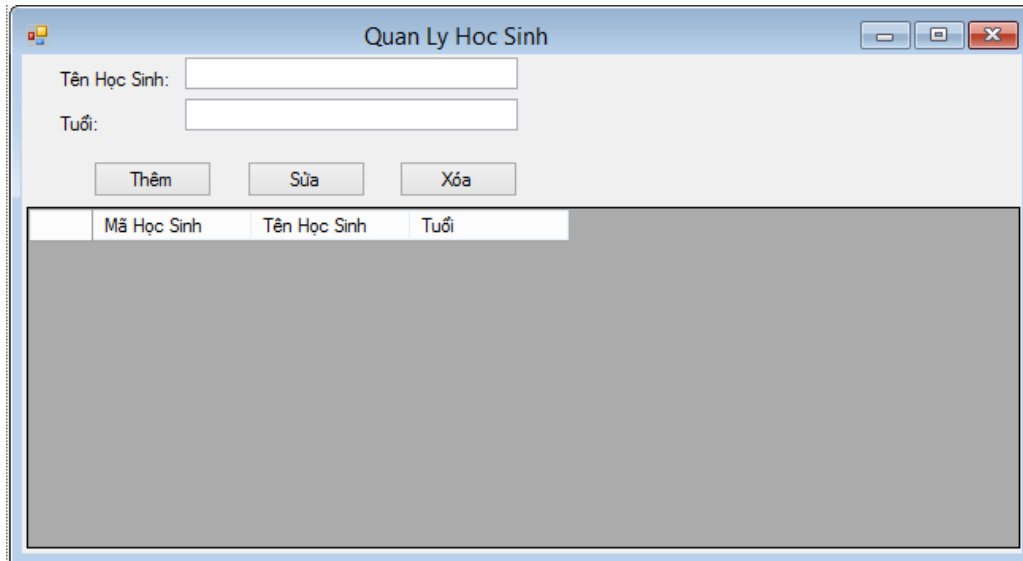


Xóa Stored Procedure: Click phải vào Stored Procedure muốn xóa và chọn Delete



3.2. Tạo ứng dụng:

Bước 1: Thiết kế giao diện như sau



Bước 2: Viết hàm LoadData() để load dữ liệu lên DataGridView

```
public void LoadData()
{
    SqlConnection connection = new SqlConnection();
    connection.ConnectionString = @"Data Source=COMPUTER\SQLSERVER2012;Initial
    Catalog=QuanLiHocSinh;Integrated Security=True";
    connection.Open();

    SqlCommand command = new SqlCommand("LietKeHS", connection);
    command.CommandType = CommandType.StoredProcedure;

    command.ExecuteNonQuery();

    SqlDataAdapter adapter = new SqlDataAdapter();
    adapter.SelectCommand = command;
    DataTable dt = new DataTable();
    adapter.Fill(dt);

    dataGridView.DataSource = dt;
    connection.Close();

    txtTenHS.Text = Convert.ToString(dataGridView.CurrentRow.Cells["TenHS"].Value);
    txtTuoi.Text = Convert.ToString(dataGridView.CurrentRow.Cells["Tuoi"].Value);
}
```

Bước 3: Viết sự kiện Load cho Form

```
private void frmQuanLiHocSinh_Load(object sender, EventArgs e)
{
    dataGridView.AutoGenerateColumns = false;
    dataGridView.Columns[0].DataPropertyName = "MaHS";
    dataGridView.Columns[1].DataPropertyName = "TenHS";
    dataGridView.Columns[2].DataPropertyName = "Tuoi";
    LoadData();
}
```

Bước 4: Viết sự kiện CellClick cho DataGridView

```
private void dataGridView_CellClick(object sender, DataGridViewCellEventArgs e)
{
    if (e.RowIndex >= 0 && e.ColumnIndex >= 0)
    {
        txtTenHS.Text = Convert.ToString(dataGridView.CurrentRow.Cells["TenHS"].Value);
        txtTuoi.Text = Convert.ToString(dataGridView.CurrentRow.Cells["Tuoi"].Value);
    }
}
```

Bước 5: Viết sự kiện Click cho nút Thêm

```
private void btnThem_Click(object sender, EventArgs e)
{
    SqlConnection connection = new SqlConnection();
    connection.ConnectionString = @"Data Source=PC;Initial Catalog=QuanLiHocSinh;Integrated
    Security=True";
    connection.Open();

    SqlCommand command = new SqlCommand("ThemHS", connection);
    command.CommandType = CommandType.StoredProcedure;

    SqlParameter p = new SqlParameter("@TenHS", txtTenHS.Text);
    command.Parameters.Add(p);
    p = new SqlParameter("@Tuoi", txtTuoi.Text);
    command.Parameters.Add(p);

    int count = command.ExecuteNonQuery();

    if (count > 0)
    {
        MessageBox.Show("Thêm thành công");
        LoadData();
    }
    connection.Close();
}
```

Bước 6: Viết sự kiện Click cho nút Sửa

```
private void btnSua_Click(object sender, EventArgs e)
{
    SqlConnection connection = new SqlConnection();
    connection.ConnectionString = @"Data Source=PC;Initial Catalog=QuanLiHocSinh;Integrated
        Security=True";
    connection.Open();

    SqlCommand command = new SqlCommand("SuaHS", connection);
    command.CommandType = CommandType.StoredProcedure;

    int id = (int)dataGridView.CurrentRow.Cells["MaHS"].Value;

    SqlParameter p = new SqlParameter("@MaHS", id);
    command.Parameters.Add(p);
    p = new SqlParameter("@TenHS", txtTenHS.Text);
    command.Parameters.Add(p);
    p = new SqlParameter("@Tuoi", txtTuoi.Text);
    command.Parameters.Add(p);

    int count = command.ExecuteNonQuery();

    if (count > 0)
    {
        MessageBox.Show("Sửa thành công");
        LoadData();
    }
    connection.Close();
}
```

Bước 7: Viết sự kiện Click cho nút Xóa

```
private void btnXoa_Click(object sender, EventArgs e)
{
    SqlConnection connection = new SqlConnection();
    connection.ConnectionString = @"Data Source=PC;Initial Catalog=QuanLiHocSinh;Integrated
        Security=True";
    connection.Open();

    SqlCommand cmd = new SqlCommand("XoaHS", connection);
    cmd.CommandType = CommandType.StoredProcedure;

    int id = (int)dataGridView.CurrentRow.Cells["MaHS"].Value;

    SqlParameter p = new SqlParameter("@MaHS", id);
    cmd.Parameters.Add(p);

    int count = cmd.ExecuteNonQuery();

    if (count > 0)
    {
        MessageBox.Show("Xóa thành công!");
        LoadData();
    }
    connection.Close();
}
```

Bước 8: Chạy thử chương trình