

**PHÁT TRIỂN ỨNG DỤNG CSDL 1**

**Tháng 9/2014**

# **LẬP TRÌNH TRUY XUẤT CƠ SỞ DỮ LIỆU VỚI C# VÀ ADO.NET**

Tóm tắt nội dung bài thực hành:

Xây dựng được một ứng dụng (C#) window form  
kết nối CSDL với các chức năng cơ bản

Bộ môn **Hệ thống thông tin**

Khoa Công nghệ thông tin

ĐH Khoa học tự nhiên TP HCM

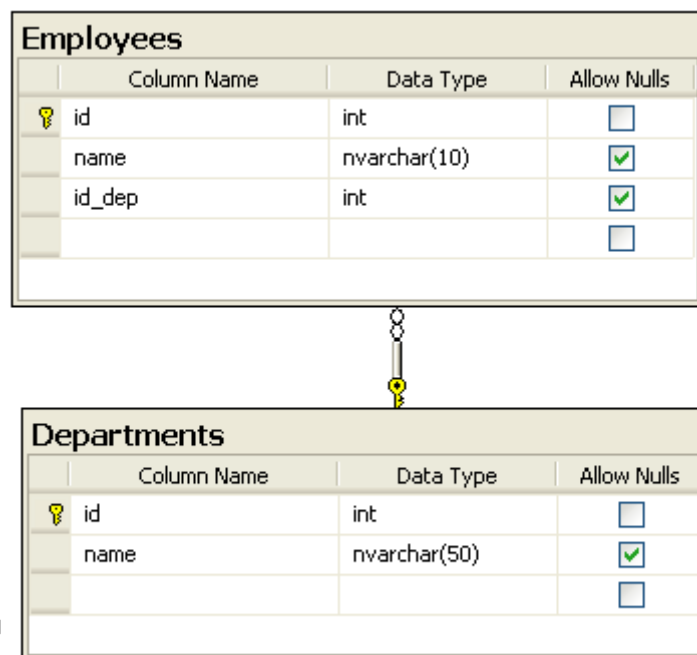


## MỤC LỤC

<b>1</b>	<b>Mục tiêu .....</b>	<b>1</b>
<b>2</b>	<b>Hướng dẫn cụ thể.....</b>	<b>1</b>
2.1	Một số lớp hỗ trợ kết nối đến CSDL cơ bản.....	1
2.2	Các bước thực hiện kết nối đến CSDL .....	2
2.3	Hướng dẫn xây dựng ứng dụng có chức năng THÊM dữ liệu: Chức năng Thêm phòng ban .....	2
2.3.1	Thiết kế giao diện: .....	2
2.3.2	Cài đặt xử lý sự kiện Click cho button btnInsert .....	2
2.3.3	Giải thích.....	3
2.3.4	Hướng dẫn lấy chuỗi Connection String.....	4
2.4	Hướng dẫn xây dựng ứng dụng có chức năng ĐỌC, CẬP NHẬT, XOÁ dữ liệu: Chức năng Đọc, Cập nhật, Xoá Phòng ban .....	6
2.4.1	Thiết kế giao diện: .....	6
2.4.2	Cài đặt xử lý Đọc, Cập nhật và Xoá .....	6
2.4.3	Giải thích.....	8
2.5	Hướng dẫn xây dựng ứng dụng có chức năng TÌM KIẾM dữ liệu: Chức năng Tìm kiếm phòng ban .....	9
2.5.1	Thiết kế giao diện: .....	9
2.5.2	Cài đặt sự kiện Click cho btnSearch:.....	9
2.5.3	Giải thích:.....	10

## 1 Mục tiêu

- Xây dựng ứng dụng kết nối CSDL cơ bản với các chức năng:
  - Thêm dữ liệu
  - Đọc dữ liệu
  - Xoá
  - Cập nhật
  - Tìm kiếm
  - Hiển thị dữ liệu lên combobox, GridView
- CSDL quản lý nhân viên sử dụng cho các ví dụ:



## 2 Hướng dẫn cụ thể

### 2.1 Một số lớp hỗ trợ kết nối đến CSDL cơ bản

<b>SqlConnection</b>	Lớp hỗ trợ tạo đối tượng kết nối đến CSDL
<b>SqlCommand</b>	Lớp hỗ trợ thực thi câu lệnh SQL hoặc các Stored Procedure
<b>SqlError</b>	Lưu trữ thông tin về lỗi và cảnh cáo (warning)

**SqlException** Các ngoại lệ trong trường hợp SQL Server lỗi và cảnh báo

**SqlReader** Cung cấp một data stream để đọc dữ liệu từ CSDL

## 2.2 Các bước thực hiện kết nối đến CSDL

Bước 1: Tạo đối tượng kết nối đến CSDL & mở kết nối

Bước 2: Xây dựng câu truy vấn SQL để thực hiện thao tác mong muốn (Thêm, xóa, sửa, cập nhật)

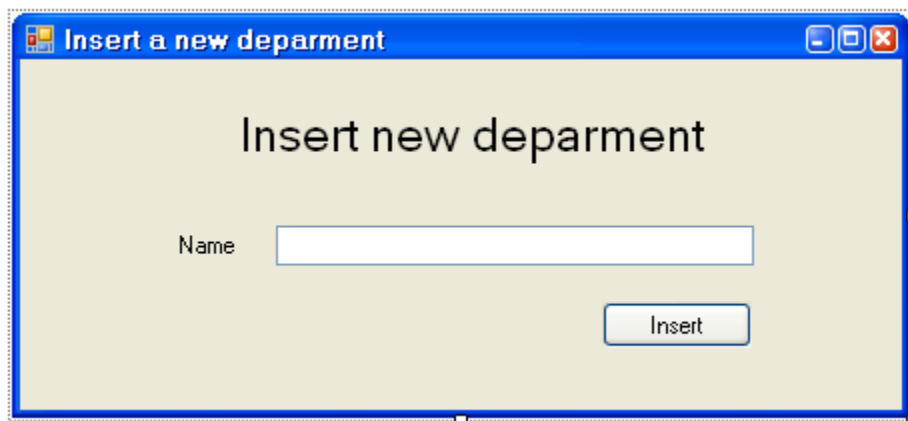
Bước 3: Tạo đối tượng command để thực thi câu truy vấn

Bước 4: Thực thi câu truy vấn và xử lý kết quả trả về

Bước 5: Đóng kết nối

## 2.3 Hướng dẫn xây dựng ứng dụng có chức năng THÊM dữ liệu: Chức năng Thêm phòng ban

### 2.3.1 Thiết kế giao diện:



- txtName: TextBox → Tên phòng ban
- btnSearch: Button → Button để thực hiện thêm phòng ban

### 2.3.2 Cài đặt xử lý sự kiện Click cho button btnInsert

```
1 using System;  
2 using System.Collections.Generic;  
3 using System.ComponentModel;  
4 using System.Data;  
5 using System.Drawing;  
6 using System.Linq;  
7 using System.Text;
```

```

8 using System.Windows.Forms;
9 using System.Data.SqlClient;
10
11 namespace DemoCRUD_EmployeeManagement
12 {
13     public partial class Form1 : Form
14     {
15         SqlConnection _connection = null;
16         SqlCommand _command = null;
17         String _connectionString = "";
18
19         public Form1()
20         {
21             InitializeComponent();
22
23             _connectionString = @"Data Source=B44-WS32\MISASME2010;Initial
24 Catalog=EmployeeDB;Integrated Security=True";
25         }
26
27         private void btnInsert_Click(object sender, EventArgs e)
28         {
29             //Bước 1: Tạo đối tượng kết nối để CSDL & mở kết nối
30             _connection = new SqlConnection(_connectionString);
31             _connection.Open();
32
33             //Bước 2: Xây dựng câu lệnh SQL để thực hiện chức năng mong muốn
34             String sql = "INSERT INTO Departments VALUES(N'" + txtName.Text +
35             "')";
36
37             //Bước 3: Tạo đối tượng thực thi câu lệnh SQL
38             _command = new SqlCommand(sql);
39             _command.Connection = _connection;
40
41             //Bước 3: Thực thi câu SQL và lấy kết quả trả về
42             int n = _command.ExecuteNonQuery();
43             if (n > 0)
44                 MessageBox.Show("Add successfully !!!");
45             else
46                 MessageBox.Show("Error !!!");
47             //Bước
48             _connection.Close();
49         }
50     }
51 }

```

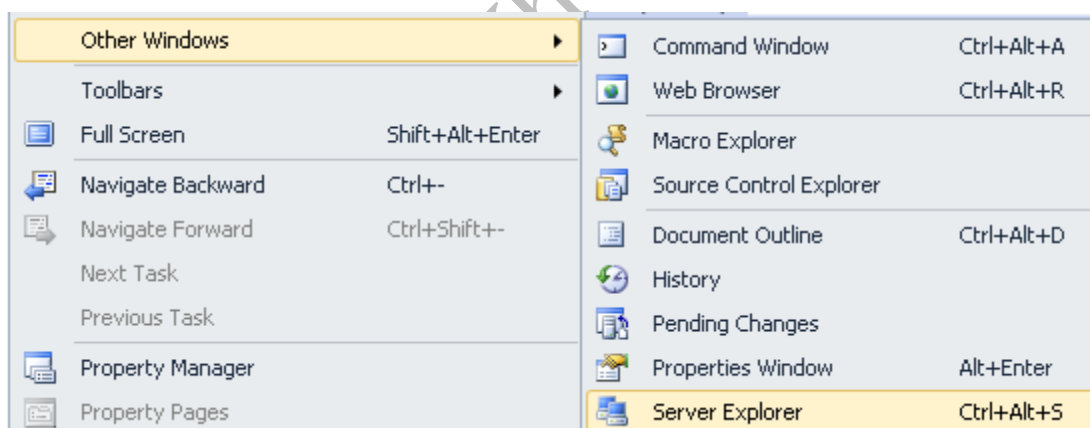
### 2.3.3 Giải thích

<code>using System.Data.SqlClient;</code>	Namespace chứa các lớp kết nối đến CSDL
<code>connectionString = @"Data Source=B44-WS32\MISASME2010;Initial Catalog=EmployeeDB;Integrated Security=True";</code>	Mô tả thông tin kết nối đến CSDL. Có thể xác định chuỗi này theo hướng dẫn trong phần <b>2.3.4</b>
<code>//Bước 1: Tạo đối tượng kết nối để CSDL &amp; mở kết nối _connection = new SqlConnection(_connectionString);</code>	

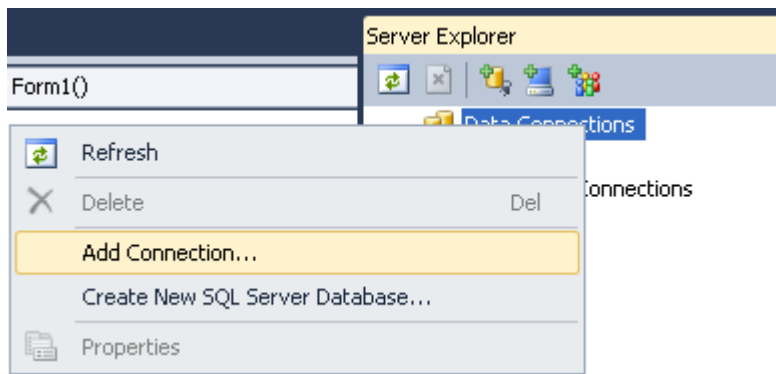
<code>_connection.Open();</code>	
<b>//Bước 2: Xây dựng câu lệnh SQL để thực hiện chức năng mong muốn</b> <code>String sql = "INSERT INTO Departments VALUES(N'" + txtName.Text + "')";</code>	Tạo câu lệnh SQL thực hiện chức năng thêm. Lưu ý câu lệnh này phải tuyệt đối tuân theo cú pháp của ngôn ngữ SQL, các bảng, thuộc tính sử dụng phải có trong CSDL
<b>//Bước 3: Tạo đối tượng thực thi câu lệnh SQL</b> <code>_command = new SqlCommand(sql);</code> <code>_command.Connection = _connection;</code>	Tạo đối tượng thực thi câu lệnh. Có thể sử dụng một lệnh để tạo đối tượng Command như sau: <code>_command = new SqlCommand(sql, _connection);</code>
<b>//Bước 4: Thực thi câu SQL và lấy kết quả trả về</b> <code>int n = _command.ExecuteNonQuery();</code> <code>if (n &gt; 0)</code> <code>    MessageBox.Show("Add successfully !!!");</code> <code>else</code> <code>    MessageBox.Show("Error !!!");</code>	Thực thi câu lệnh SQL. Trong trường hợp này, n là số dòng được thêm vào trong CSDL. Trong các chức năng cập nhật và xóa thì n là số dòng được cập nhật hoặc bị xóa đi. Có thể dùng n để kiểm tra việc thêm vào thành công hay không.

#### 2.3.4 Hướng dẫn lấy chuỗi Connection String

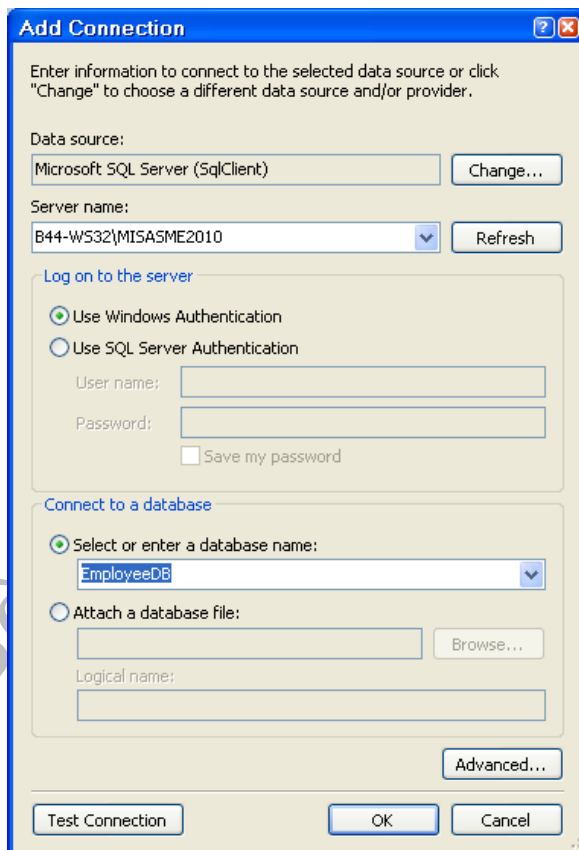
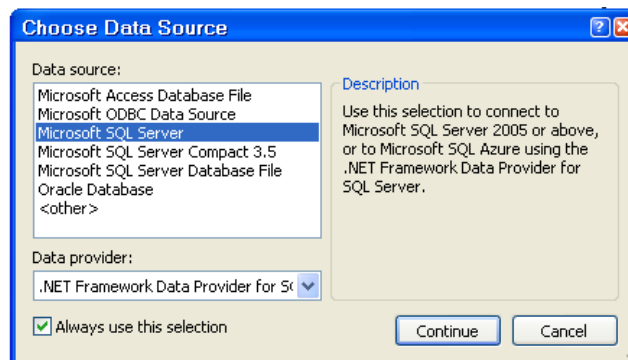
B1: Chọn View → Other Windows → Server Explorer hoặc View→Server Explorer



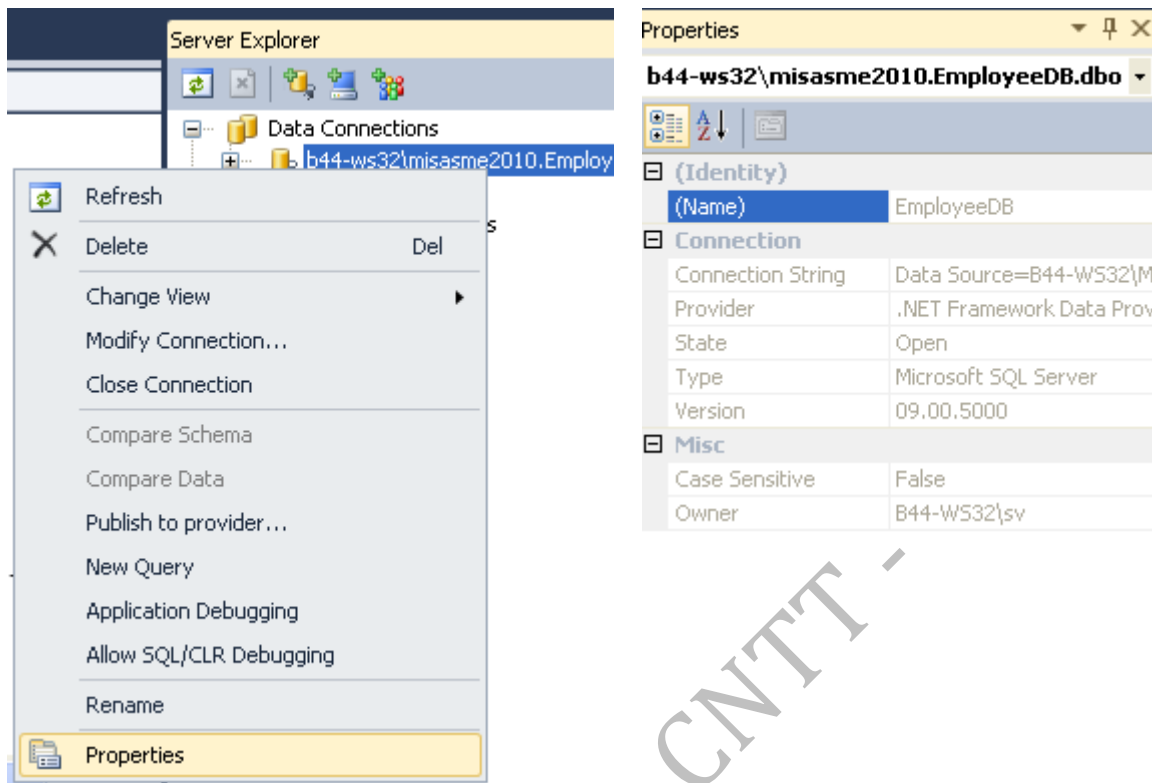
B2: Thêm một kết nối đến CSDL mong muốn: Data Connections → Add Connection



### B3: Kết nối vào CSDL SQL Server



B4: Lấy chuỗi kết nối để sử dụng (Copy / Paste)



## 2.4 Hướng dẫn xây dựng ứng dụng có chức năng ĐỌC, CẬP NHẬT, XOÁ dữ liệu: Chức năng Đọc, Cập nhật, Xoá Phòng ban

### 2.4.1 Thiết kế giao diện:

The screenshot shows a Windows Form titled 'Frm\_ReadUpdateDelete'. The form has a title bar with standard Windows controls. The main area contains the text 'Read, Update and Delete Deparments'. Below this text, there are two input fields: 'ID' and 'Name'. The 'ID' field is a text box, and the 'Name' field is a text box with a dropdown arrow on the right. To the right of the 'ID' field is a 'Read' button. Below the 'Name' field are two buttons: 'Update' and 'Delete'.

### 2.4.2 Cài đặt xử lý Đọc, Cập nhật và Xoá

```
1 using System;  
2 using System.Collections.Generic;
```



```

3 using System.ComponentModel;
4 using System.Data;
5 using System.Drawing;
6 using System.Linq;
7 using System.Text;
8 using System.Windows.Forms;
9 using System.Data.SqlClient;
10
11 namespace DemoCRUD_EmployeeManagement
12 {
13     public partial class Frm_ReadUpdateDelete : Form
14     {
15         SqlConnection _connection = null;
16         SqlCommand _command = null;
17         String _connectionString = "";
18         public Frm_ReadUpdateDelete()
19         {
20             InitializeComponent();
21             _connectionString = @"Data Source=B44-WS32\MISASME2010;Initial
22 Catalog=EmployeeDB;Integrated Security=True";
23         }
24         private void btnRead_Click(object sender, EventArgs e)
25         {
26             //Bước 1: Tạo đối tượng kết nối để CSDL & mở kết nối
27             _connection = new SqlConnection(_connectionString);
28             _connection.Open();
29             //Bước 2: Xây dựng câu lệnh SQL để thực hiện chức năng mong muốn
30             String sql = String.Format("SELECT * FROM Departments where id={0}",
31 txtID.Text);
32             //Bước 3: Tạo đối tượng thực thi câu lệnh SQL
33             _command = new SqlCommand(sql);
34             _command.Connection = _connection;
35             //Bước 4: Thực thi câu SQL và lấy kết quả trả về
36             SqlDataReader reader = _command.ExecuteReader();
37             if (reader.Read())
38             {
39                 txtName.Text = (String)reader["name"];
40             }
41             else
42                 MessageBox.Show("Not found !!!");
43             reader.Close();
44             //Bước 5: Đóng kết nối
45             _connection.Close();
46         }
47
48         private void btnDelete_Click(object sender, EventArgs e)
49         {
50             //Bước 1: Tạo đối tượng kết nối để CSDL & mở kết nối
51             _connection = new SqlConnection(_connectionString);
52             _connection.Open();
53             //Bước 2: Xây dựng câu lệnh SQL để thực hiện chức năng mong muốn
54             String sql = String.Format("DELETE FROM Departments WHERE id={0}",
55 txtID.Text);
56             //Bước 3: Tạo đối tượng thực thi câu lệnh SQL
57             _command = new SqlCommand(sql);
58             _command.Connection = _connection;
59             //Bước 4: Thực thi câu SQL và lấy kết quả trả về
60             int n = _command.ExecuteNonQuery();
61         }
62     }

```

```

63         if (n > 0)
64         {
65             MessageBox.Show("Delete successfully !!!");
66             txtID.Text = "";
67         }
68         else
69             MessageBox.Show("No row deleted !!!");
70         //Bước 5: Đóng kết nối
71         _connection.Close();
72     }
73
74     private void btnUpdate_Click(object sender, EventArgs e)
75     {
76         //Bước 1: Tạo đối tượng kết nối để CSDL & mở kết nối
77         _connection = new SqlConnection(_connectionString);
78         _connection.Open();
79         //Bước 2: Xây dựng câu lệnh SQL để thực hiện chức năng mong muốn
80         String sql = String.Format("UPDATE Departments SET name=N'{0}' WHERE
81 id={1}", txtName.Text, txtID.Text);
82         //Bước 3: Tạo đối tượng thực thi câu lệnh SQL
83         _command = new SqlCommand(sql);
84         _command.Connection = _connection;
85         //Bước 4: Thực thi câu SQL và lấy kết quả trả về
86         int n = _command.ExecuteNonQuery();
87         if (n > 0)
88         {
89             MessageBox.Show("Update successfully: " + n.ToString() + " row
90 !!!");
91         }
92         else
93             MessageBox.Show("No row updated !!!");
94         //Bước 5: Đóng kết nối
95         _connection.Close();
96     }
97 }
98 }
99
100

```

### 2.4.3 Giải thích

```

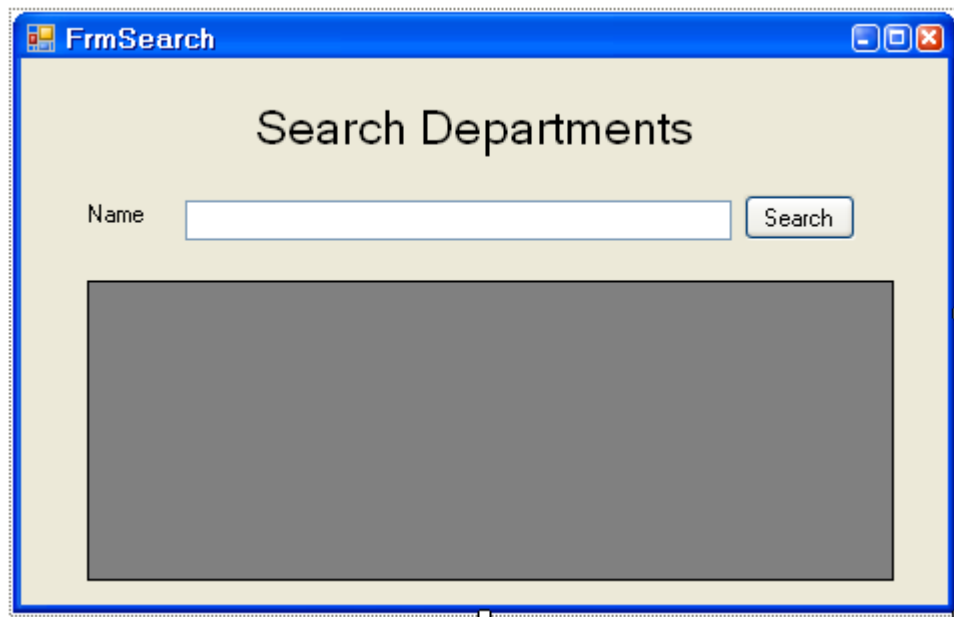
//Bước 4: Thực thi câu SQL và lấy kết quả trả
về
SqlDataReader reader = _command.ExecuteReader();
if (reader.Read())
{
    txtName.Text = (String)reader["name"];
}
else
    MessageBox.Show("Not found !!!");
reader.Close();

```

Thực thi câu truy vấn đọc dữ liệu, sau đó có thể đọc dữ liệu từ đối tượng của lớp. SqlDataReader. Hàm **reader.Read()** sẽ cho kết quả là **true** nếu có dữ liệu. Nếu có dữ liệu thì có thể lấy dòng dữ liệu hiện tại sử dụng đối tượng của lớp SqlDataReader: **txtName.Text = (String)reader["name"];**

## 2.5 Hướng dẫn xây dựng ứng dụng có chức năng TÌM KIẾM dữ liệu: Chức năng Tìm kiếm phòng ban

### 2.5.1 Thiết kế giao diện:



### 2.5.2 Cài đặt sự kiện Click cho btnSearch:

```
1 using System;
2 using System.Collections.Generic;
3 using System.ComponentModel;
4 using System.Data;
5 using System.Drawing;
6 using System.Linq;
7 using System.Text;
8 using System.Windows.Forms;
9 using System.Data.SqlClient;
10
11 namespace DemoCRUD_EmployeeManagement
12 {
13     public partial class FrmSearch : Form
14     {
15         SqlConnection _connection = null;
16         SqlCommand _command = null;
17         String _connectionString = "";
18
19         public FrmSearch()
20         {
21             InitializeComponent();
22             _connectionString = @"Data Source=B44-WS32\MISASME2010;Initial
23 Catalog=EmployeeDB;Integrated Security=True";
24         }
25
26         private void btnSearch_Click(object sender, EventArgs e)
27         {
28             //Bước 1: Tạo đối tượng kết nối để CSDL & mở kết nối
```

```

29         _connection = new SqlConnection(_connectionString);
30         _connection.Open();
31         //Bước 2: Xây dựng câu lệnh SQL để thực hiện chức năng mong muốn
32         String sql = String.Format("SELECT * FROM Departments where name LIKE
33 N'%{0}%'", txtKeyword.Text);
34         //Bước 3: Tạo đối tượng thực thi câu lệnh SQL
35         _command = new SqlCommand(sql);
36         _command.Connection = _connection;
37         //Bước 4: Thực thi câu SQL và lấy kết quả trả về
38         DataTable table = new DataTable();
39         table.Columns.Add("id", typeof(int));
40         table.Columns.Add("name", typeof(String));
41         SqlDataReader reader = _command.ExecuteReader();
42         DataRow row = null;
43         while (reader.Read())
44         {
45             row = table.NewRow();
46             row["name"] = reader["name"];
47             row["id"] = reader["id"];
48             table.Rows.Add(row);
49         }
50         reader.Close();
51
52         dtgResult.DataSource = table;
53         //Bước 5: Đóng kết nối
54         _connection.Close();
55     }
56 }
57 }

```

### 2.5.3 Giải thích:

<pre> DataTable table = new DataTable(); table.Columns.Add("id", typeof(int)); table.Columns.Add("name", typeof(String)); </pre>	Tạo đối tượng table để chứa dữ liệu đọc được
<pre> DataRow row = null; while (reader.Read()) {     row = table.NewRow();     row["name"] = reader["name"];     row["id"] = reader["id"];     table.Rows.Add(row); } </pre>	Đọc dữ liệu trong đối tượng của lớp SqlDataReader và đưa vào Table