



ĐẠI HỌC THÁI NGUYÊN
TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP
KHOA ĐIỆN TỬ
BỘ MÔN: CÔNG NGHỆ THÔNG TIN



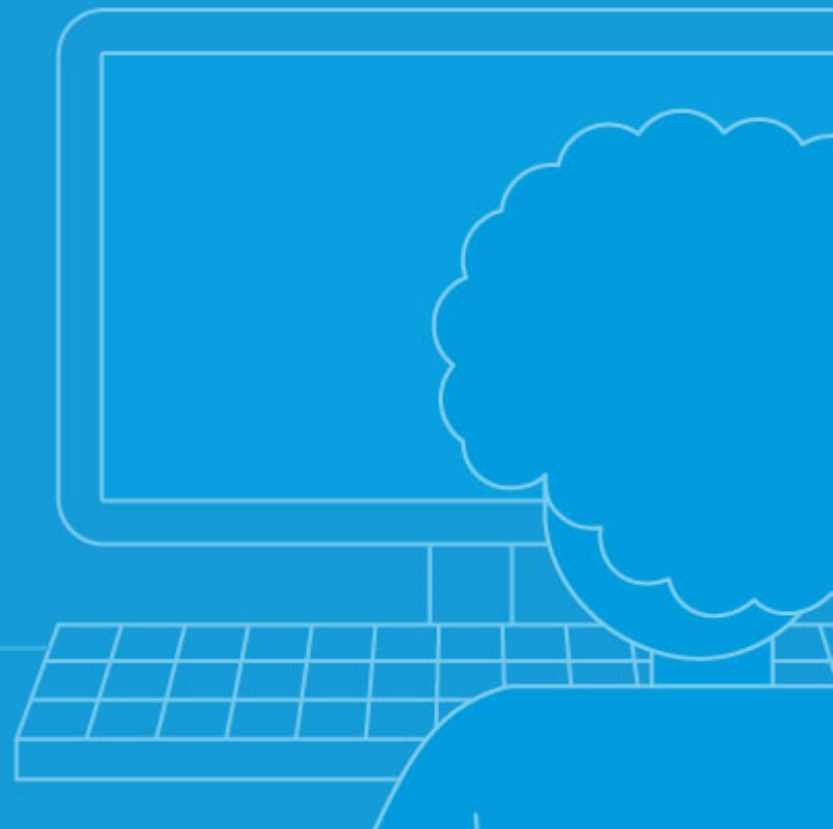
TÀI LIỆU THỰC HÀNH

LẬP TRÌNH TRONG MÔI TRƯỜNG WINDOW

MÃ HP: TEE320, SỐ TÍN CHỈ: 03



Thái Nguyên, 2024



❑ NỘI DUNG THỰC HÀNH

- ❖ BÀI 1: LẬP TRÌNH CONSOLE TRONG C#
- ❖ BÀI 2: LẬP TRÌNH ĐỐI TƯỢNG C#
- ❖ BÀI 3: LẬP TRÌNH WINDOW FORM C#
- ❖ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET
- ❖ BÀI 5: LẬP TRÌNH LINQ TRONG C#



✓ MỤC TIÊU

Cung cấp cho sinh viên kỹ năng về lập trình trên môi trường Window với Visual Studio.Net bao gồm:

- Hiểu và biết cách vận dụng lập trình hướng đối tượng với C#
- Viết các chương trình trên ứng dụng Console
- Xây dựng được các ứng dụng dạng Window form
- Xây dựng được các ứng dụng Window form có kết nối và giao tiếp với cơ sở dữ liệu

✓ YÊU CẦU VỚI SINH VIÊN

- Học và nắm bắt nội dung lý thuyết Bài giảng về lập trình trên môi trường window
- Chuẩn bị tài liệu thực hành trước khi thực hành
- Tham gia thực hành đầy đủ theo thời khóa biểu và hướng dẫn của giảng viên
- Thực hiện các nội quy, quy định của phòng thực hành
- Ngoài ra, sinh viên có thể thực hành thêm trên máy tính cá nhân tại nhà với các bài tập thực hành giao về nhà.



▪ Mục đích, yêu cầu

- Cài đặt và thiết lập môi trường lập trình C#;
- Sinh viên làm quen với môi trường và ngôn ngữ lập trình C#, các thành phần cơ bản, các kiểu dữ liệu và các phép toán trong C#;
- Viết chương trình trên ứng dụng Console để giải các bài toán cụ thể.

▪ Các bài tập vận dụng

1.1. Cài đặt và khởi tạo môi trường lập trình C#

❖ Tải phần mềm **Visual Studio Code**

- [Visual Studio Code November 2022](#)
- [Visual Studio Code November 2023](#)
- [Visual Studio Code March 2024](#)

❖ Cài đặt phần mềm

- [Hướng dẫn cài đặt phiên bản Visual Studio 2019](#)
- [Link1: Hướng dẫn cài đặt phiên bản Visual Studio 2022](#)
- [Link2: Hướng dẫn cài đặt phiên bản Visual Studio 2022](#)

❖ Khởi tạo ứng dụng

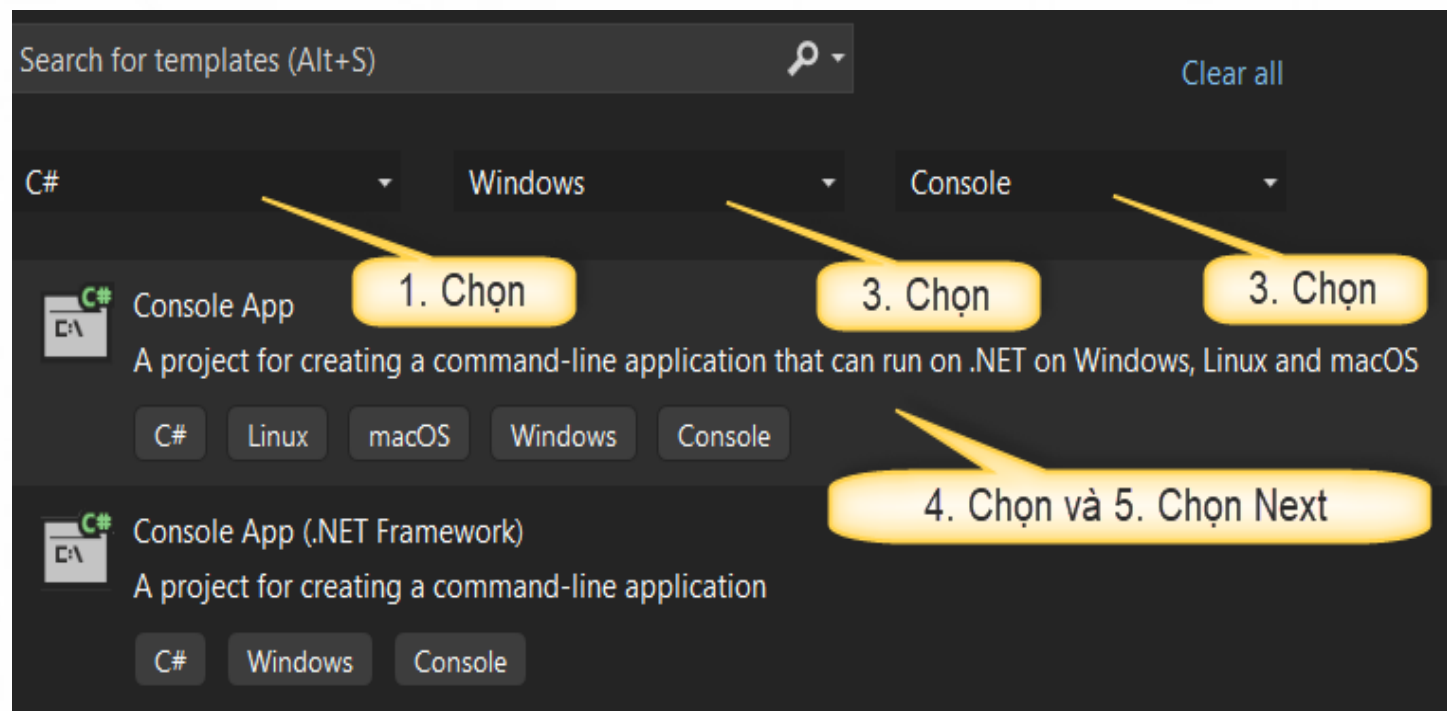
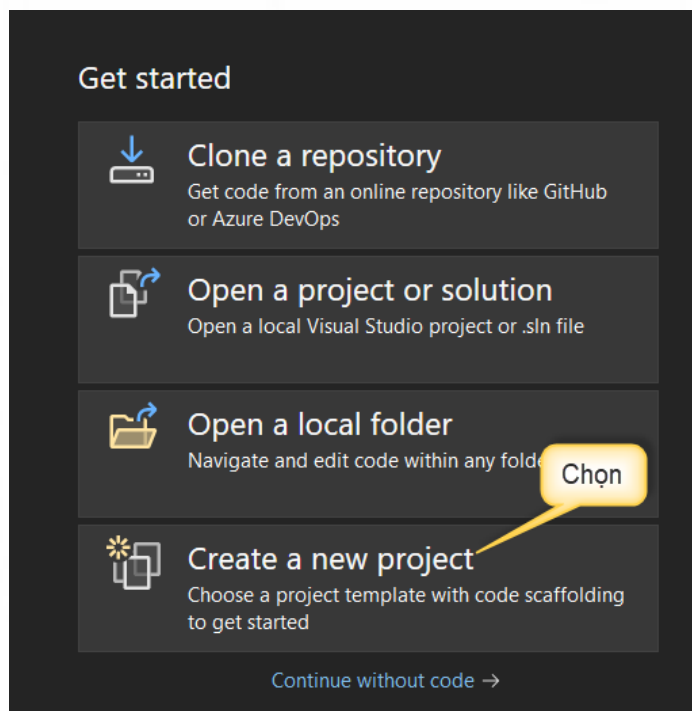
- Ứng dụng Console
- Ứng dụng Desktop

❑ BÀI 1: LẬP TRÌNH CONSOLE TRONG C#

1.1. Cài đặt và khởi tạo môi trường lập trình C#

❑ Khởi tạo ứng dụng Console

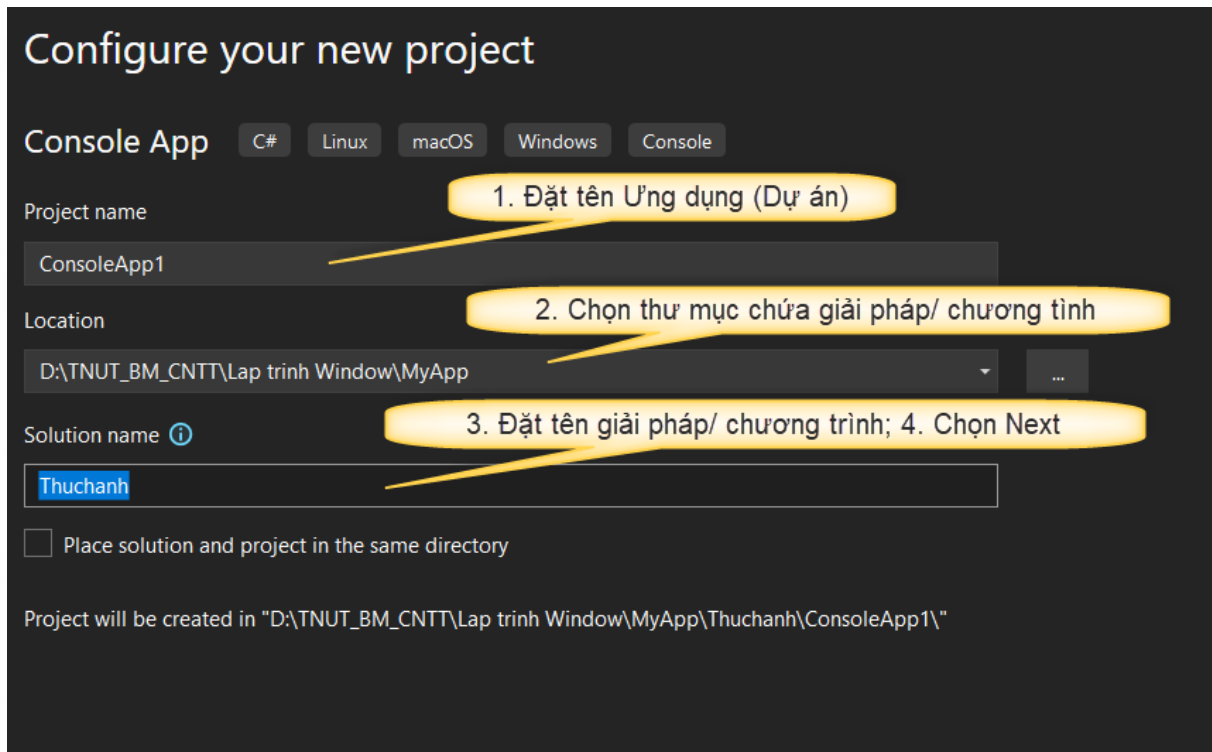
- Khởi động Visual Studio 2022/Crete new project;
- Chọn ngôn ngữ lập trình C#, hệ điều hành Window, ứng dụng Console
- Chọn thư viện Framework



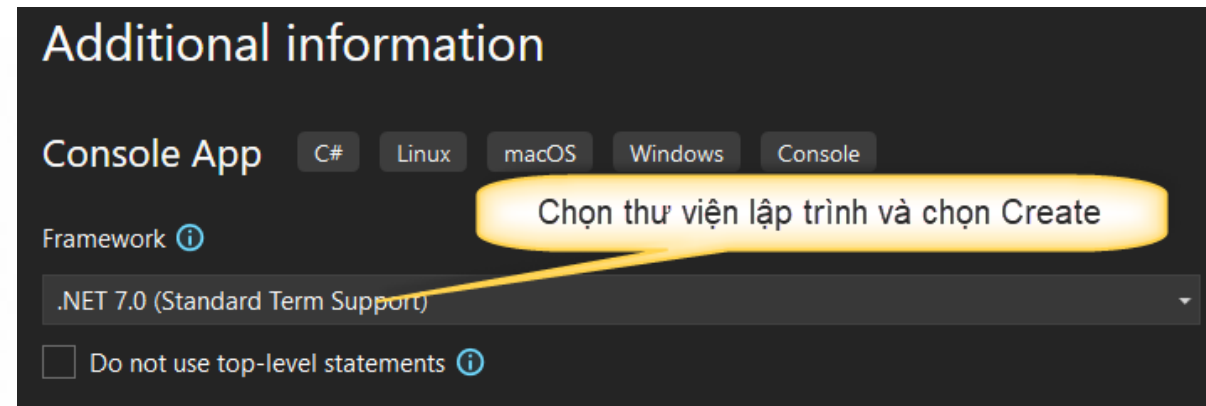
❑ BÀI 1: LẬP TRÌNH CONSOLE TRONG C#

1.1. Cài đặt và khởi tạo môi trường lập trình C#

❑ Khởi tạo ứng dụng Console



Hình: 3

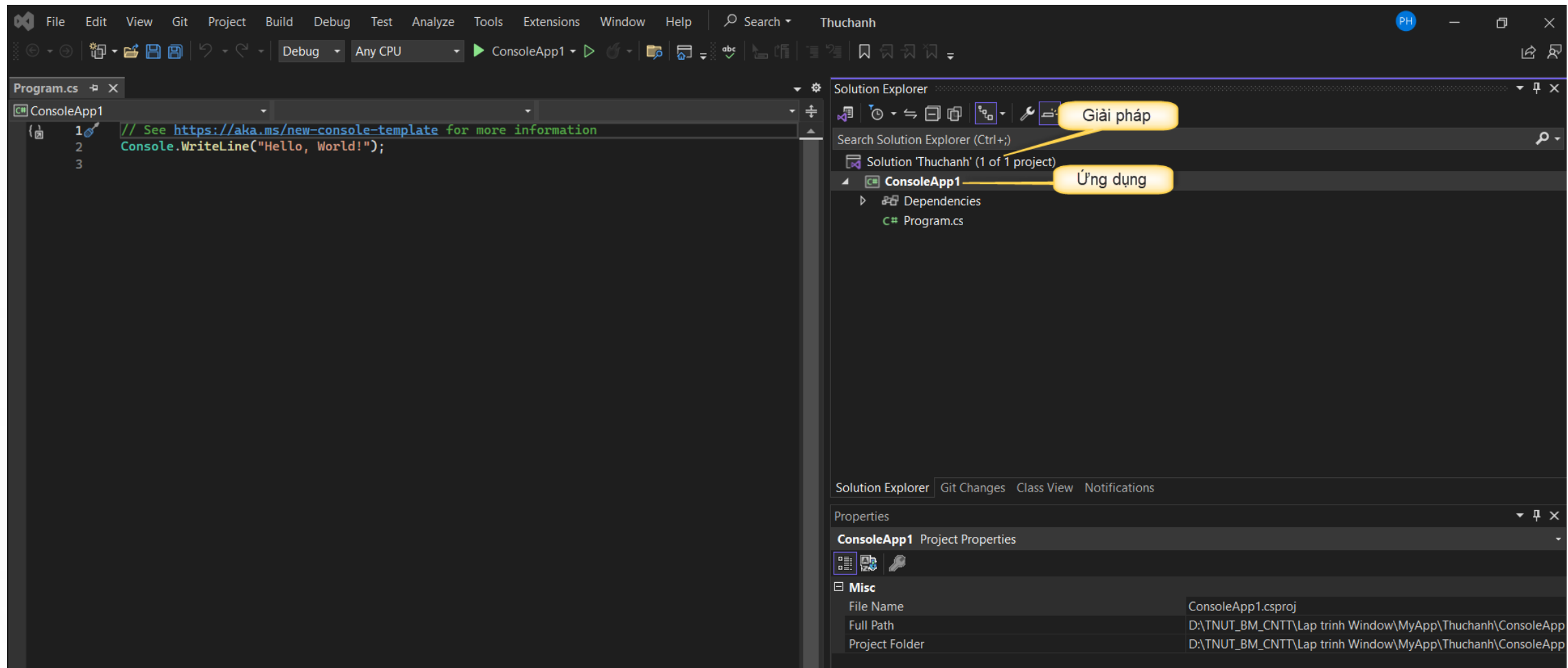


Hình: 4

❑ BÀI 1: LẬP TRÌNH CONSOLE TRONG C#

1.1. Cài đặt và khởi tạo môi trường lập trình C#

❑ Khởi tạo ứng dụng Console



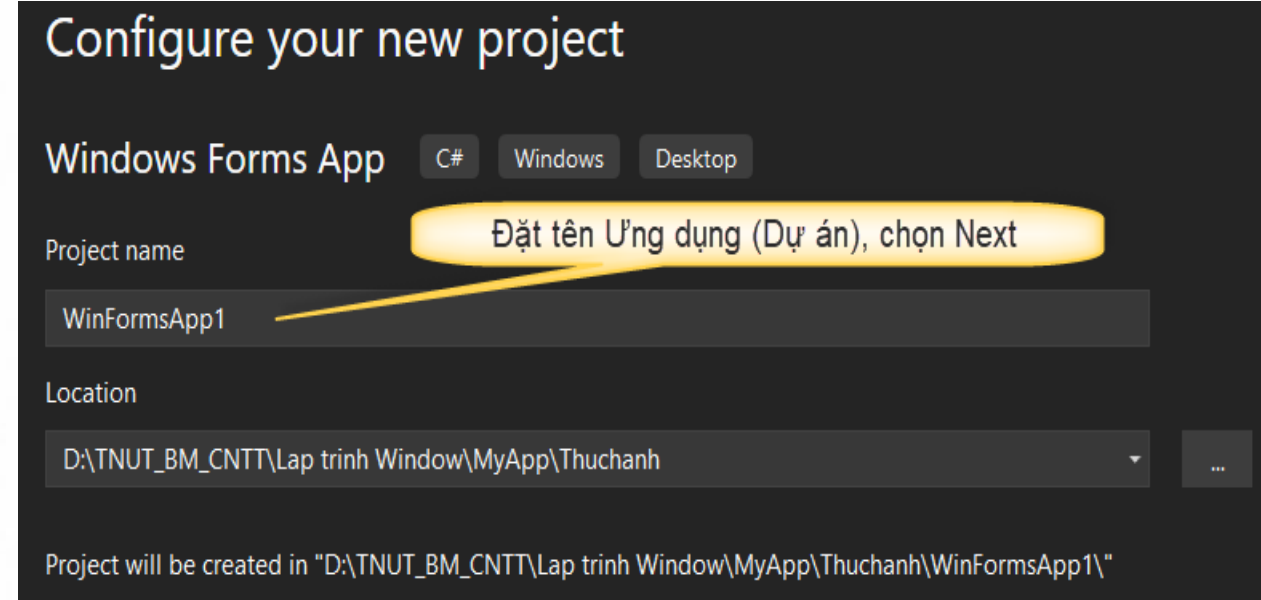
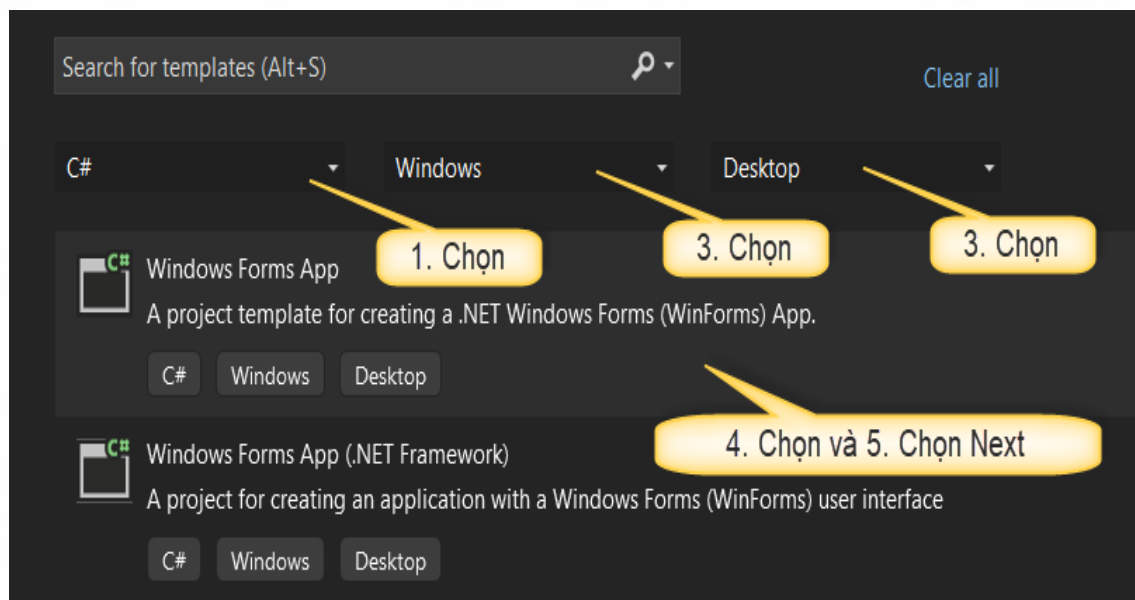
Hình 5: Kết quả tạo ứng dụng Console

❑ BÀI 1: LẬP TRÌNH CONSOLE TRONG C#

1.1. Cài đặt và khởi tạo môi trường lập trình C#

❑ Khởi tạo ứng dụng Desktop

- **Cách 1:** Thêm dự án/ ứng dụng vào giải pháp đã có sẵn
- Phải chuột ***Solution Thuchanh/Add/New Project***, đặt tên cho dự án **WinFormsApp1**
- Chọn ngôn ngữ lập trình C#, hệ điều hành Window, ứng dụng Desktop

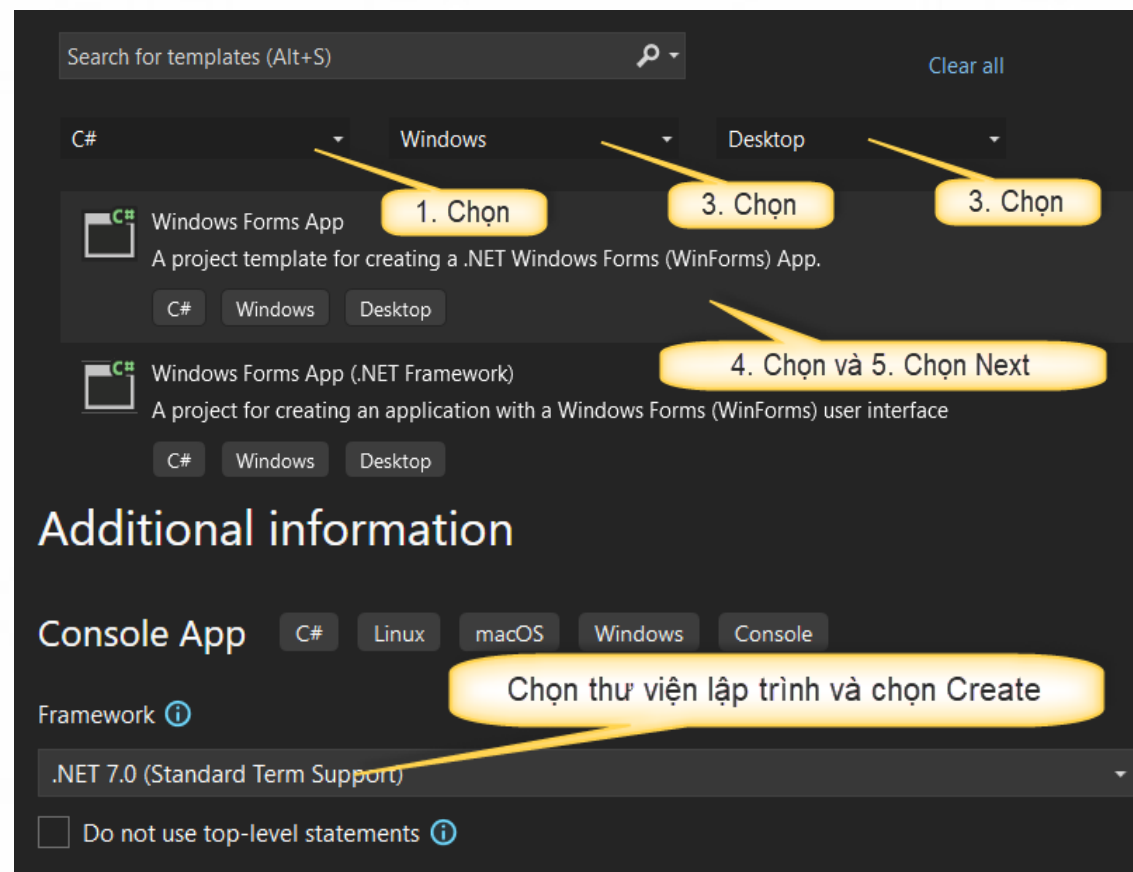
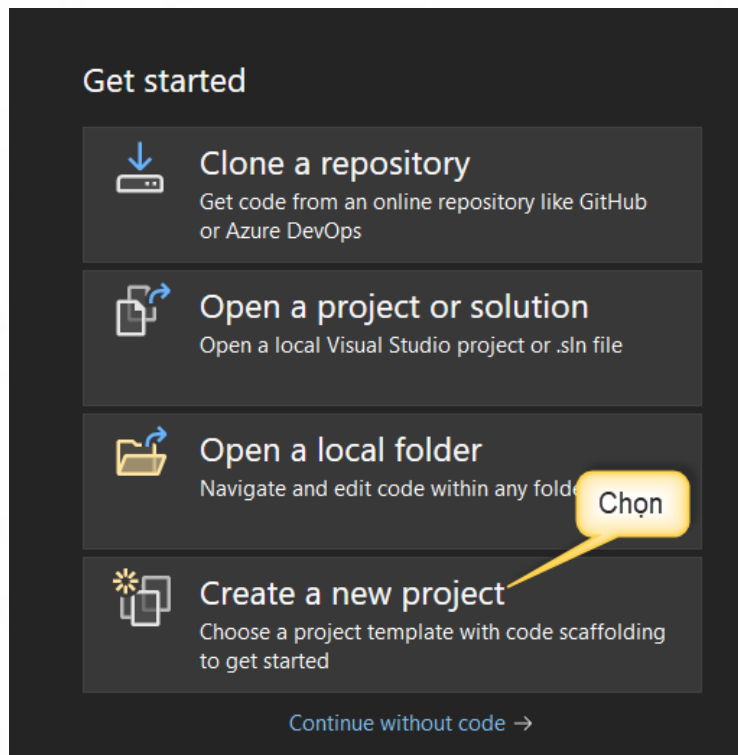


❑ BÀI 1: LẬP TRÌNH CONSOLE TRONG C#

1.1. Cài đặt và khởi tạo môi trường lập trình C#

❑ Khởi tạo ứng dụng Desktop

- **Cách 1:** Phải chuột ***Solution Thuchanh/Add/New Project***, đặt tên cho dự án **WinFormsApp1**
- Chọn ngôn ngữ lập trình C#, hệ điều hành Window, ứng dụng Desktop
- Chọn thư viện Framework

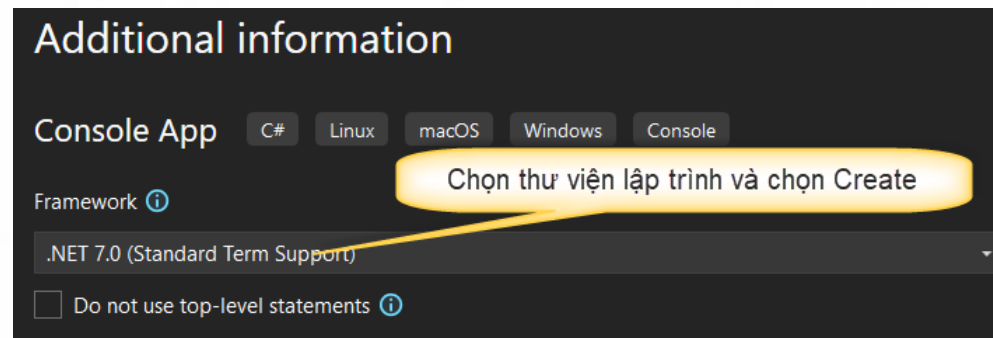
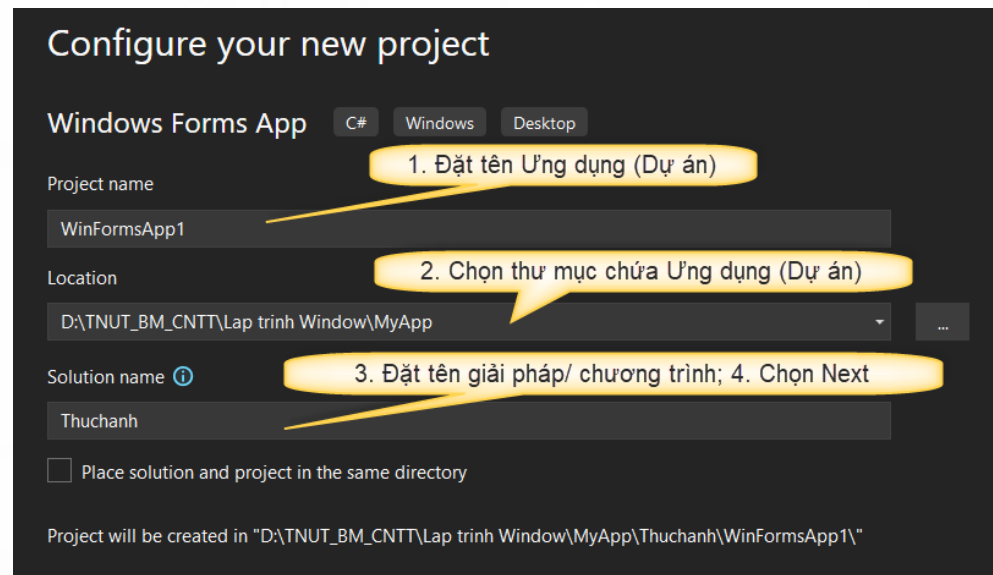
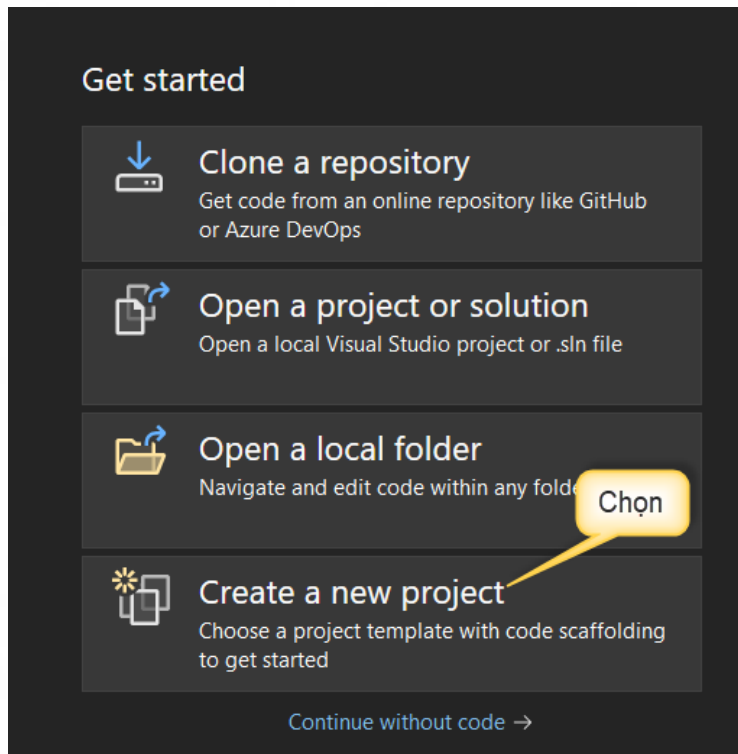


❑ BÀI 1: LẬP TRÌNH CONSOLE TRONG C#

1.1. Cài đặt và khởi tạo môi trường lập trình C#

❑ Khởi tạo ứng dụng Desktop

- **Cách 2:** Khởi tạo ứng dụng *Visual Studio/Create a new project*
- Chọn ngôn ngữ lập trình C#, hệ điều hành Window, ứng dụng Desktop
- Chọn thư viện Framework



❑ BÀI 1: LẬP TRÌNH CONSOLE TRONG C#

1.1. Cài đặt và khởi tạo môi trường lập trình C#

❑ Khởi tạo ứng dụng Desktop

The screenshot displays the Visual Studio IDE with a solution named 'Thuchanh' containing two projects: 'ConsoleApp1' and 'WinFormsApp1'. The 'WinFormsApp1' project is selected, and its 'Form1' is visible in the design view on the left. The Solution Explorer on the right shows the project structure with callouts: 'Ứng dụng 1: Console' pointing to 'ConsoleApp1' and 'Ứng dụng 1: Window Form (Desktop)' pointing to 'WinFormsApp1'. The Properties window at the bottom right shows the properties for 'Form1' (System.Windows.Forms.Form), including 'Font' (Segoe UI, 9pt), 'Text' (Form1), and 'Text' (The text associated with the control).

Property	Value
BackgroundImageLayout	Tile
Cursor	Default
Font	Segoe UI, 9pt
ForeColor	ControlText
FormBorderStyle	Sizable
RightToLeft	No
RightToLeftLayout	False
Text	Form1

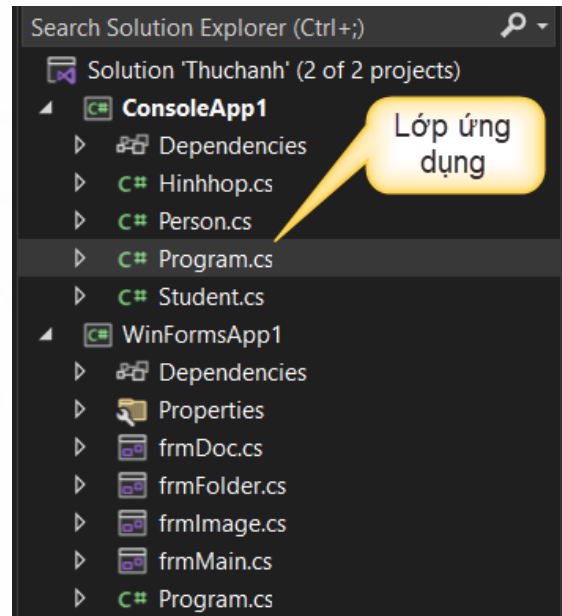
❑ BÀI 1: CÀI ĐẶT VÀ THIẾT LẬP MÔI TRƯỜNG LẬP TRÌNH

1.2. Các bài tập ứng dụng trên Console

Bài mẫu: Viết hàm (phương thức) `M01()` tính tổng, hiệu, tích, thương của 2 số. Thương làm tròn đến 2 số thập phân.

a) Cấu trúc chương trình

- **Program:** Lớp ứng dụng
- **Hàm `Main()`:** Hàm chính, mặc định chạy đầu tiên trong ứng dụng
- **`MAU01()`:** Hàm tự định nghĩa



```
namespace ConsoleApp1
{
    0 references
    internal static class Program
    {
        0 references
        static void Main()
        {
            try
            {
                // ....Khai báo biến, hằng.....
                // ....nhập giá trị cho biến.....
                // ....gọi hàm MAU01() để tính toán và hiển thị kết quả.....
                Console.ReadKey();
            }
            catch (Exception ex)
            {
                Console.Write(ex.ToString()); // Hiển thị lỗi nếu có
            }
        }

        //Xây dựng hàm
        0 references
        public static double MAU01(int a, int b, string pheptinh)
        {
            double value = 0;
            //Code nội dung hàm
            return value;
        }
    }
}
```

❑ BÀI 1: LẬP TRÌNH CONSOLE TRONG C#

1.2. Các bài tập ứng dụng trên Console

b) Mã nguồn chương trình:

```
static void Main()
{
    try
    {
        // ...Khai báo biến kiểu nguyên.....
        int a, b;
        // ...Hiển thị tiêu đề về bài tập.....
        Console.WriteLine("...Bai tap mau lap trnh tren Console c#\n");
        // ...Nhập giá trị cho biến a.....
        Console.WriteLine("Nhap so a = "); a = Convert.ToInt32(Console.ReadLine());
        // ...Nhập giá trị cho biến b.....
        Console.WriteLine("Nhap so b = "); b = Convert.ToInt32(Console.ReadLine());

        // ...Gọi hàm và truyền tham số để tính TỔNG
        Console.WriteLine("Ket qua tinh toan:\n");
        Console.WriteLine(a.ToString() + " + " + b.ToString() + " = {0}", MAU01(a, b, "TONG"));
        // ...Gọi hàm và truyền tham số để tính HIỆU
        Console.WriteLine(a.ToString() + " - " + b.ToString() + " = {0}", MAU01(a, b, "HIEU"));
        // ...Gọi hàm và truyền tham số để tính TÍCH
        Console.WriteLine(a.ToString() + " x " + b.ToString() + " = {0}", MAU01(a, b, "TICH"));
        // ...Gọi hàm và truyền tham số để tính THƯƠNG
        Console.WriteLine(a.ToString() + " / " + b.ToString() + " = {0}", MAU01(a, b, "THUONG"));
        //....Dừng màn hình cho đến khi bấm phím bất kỳ
        Console.ReadKey();
    }
    catch (Exception ex)
    {
        Console.WriteLine(ex.ToString()); // Hiển thị lỗi nếu có
    }
}
```

```
//Hàm tự định nghĩa
4 references
public static double MAU01(int a, int b, string pheptinh)
{
    double value = 0;
    switch (pheptinh)
    {
        case "TONG":
            value = a + b;
            break;
        case "HIEU":
            value = a - b;
            break;
        case "TICH":
            value = a * b;
            break;
        case "THUONG":
            value = Math.Round((double)a / b, 2); // Làm tròn 2 số thập phân
            break;
    }
    return value;
}
```

□ BÀI 1: LẬP TRÌNH CONSOLE TRONG C#

1.2. Các bài tập ứng dụng trên Console

c) Kết quả thực hiện chương trình:

```
....Bai tap mau lap trnh tren Console c#...  
Nhap so a = 50  
Nhap so b = 30  
Ket qua tinh toan:  
50 + 30 = 80  
50 - 30 = 20  
50 x 30 = 1500  
50 / 30 = 1.67
```

```
....Bai tap mau lap trnh tren Console c#...  
Nhap so a = -50  
Nhap so b = 20  
Ket qua tinh toan:  
-50 + 20 = -30  
-50 - 20 = -70  
-50 x 20 = -1000  
-50 / 20 = -2.5
```

```
....Bai tap mau lap trnh tren Console c#...  
Nhap so a = 30  
Nhap so b = 50  
Ket qua tinh toan:  
30 + 50 = 80  
30 - 50 = -20  
30 x 50 = 1500  
30 / 50 = 0.6
```

```
....Bai tap mau lap trnh tren Console c#...  
Nhap so a = 100  
Nhap so b = 50  
Ket qua tinh toan:  
100 + 50 = 150  
100 - 50 = 50  
100 x 50 = 5000  
100 / 50 = 2
```


1.2. Các bài tập ứng dụng trên Console

Câu 1.2.1: Viết hàm `CS01()` tính tổng dãy số, tổng các số chẵn, tổng các số lẻ từ 1 đến n ($n \geq 1$); hiển thị dãy số chẵn và tổng dãy chẵn, hiển thị dãy số lẻ và tổng dãy lẻ, hiển thị tổng của dãy số.

Các hàm sử dụng:

- `public static void CS01(int n)`: Hàm tự định nghĩa
- `Console.Write(...)`: Hiển thị nội dung; toán tử `%` kiểm tra số là chẵn hoặc lẻ
- `if (n % 2 == 0)`, n là chẵn, else n là lẻ

```
Cau 2.1: Tinh tong cac so tu nhien trong C#:
```

```
Nhap n= 20
```

```
-----  
Day cac so CHAN:
```

```
2, 4, 6, 8, 10, 12, 14, 16, 18, 20,
```

```
Tong CHAN: 110
```

```
Day cac so LE:
```

```
1, 3, 5, 7, 9, 11, 13, 15, 17, 19,
```

```
Tong LE: 100
```

```
Tong CHAN va LE: 210
```

```
Cau 2.1: Tinh tong cac so tu nhien trong C#:
```

```
Nhap n= 30
```

```
-----  
Day cac so CHAN:
```

```
2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30,
```

```
Tong CHAN: 240
```

```
Day cac so LE:
```

```
1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29,
```

```
Tong LE: 225
```

```
Tong CHAN va LE: 465
```

1.2. Các bài tập ứng dụng trên Console

Câu 1.2.2: Viết hàm CS02() nhập 3 cạnh a, b, c của một tam giác là các số nguyên dương; Kiểm tra tam giác là tam giác đều nếu $a=b$ và $b=c$; Kiểm tra là tam giác cân nếu $a=b$ hoặc $a=c$ hoặc $b=c$.

```
Cau 2.2: Kiem tra tam giac deu, can, thuong trong C#:  
-----
```

```
Nhap canh a: 15  
Nhap canh b: 15  
Nhap canh c: 15  
Day la tam giac DEU.
```

```
Cau 2.2: Kiem tra tam giac deu, can, thuong trong C#:  
-----
```

```
Nhap canh a: 15  
Nhap canh b: 15  
Nhap canh c: 20  
Day la tam giac CAN.
```

```
Cau 2.2: Kiem tra tam giac deu, can, thuong trong C#:  
-----
```

```
Nhap canh a: 15  
Nhap canh b: 17  
Nhap canh c: 20  
Day la tam giac THUONG.
```

Các hàm sử dụng:

- **public static void CS02(int canha, int canhb, int canhc):** Hàm tự định nghĩa

1.2. Các bài tập ứng dụng trên Console

Câu 1.2.3: Viết hàm CS03() nhập mảng một chiều a[] gồm n phần tử từ bàn phím; Tách mảng a[] ra thành 02 mảng gồm mảng b[] chứa các phần tử chẵn và mảng c[] chứa các phần tử lẻ. Hiển thị mảng a[], b[], c[] và tổng các phần tử của từng mảng.

```
Cau 2.3: Chia mang thanh mang chan, mang le trong C#:  
-----
```

```
Nhap n= 10  
a[0]=0  
a[1]=2  
a[2]=5  
a[3]=7  
a[4]=9  
a[5]=12  
a[6]=20  
a[7]=8  
a[8]=13  
a[9]=10  
  
Cac phan cua mang:  
0 2 5 7 9 12 20 8 13 10  
Tong: 86  
Cac phan tu chan:  
0 2 12 20 8 10  
Tong: 52  
Cac phan tu le:  
5 7 9 13 nTong: 34
```

Các hàm sử dụng:

- **public static void CS03(int n, int[] a, int[] b, int[] c):** Hàm tự định nghĩa
- **Convert.ToInt32(Console.ReadLine()):** Chuyển kiểu string sang kiểu số nguyên khi nhập số phần tử của mảng và giá trị cho các phần tử của mảng
- **a[i] = Convert.ToInt32(Console.ReadLine()):** nhập giá trị cho phần tử thứ i của mảng
- **Console.Write("{0} ", a[i]):** Hiển thị phần tử thứ i của mảng

1.2. Các bài tập ứng dụng trên Console

Câu 1.2.4: Viết hàm CS04() nhập ma trận $a[n][m]$ với các phần tử được nhập từ bàn phím; Hiển thị các phần tử của mảng; Tính và hiển thị tổng các phần tử trên đường chéo chính của mảng.

```
Cau 2.4: Tinh tong cac phan tu tren duong cheo chinh cua ma tran VUONG C#:  
-----  
Nhap kích thước ma tran vuong n= 4  
a[0],[0]= 1  
a[0],[1]= 4  
a[0],[2]= 3  
a[0],[3]= 2  
a[1],[0]= 6  
a[1],[1]= 5  
a[1],[2]= 4  
a[1],[3]= 3  
a[2],[0]= 8  
a[2],[1]= 7  
a[2],[2]= 3  
a[2],[3]= 4  
a[3],[0]= 5  
a[3],[1]= 6  
a[3],[2]= 7  
a[3],[3]= 8  
In ma tran:  
1 4 3 2  
6 5 4 3  
8 7 3 4  
5 6 7 8  
Tong cac phan tu tren duong cheo chinh la: 17
```

Các hàm sử dụng:

- **public static void CS04(int n, int[,] a):** Hàm tự định nghĩa
- **Convert.ToInt32(Console.ReadLine()):** Chuyển kiểu string sang kiểu số nguyên khi nhập số phần tử của mảng và giá trị cho các phần tử của mảng
- **a[i,j]=Convert.ToInt32(Console.ReadLine()):** Nhập giá trị cho phần tử ở dòng i, cột j của mảng
- **Console.Write("{0} ", a[i, j]):** Hiển thị giá trị phần tử ở dòng i, cột j của mảng.

❑ BÀI 1: LẬP TRÌNH CONSOLE TRONG C#

1.2. Các bài tập ứng dụng trên Console

Câu 1.2.5: Viết hàm CS05() nhập một chuỗi ký tự đầu vào từ bàn phím; Đếm và hiển thị số chữ cái, chữ số và ký tự đặc biệt trong chuỗi.

Các hàm sử dụng:

- **public static void CS05(string str):** Hàm tự định nghĩa
- **str.Length:** Lấy chiều dài của chuỗi ký tự
- **if ((str[i] >= 'a' && str[i] <= 'z') || (str[i] >= 'A' && str[i] <= 'Z')):** Kiểm tra ký tự là các chữ cái
- **if (str[i] >= '0' && str[i] <= '9'):** Kiểm tra ký tự là chữ số
- Còn lại là các ký tự đặc biệt.

```
Cau 2.5: Xy ly chuoì trong C#:
```

```
-----  
Nhap chuoì: phamphuonglinh^%$#@778899  
-----
```

```
Chuoì ky tu nhap:  
phamphuonglinh^%$#@778899  
So luong: 25  
-----
```

```
Day chu cai:  
phamphuonglinh  
So luong: 14  
-----
```

```
Day chu so:  
778899  
So luong: 6  
-----
```

```
Day ky tu dac biet:  
^%$#@  
So luong: 5
```

```
Cau 2.5: Xy ly chuoì trong C#:
```

```
-----  
Nhap chuoì: Daihoc!@123Kythuat#$456Congnghiepj%^&*789  
-----
```

```
Chuoì ky tu nhap:  
Daihoc!@123Kythuat#$456Congnghiepj%^&*789  
So luong: 41  
-----
```

```
Day chu cai:  
DaihocKythuatCongnghiepj  
So luong: 24  
-----
```

```
Day chu so:  
123456789  
So luong: 9  
-----
```

```
Day ky tu dac biet:  
!@#$%^&*  
So luong: 8
```

1.2. Các bài tập ứng dụng trên Console

Câu 1.2.6: Viết hàm CS06() nhập chuỗi ký tự là Họ và Tên đầy đủ của một sinh viên; Hãy tách và hiển thị Họ, Tên đệm, Tên của sinh viên; Nhập Tên mới, thay thế Tên sinh viên cũ thành Tên sinh viên mới, hiển thị Họ và tên đầy đủ của sinh viên mới ở dạng chữ hoa và số ký tự của Họ và tên.

Các hàm sử dụng:

- **public static void CS06():** Hàm tự định nghĩa
- **str = Console.ReadLine():** Nhập chuỗi ký tự cho biến str
- **string[] str1 = str.Split(' '):** Tách các từ trong chuỗi ký tự cách nhau bởi dấu cách; **str1[0]:** lấy chữ đầu tiên sau khi tách.
- **Substring(...):** Lấy chuỗi ký tự con trong chuỗi ký tự; **Replace(...):** Thay thế chuỗi ký tự; **ToUpper():** Chuyển chuỗi ký tự thành chữ hoa; **Length:** Cho biết chiều dài chuỗi ký tự.

```
Cau 2.6: Xy ly chuoai trong C#:  
-----  
Nhap Ho va ten: Pham Phuong Anh  
  
Ho: Pham  
  
Ten dem: Phuong  
  
Ho ten dem: Pham Phuong  
  
Ten: Anh  
.....  
Nhap ten moi: linh  
  
Ho ten moi: PHAM PHUONG LINH  
So ky tu: 16
```

```
Cau 2.6: Xy ly chuoai trong C#:  
-----  
Nhap Ho va ten: Pham Nguyen Phuong Anh  
  
Ho: Pham  
  
Ten dem: Nguyen Phuong  
  
Ho ten dem: Pham Nguyen Phuong  
  
Ten: Anh  
.....  
Nhap ten moi: linh  
  
Ho ten moi: PHAM NGUYEN PHUONG LINH  
So ky tu: 23
```

1.2. Các bài tập ứng dụng trên Console

Câu 1.2.7: Viết hàm CS07() tính tổng sau (làm tròn đến 3 chữ số)

Các hàm sử dụng:

- **public static double CS07(int n, double x)**: Hàm tự định nghĩa
- **Math.Round(...)**: Làm tròn số thực.
- Sử dụng vòng lặp $i = 1$ đến n để tính các tổng thành phần S_{tp} , tổng $S = S + S_{tp}$, in S_{tp} . Kết thúc vòng lặp hàm trả về giá trị là S .

$$S(x, n) = x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!}$$

Câu 2.7: Tinh tong trong C#:

```
Nhap n= 10
Nhap x= 5
S(1) = 5
S(2) = 12.5
S(3) = 20.833
S(4) = 26.042
S(5) = 26.042
S(6) = 21.701
S(7) = 15.501
S(8) = 9.688
S(9) = 5.382
S(10) = 2.691
Tong S = 145
```

Câu 2.7: Tinh tong trong C#:

```
Nhap n= 10
Nhap x= -2
S(1) = -2
S(2) = 2
S(3) = -1.333
S(4) = 0.667
S(5) = -0.267
S(6) = 0.089
S(7) = -0.025
S(8) = 0.006
S(9) = -0.001
S(10) = 0
Tong S = -1
```

1.2. Các bài tập ứng dụng trên Console

BÀI TẬP

Câu 1: Viết chương trình C# cho phép nhập vào 2 ma trận và tính tổng hai ma trận và sau đó in ma trận kết quả trên màn hình.

Câu 2: Viết chương trình nhập vào một mảng một chiều và thực hiện:

Liệt kê các phần tử dương

Liệt kê các phần tử lẻ ở vị trí chẵn.

Liệt kê các số nguyên tố, số chính phương, số hoàn hảo trong mảng.

Tìm phần tử lớn nhất, nhỏ nhất trong mảng một chiều

Xóa một phần tử khỏi mảng

Xóa tất cả phần tử = x.

Sắp xếp mảng theo chiều tăng, giảm dần.

Câu 3: Viết chương trình giải phương trình bậc nhất có dạng $ax + b = 0$, với các hệ số a, b được nhập từ bàn phím.

Câu 4: Viết chương trình nhập vào một số nguyên n. Kiểm tra n có phải số nguyên tố hay không? Hiển thị các số nguyên tố nhỏ hơn n. Hiển thị n số nguyên tố đầu tiên

Câu 5: Viết chương trình C# cho phép nhập vào 2 ma trận và tính tổng hai ma trận và sau đó in ma trận kết quả trên màn hình.

□ BÀI 2: LẬP TRÌNH ĐỐI TƯỢNG TRONG C#

▪ Mục đích, yêu cầu

- Tìm hiểu về lớp, đối tượng, thuộc tính, phương thức, thừa kế,... trong C#
- Xây dựng các lớp đối tượng cụ thể và sử dụng các lớp đối tượng trong C#

▪ Các bài tập vận dụng



Câu 2.1:

a) Xây dựng lớp đối tượng về người có tên là **Person** gồm các thuộc tính (**hovaten, gioitinh, tuoi**). Trong đó có một phương thức có tên **Luatuoi()** kiểu **void** xác định và hiển thị lứa tuổi sau khi nhập tuổi (**tuoi**) cho đối tượng cụ thể. Lứa tuổi được chia như sau:

- $0 < \text{Tuổi} < 6$: Trẻ em, mầm non, mẫu giáo
- $6 \leq \text{Tuổi} < 11$: Nhi đồng
- $11 \leq \text{Tuổi} < 15$: Thiếu niên
- $15 \leq \text{Tuổi} < 28$: Thanh niên
- $28 \leq \text{Tuổi} < 60$: Trưởng thành
- $60 \leq \text{Tuổi}$: Về già

b) Xây dựng lớp đối tượng về sinh viên có tên là **Student** thừa kế lớp **Person** gồm các thuộc tính (**masv, malop, diemtichluy**). Trong đó có một phương thức có tên **Xeploaihocluc()** kiểu **string** xác định và hiển thị xếp loại học lực sau khi nhập điểm trung bình tích lũy hệ 4 (**dientichluy**) cho đối tượng cụ thể. Xếp loại học lực được chia như sau:

- $0.00 < \text{điểm tích lũy} \leq 1.99$: Chưa xếp loại
- $2.00 \leq \text{điểm tích lũy} \leq 2.99$: Trung bình
- $2.30 \leq \text{điểm tích lũy} \leq 2.49$: Trung bình khá
- $2.50 \leq \text{điểm tích lũy} \leq 3.19$: Khá
- $3.20 \leq \text{điểm tích lũy} \leq 3.59$: Giỏi
- $3.60 \leq \text{điểm tích lũy} \leq 4.00$: Xuất sắc

□ BÀI 2: LẬP TRÌNH ĐỐI TƯỢNG TRONG C#

c) Tạo lớp **Program**, trong đó xây dựng hàm **Main()** thực hiện:

- Gọi và tạo mới đối tượng **Student**
- Nhập thông tin từ bàn phím gồm: **Họ và tên, tuổi, giới tính, mã sinh viên, mã lớp, điểm trung bình tích lũy hệ 4.**
- **Hiển thị thông tin: Họ và tên, tuổi, giới tính, mã sinh viên, mã lớp, điểm trung bình tích lũy hệ 4, lứa tuổi, xếp loại học lực.**

❑ BÀI 2: LẬP TRÌNH ĐỐI TƯỢNG TRONG C#

▪ Xây dựng lớp **Person**

```
namespace ConsoleApp1
{
    1 reference
    class Person
    {
        3 references
        public int tuoi { get; set; } // thuộc tính
        //...định nghĩa các thuộc tính còn lại

        0 references
        public void Luatuoi() //Phương thức kiểm tra lứa tuổi
        {
            switch (tuoi)
            {
                case var expression when (tuoi > 0 && tuoi < 6):
                    Console.WriteLine("Lua tuoi: Tre em, mam non, mau giao");
                    break;
                    //.....viết Code còn lại
                default:
                    Console.WriteLine("Lua tuoi: Ve gia");
                    break;
            }
        }
    }
}
```

▪ Xây dựng lớp **Student**

```
namespace ConsoleApp1
{
    0 references
    class Student : Person
    {
        1 reference
        public double diemtichluy { get; set; } // thuộc tính
        // định nghĩa các thuộc tính còn lại.....

        0 references
        public string Xeploaihocluc() //Phương thức
        {
            string loai = "";
            switch (diemtichluy)
            {
                case var case1 when case1 >= 0.00 && case1 <= 1.99:
                    loai = "CHUA XL";
                    break;
                    //....Viết Code còn lại.....
            }
            return loai;
        }
    }
}
```

❑ BÀI 2: LẬP TRÌNH ĐỐI TƯỢNG TRONG C#

▪ Xây dựng lớp **Program** và hàm **Main()**

```
namespace ConsoleApp1
{
    0 references
    internal static class Program
    {
        0 references
        static void Main()
        {
            try
            {
                //Câu 2.1:
                Student mySinhvien = new Student();
                Console.WriteLine("***** CAU 2.1: DOI TUONG C# - THUA KE *****\n");
                Console.WriteLine(".....Nhap doi tuong sinh vien.....\n");
                Console.WriteLine("Ho va ten = ");
                mySinhvien.hovaten = Console.ReadLine();
                Console.WriteLine("Tuoi = ");
                mySinhvien.tuoi = Convert.ToInt32(Console.ReadLine());
                Console.WriteLine("Gioi tinh = ");
                mySinhvien.gioitinh = Console.ReadLine();
                Console.WriteLine("Ma SV= ");
                mySinhvien.masv = Console.ReadLine();
                Console.WriteLine("Ma lop SV = ");
                mySinhvien.malop = Console.ReadLine();
                Console.WriteLine("Diem TB tích lũy he 4 = ");
                mySinhvien.diemtichluy = Convert.ToDouble(Console.ReadLine());
                Console.WriteLine(".....Hien thi doi tuong sinh vien.....\n");
                Console.WriteLine("Ho va ten: {0}", mySinhvien.hovaten.ToString());
                Console.WriteLine("Tuoi: {0}", mySinhvien.tuoi);
                Console.WriteLine("Gioi tinh: {0}", mySinhvien.gioitinh);
                mySinhvien.Luatuai();
                Console.WriteLine("Ma so SV: {0}", mySinhvien.masv);
                Console.WriteLine("Ma lop sv: {0}", mySinhvien.malop);
                Console.WriteLine("Diem tích lũy he 4: {0}", mySinhvien.diemtichluy);
                Console.WriteLine("Xep loai hoc luc: {0}", mySinhvien.Xeploaihocluc());
                Console.ReadKey();
            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.ToString());
            }
        }
    }
}
```

❑ BÀI 2: LẬP TRÌNH ĐỐI TƯỢNG TRONG C#

❖ Kết quả thực hiện chương trình Câu 2.1:

```
***** CAU 2.1: DOI TUONG C# - THUA KE *****
.....Nhap doi tuong sinh vien.....
Ho va ten = Pham Phuong Linh
Tuoi = 22
Gioi tinh = Nu
Ma SV= SV001
Ma lop SV = LOP001
Diem TB tích lũy hệ 4 = 3.65
.....Hien thi doi tuong sinh vien.....
Ho va ten: Pham Phuong Linh
Tuoi: 22
Gioi tinh: Nu
Lua tuoi: Thanh nien
Ma so SV: SV001
Ma lop sv: LOP001
Diem tích lũy hệ 4: 3.65
Xep loai hoc luc: XUATSAC
```

```
***** CAU 2.1: DOI TUONG C# - THUA KE *****
.....Nhap doi tuong sinh vien.....
Ho va ten = Pham Duc Anh
Tuoi = 33
Gioi tinh = Nam
Ma SV= SV002
Ma lop SV = LOP009
Diem TB tích lũy hệ 4 = 2.86
.....Hien thi doi tuong sinh vien.....
Ho va ten: Pham Duc Anh
Tuoi: 33
Gioi tinh: Nam
Lua tuoi: Truong thanh
Ma so SV: SV002
Ma lop sv: LOP009
Diem tích lũy hệ 4: 2.86
Xep loai hoc luc: KHA
```

Câu 2.2 :

a) Xây dựng lớp đối tượng về hình hộp có tên là **Hinhhop** gồm:

- Các thuộc tính `chieu_dai`, `chieu_rong`, `chieu_cao`. Trong đó:
- Có một phương thức **`tinhTheTich()`** kiểu `double` tính thể tích cho hình hộp, với thể tích = `chieu_dai * chieu_rong * chieu_cao`
- Có phương thức nạp chồng toán tử **`Hinhhop(...)`** kiểu `static` để tính tổng thể tích của các hình hộp.
- Có phương thức nạp chồng hàm **`inKetQua(...)`** kiểu `void` để tính toán và in các thông số và thể tích của Hình hộp thứ 1 và Hình hộp thứ 2.
- Có phương thức nạp chồng hàm **`inKetQua(...)`** kiểu `void` để tính toán và in các thông số và thể tích của hình hộp thứ 3 là tổng của Hình hộp thứ 1 và Hình hộp thứ 2.

b) Xây dựng lớp **Tinhtoan** trong đó có phương thức **`Main()`** thực hiện:

- Gọi và tạo mới 3 hình hộp
- Nhập các thông số chiều dài, chiều rộng, chiều cao cho Hình hộp thứ 1 và Hình hộp thứ 2
- Gọi hàm **`inKetQua(myHinhhop, myThutuhinhhop)`** để tính toán và hiển thị thể tích của Hình hộp thứ 1 và Hình hộp thứ 2
- Gọi hàm **`inKetQua(myHinhhop1, myHinhhop2)`** để tính toán và hiển thị thể tích của Hình hộp thứ 3 là tổng của Hình hộp thứ nhất và Hình hộp thứ 2.

❑ BÀI 2: LẬP TRÌNH ĐỐI TƯỢNG TRONG C#

❖ Trong lớp **Hinhhop**

- Định nghĩa các thuộc tính

```
0 references
public double chieu_dai { get; set; } // Chiều dài của hộp
8 references
public double chieu_rong { get; set; } // Chiều rộng của hộp
8 references
public double chieu_cao { get; set; } // Chiều cao của hộp
```

- Xây dựng phương thức tính thể tích

```
2 references
public double tinhTheTich() // Phương thức tính thể tích
{
    return chieu_dai * chieu_rong * chieu_cao;
}
```

- Xây dựng phương thức nạp chồng toán tử

```
//Nạp chồng toán tử
1 reference
public static Hinhhop operator + (Hinhhop b, Hinhhop c)
{
    Hinhhop hopA = new Hinhhop();
    hopA.chieu_dai = b.chieu_dai + c.chieu_dai;
    hopA.chieu_rong = b.chieu_rong + c.chieu_rong;
    hopA.chieu_cao = b.chieu_cao + c.chieu_cao;
    return hopA;
}
```

- Xây dựng phương thức nạp chồng hàm

```
// Nạp chồng hàm: tính và in thông số, thể tích của Hình hộp 1 và 2
2 references
public void inKetQua(Hinhhop Hop, int thu)
{
    Console.WriteLine("THE TICH HOP " + thu.ToString() + " (Dai x Rong x Cao): {0}",
        Hop.chieu_dai.ToString()
        + " x " + Hop.chieu_rong.ToString()
        + " x " + Hop.chieu_cao.ToString()
        + " = " + Hop.tinhTheTich().ToString());
}

// Nạp chồng hàm: Tính và in thông số, thể tích của Hình hộp thứ 3
1 reference
public void inKetQua(Hinhhop HopA, Hinhhop HopB)
{
    Hinhhop HopC;
    HopC = HopA + HopB;
    Console.WriteLine("TONG THE TICH 2 HOP TREN (Dai x Rong x Cao): {0}",
        HopC.chieu_dai.ToString()
        + " x " + HopC.chieu_rong.ToString()
        + " x " + HopC.chieu_cao.ToString()
        + " = " + HopC.tinhTheTich().ToString());
}
```


❑ BÀI 2: LẬP TRÌNH ĐỐI TƯỢNG TRONG C#

❖ Trong lớp Tinhtoan

```
class Tinhtoan
{
    0 references
    static void Main(string[] args)
    {
        Hinhhop Hop1 = new Hinhhop(); // Khai báo Hop1 thuộc loại Hinhhop
        Hinhhop Hop2 = new Hinhhop(); // Khai báo Hop2 thuộc loại Hinhhop
        Hinhhop Hop3 = new Hinhhop(); // Khai báo Hop3 thuộc loại Hinhhop

        // Thông số của Hop1
        Console.WriteLine("***** Cau 2.2: DOI TUONG C# - NAP CHONG VA GHI DE *****\n");
        Console.WriteLine(".....Nhap thong so Hinh hop thứ 1.....\n");
        Console.WriteLine("Chieu dai = ");
        Hop1.chieu_dai = Convert.ToDouble(Console.ReadLine());
        Console.WriteLine("Chieu rong = ");
        Hop1.chieu_rong = Convert.ToDouble(Console.ReadLine());
        Console.WriteLine("Chieu cao = ");
        Hop1.chieu_cao = Convert.ToDouble(Console.ReadLine());

        // Thông số của Hop2
        Console.WriteLine(".....Nhap thong so Hinh hop thứ 2.....\n");
        Console.WriteLine("Chieu dai = ");
        Hop2.chieu_dai = Convert.ToDouble(Console.ReadLine());
        Console.WriteLine("Chieu rong = ");
        Hop2.chieu_rong = Convert.ToDouble(Console.ReadLine());
        Console.WriteLine("Chieu cao = ");
        Hop2.chieu_cao = Convert.ToDouble(Console.ReadLine());

        // Hiển thị thông số, tính toán thể tích và hiển thị thể tích
        Console.WriteLine(".....In ket qua.....\n");
        Hop1.inKetQua(Hop1, 1); //In thông số và thể tích của Hình hộp thứ 1
        Hop2.inKetQua(Hop2, 2); //In thông số và thể tích của Hình hộp thứ 2
        Hop3.inKetQua(Hop1, Hop2); //In thông số và thể tích của Hình hộp thứ 3, là tổng của Hình hộp thứ 1 và thứ 2
        Console.ReadKey();
    }
}
```

□ BÀI 2: LẬP TRÌNH ĐỐI TƯỢNG TRONG C#

❖ Kết quả thực hiện chương trình Câu 2.2:

```
***** Cau 2.2: DOI TUONG C# - NAP CHONG VA GHI DE *****  
.....Nhap thong so Hinh hop thu 1.....  
Chieu dai = 5  
Chieu rong = 5.5  
Chieu cao = 3  
.....Nhap thong so Hinh hop thu 2.....  
Chieu dai = 6  
Chieu rong = 4  
Chieu cao = 2  
.....In ket qua.....  
THE TICH HOP 1 (Dai x Rong x Cao): 5 x 5.5 x 3 = 82.5  
THE TICH HOP 2 (Dai x Rong x Cao): 6 x 4 x 2 = 48  
TONG THE TICH 2 HOP TREN (Dai x Rong x Cao): 11 x 9.5 x 5 = 522.5
```

BÀI TẬP

Câu 1: Viết chương trình xây dựng lớp phân số cho phép nhập, xuất phân số, cộng 2 phân số và trả về là một phân số tối giản.

Câu 2: Viết một chương trình để lưu trữ các sinh viên gồm: mã sinh viên, họ và tên và điểm trung bình của N (N nhập từ bàn phím) sinh viên. Hãy sắp xếp danh sách sinh viên này theo thứ tự điểm trung bình giảm dần. Hiển thị 3 sinh viên có điểm trung bình cao nhất.

Câu 3:

- Xây dựng các lớp đối tượng hình học như: điểm, đoạn thẳng, đường tròn, hình chữ nhật, hình vuông, tam giác, hình bình hành, hình thoi. Mỗi lớp có các thuộc tính riêng để xác định được hình vẽ biểu diễn của nó như đoạn thẳng thì có điểm đầu, điểm cuối....
- Mỗi lớp thực thi một phương thức Draw() ghi đè (overriding) phương thức Draw() của lớp cơ sở gốc của các hình mà nó dẫn xuất. Hãy xây dựng lớp cơ sở của các lớp trên và thực thi đa hình với phương thức Draw(), sau đó tạo lớp Tester cùng với hàm Main() để thử nghiệm.

Câu 4: Xây dựng lớp Stack để mô phỏng một stack bao gồm:

- Phương thức khởi tạo không tham số với việc khởi tạo stack ngầm định có 20 phần tử và stack rỗng
- Phương thức khởi tạo một tham số với việc khởi tạo số phần tử của stack được truyền vào thông qua đối số của phương thức và stack rỗng
- Phương thức IsEmpty kiểm tra xem stack có rỗng không
- Phương thức IsFull kiểm tra xem stack có đầy không
- Phương thức Push và Pop để thêm vào, lấy ra một phần tử
- Hãy viết chương trình cài đặt lớp Stack và lớp sử dụng Stack.

▪ Mục đích, yêu cầu

- Thao tác với Form và các công cụ điều khiển Control
- Xây dựng các ứng dụng cụ thể trên Window form

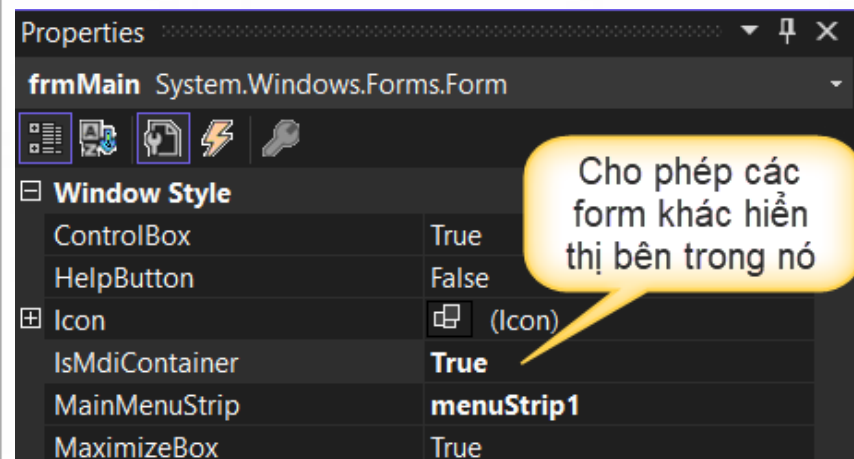
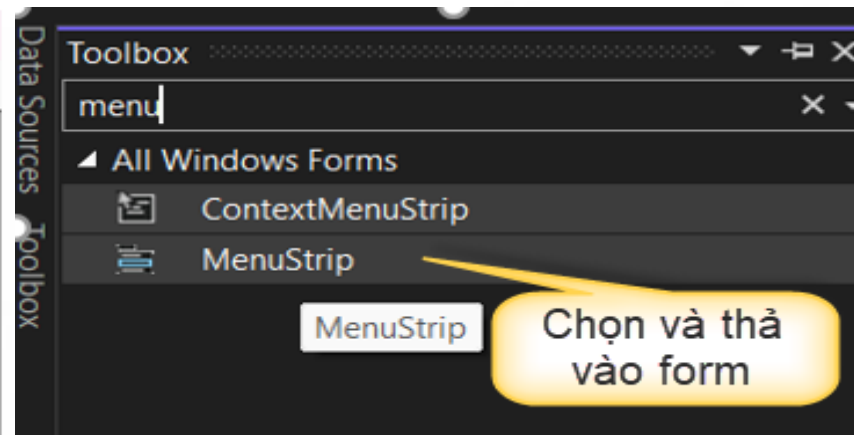
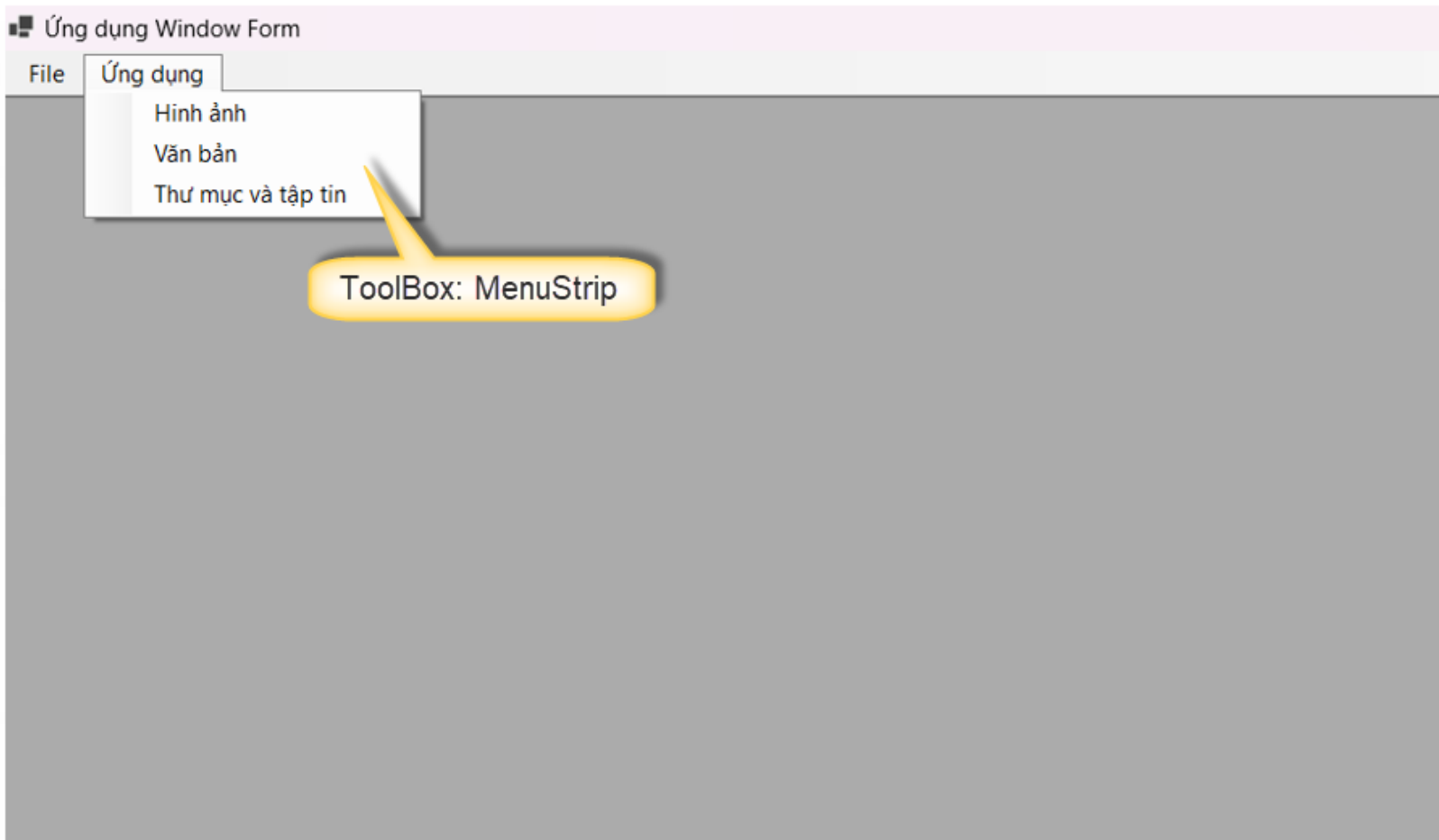
▪ Các bài tập vận dụng



□ BÀI 3: LẬP TRÌNH WINDOW FORM C#

Câu 3.1 :

Thiết kế form chính có tên là **frmMain()** với các menu chức năng như hình. Khi chọn với chức năng trên menu thì hiển thị form tương ứng với tính năng mới và nằm trong **frmMain()**



❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

- Sau khi đã thiết kế các form chức năng, các phương thức tương ứng với từng chức năng trên menu của frmMain() như sau:

```
//Hiển thị form hình ảnh frmImage()
1 reference
private void ShowImageToolStripMenuItem_Click(object sender, EventArgs e)
{
    frmImage frm = new frmImage();
    frm.MdiParent = frmMain.ActiveForm;
    frm.Show();
}

//Hiển thị form văn bản frmDoc()
1 reference
private void ShowDocToolStripMenuItem_Click(object sender, EventArgs e)
{
    frmDoc frm = new frmDoc();
    frm.MdiParent = frmMain.ActiveForm;
    frm.Show();
}

//Hiển thị form quản lý thư mục frmFolder()
1 reference
private void ShowFolderToolStripMenuItem_Click(object sender, EventArgs e)
{
    frmFolder frm = new frmFolder();
    frm.MdiParent = frmMain.ActiveForm;
    frm.Show();
}

//Đóng form frmMain()
1 reference
private void quitToolStripMenuItem_Click(object sender, EventArgs e)
{
    Application.Exit();
}
```

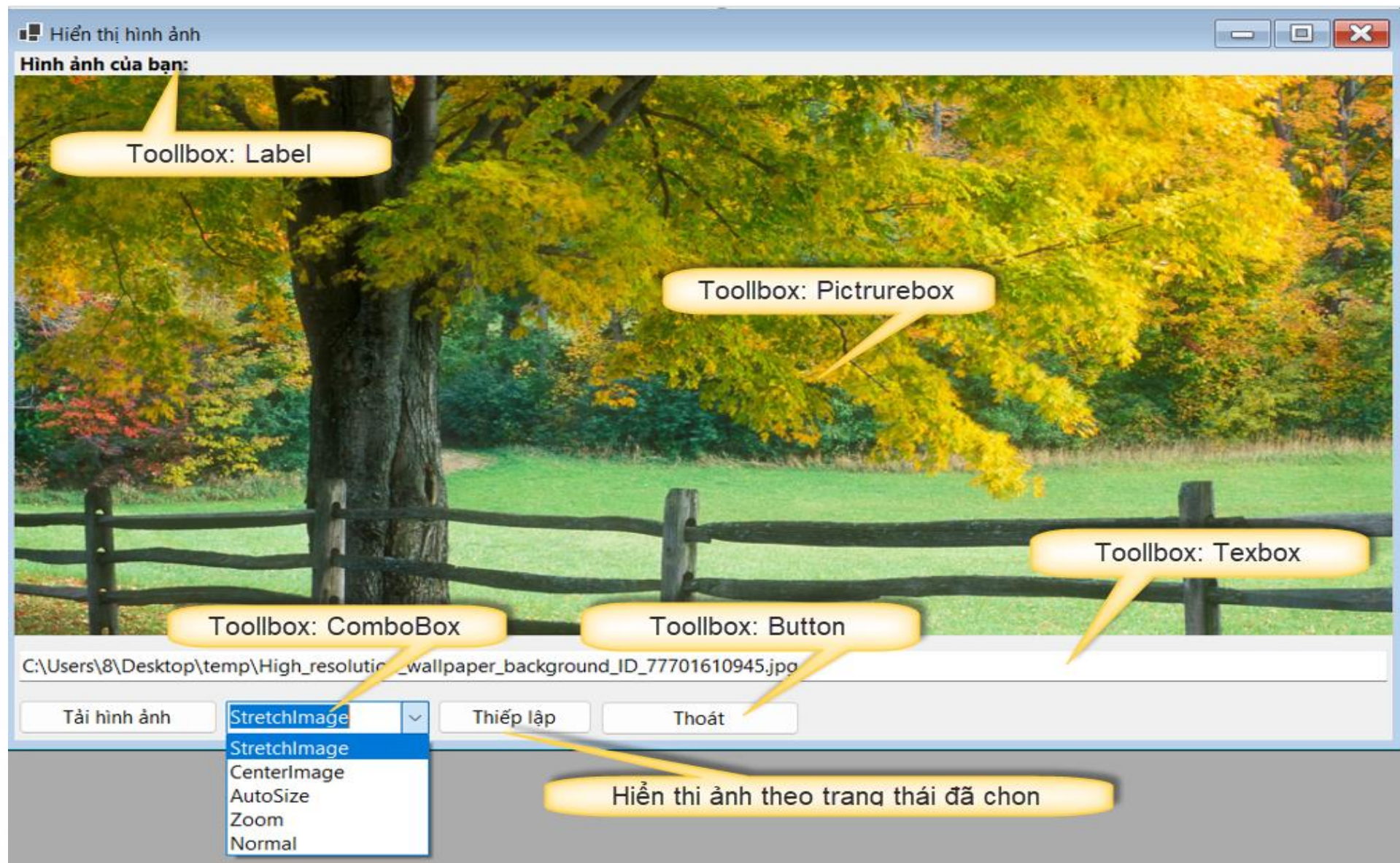
Câu 3.2 :

Thiết kế form ứng dụng xem ảnh có tên `frmImage()` như hình, thực hiện các chức năng sau:

- Chạy form ứng dụng trên menu tính năng của `frmMain()`
- Chỉ cho phép chọn và tải các file là hình ảnh
- Hiển thị đầy đủ đường dẫn và file ảnh
- Cho phép thiết lập các chế độ xem ảnh
- Hiển thị file và thư mục chứa ảnh
- Cho phép đóng/ thoát form (ứng dụng)

❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

- Công cụ sử dụng trên form:



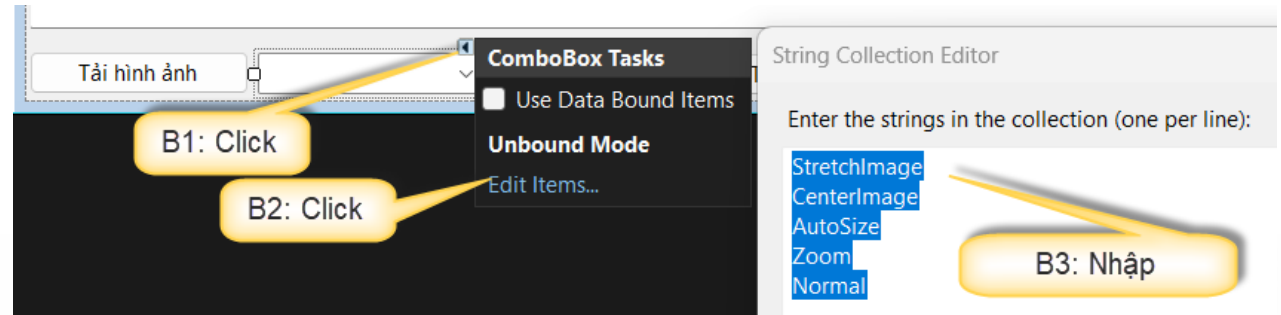
❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

■ Các phương thức trong form:

```
// Nút bấm Thiết lập trạng thái hiển thị hình ảnh
1 reference
private void cmdChangeSize_Click(object sender, EventArgs e)
{
    try
    {
        switch (combModeImage.Text)
        {
            case "StretchImage":
                myPictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;
                break;
            case "CenterImage":
                myPictureBox1.SizeMode = PictureBoxSizeMode.CenterImage;
                break;
            case "AutoSize":
                myPictureBox1.SizeMode = PictureBoxSizeMode.AutoSize;
                break;
            case "Zoom":
                myPictureBox1.SizeMode = PictureBoxSizeMode.Zoom;
                break;
            case "Normal":
                myPictureBox1.SizeMode = PictureBoxSizeMode.Normal;
                break;
        }
    }
    catch {
        MessageBox.Show("Vui lòng chọn chế độ hiển thị hình ảnh");
    }
}
```

■ Phương thức trong frmMain() để hiển thị frmImage():

```
//Hiển thị form hình ảnh frmImage()
1 reference
private void ShowImageToolStripMenuItem_Click(object sender, EventArgs e)
{
    frmImage frm = new frmImage();
    frm.MdiParent = frmMain.ActiveForm;
    frm.Show();
}
```

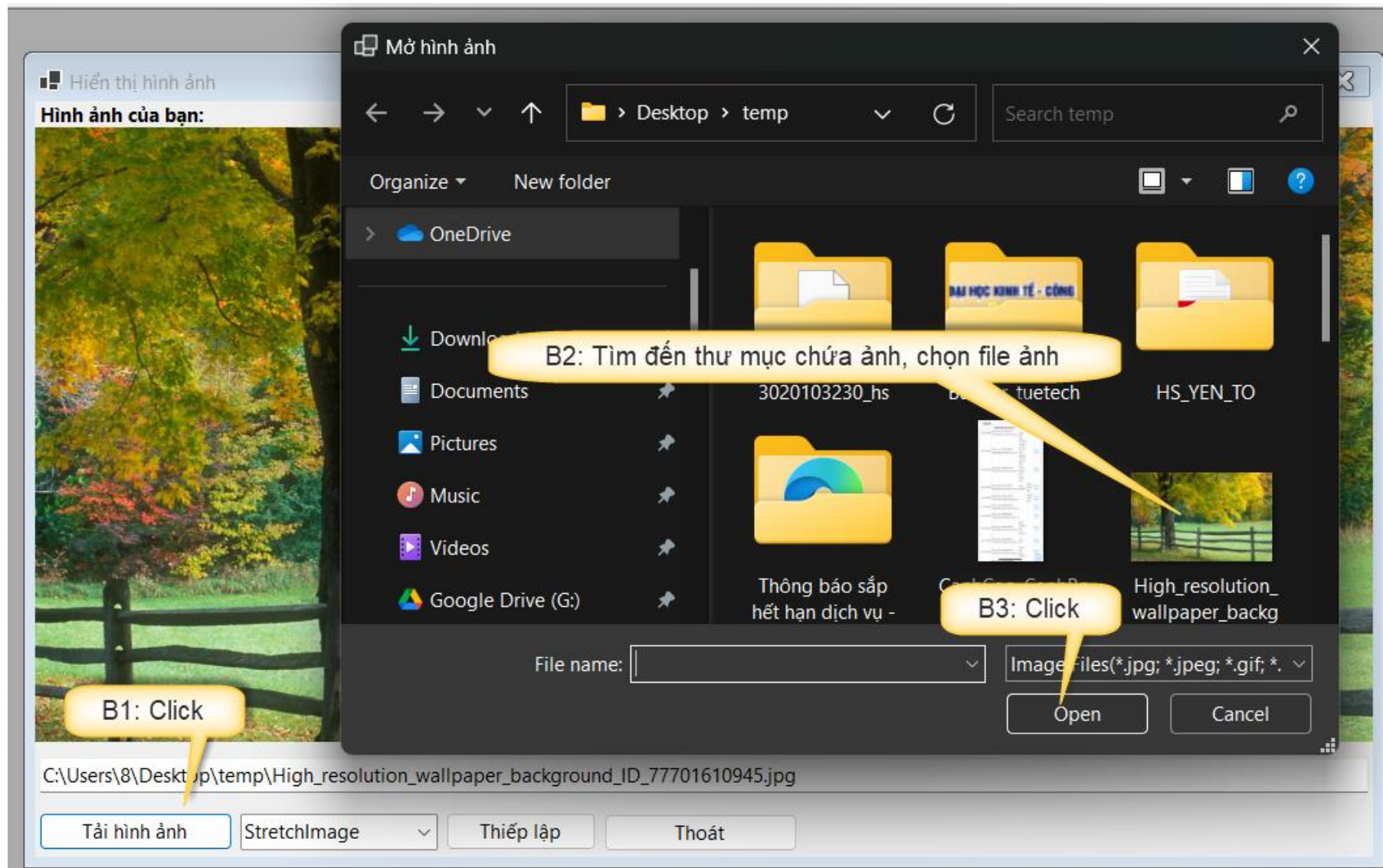


```
// Nút bấm mở trình Browse và chọn hình ảnh
1 reference
private void cmdLoadImage_Click(object sender, EventArgs e)
{
    var dialog = new OpenFileDialog();
    dialog.Title = "Mở hình ảnh";
    dialog.Filter = "Image Files(*.jpg; *.jpeg; *.gif; *.bmp)|*.jpg; *.jpeg; *.gif; *.bmp";
    if (dialog.ShowDialog() == DialogResult.OK)
    {
        myPictureBox1.Image = new Bitmap(dialog.FileName);
        txtPathFile.Text = dialog.FileName;
    }
    dialog.Dispose();
}
```

```
//Nút đóng form ứng dụng
1 reference
private void cmdCloseForm_Click(object sender, EventArgs e)
{
    this.Close();
}
```

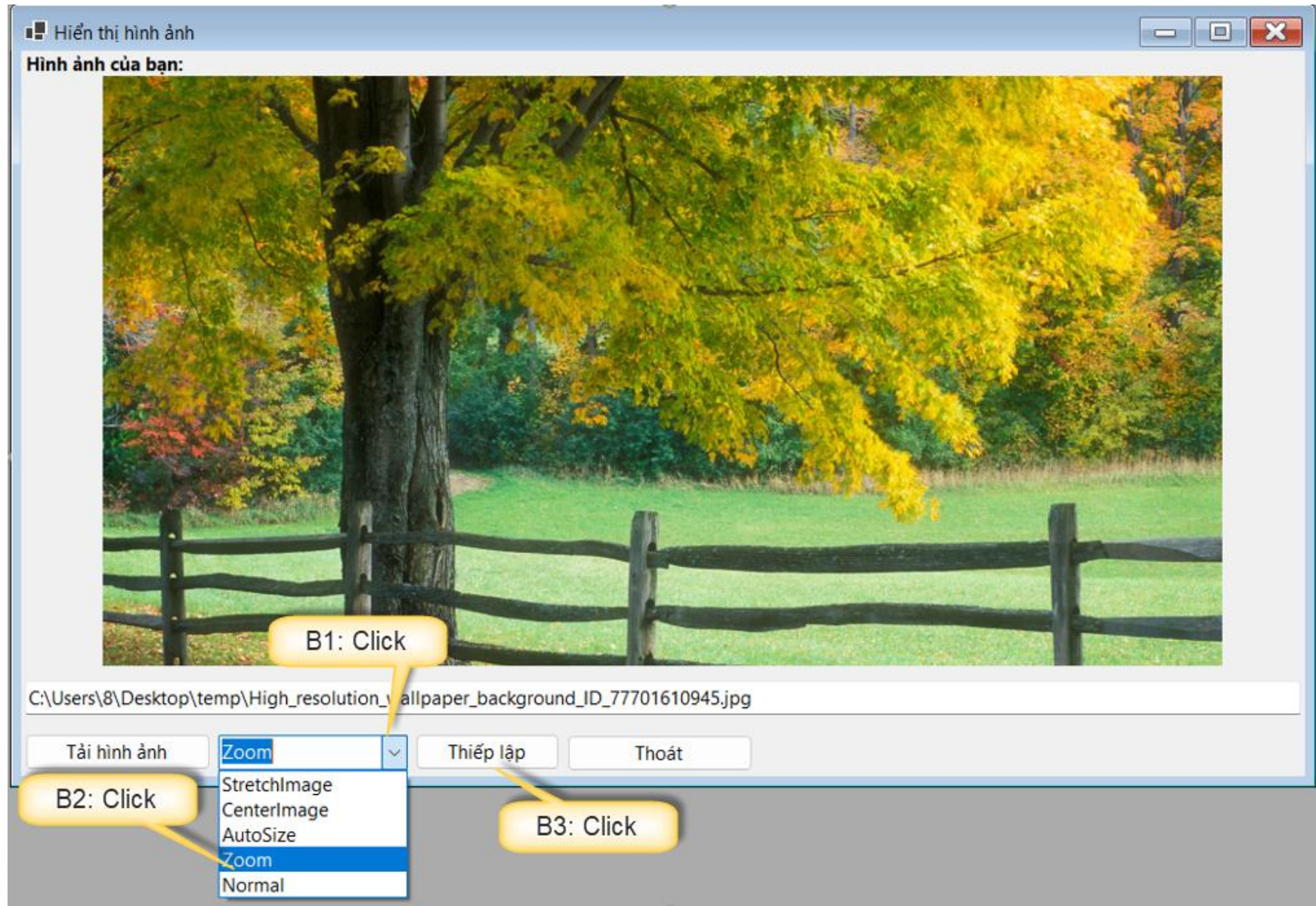
❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

- Kết quả và các bước thao tác ứng dụng:



❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

- Kết quả và các bước thao tác ứng dụng:



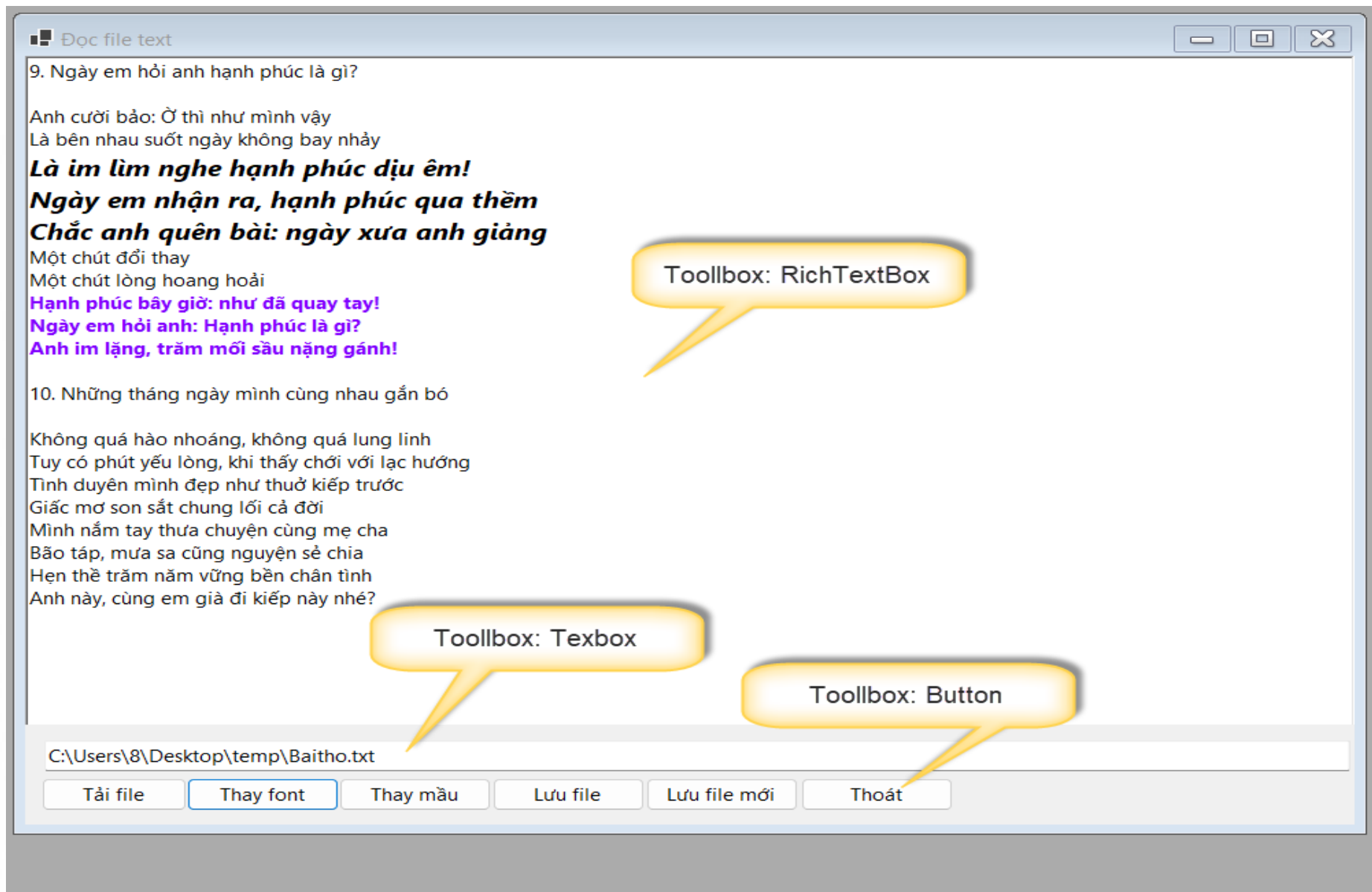
Câu 3.3 :

Thiết kế form ứng dụng đọc file văn bản (.txt) có tên **frmDoc()** như hình, thực hiện các chức năng sau:

- Chạy form ứng dụng trên menu tính năng của **frmMain()**
- Chỉ cho phép chọn và tải các file văn bản là .txt
- Chọn và thiết lập (định dạng) kiểu font chữ
- Chọn và thiết lập (định dạng) màu chữ
- Lưu văn bản; lưu văn bản với tên file mới vào thư mục chọn từ của sổ browse
- Cho phép đóng/ thoát form (ứng dụng)

❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

- Công cụ sử dụng trong form:



❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

■ Các phương thức trong form:

```
// Nút bấm Tải file: Mở trình Browse và chọn file txt
1 reference
private void cmdLoadText_Click(object sender, EventArgs e)
{
    var dialog = new OpenFileDialog();

    dialog.Title = "Mở file text";
    dialog.Filter = "txt files (*.txt)|*.txt|All files (*.*)|*.*";

    if (dialog.ShowDialog() == DialogResult.OK)
    {
        string file = dialog.FileName;
        try
        {
            myTextBox.Text = File.ReadAllText(file);
            txtPathFileText.Text = dialog.FileName;
        }
        catch (IOException) { }
    }
    dialog.Dispose();
}
```

```
// Nút bấm Lưu file mới: Mở trình Browse và đặt tên file mới
1 reference
private void cmdSaveTextFile_Click(object sender, EventArgs e)
{
    SaveFileDialog dialog = new SaveFileDialog();
    dialog.Filter = "txt files (*.txt)|*.txt|All files (*.*)|*.*";
    dialog.RestoreDirectory = true;
    if (dialog.ShowDialog() == DialogResult.OK)
    {
        File.WriteAllText(dialog.FileName, myTextBox.Text);
        MessageBox.Show(dialog.FileName);
    }
}
```

```
// Nút bấm Thoát: Đóng form
1 reference
private void cmdThoat_Click(object sender, EventArgs e)
{
    this.Close();
}
```

```
// Nút bấm Lưu file: Lưu vào file đang mở
1 reference
private void cmdSaveTextFileOld_Click(object sender, EventArgs e)
{
    SaveFileDialog dialog = new SaveFileDialog();
    string pathFile = txtPathFileText.Text.ToString();
    File.WriteAllText(pathFile, myTextBox.Text);
    MessageBox.Show("Đã ghi vào file: "+pathFile);
}
```

```
// Nút bấm Thay font: Mở hộp thoại chọn font cần định dạng
1 reference
private void cmdChangeFont_Click(object sender, EventArgs e)
{
    FontDialog dialog = new FontDialog();
    if (dialog.ShowDialog() == DialogResult.OK)
    {
        myTextBox.SelectionFont = dialog.Font;
    }
}
```

```
// Nút bấm Thay màu: Mở hộp thoại chọn màu cần định dạng
1 reference
private void cmdChangeColor_Click(object sender, EventArgs e)
{
    ColorDialog dialog = new ColorDialog();
    if (dialog.ShowDialog() == DialogResult.OK)
    {
        myTextBox.SelectionColor = dialog.Color;
    }
}
```

■ Phương thức trong frmMain() để hiển thị frmDoc():

```
//Hiển thị form văn bản frmDoc()
1 reference
private void ShowDocToolStripMenuItem_Click(object sender, EventArgs e)
{
    frmDoc frm = new frmDoc();
    frm.MdiParent = frmMain.ActiveForm;
    frm.Show();
}
```

❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

- Kết quả và các bước thao tác ứng dụng:

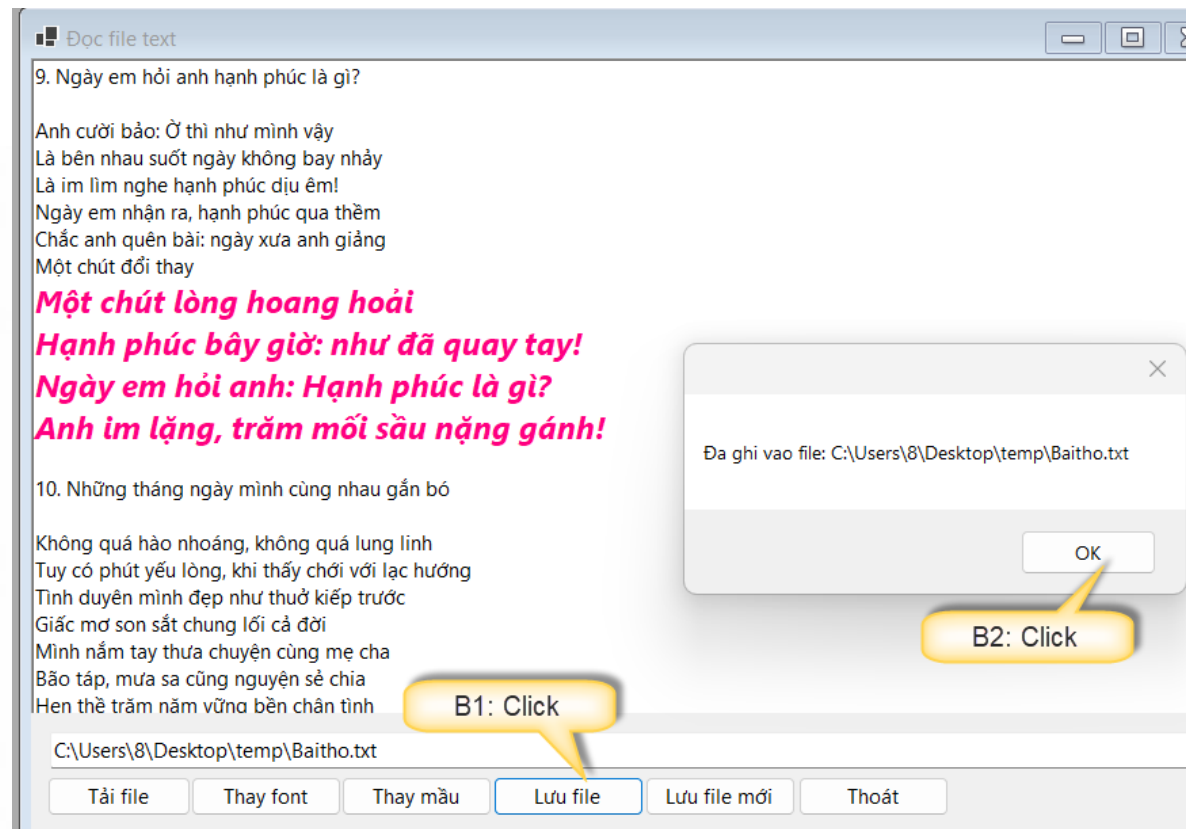
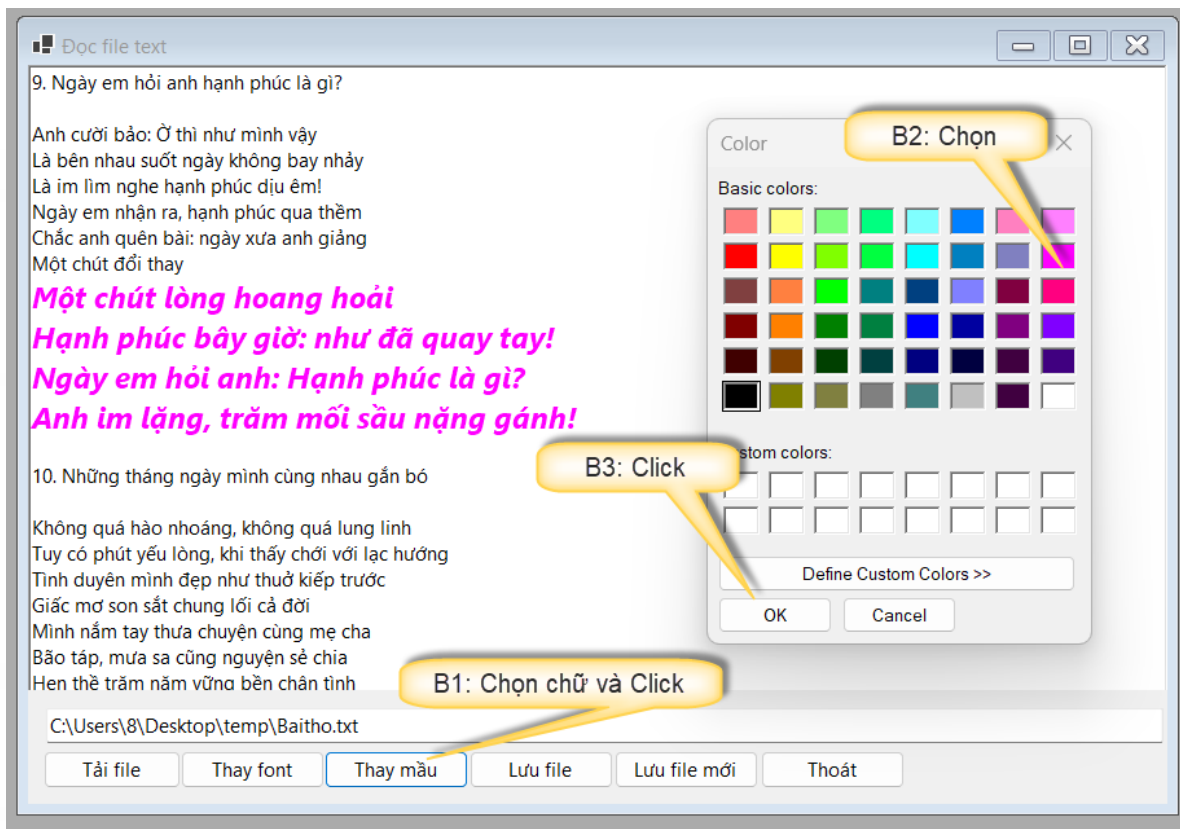
The image displays two screenshots of a Windows application interface, illustrating the steps for file selection and font formatting.

Left Screenshot: Shows a file explorer window titled "Mở file text" (Open text file) with the path "Desktop > temp". The file list includes folders like "3020103230_hs" and files like "Baitho". A yellow callout bubble labeled "B2: Tìm, chọn file" (B2: Find, select file) points to the "Baitho" file. Below the file explorer, a file name field contains "Baitho" and the file type is set to "txt files (*.txt)". A yellow callout bubble labeled "B3: Click" (B3: Click) points to the "Open" button. At the bottom, a toolbar contains buttons: "Tải file" (Load file), "Thay font" (Change font), "Thay màu" (Change color), "Lưu file" (Save file), "Lưu file mới" (Save new file), and "Thoát" (Exit). A yellow callout bubble labeled "B1: Click" (B1: Click) points to the "Tải file" button.

Right Screenshot: Shows a text editor window titled "Đọc file text" (Read text file) displaying the content of the selected file. The text includes a question "9. Ngày em hỏi anh hạnh phúc là gì?" and a poem. A yellow callout bubble labeled "B2: Chọn các thông số" (B2: Select parameters) points to the font settings dialog box. The dialog box shows "Font" set to "Segoe UI", "Font style" set to "Bold Italic", and "Size" set to "14". A yellow callout bubble labeled "B3: Click" (B3: Click) points to the "OK" button in the font dialog. Below the text editor, a toolbar contains buttons: "Tải file" (Load file), "Thay font" (Change font), "Thay màu" (Change color), "Lưu file" (Save file), "Lưu file mới" (Save new file), and "Thoát" (Exit). A yellow callout bubble labeled "B1: chọn chữ và Click" (B1: select text and Click) points to the "Thay font" button.

❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

- Kết quả và các bước thao tác ứng dụng:



❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

- Kết quả và các bước thao tác ứng dụng:

The screenshot displays a Windows application window titled "Đọc file text" with the following content:

9. Ngày em hỏi anh hạnh phúc là gì?
Anh cười bảo: Ở thì như mình vậy
Là bên nhau suốt ngày không bay nhảy
Là im lìm nghe hạnh phúc dịu êm!
Ngày em nhận ra, hạnh phúc qua thềm
Chắc anh quên bài: ngày xưa anh giảng
Một chút đổi thay
Một chút lòng hoang hoải
Hạnh phúc bây giờ: như đã quay tay!
Ngày em hỏi anh: Hạnh phúc là gì?
Anh im lặng, trăm mối sầu nặng trĩu

10. Những tháng ngày mình cùng nhau gắn bó
Không quá hào nhoáng, không quá lung linh
Tuy có phút yếu lòng, khi thấy chơi với lạc hướng
Tình duyên mình đẹp như thủa kiếp trước
Giấc mơ son sắt chung lối cả đời
Mình nắm tay thưa chuyện cùng mẹ cha
Bão táp, mưa sa cũng nguyện sẽ chia
Hen thề trăm năm vững bền chân tình

The "Save As" dialog box is open, showing the file path "C:\Users\8\Desktop\temp\Baitho.txt" and the file name "Baitho_update". The "Save as type" is set to "txt files (*.txt)".

Three yellow callout boxes indicate the steps:

- B1: Click (pointing to the "Lưu file mới" button)
- B2: Chọn thư mục, đặt tên file (pointing to the file name field)
- B3: Click (pointing to the "Save" button)

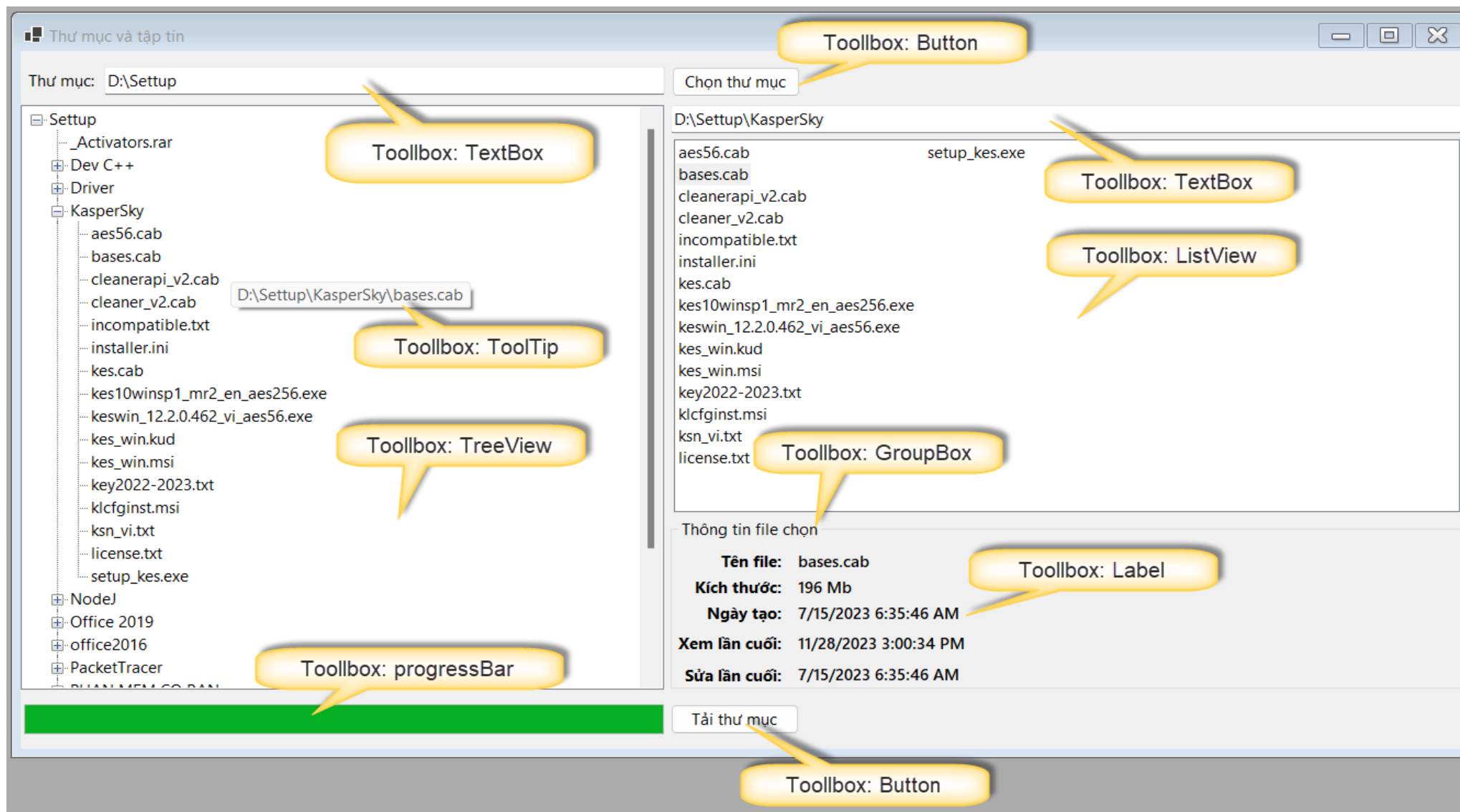
Câu 3.4 :

Thiết kế form ứng dụng quản lý folder và file có tên **frmFolder()** như hình, thực hiện các chức năng sau:

- Chạy form ứng dụng trên menu tính năng của **frmMain()**;
- Cho phép chọn, tải thư mục bất kỳ trên ổ cứng và hiển thị trên cửa sổ dạng hình cây thư mục;
- Hiển thị thư mục đã chọn;
- Khi di chuyển chuột đến thư mục hoặc file trong cây thư mục thì hiển thị đầy đủ đường dẫn và file
- Khi bấm chọn thư mục bất kỳ trong cây thư mục thì cho phép hiển thị các file trong thư mục đó ở cửa sổ bên phải;
- Hiển thị thư mục trong cây thư mục đã chọn;
- Khi chọn file bất kỳ thì hiển thị các thông số (thuộc tính) của file đó: Tên file, kích thước file, ngày tạo file, ngày xem file lần cuối, ngày sửa file lần cuối;
- Cho phép đóng/ thoát form (ứng dụng)

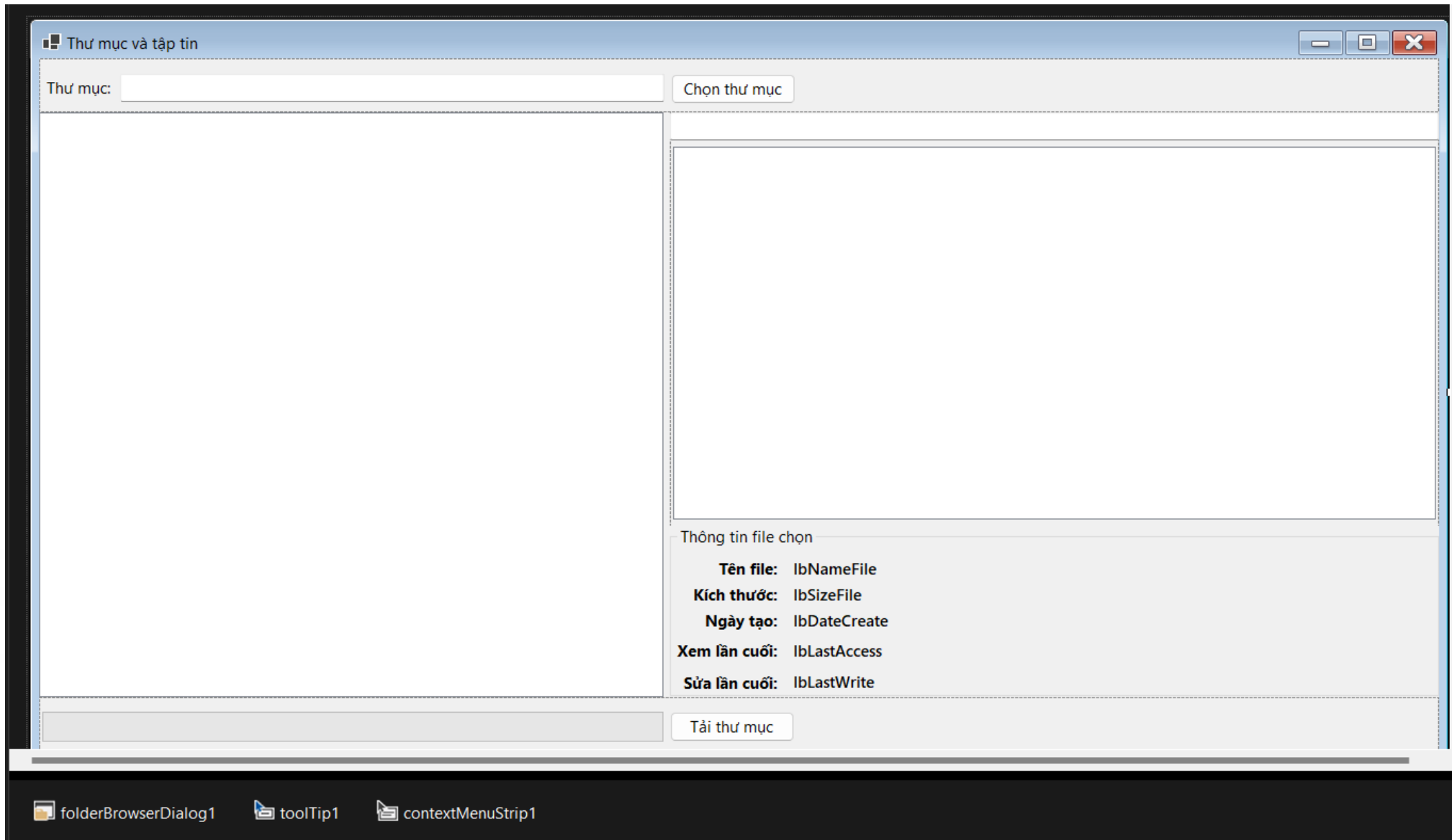
❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

- Công cụ sử dụng trong form, ngoài ra sử dụng điều khiển **folderBrowserDialog**, **toolTip**



□ BÀI 3: LẬP TRÌNH WINDOW FORM C#

- Công cụ sử dụng trong form, ngoài ra sử dụng điều khiển **folderBrowserDialog**, **toolTip**



❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

- Các phương thức trong form:

```
//Tải thư mục cha
1 reference
public void LoadDirectory(string Dir)
{
    DirectoryInfo di = new DirectoryInfo(Dir);
    //Setting ProgressBar Maximum Value
    progressBar1.Maximum = Directory.GetFiles(Dir, "**.*", SearchOption.AllDirectories).Length +
        Directory.GetDirectories(Dir, "**", SearchOption.AllDirectories).Length;
    TreeNode tds = treeView1.Nodes.Add(di.Name);
    tds.Tag = di.FullName;
    tds.StateImageIndex = 0;
    LoadFiles(Dir, tds); //Tải các file trong thư mục
    LoadSubDirectories(Dir, tds); //Tải các thư mục con
}
```

```
//Tải các file trong các thư mục
2 references
private void LoadFiles(string dir, TreeNode td)
{
    // lấy tất cả các file trong thư mục
    string[] Files = Directory.GetFiles(dir, "**.*");
    // Duyệt các file
    foreach (string file in Files)
    {
        FileInfo fi = new FileInfo(file);
        TreeNode tds = td.Nodes.Add(fi.Name);
        tds.Tag = fi.FullName;
        tds.StateImageIndex = 1;
        UpdateProgress();
    }
}
```

```
//Hiển thị trạng thái khi bấm nút Tải thư mục
2 references
private void UpdateProgress()
{
    if (progressBar1.Value < progressBar1.Maximum)
    {
        progressBar1.Value++;
        int percent = (int)(((double)progressBar1.Value / (double)progressBar1.Maximum) * 100);
        progressBar1.CreateGraphics().DrawString(percent.ToString() + "%", new Font("Arial", (float)8.25, FontStyle.Regular),
            Brushes.Black, new PointF(progressBar1.Width / 2 - 10, progressBar1.Height / 2 - 7));
        Application.DoEvents();
    }
}
```

```
//Tải các thư mục con
2 references
private void LoadSubDirectories(string dir, TreeNode td)
{
    // lấy tất cả thư mục con trong thư mục cha
    string[] subdirectoryEntries = Directory.GetDirectories(dir);
    // Duyệt thư mục
    foreach (string subdirectory in subdirectoryEntries)
    {
        DirectoryInfo di = new DirectoryInfo(subdirectory);
        TreeNode tds = td.Nodes.Add(di.Name);
        tds.StateImageIndex = 0;
        tds.Tag = di.FullName;
        LoadFiles(subdirectory, tds);
        LoadSubDirectories(subdirectory, tds);
        UpdateProgress();
    }
}
```

❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

- Các phương thức trong form:

```
// Nút bấm Chọn thư mục: Mở trình Browse và chọn thư mục cần hiển thị
1 reference
private void cmdDirectoryPath_Click(object sender, EventArgs e)
{
    try
    {
        folderBrowserDialog1.SelectedPath = txtDirectoryPath.Text;
        DialogResult drResult = folderBrowserDialog1.ShowDialog();
        if (drResult == DialogResult.OK)
            txtDirectoryPath.Text = folderBrowserDialog1.SelectedPath;
    }
    catch {
        MessageBox.Show("Thu mục nay khong cho phép truy cap ...");
    }
}
```

```
// Hiển thị các trong ListView file khi kích chọn một thư mục trong cây thư mục TreeView
1 reference
private void treeView1_MouseClick(object sender, MouseEventArgs e)
{
    try
    {
        listView1.Items.Clear();
        listView1.View = View.List;
        TreeNode theNode = treeView1.GetNodeAt(e.X, e.Y);
        txtDirectoryPath1.Text = theNode.Tag.ToString();
        string[] Files = Directory.GetFiles(theNode.Tag.ToString());
        var i = 0;
        foreach (string file in Files)
        {
            i++;
            string fileName = Path.GetFileName(file);
            ListViewItem item = new ListViewItem(fileName);
            //item.Tag = file;
            listView1.Items.Add(item);
        }
    }
    catch { }
}
```

```
// Nút bấm Tải thư mục: Tải thư mục vào cây thư mục TreeView
1 reference
private void cmdLoadDirectory_Click(object sender, EventArgs e)
{
    try
    {
        progressBar1.Value = 0; //Khởi tạo giá trị ban đầu cho progressBar1
        treeView1.Nodes.Clear(); // Xóa tất cả các nút trên cây đã có / khởi tạo cây
        tooltip1.ShowAlways = true;
        if (txtDirectoryPath.Text != "" && Directory.Exists(txtDirectoryPath.Text))
            LoadDirectory(txtDirectoryPath.Text);
        else
            MessageBox.Show("Select Directory!!");
    }
    catch
    {
        MessageBox.Show("Thu mục nay khong cho phép truy cap ...");
    }
}
```

```
//Hiển thị đường dẫn thi mục khi di chuyển chuột đến thư mục nào đó
1 reference
private void treeView1_MouseMove(object sender, MouseEventArgs e)
{
    // Lấy vị trí của chuột khi di chuyển trên TreeView
    TreeNode theNode = this.treeView1.GetNodeAt(e.X, e.Y);

    if (theNode != null && theNode.Tag != null)
    {
        // Hiển thị khi di chuyển chuột đến vị trí tiếp theo
        if (theNode.Tag.ToString() != this.tooltip1.GetToolTip(this.treeView1))
            this.tooltip1.SetToolTip(this.treeView1, theNode.Tag.ToString());
    }
    else // Tắt hiển thị khi di chuyển chuột ra chỗ khác
    {
        this.tooltip1.SetToolTip(this.treeView1, "");
    }
}
```

❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

- Các phương thức trong form:

```
// Hiển thị các thuộc tính của file khi kích chọn một file trên ListView
1 reference
private void listView1_MouseClick(object sender, MouseEventArgs e)
{
    var focusedItem = listView1.FocusedItem;
    ListViewHitTestInfo info = listView1.HitTest(e.X, e.Y);
    string fullPath = Path.Combine(txtDirectoryPath1.Text.ToString(), info.Item.Text);
    FileInfo oFileInfo = new FileInfo(fullPath);
    lbNameFile.Text = oFileInfo.Name; // Tên file
    lbSizeFile.Text = Formatter(oFileInfo.Length).ToString(); //Kích thước file
    lbDateCreate.Text = oFileInfo.CreationTime.ToString(); //Ngày tạo file
    lbLastAccess.Text = oFileInfo.LastAccessTime.ToString(); //Truy cập lần cuối
    lbLastWrite.Text = oFileInfo.LastWriteTime.ToString(); //Chỉnh sửa lần cuối
}
```

```
// Đo kích thước của file
1 reference
public string Formatter(long input)
{
    string output;
    switch (input.ToString().Length)
    {
        case > 12:
            output = input / 1000000000000 + " Tb";
            break;
        case > 9:
            output = input / 1000000000 + " Gb";
            break;
        case > 6:
            output = input / 1000000 + " Mb";
            break;
        case > 3:
            output = input / 1000 + " Kb";
            break;
        default:
            output = input + " b";
            break;
    }
    return output;
}
```

```
//Nút Thoát ứng dụng
1 reference
private void frmClose_Click(object sender, EventArgs e)
{
    this.Close();
}
```

- Phương thức trong frmMain() để hiển thị frmFolder():

```
//Hiển thị form quản lý thư mục frmFolder()
1 reference
private void ShowFolderToolStripMenuItem_Click(object sender, EventArgs e)
{
    frmFolder frm = new frmFolder();
    frm.MdiParent = frmMain.ActiveForm;
    frm.Show();
}
```

❑ BÀI 3: LẬP TRÌNH WINDOW FORM C#

- Kết quả và các bước thao tác ứng dụng:

The screenshot shows a Windows Explorer window titled "Thư mục và tập tin" with the address bar set to "D:\Setup". The left pane shows a tree view with "KasperSky" selected. The right pane shows the contents of "D:\Setup\KasperSky", including files like "aes56.cab", "bases.cab", "cleanerapi_v2.cab", "cleaner_v2.cab", "incompatible.txt", "installer.ini", "kes.cab", "kes10winsp1_mr2_en_aes256.exe", "keswin_12.2.0.462_vi_aes56.exe", "kes_win.kud", "kes_win.msi", "key2022-2023.txt", "klcfginst.msi", "ksn_vi.txt", and "license.txt". The file "bases.cab" is selected. Below the file list, the "Thông tin file chọn" section displays details for "bases.cab":
Tên file: bases.cab
Kích thước: 196 Mb
Ngày tạo: 7/15/2023 6:35:46 AM
Xem lần cuối: 11/28/2023 3:00:34 PM
Sửa lần cuối: 11/28/2023 3:00:34 PM
Yellow callout boxes indicate the following steps: "B1: Click" points to the "Chọn thư mục" button; "B2: Click" points to the "Sửa lần cuối:" field; "B3: Click" points to the "KasperSky" folder in the left pane; "B4: Click" points to the "bases.cab" file in the right pane. A green bar is visible at the bottom of the window.

BÀI TẬP

Câu 1: Xây dựng ứng dụng Window Form thực hiện giải phương trình bậc nhất $Ax + B = 0$

Câu 2: Xây dựng ứng dụng Window Form thực hiện giải phương trình bậc hai $Ax^2 + Bx + C = 0$

Câu 3: Xây dựng ứng dụng Window Form thực hiện các phép tính của 2 số với giao diện dưới đây:

Xử lý các sự kiện:

- Ở ô Textbox thêm điều kiện chỉ cho phép người dùng nhập số và xóa lùi trên Textbox
- Xử lý các phép toán cộng, trừ, nhân, chia tương ứng với các Button.
- Khi người dùng nhấn Del thì các số ở trên Textbox sẽ được xóa hết.
- Kết quả sẽ được hiển thị ở ô Textbox "Kết quả"

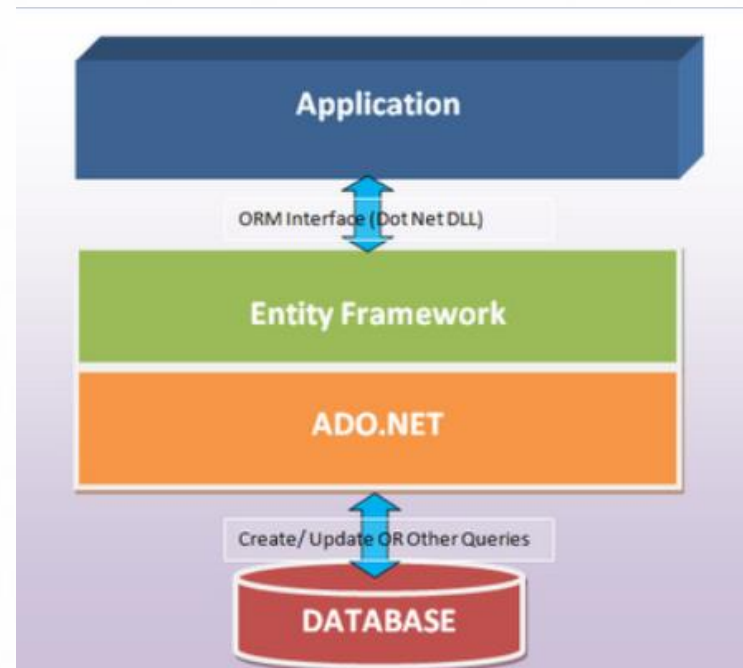
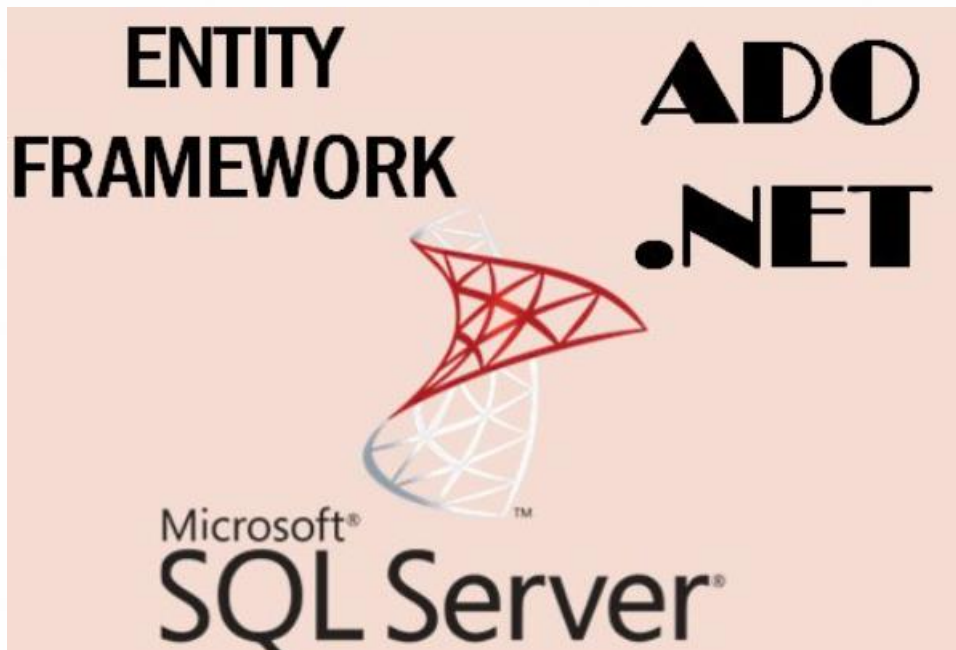
The screenshot shows a Windows Form application window titled "Form1". The form has a light gray background and contains the following elements:

- Two text boxes for input, labeled "Số thứ nhất" and "Số thứ hai".
- A row of five buttons: "+", "-", "*", "/", and "Del".
- A text box for the output, labeled "Kết quả".

▪ Mục đích, yêu cầu

- Tìm hiểu và thao tác với lớp đối tượng ADO.Net trong C#
- Xây dựng chương trình ứng dụng trên window form giao tiếp với cơ sở dữ liệu SQL Server, sử dụng ADO.Net để kết nối và thao tác với cơ sở dữ liệu.

▪ Các bài tập vận dụng



□ Yêu cầu thực hành: Xây dựng chương trình ứng dụng **Quản lý sinh viên**, sử dụng cơ sở dữ liệu **SQL Server**, gồm các tính năng:

- **Quản lý danh mục** gồm: Khoa, Ngành, Lớp, Học phần, Học kỳ, Dân tộc; Cho phép thao tác nhập, sửa, xóa, in, xuất Excel danh mục
- **Quản lý sinh viên** gồm: Quản lý thông tin các sinh viên trong trường; Cho phép thao tác nhập, sửa, in, xuất Excel thông tin sinh viên
- **Quản lý điểm** gồm: Đăng ký môn học cho sinh viên, nhập điểm theo môn học, xử lý kết quả học tập; Cho phép thao tác nhập, sửa, in, xuất Excel kết quả đăng ký học và học tập của từng sinh viên, toàn bộ các sinh viên.

❖ Tạo cơ sở dữ liệu **Thuchanh** trong SQL Server, gồm các bảng dữ liệu:

- **Sinhvien**: Lưu trữ thông tin sinh viên
- **Khoa**: Danh mục khoa trong trường
- **Lop**: Danh mục lớp thuộc khoa trong trường
- **Nganh**: Danh mục ngành học
- **Hocky**: Danh mục học kỳ sinh viên đăng ký học
- **Monhoc**: Danh mục các môn học trong trường
- **Dantoc**: Danh mục dân tộc
- **Dangkyhoc**: Kết quả học tập của từng sinh viên trong kỳ

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

✓ Cấu trúc các bảng dữ liệu trong CSDL **Thuchanh**

Object Explorer: thuchanh

- Database Diagrams
- Tables
 - System Tables
 - FileTables
 - dbo.Dangkyhoc
 - dbo.Dantoc
 - dbo.DIEM
 - dbo.Diemhocktap
 - dbo.Khoa
 - dbo.Lop
 - dbo.Monhoc
 - dbo.Nganh
 - dbo.Sinhvien**
 - dbo.tb_nhanvien
 - dbo.tb_phongban
 - dbo.tbl_DatHang
 - dbo.tbl_KhoHang
 - dbo.Thangdiem

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
MaSV	nvarchar(50)	<input type="checkbox"/>
HoTen	nvarchar(100)	<input checked="" type="checkbox"/>
Ngaysinh	nvarchar(20)	<input checked="" type="checkbox"/>
GioiTinh	nvarchar(10)	<input checked="" type="checkbox"/>
Dantoc	nvarchar(20)	<input checked="" type="checkbox"/>
HoKhu	nvarchar(100)	<input checked="" type="checkbox"/>
MaLop	nvarchar(50)	<input checked="" type="checkbox"/>

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
MaKhoa	nvarchar(50)	<input type="checkbox"/>
TenKhoa	nvarchar(100)	<input checked="" type="checkbox"/>

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
MaLop	nvarchar(50)	<input type="checkbox"/>
TenLop	nvarchar(100)	<input checked="" type="checkbox"/>
NiemKhoa	nvarchar(50)	<input checked="" type="checkbox"/>
MaKhoa	nvarchar(50)	<input checked="" type="checkbox"/>
MaNghanh	nvarchar(50)	<input checked="" type="checkbox"/>

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
MaNghanh	nvarchar(50)	<input type="checkbox"/>
TenNghanh	nvarchar(100)	<input checked="" type="checkbox"/>

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
MaMonhoc	nvarchar(50)	<input type="checkbox"/>
TenMonhoc	nvarchar(100)	<input checked="" type="checkbox"/>
SoTinchi	int	<input checked="" type="checkbox"/>
Mabm	nvarchar(50)	<input checked="" type="checkbox"/>
Ghichu	varchar(200)	<input checked="" type="checkbox"/>

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

✓ Cấu trúc các bảng dữ liệu trong CSDL **Thuchanh**

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
MaSV	nvarchar(50)	<input checked="" type="checkbox"/>
MaMonhoc	nvarchar(50)	<input checked="" type="checkbox"/>
Malophocphan	nvarchar(50)	<input checked="" type="checkbox"/>
HockyDangKy	nvarchar(50)	<input checked="" type="checkbox"/>
PhantramKT	float	<input checked="" type="checkbox"/>
DiemKT	float	<input checked="" type="checkbox"/>
DiemTH	float	<input checked="" type="checkbox"/>
DiemTB10	float	<input checked="" type="checkbox"/>
DiemTB4	float	<input checked="" type="checkbox"/>
DiemTBC	char(10)	<input checked="" type="checkbox"/>

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
tudiem	decimal(3, 1)	<input type="checkbox"/>
dendiem	decimal(3, 1)	<input checked="" type="checkbox"/>
diemso4	decimal(3, 1)	<input checked="" type="checkbox"/>
diemchu	char(2)	<input checked="" type="checkbox"/>

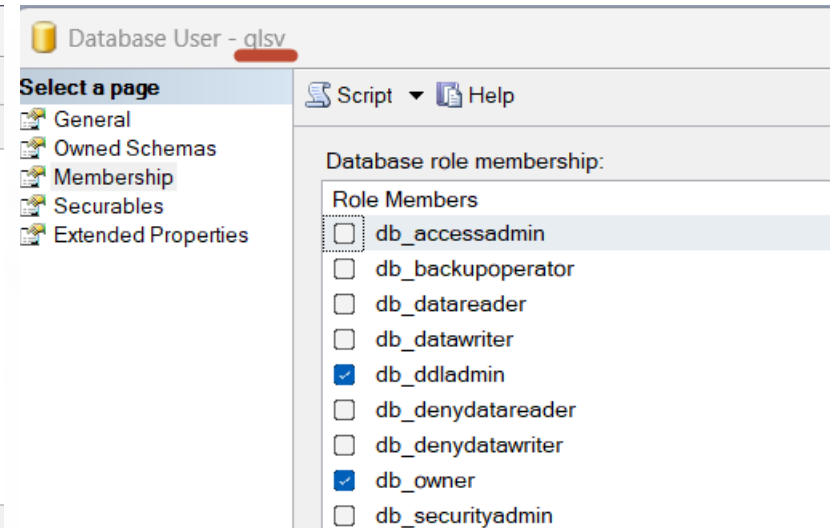
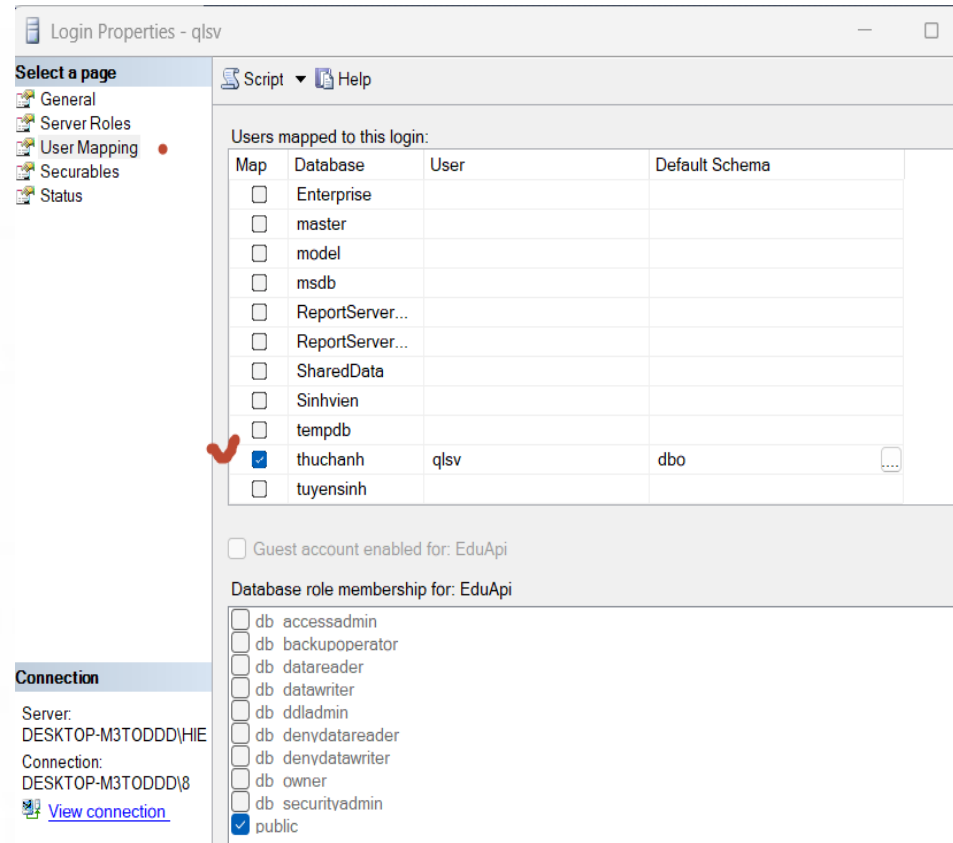
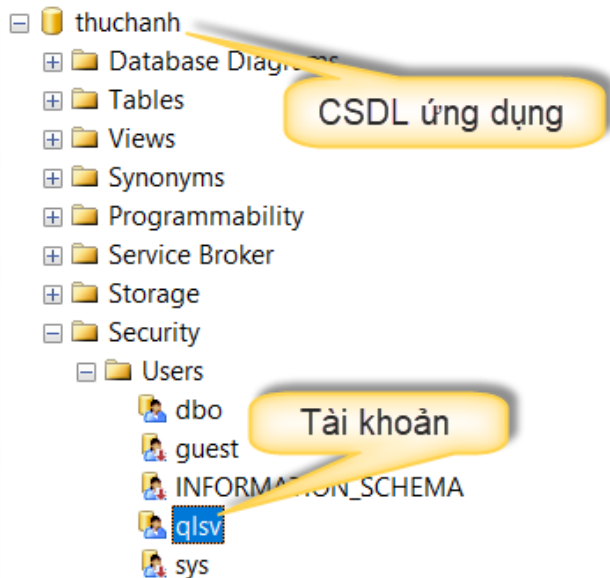
The screenshot shows the Object Explorer in SQL Server. The 'thuchanh' database is expanded, showing a list of tables in the 'dbo' schema. The tables listed are: dbo.Dangkyhoc, dbo.Dantoc, dbo.DIEM, dbo.Diemhoctap, dbo.Khoa, dbo.Lop, dbo.Monhoc, dbo.Nganh, dbo.Sinhvien, dbo.tb_nhanvien, dbo.tb_phongban, dbo.tbl_DatHang, dbo.tbl_KhoHang, and dbo.Thangdiem. The 'Dangkyhoc' table is selected, and its structure is shown in the right-hand pane.

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
MaSV	nvarchar(50)	<input checked="" type="checkbox"/>
MaMonhoc	nvarchar(50)	<input checked="" type="checkbox"/>
Malophocphan	nvarchar(50)	<input checked="" type="checkbox"/>
HockyDangKy	nvarchar(50)	<input checked="" type="checkbox"/>
PhantramKT	float	<input checked="" type="checkbox"/>
DiemKT	float	<input checked="" type="checkbox"/>
DiemTH	float	<input checked="" type="checkbox"/>
DiemTB10	float	<input checked="" type="checkbox"/>
DiemTB4	float	<input checked="" type="checkbox"/>
DiemTBC	char(10)	<input checked="" type="checkbox"/>

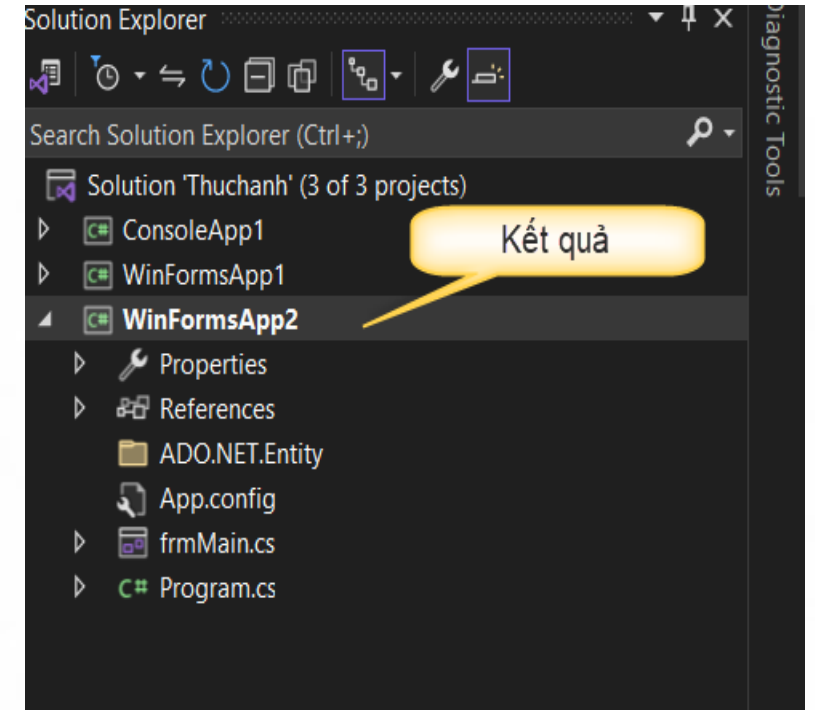
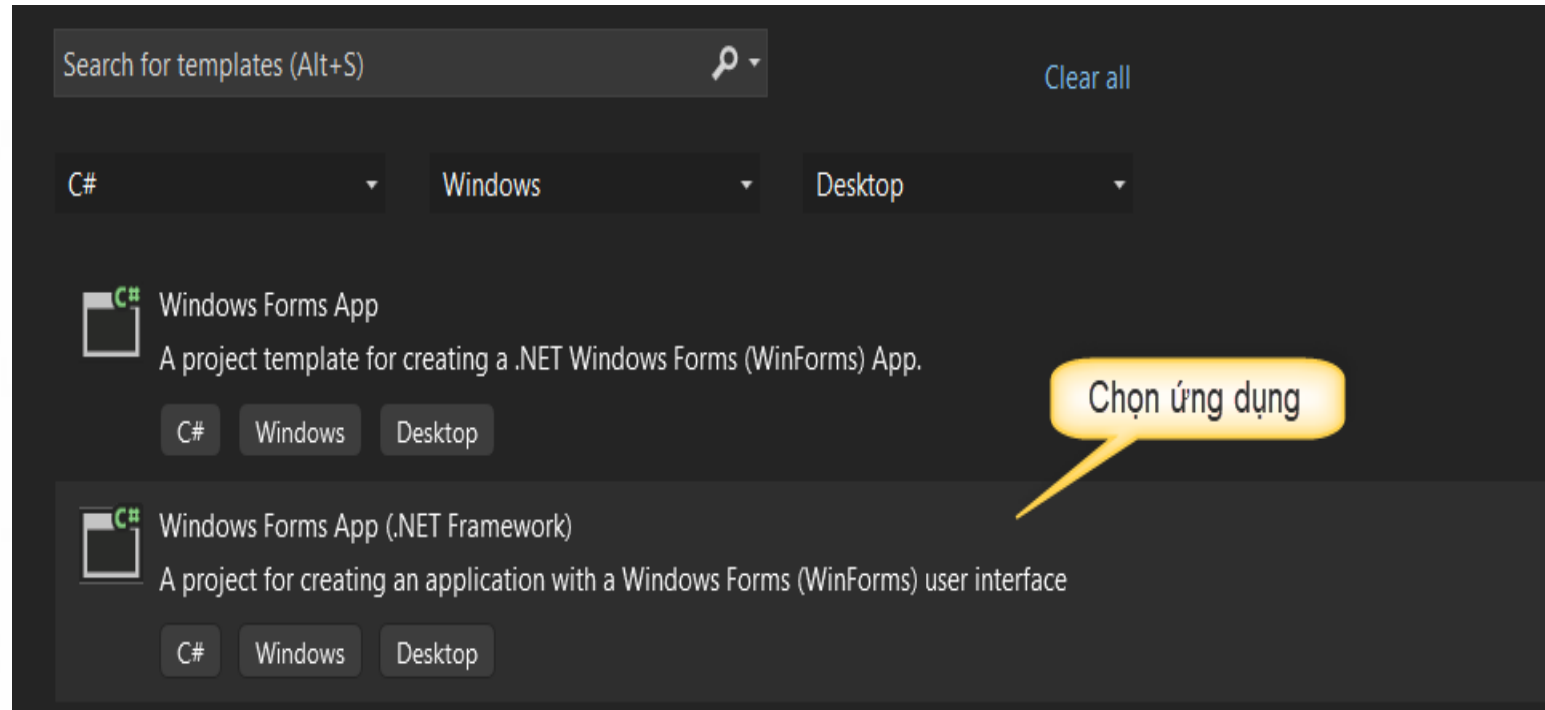
Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
madantoc	nvarchar(5)	<input type="checkbox"/>
tendantoc	nvarchar(20)	<input checked="" type="checkbox"/>

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

❖ Tạo tài khoản **qlsv** trong SQL Server, mật khẩu là **123456** Quản trị CSDL **Thuchanh**, kết nối với ứng dụng trên window form.



❖ Tạo dự án (Project) WinFormsApp2



❖ Thiết kế form chính **frmMain()** và **Menu** tính năng chương trình

The image shows a screenshot of Visual Studio with two main components:

- Design View (Left):** Displays the main form titled "Quản lý sinh viên". It features a menu bar with four items: "Quản lý danh mục", "Quản lý sinh viên", "Quản lý điểm", and "Hệ thống". The "Quản lý danh mục" menu is open, showing a list of options: "Danh mục khoa", "Danh mục ngành", "Danh mục lớp", "Danh mục học phần", "Danh mục dân tộc", and "Danh mục học kỳ". Below the list is a text box labeled "Type Here". A yellow callout bubble points to the form with the text "frmMain()".
- Solution Explorer (Right):** Shows the project structure for "Solution 'Thuchanh' (3 of 3 projects)". The expanded project is "WinFormsApp2", which contains:
 - Properties
 - References
 - ADO.NET.Entity
 - App.config
 - frmMain.cs (highlighted with a yellow callout bubble labeled "Form chính")
 - Program.cs

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

❖ Kết nối CSDL ADO.NET với SQL Server

Add New Item - WinFormsApp2

Sort by: Default

Search (Ctrl+E)

Item Name	Type
ADO.NET Entity Data Model	C# Items
DataSet	C# Items
EF 5.x DbContext Generator	C# Items
EF 6.x DbContext Generator	C# Items
Service-based Database	C# Items
XML File	C# Items
XML Schema	C# Items
XSLT File	C# Items

Type: C# Items
A project item for creating an ADO.NET Entity Data Model.

Name: dataModel

Show Compact View

Add Cancel

Entity Data Model Wizard

Choose Model Contents

What should the model contain?

- EF Designer from data...
- Empty EF Designer model
- Empty Code First model
- Code First from database

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

Connection Properties

Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.

Data source: Microsoft SQL Server (SqlClient) **Máy chủ** Change...

Server name: DESKTOP-M3TODDD\HIEUPHAM Refresh

Log on to the server

Authentication: SQL Server Authentication **Tài khoản**

User name: qlsv

Password: ●●●●●●

Encrypt: Optional (False)

Trust Server Certificate

Save my password

Connect to a database **CSDL**

Select or enter a database name: thuchanh

Attach a database file: Browse...

Logical name:

Click Advanced...

Test Connection OK Cancel

Entity Data Model Wizard

Choose Your Data Connection

Which data connection should your application use to connect to the database?

desktop-m3toddd\hieupham.thuchanh.dbo New Connection...

This connection string appears to contain sensitive data (for example, a password) that is required to connect to the database. Storing sensitive data in the connection string can be a security risk. Do you want to include this sensitive data in the connection string?

No, exclude sensitive data from the connection string. I will set it in my application code.

Yes, include the sensitive data in the connection string.

Connection string:

```
metadata=res://*/ADO.NET.Entity.dataModel.csdll|res://*/ADO.NET.Entity.dataModel.ssdll|res://*/ADO.NET.Entity.dataModel.msl;provider=System.Data.SqlClient;provider connection string="data source=DESKTOP-M3TODDD\HIEUPHAM;initial catalog=thuchanh;user id=qlsv;password=*****;encrypt=False;MultipleActiveResultSets=True;App=EntityFramework"
```

Save connection settings in App.Config as:

thuchanhEntities

< Previous Next > Finish Cancel

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

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

i It is also possible to install and use other versions of Entity Framework.

[Learn more about this](#)

Entity Data Model Wizard



Choose Your Database Objects and Settings

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

- dbo
 - Dangkyhoc
 - DIEM
 - Diemhoctap
 - Khoa
 - Lop
 - Monhoc
 - Nganh
 - Sinhvien
 - tb_nhanvien
 - tb_phongban
 - tbl_DatHang
 - tbl_KhoHang

Chọn các bảng dữ liệu

> Views

Pluralize or singularize generated object names

Include foreign key columns in the model

Import selected stored procedures and functions into the entity model

Model Namespace:

thuchanhModel

Click

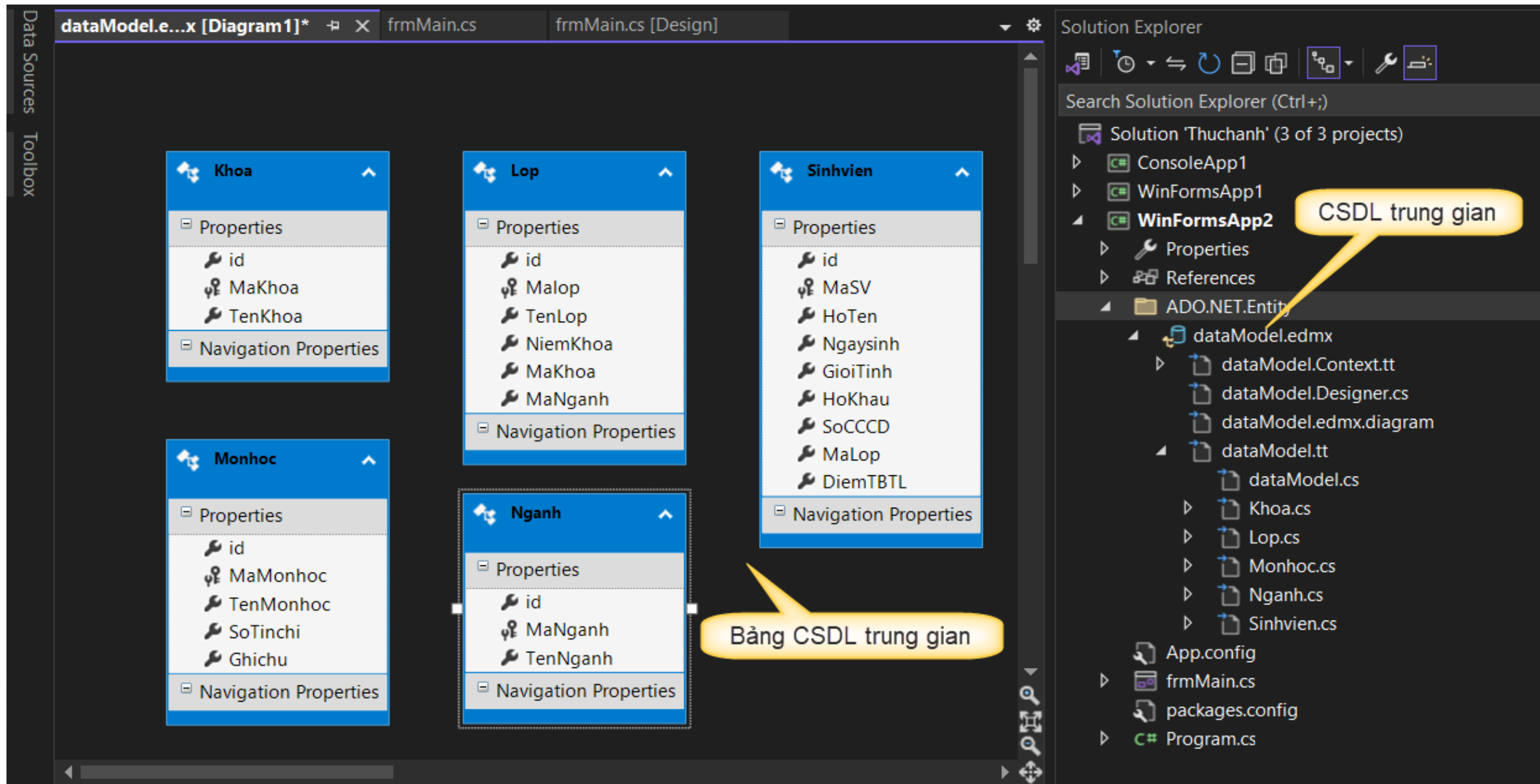
< Previous

Next >

Finish

Cancel

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET



The screenshot displays the Visual Studio environment with the Entity Data Model (EDM) diagram and the Solution Explorer.

Entity Data Model (EDM) Diagram:

- Khoa:** Properties: id, MaKhoa, TenKhoa. Navigation Properties: (empty).
- Lop:** Properties: id, MaLop, TenLop, NiemKhoa, MaKhoa, MaNganh. Navigation Properties: (empty).
- Sinhvien:** Properties: id, MaSV, HoTen, Ngaysinh, GioiTinh, HoKha, SoCCCD, MaLop, DiemTBTL. Navigation Properties: (empty).
- Monhoc:** Properties: id, MaMonhoc, TenMonhoc, SoTinchi, Ghichu. Navigation Properties: (empty).
- Nganh:** Properties: id, MaNganh, TenNganh. Navigation Properties: (empty).

Solution Explorer:

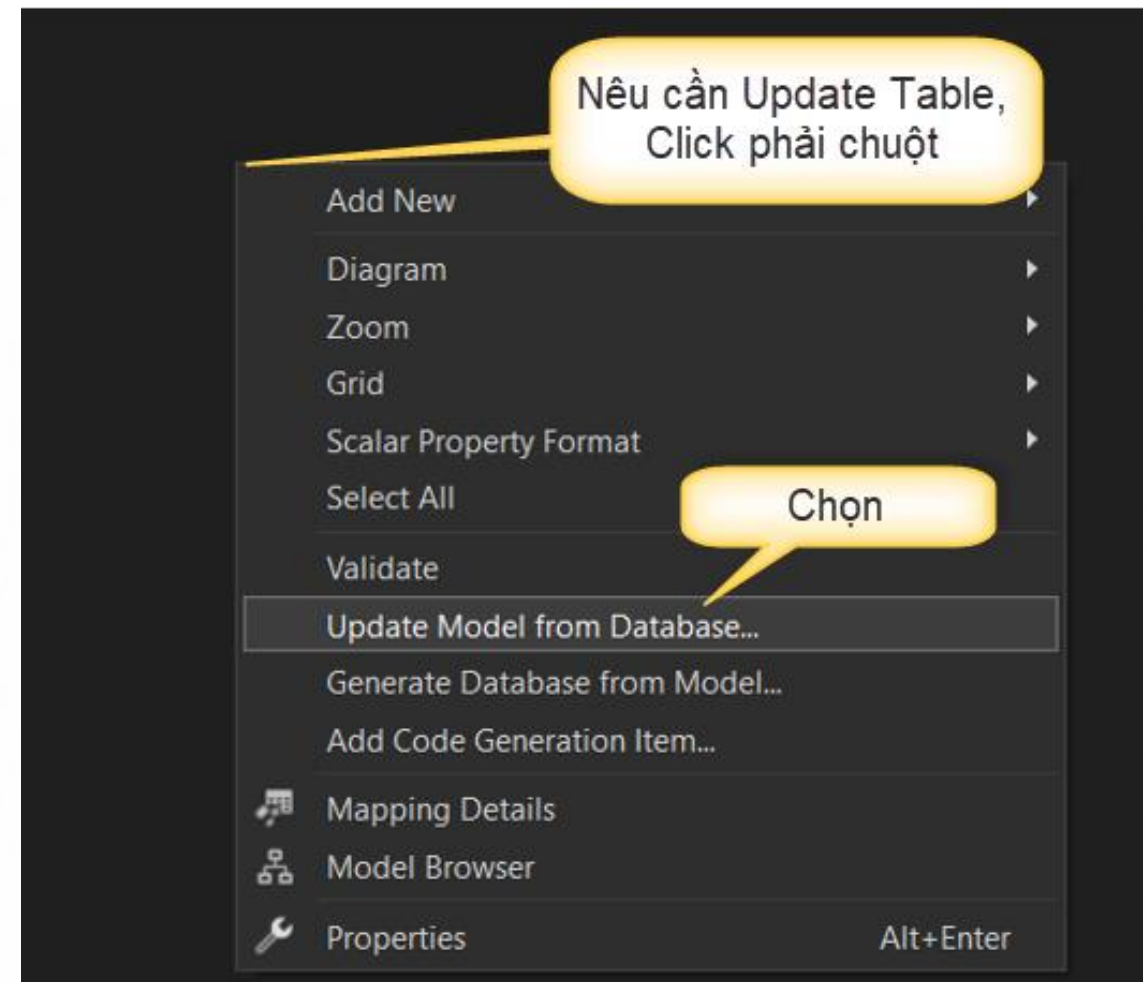
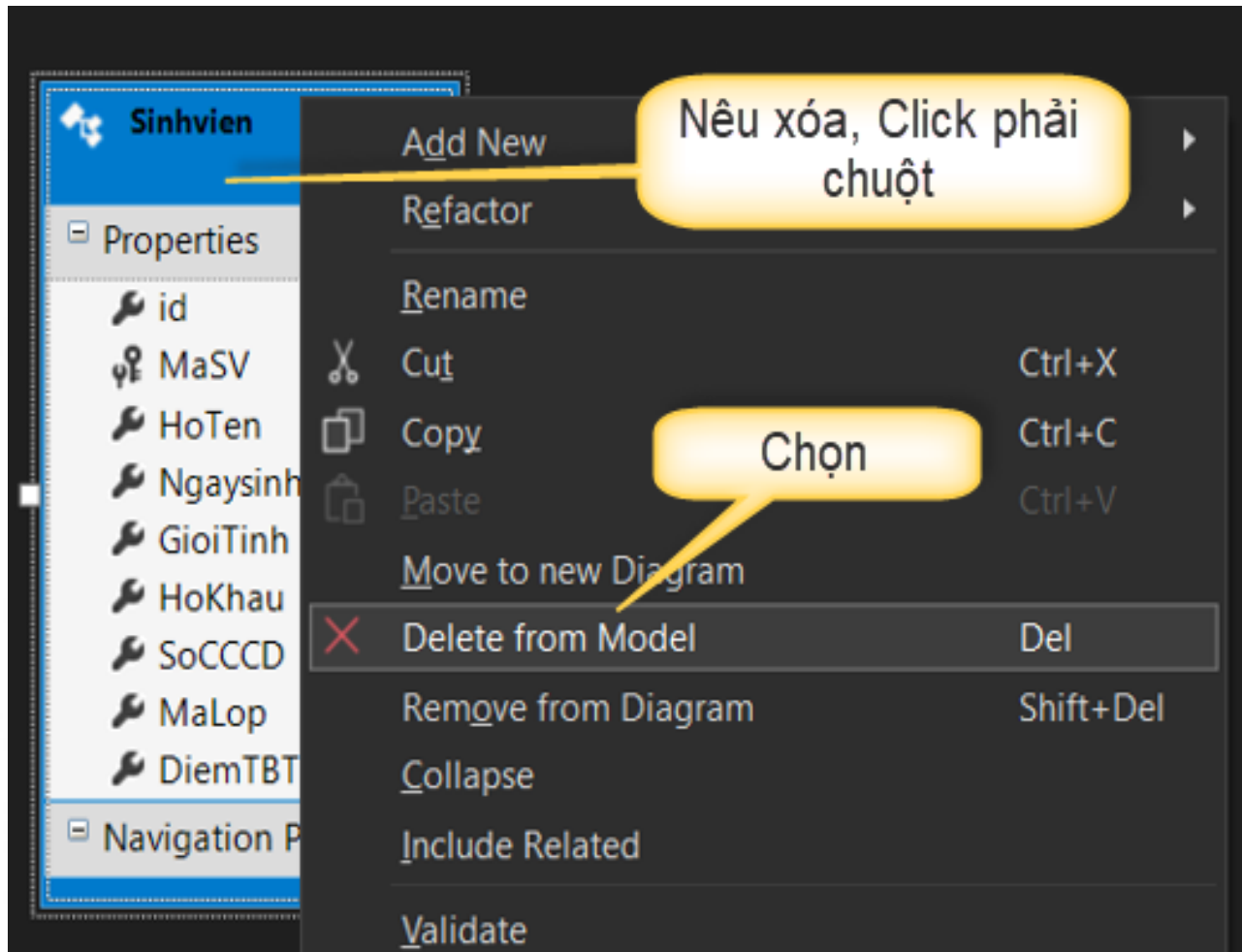
- Solution 'Thuchanh' (3 of 3 projects)
 - ConsoleApp1
 - WinFormsApp1
 - WinFormsApp2** (C#)
 - Properties
 - References
 - ADO.NET.Entity
 - dataModel.edmx
 - dataModel.Context.tt
 - dataModel.Designer.cs
 - dataModel.edmx.diagram
 - dataModel.tt
 - dataModel.cs
 - Khoa.cs
 - Lop.cs
 - Monhoc.cs
 - Nganh.cs
 - Sinhvien.cs
 - App.config
 - frmMain.cs
 - packages.config
 - Program.cs

Callouts:

- CSDL trung gian** (Intermediate Database) points to the ADO.NET.Entity folder.
- Bảng CSDL trung gian** (Intermediate Database Table) points to the Nganh entity type.

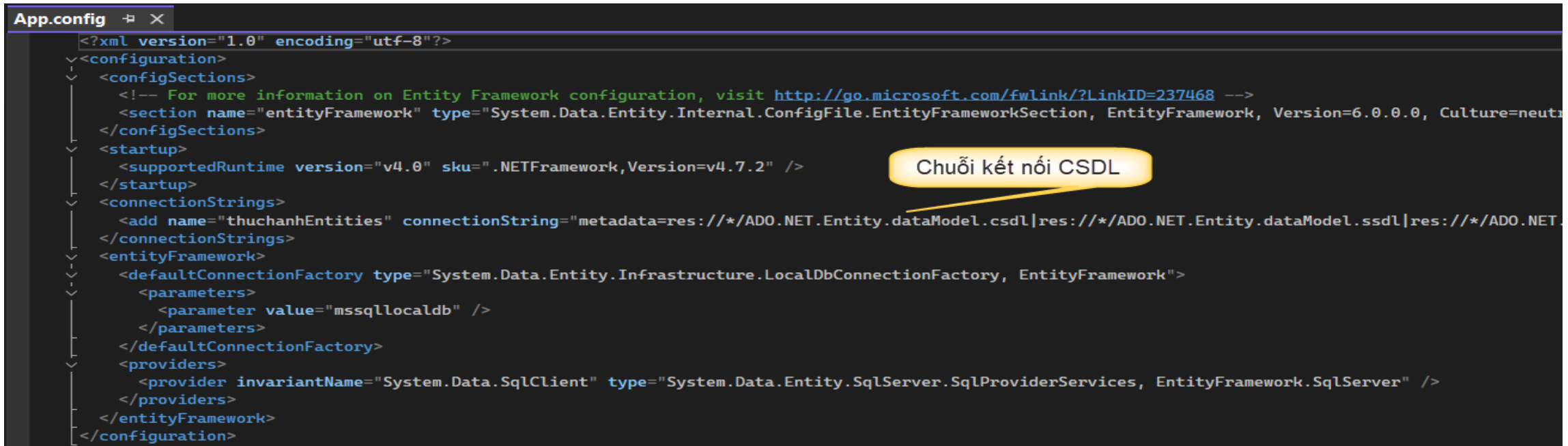
❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

Xóa và cập nhật bảng dữ liệu trung gian nếu bảng dữ liệu trong SQL Server thay đổi



❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

File **App.config** : Lưu thiết lập cấu hình chương trình ứng dụng và kết nối đến CSDL



```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <configSections>
    <!-- For more information on Entity Framework configuration, visit http://go.microsoft.com/fwlink/?LinkID=237468 -->
    <section name="entityFramework" type="System.Data.Entity.Internal.ConfigFile.EntityFrameworkSection, EntityFramework, Version=6.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" />
  </configSections>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.7.2" />
  </startup>
  <connectionStrings>
    <add name="thuchanhEntities" connectionString="metadata=res://*/ADO.NET.Entity.dataModel.csd|res://*/ADO.NET.Entity.dataModel.ssd|res://*/ADO.NET.Entity.dataModel.msl;provider=System.Data.SqlClient;provider connection string="data source=localhost;initial catalog=thuchanh;user id=qlsv;password=123456;encrypt=False;MultipleActiveResultSets=True;App=EntityFramework";providerName="System.Data.EntityClient" />
  </connectionStrings>
  <entityFramework>
    <defaultConnectionFactory type="System.Data.Entity.Infrastructure.LocalDbConnectionFactory, EntityFramework">
      <parameters>
        <parameter value="mssqllocaldb" />
      </parameters>
    </defaultConnectionFactory>
    <providers>
      <provider invariantName="System.Data.SqlClient" type="System.Data.Entity.SqlServer.SqlProviderServices, EntityFramework.SqlServer" />
    </providers>
  </entityFramework>
</configuration>
```

```
<connectionStrings>
```

```
  <add name="thuchanhEntities"
```

```
  connectionString="metadata=res://*/ADO.NET.Entity.dataModel.csd|res://*/ADO.NET.Entity.dataModel
```

```
  .ssd|res://*/ADO.NET.Entity.dataModel.msl;provider=System.Data.SqlClient;provider connection
```

```
  string=&quot;data source=localhost;initial catalog=thuchanh;user
```

```
  id=qlsv;password=123456;encrypt=False;MultipleActiveResultSets=True;App=EntityFramework&quot;;
```

```
  providerName="System.Data.EntityClient" />
```

```
</connectionStrings>
```

❖ Xây dựng các Form chức năng của chương trình

- Tạo form **frmKhoa**: quản lý danh mục các khoa, gồm các Toolbox chức năng sau:

The screenshot shows the 'frmKhoa' application window with the following components:

- Form Fields:**
 - Mã khoa: (Callout: TextBox: txtMaKhoa)
 - Tên khoa: (Callout: TextBox: txtTenKhoa)
- Toolbar:**
 - Bỏ qua
 - Thêm (Callout: Button: cmdAdd)
 - Lưu thêm (Callout: Button: cmdAddSave)
 - Lưu sửa (Callout: Button: cmdEditSave)
 - Xóa (Callout: Button: cmdDelete)
 - Xuất Excel (Callout: Button: cmdExportExcel)
 - Thoát (Callout: Button: cmdExit)
- DataGridView:** myGridView (Callout: DataGridView: myGridView)

ID	Mã khoa	Tên khoa
10	CB	Khoa học Cơ bản và Ứng dụng
11	CK	Cơ khí
12	DT	Kỹ thuật Ô tô & Máy Động lực
13	KT	Điện
14	QC	Kinh tế Công nghiệp
11	QT	Quốc tế
12	SP	Công nghệ cơ điện và điện tử
13	XD	Xây dựng và Môi trường

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

▪ Các phương thức trong frmKhoa

```
namespace WinFormsApp2
{
    4 references
    public partial class frmKhoa : Form
    {
        thuchanhEntities db = new thuchanhEntities();
        int idRow = -1;
        1 reference
        public frmKhoa()
        {
            InitializeComponent();
        }
    }
}
```

Khai báo và khởi tạo biến kết nối dữ liệu

```
//Tự động gọi và thực hiện khi mở Form ứng dụng
1 reference
private void frmKhoa_Load(object sender, EventArgs e)
{
    try
    {
        //Thiết lập trạng thái
        statusWindow("Load");
        //Đọc dữ liệu vào GridView
        loadDataGridView();
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString());
    }
}
```

```
//Đọc dữ liệu và GridView
3 references
private void loadDataGridView()
{
    var source = new BindingSource();
    List<Khoa> list = new List<Khoa>();
    list = db.Khoas.ToList();
    source.DataSource = list;
    myGridView.DataSource = source;
    myGridView.AutoGenerateColumns = false;
}
```

```
//Bấm nút Thêm
1 reference
private void cmdAdd_Click(object sender, EventArgs e)
{
    statusWindow("Add");//Thiết lập trạng thái trên Form
}

//Bấm nút Lưu thêm
1 reference
private void cmdSaveAdd_Click(object sender, EventArgs e)
{
    try
    {
        // Yêu cầu cần nhập Mã là khóa chính
        if (string.IsNullOrEmpty(txtMaKhoa.Text))
        {
            MessageBox.Show("Nhập mã khoa...!", "Kết quả");
            txtMaKhoa.Focus();
            return;
        }
        //Kiểm tra và Cập nhật bổ sung thêm thông tin
        var obj = db.Khoas.Where(x => x.MaKhoa == txtMaKhoa.Text).FirstOrDefault();
        if (obj == null) //Nếu chưa có thông tin cần nhập
        {
            var obj1 = new Khoa();
            obj1.MaKhoa = txtMaKhoa.Text;
            obj1.TenKhoa = txtTenKhoa.Text;
            db.Khoas.Add(obj1);
            db.SaveChanges();
            loadDataGridView(); //Đọc lại dữ liệu và GridView
            statusWindow("AddSave");//Thiết lập trạng thái trên Form
            MessageBox.Show("Bổ sung thông tin thành công...!", "Kết quả");
        }
        else//Đã tồn tại thông tin cần nhập
        {
            MessageBox.Show("Thông tin đã có...!", "Kết quả");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString());
    }
}
```

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

▪ Các phương thức trong frmKhoa

```
//Bấm nút Lưu sửa
1 reference
private void cmdEditSave_Click(object sender, EventArgs e)
{
    try
    {
        statusWindow("EditSave");//Thiết lập trạng thái trên Form
        MessageBox.Show(EditSave(),"Kết quả");//Lưu dữ liệu chỉnh sửa và thông báo kết quả
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString());
    }
}
```

```
//Lưu dữ liệu chỉnh sửa
1 reference
private string EditSave()
{
    string str = "";
    if(idRow != -1) //Kiểm tra đã bấm vào dòng trên GridView chưa
    {
        if(!string.IsNullOrEmpty(txtMaKhoa.Text))
        {
            var obj = db.Khoas.Where(x => x.MaKhoa==txtMaKhoa.Text).FirstOrDefault();
            obj.TenKhoa = txtTenKhoa.Text;
            db.Entry(obj).State = EntityState.Modified;
            db.SaveChanges();
            myGridView.Refresh();//Làm mới GridView
            str = "Ghi dữ liệu sửa thành công...!";
        }
        else
        {
            str = "Vui lòng chọn nội dung cần sửa...!";
        }
    }
    return str;
}
```

```
//Hiển thị dòng thông tin Gridview lên các Textbox khi click chọn dòng trong GridView
1 reference
private void selectRowGridview(int idRow)
{
    try
    {
        txtMaKhoa.Text = myGridView.Rows[idRow].Cells[1].Value.ToString().Trim();
        txtTenKhoa.Text = myGridView.Rows[idRow].Cells[2].Value.ToString().Trim();
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString());
    }
}
```

```
//Khi click chọn dòng trên dòng của GridView
1 reference
private void myGridView_RowEnter(object sender, DataGridViewCellEventArgs e)
{
    idRow = e.RowIndex;//Lấy chỉ số của dòng
    selectRowGridview(idRow);//Hiển thị dòng thông tin Gridview lên các Textbox
}
```

```
//Bấm nút Bỏ qua khi đã bấm nút Thêm
1 reference
private void cmdCancel_Click(object sender, EventArgs e)
{
    statusWindow("Cancel");//Thiết lập trạng thái trên Form
}
```

```
// Thoát tính năng
1 reference
private void cmdExit_Click(object sender, EventArgs e)
{
    this.Close();
}
```

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

▪ Các phương thức trong `frmKhoa`

```
namespace WinFormsApp2
{
    1 reference
    class Service
    {
        //Xuất dữ liệu từ GridView ra Excel
        1 reference
        public static void exportExcel(DataGridView myGridView)
        {
            try
            {
                Microsoft.Office.Interop.Excel._Application app = new Microsoft.Office.Interop.Excel.Application();
                Microsoft.Office.Interop.Excel._Workbook workbook = app.Workbooks.Add(Type.Missing);
                Microsoft.Office.Interop.Excel._Worksheet worksheet = null;
                app.Visible = true;
                worksheet = workbook.Sheets["Sheet1"];
                worksheet = workbook.ActiveSheet;
                worksheet.Name = "Data";
                try
                {
                    for (int i = 1; i < myGridView.Columns.Count + 1; i++)
                    {
                        worksheet.Cells[1, i] = myGridView.Columns[i - 1].HeaderText;
                    }
                    for (int i = 0; i < myGridView.Rows.Count - 0; i++)
                    {
                        for (int j = 0; j < myGridView.Columns.Count; j++)
                        {
                            if (myGridView.Rows[i].Cells[j].Value != null)
                            {
                                worksheet.Cells[i + 2, j + 1] = myGridView.Rows[i].Cells[j].Value.ToString();
                            }
                            else
                            {
                                worksheet.Cells[i + 2, j + 1] = "";
                            }
                        }
                    }
                }
            }
            //Mở hộp thoại để lưu file excel vào thư mục
        }
    }
}
```

```
//Mở hộp thoại để lưu file excel vào thư mục
SaveFileDialog saveDialog = new SaveFileDialog();
saveDialog.Filter = "Excel files (*.xlsx)|*.xlsx|All files (*.*)|*.*";
saveDialog.FilterIndex = 2;

if (saveDialog.ShowDialog() == DialogResult.OK)
{
    string pathFull = saveDialog.FileName.ToString();
    workbook.SaveAs(saveDialog.FileName);
    MessageBox.Show("Xuất dữ liệu thành công: "+pathFull, "Kết quả",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
}
catch (System.Exception ex)
{
    MessageBox.Show(ex.Message);
}
finally
{
    app.Quit();
    workbook = null;
    worksheet = null;
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message.ToString());
}
}
```

```
namespace WinFormsApp2
{
    1 reference
    class Service
    {
        //Xuất dữ liệu từ GridView ra Excel
        1 reference
        public static void exportExcel(Data
        {
            try
            {
                Microsoft.Office.Interop.Ex
                Microsoft.Office.Interop.Ex
                Microsoft.Office.Interop.Ex
                app.Visible = true;
            }
        }
    }
}
```

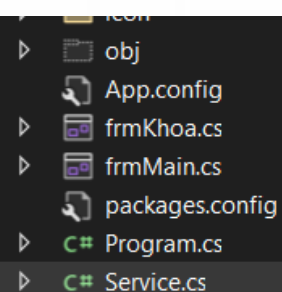
- References
- ADO.NET.Entity
- bin
- icon
- obj
- App.config
- frmKhoa.cs
- frmMain.cs
- packages.config
- C# Program.cs
- C# Service.cs

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

▪ Các phương thức trong frmKhoa

```
// Bấm nút Xóa
1 reference
private void cmdDelete_Click(object sender, EventArgs e)
{
    try
    {
        // Kiểm tra xem đã chọn dòng cần xóa chưa
        if (string.IsNullOrEmpty(txtMaKhoa.Text))
        {
            MessageBox.Show("Chưa chọn thông tin cần xóa!", "Kết quả");
            return;
        }
        if (DialogResult.Yes == MessageBox.Show("Có chắc chắn xóa thông tin không...?",
            "Thông báo", MessageBoxButtons.YesNo, MessageBoxIcon.Warning))
        {
            var obj = db.Khoas.Where(x => x.MaKhoa == txtMaKhoa.Text).FirstOrDefault();
            db.Khoas.Remove(obj);
            db.SaveChanges();
            loadDataGridView(); //Đọc lại dữ liệu và GridView
            MessageBox.Show("Xóa thông tin thành công...!", "Kết quả");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString());
    }
}
```

```
// Nút bấm Xuất Excel
1 reference
private void cmdExportExcel_Click(object sender, EventArgs e)
{
    try
    {
        Service.exportExcel(myGridView); //Gọi phương thức xuất Excel từ lớp Service
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message.ToString());
    }
}
```



```
obj
App.config
frmKhoa.cs
frmMain.cs
packages.config
C# Program.cs
C# Service.cs
```

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

▪ Các phương thức trong frmKhoa

```
//Thiết lập trạng thái Form
5 references
private void statusWindow(string click)
{
    switch(click)
    {
        //Khi mở Form
        case "Load":
            cmdAddSave.Enabled = false;
            cmdCancel.Enabled = false;
            txtMaKhoa.ReadOnly = true;
            myGridView.ReadOnly = true; //Chỉ cho phép đọc
            myGridView.AllowUserToAddRows = false; //Không cho thêm dòng trực tiếp
            myGridView.AllowUserToDeleteRows = false; //Không cho xóa dòng trực tiếp
            myGridView.AllowUserToOrderColumns = true; //Cho phép sắp xếp
            myGridView.AutoGenerateColumns = false;
            break;

        //Khi bấm nút Thêm
        case "Add":
            txtMaKhoa.Text = "";
            txtTenKhoa.Text = "";
            txtMaKhoa.ReadOnly = false;
            txtMaKhoa.Focus();
            cmdAdd.Enabled = false;
            cmdAddSave.Enabled = true;
            cmdEditSave.Enabled = true;
            cmdCancel.Enabled = true;
            cmdEditSave.Enabled = false;
            cmdDelete.Enabled = false;
            myGridView.Enabled = false;
            break;

        //Khi bấm nút Lưu thêm
        case "AddSave":
            cmdEditSave.Enabled = true;
            cmdCancel.Enabled = false;
            cmdAdd.Enabled = true;
            cmdAddSave.Enabled = false;
            cmdDelete.Enabled = true;
            txtMaKhoa.ReadOnly = true;
            myGridView.Enabled = true;
            myGridView.Refresh();
            break;
    }
}
```

```
//Khi bấm nút Lưu thêm
case "AddSave":
    cmdEditSave.Enabled = true;
    cmdCancel.Enabled = false;
    cmdAdd.Enabled = true;
    cmdAddSave.Enabled = false;
    cmdDelete.Enabled = true;
    txtMaKhoa.ReadOnly = true;
    myGridView.Enabled = true;
    myGridView.Refresh();
    break;

//Khi bấm nút Lưu sửa
case "EditSave":
    cmdAdd.Enabled = true;
    cmdCancel.Enabled = false;
    txtMaKhoa.ReadOnly = true;
    myGridView.Refresh();
    break;

//Khi bấm nút Bỏ sửa
case "Cancel":
    txtMaKhoa.Text = "";
    txtTenKhoa.Text = "";
    cmdCancel.Enabled = false;
    cmdAdd.Enabled = true;
    cmdAddSave.Enabled = false;
    cmdEditSave.Enabled = true;
    cmdDelete.Enabled = true;
    txtMaKhoa.ReadOnly = true;
    myGridView.Enabled = true;
    break;
}
```

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

- **Tạo form frmNganh:** quản lý danh mục các ngành đào tạo. Thực hiện tương tự như form frmKhoa

Quản lý sinh viên

Quản lý danh mục | Quản lý sinh viên | Quản lý điểm | Hệ thống

Danh mục khoa
Danh mục ngành
Danh mục lớp
Danh mục học phần
Danh mục dân tộc
Danh mục học kỳ

QLSV

QUẢN LÝ DANH MỤC NGÀNH

Mã ngành: 00

Tên ngành: Đại cương

Bỏ qua | Thêm | Lưu thêm | Lưu sửa | Xóa | Xuất Excel | Thoát

ID	Mã ngành	Tên ngành
1129	00	Đại cương
1003	11	Cơ khí Chế tạo Máy
1004	110	Kỹ thuật cơ khí
1005	113	Kỹ thuật Cơ điện tử
1006	116	Kỹ thuật vật liệu
1007	118	Công nghệ kỹ thuật cơ khí
1008	14NC	Cơ khí động lực
1009	17	Công nghệ ô tô
1010	18	Công nghệ gia công cắt gọt
1011	19	Liên Ngành Cơ - Điện
1012	200	Kỹ thuật Môi trường
1013	220	Xây dựng dân dụng và công nghiệp
1014	222	Kiến trúc
1015	231	Quản lý môi trường công nghiệp và đô thị
1016	40	Kỹ thuật Điện

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

- **Tạo form frmLop:** quản lý danh mục các lớp trong trường. Thực hiện tương tự như form **frmKhoa**

Quản lý sinh viên

Quản lý danh mục | Quản lý sinh viên | Quản lý điểm | Hệ thống

Danh mục khoa
Danh mục ngành
Danh mục lớp
Danh mục học phần
Danh mục dân tộc
Danh mục học kỳ

QLSV

QUẢN LÝ DANH SÁCH LỚP

Mã lớp:

Tên lớp:

Niên khóa:

Khoa:

Ngành:

Combobox: txtMakhoa
Combobox: txtMang

Bỏ qua | Thêm | Lưu thêm | Lưu sửa | Xóa | Xuất Excel | Thoát

ID	MALOP	Tên lớp	Niên khóa	MAKH	Tên khoa	MANGANH	Tên ngành
▶ 2306	111102141	K38MD(K38CDL.01)	2002-2007	CK	Cơ khí	14NC	Cơ khí động lực
2307	111106991	Ghép 1	2006-2011	CB	Khoa học Cơ bản và Ứng dụng	110	Kỹ thuật cơ khí
2308	111106992	Ghép 2	2006-2011	CB	Khoa học Cơ bản và Ứng dụng	110	Kỹ thuật cơ khí
2309	111106993	Ghép 3	2006-2011	CB	Khoa học Cơ bản và Ứng dụng	110	Kỹ thuật cơ khí
2310	111109203	K45KXC.03	2009-2014	XD	Xây dựng và Môi trường	A20	Kỹ thuật Xây dựng Công trình
2311	111110201	K46KXC.01	2010-2015	XD	Xây dựng và Môi trường	A20	Kỹ thuật Xây dựng Công trình
2312	111110541	K46ĐĐK.01	2010-2015	DT	Điện tử	A54	Kỹ thuật điều khiển
2313	111110542	K46ĐĐK.02	2010-2015	DT	Điện tử	A54	Kỹ thuật điều khiển
2314	111111521	K47KĐT.01	2011-2016	DT	Điện tử	B52	Kỹ thuật điện tử
2315	111112201	K48KXC.01	2012-2017	XD	Xây dựng và Môi trường	B20	Xây dựng dân dụng và công nghiệp
2316	111113111	K49CCM.01	2013-2018	CK	Cơ khí	B11	Cơ khí chế tạo máy
2317	111113112	K49CCM.02	2013-2018	CK	Cơ khí	B11	Cơ khí chế tạo máy
2318	111113113	K49CCM.03	2013-2018	CK	Cơ khí	B11	Cơ khí chế tạo máy
2319	111113151	K49KTK.01	2013-2018	CK	Cơ khí	B15	Thiết kế và chế tạo cơ khí

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

- Tạo thủ tục `spDmLop`: Thủ tục kết nối các bảng dữ liệu để hiển thị đầy đủ thông tin lớp trên Gridview

```
USE [thuchanh]
GO
/***** Object: StoredProcedure [dbo].[spDmLop]    Script Date: 5/11/2024 4:29:12 PM *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

ALTER PROCEDURE [dbo].[spDmLop]
AS
BEGIN
SELECT dbo.Lop.id, dbo.Lop.MaLop, dbo.Lop.TenLop, dbo.Lop.NiemKhoa, dbo.Lop.MaKhoa, dbo.Khoa.TenKhoa, dbo.Lop.MaNganh, dbo.Nganh.TenNganh
FROM    dbo.Lop INNER JOIN
        dbo.Khoa ON dbo.Lop.MaKhoa = dbo.Khoa.MaKhoa INNER JOIN
        dbo.Nganh ON dbo.Lop.MaNganh = dbo.Nganh.MaNganh
END
```

100 % <

Results Messages

Kết quả thực thi thủ tục

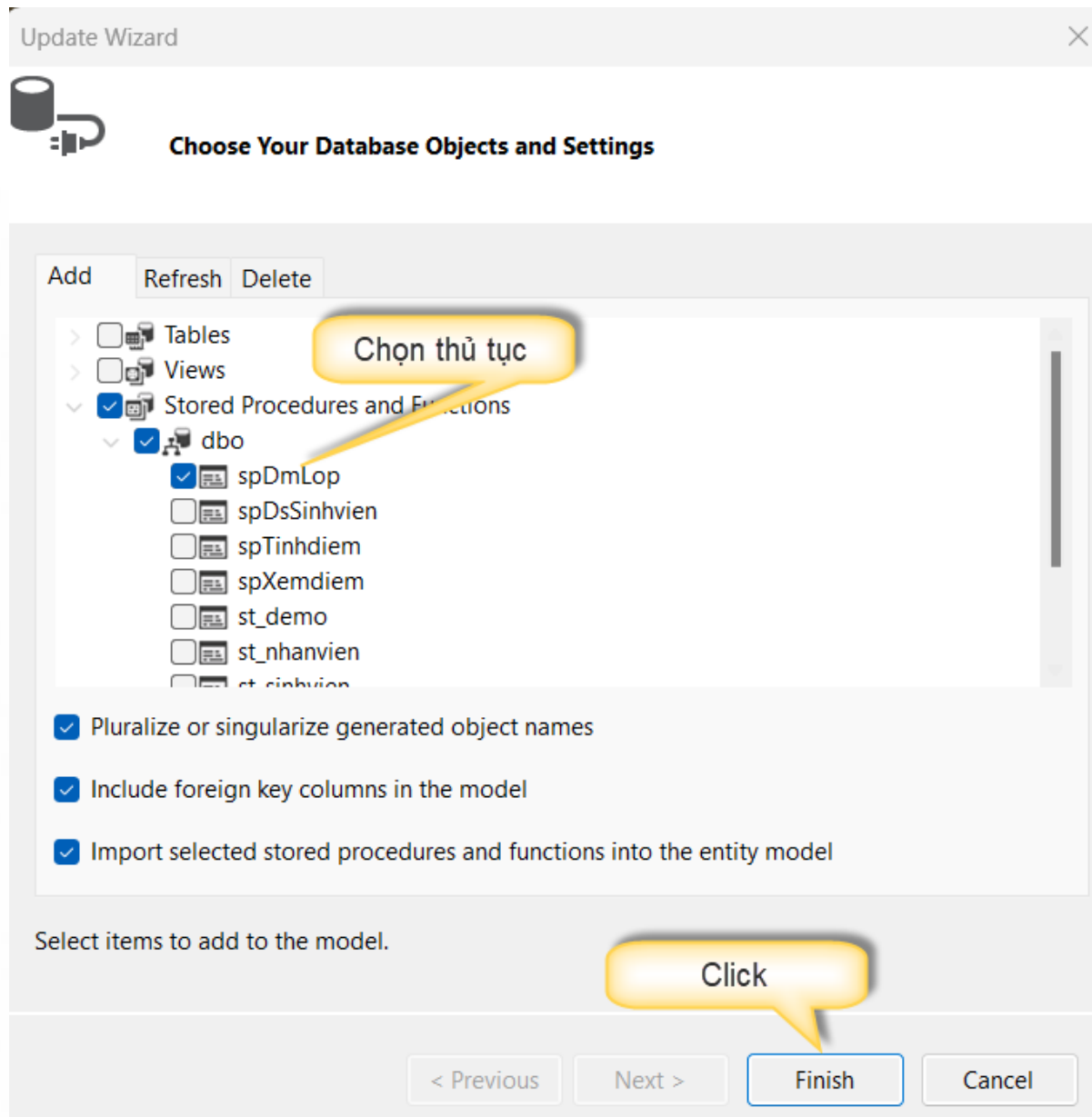
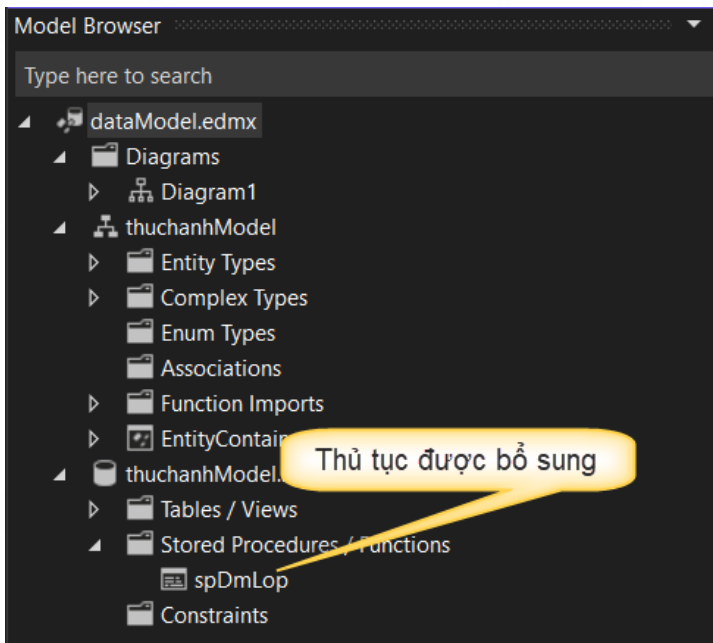
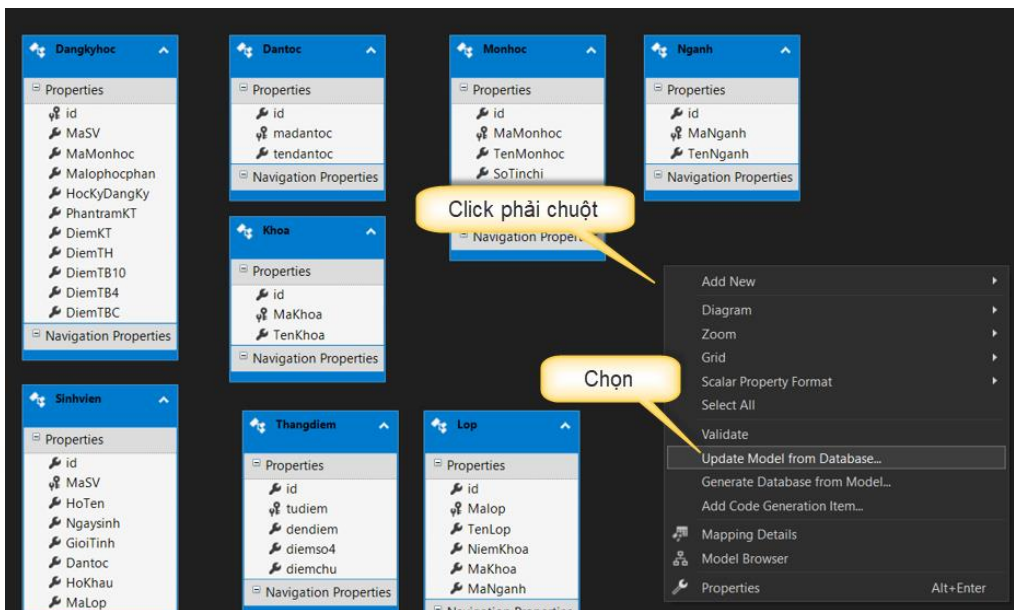
	id	Malop	TenLop	NiemKhoa	MaKhoa	TenKhoa	MaNganh	TenNganh
1	2306	111102141	K38MD(K38CĐL.01)	2002-2007	CK	Cơ khí	14NC	Cơ khí động lực
2	2307	111106991	Ghép 1	2006-2011	CB	Khoa học Cơ bản và Ứng dụng	110	Kỹ thuật cơ khí
3	2308	111106992	Ghép 2	2006-2011	CB	Khoa học Cơ bản và Ứng dụng	110	Kỹ thuật cơ khí
4	2309	111106993	Ghép 3	2006-2011	CB	Khoa học Cơ bản và Ứng dụng	110	Kỹ thuật cơ khí
5	2310	111109203	K45KXC.03	2009-2014	XD	Xây dựng và Môi trường	A20	Kỹ thuật Xây dựng Công trình
6	2311	111110201	K46KXC.01	2010-2015	XD	Xây dựng và Môi trường	A20	Kỹ thuật Xây dựng Công trình
7	2312	111110541	K46ĐĐK.01	2010-2015	DT	Điện tử	A54	Kỹ thuật điều khiển
8	2313	111110542	K46ĐĐK.02	2010-2015	DT	Điện tử	A54	Kỹ thuật điều khiển
9	2314	111111521	K47KĐT.01	2011-2016	DT	Điện tử	B52	Kỹ thuật điện tử
10	2315	111112201	K48KXC.01	2012-2017	XD	Xây dựng và Môi trường	B20	Xây dựng dân dụng và công nghiệp

Return Val...

1	0
---	---

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

- Bổ sung thủ tục **spDmLop** vào CSDL



❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

- Một số phương thức bổ sung

```
//Tự động gọi và thực hiện khi mở Form ứng dụng
1 reference
private void frmLop_Load(object sender, EventArgs e)
{
    try
    {
        //Thiết lập trạng thái trên Form
        statusWindow("Load");
        //Tải dữ liệu vào Combobox
        var listKhoa = db.Khoas.OrderBy(x => x.TenKhoa).ToList();
        Service.loadCombobox(txtMakhoa, listKhoa, "TenKhoa", "MaKhoa");
        var listNganh = db.Nganhs.OrderBy(x => x.TenNganh).ToList();
        Service.loadCombobox(txtMang, listNganh, "TenNganh", "MaNganh");
        //Đọc dữ liệu vào GridView
        loadDataGridView();
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString());
    }
}
```

```
//Hiển thị item tương ứng khi được chọn ở GridView
2 references
public static void selectedCombox(string textDisplay, ComboBox myCombobox)
{
    for (int i = 0; i < myCombobox.Items.Count; i++)
    {
        object a = myCombobox.Items[i];
        object ae = myCombobox.GetItemText(a);
        if (ae.ToString() == textDisplay)
        {
            myCombobox.SelectedIndex = i;
            break;
        }
    }
}
```

```
//Tải dữ liệu vào Combobox
2 references
public static void loadCombobox(ComboBox myCombobox, object obj, string DisplayMember, string ValueMember)
{
    myCombobox.DataSource = new BindingSource(obj, null);
    myCombobox.DisplayMember = DisplayMember;
    myCombobox.ValueMember = ValueMember;
}
```

```
//Hiển thị dòng thông tin Gridview lên các Textbox, Combobox khi click chọn dòng trong GridView
1 reference
private void selectRowGridview(int idRow)
{
    try
    {
        txtMaLop.Text = Convert.ToString(myGridView.Rows[idRow].Cells[1].Value);
        txtTenLop.Text = Convert.ToString(myGridView.Rows[idRow].Cells[2].Value);
        txtNienKhoa.Text = Convert.ToString(myGridView.Rows[idRow].Cells[3].Value);
        //Lấy tên khoa, ngành ở dòng tương ứng trong GridView
        string tenkhoa = Convert.ToString(myGridView.Rows[idRow].Cells[5].Value);
        string tenng = Convert.ToString(myGridView.Rows[idRow].Cells[7].Value);
        //Gọi phương thức để hiển thị lựa chọn khoa, ngành trong Combobox txtMakhoa và txtMang
        Service.selectedCombox(tenkhoa, txtMakhoa);
        Service.selectedCombox(tenng, txtMang);
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString());
    }
}

//Khi click chọn dòng trên dòng của GridView
1 reference
private void myGridView_RowEnter(object sender, DataGridViewCellEventArgs e)
{
    idRow = e.RowIndex; //Lấy chỉ số của dòng
    selectRowGridview(idRow); //Hiển thị dòng thông tin Gridview lên các Textbox
}
```

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

- Một số phương thức bổ sung

```
private void loadDataGridView()  
{  
  
    using (thuchanhEntities context = new thuchanhEntities())  
    {  
        IEnumerable<lopModel> list = context.Database.SqlQuery<lopModel>("spDmLop").ToList();  
        var source = new BindingSource();  
        source.DataSource = list;  
        myGridView.DataSource = source;  
        myGridView.AutoGenerateColumns = false;  
    }  
}
```

Thủ tục được gọi

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

- Tạo form **frmHocphan**: quản lý danh mục các học phần. Thực hiện tương tự như form **frmKhoa**

Quản lý sinh viên

Quản lý danh mục | Quản lý sinh viên | Quản lý điểm | Hệ thống

- Danh mục khoa
- Danh mục ngành
- Danh mục lớp
- Danh mục học phần**
- Danh mục dân tộc
- Danh mục học kỳ

QLSV

QUẢN LÝ DANH MỤC HỌC PHẦN

Mã học phần:

Tên học phần:

Số tín chỉ:

ID	Mã học phần	Tên học phần	Tín chỉ
▶ 3560	010000	Hình họa	1
2712	010001	Vẽ kỹ thuật 1a	2
5706	010002	Vẽ kỹ thuật 2a	2
1274	010003	Vẽ kỹ thuật 3	2
1191	010004	Hình họa - Vẽ kỹ thuật	2
1356	010005	Vẽ kỹ thuật 2	2
47	010011	Vẽ kỹ thuật 1	3
5705	010045	Vẽ kỹ thuật	2
5028	010102	Sức bền vật liệu 1a	3
6937	010103	Sức bền vật liệu 2a	2
3301	010132	Sức bền vật liệu	5
5399	010200	Kỹ thuật gia công cơ khí 1	2
6028	010201	Kỹ thuật gia công cơ khí 2	2
5820	010202	Công nghệ gia công phôi 1	2
5538	010203	Công nghệ gia công phôi 2	2
4695	010205	Kỹ thuật gia công cơ khí	4
1547	010200	Kỹ thuật gia công cơ khí 1	2

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

- Tạo form **frmDantoc**: quản lý danh mục các dân tộc. Thực hiện tương tự như form **frmKhoa**

Quản lý sinh viên

Quản lý danh mục | Quản lý sinh viên | Quản lý điểm | Hệ thống

- Danh mục khoa
- Danh mục ngành
- Danh mục lớp
- Danh mục học phần
- Danh mục dân tộc**
- Danh mục học kỳ

QLSV

QUẢN LÝ DANH MỤC DÂN TỘC

Mã dân tộc:

Tên dân tộc:

ID	Mã dân tộc	Tên dân tộc
▶ 1	1	Kinh
10	10	Gia-rai
11	11	Ngái
12	12	Ê-đê
13	13	Ba na
14	14	Xơ-Đăng
15	15	Sán Chay
16	16	Cơ-ho
17	17	Chăm
18	18	Sán Diu
19	19	Hrê
2	2	Tày
20	20	Mnông
21	21	Ra-glai
22	22	Xtiêng

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

- **Tạo form frmHocky:** quản lý danh mục các học kỳ. Thực hiện tương tự như form **frmKhoa**

Quản lý sinh viên

Quản lý danh mục | Quản lý sinh viên | Quản lý điểm | Hệ thống

- Danh mục khoa
- Danh mục ngành
- Danh mục lớp
- Danh mục học phần
- Danh mục dân tộc
- Danh mục học kỳ**

QLSV

QUẢN LÝ DANH MỤC HỌC KỲ

Mã học kỳ:

Tên học kỳ:

ID	Mã học kỳ	Tên học kỳ
1	3-2023-2024	3-2023-2024
▶ 2	1-2023-2024	1-2023-2024
3	2-2023-2024	2-2023-2024

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

- Tạo form **frmSinhvien**: quản lý thông tin sinh viên.

The screenshot displays the 'frmSinhvien' application window, titled 'Quản lý sinh viên'. It features a menu bar with 'Quản lý danh mục', 'Quản lý sinh viên', 'Quản lý điểm', and 'Hệ thống'. A 'Danh sách sinh viên' button is visible. The main area is divided into several sections:

- Thông tin sinh viên:** A form for entering student details. Fields include 'Mã số' (K215480106124), 'Họ tên' (Cao Đức Thành), 'Ngày sinh' (11/08/2003), 'Giới tính' (Nam), 'Dân tộc' (Kinh), and 'Hộ khẩu' (Thái Nguyên). Dropdown menus for 'Mã lớp' (K57KMT.01), 'Khoa' (Điện tử), and 'Ngành' (Tin học công nghiệp) are also present.
- Search and Action Buttons:** A search section with 'Khoa' (Điện tử) and 'Mã lớp' (111121531) dropdowns, and buttons for 'Tìm lớp', 'Vào lớp', and 'Xuất Excel'.
- DataGridView (myGridView1):** A table displaying student records with columns: ID, MALOP, TENLOP, TENKHOA, TENNGANH, NiemKhoa, and others. A callout points to the grid with the text 'DataGridView: myGridView1'.
- Control Callouts:** Several yellow callout boxes identify specific controls: 'TextBox: txtMasv; TextBox: txtHoten; TextBox: txtNgaysinh; ComboBox: txtGioitinh; ComboBox: txtDantoc; TextBox: txtHokhau; ComboBox: txtMaLop; ComboBox: txtMakhoa; ComboBox: txtMang' (pointing to the form fields); 'ComboBox: txtMakhoa' (pointing to the search dropdown); 'TextBox: txtMaLop' (pointing to the search dropdown); 'Button: cmdGoinLop' (pointing to the 'Vào lớp' button); 'Button: cmdExportExcel1' (pointing to the 'Xuất Excel' button); and 'Button: cmdFindLop' (pointing to the 'Tìm lớp' button).
- Bottom DataGridView:** A larger table at the bottom of the window, also labeled 'DataGridView: myGridView', showing a list of students with columns: ID, MSV, HOTEN, NGAYSINH, GIOI, DANTOC, HOKHA, MALOP, TENLOP, NIENKHOA, MAKHOA, TENKHOA, MANGANH, and TENNGANH.

BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

1. Click

2. Click

3. Click: Thêm các cột dữ liệu trong Griview

4. Đặt tên cho cột

5. Tên cột lấy thông tin trong bảng SQL

6. Tên cột trong Gridview

1. Click

2. Click

3. Click: Thêm các cột dữ liệu trong Griview

1. Click

2. Click

3. Nhập

4. Click

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

■ Các phương thức trong form

```
18 using WinFormsApp2.ADO.NET.Entity;
19 using WinFormsApp2.model;
20 using static System.Windows.Forms.VisualStyles.VisualStyleElement;
21
22 namespace WinFormsApp2
23 {
24     - references
25     public partial class frmSinhvien : Form
26     {
27         thuchanhEntities db = new thuchanhEntities();
28         int idRow = -1;
29         - references
30         public frmSinhvien()
31         {
32             InitializeComponent();
33         }
34         //Tự động gọi và thực hiện khi mở Form ứng dụng
35         - references
36         private void frmSinhvien_Load(object sender, EventArgs e)
37         {
38             try
39             {
40                 statusWindow("Load");//Thiết lập trạng thái trên Form
41                 //Tải dữ liệu vào Combobox
42                 var listDantoc = db.Dantocs.OrderBy(x => x.tendantoc).ToList();
43                 Service.loadCombobox(txtDantoc, listDantoc, "tendantoc", "madantoc");
44                 var listLop = db.Lops.OrderBy(x => x.TenLop).ToList();
45                 Service.loadCombobox(txtMaLop, listLop, "TenLop", "MaLop");
46                 var listKhoa = db.Khoas.OrderBy(x => x.TenKhoa).ToList();
47                 Service.loadCombobox(txtMakhoa, listKhoa, "TenKhoa", "MaKhoa");
48                 Service.loadCombobox(txtMakhoaf, listKhoa, "TenKhoa", "MaKhoa");
49                 var listNganh = db.Nganhs.OrderBy(x => x.TenNganh).ToList();
50                 Service.loadCombobox(txtMang, listNganh, "TenNganh", "MaNganh");
51             }
52             catch (Exception ex)
53             {
54                 MessageBox.Show(ex.ToString());
55             }
56         }
57
58         //Bấm nút Thêm
59         - references
60         private void cmdAdd_Click(object sender, EventArgs e)
61         {
62             statusWindow("Add");//Thiết lập trạng thái trên Form
63         }
64     }
65 }
```

```
61 //Bấm nút Lưu thêm
62 - references
63 private void cmdSaveAdd_Click(object sender, EventArgs e)
64 {
65     try
66     {
67         // Yêu cầu cần phải nhập dữ liệu
68         if (string.IsNullOrEmpty(txtMasv.Text))
69         {
70             MessageBox.Show("Nhập mã số..!", "Kết quả");
71             txtMakhoa.Focus();
72             return;
73         }
74         if (string.IsNullOrEmpty(txtMaLop.Text))
75         {
76             MessageBox.Show("Chọn lớp..!", "Kết quả");
77             txtMaLop.Focus();
78             return;
79         }
80         //Kiểm tra và Cập nhật bổ sung thêm thông tin
81         var obj = db.Sinhviens.Where(x => x.MaSV == txtMasv.Text).FirstOrDefault();
82         if (obj == null) //Nếu chưa có thông tin cần nhập
83         {
84             var obj1 = new Sinhvien();
85             obj1.MaSV = txtMasv.Text;
86             obj1.HoTen = txtHoten.Text;
87             obj1.Ngaysinh = txtNgaysinh.Text;
88             obj1.GioiTinh = txtGioitinh.Text;
89             obj1.Dantoc = txtDantoc.Text;
90             obj1.HoKhau = txtHokhau.Text;
91             obj1.MaLop = Convert.ToString(txtMaLop.SelectedValue);
92             db.Sinhviens.Add(obj1);
93             db.SaveChanges();
94             string malop = Convert.ToString(txtMaLop.SelectedValue);
95             loadDataGridView(malop); //Đọc lại dữ liệu và GridView
96             statusWindow("AddSave");//Thiết lập trạng thái trên Form
97             MessageBox.Show("Bổ sung thông tin thành công...!", "Kết quả");
98         }
99         else//Đã tồn tại thông tin cần nhập
100         {
101             MessageBox.Show("Thông tin đã có...!", "Kết quả");
102         }
103     }
104     catch (Exception ex)
105     {
106         MessageBox.Show(ex.ToString());
107     }
108 }
```

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

■ Các phương thức trong form

```
108 //Bấm nút Lưu sửa
109 -references
110 private void cmdEditSave_Click(object sender, EventArgs e)
111 {
112     try
113     {
114         statusWindow("EditSave");//Thiết lập trạng thái trên Form
115         MessageBox.Show(EditSave(),"Kết quả");//Lưu dữ liệu chỉnh sửa và thông báo kết quả
116     }
117     catch (Exception ex)
118     {
119         MessageBox.Show(ex.ToString());
120     }
121 }
122 //Bấm nút Bỏ qua khi đã bấm nút Thêm
123 -references
124 private void cmdCancel_Click(object sender, EventArgs e)
125 {
126     statusWindow("Cancel");//Thiết lập trạng thái trên Form
127 }
128 // Thoát tính năng
129 -references
130 private void cmdExit_Click(object sender, EventArgs e)
131 {
132     this.Close();
133 }
```

```
132 // Bấm nút Xóa
133 -references
134 private void cmDelete_Click(object sender, EventArgs e)
135 {
136     try
137     {
138         // Kiểm tra xem đã chọn dòng cần xóa chưa
139         if (string.IsNullOrEmpty(txtMaLop.Text))
140         {
141             MessageBox.Show("Chưa chọn thông tin cần xóa!", "Kết quả");
142             return;
143         }
144         if (DialogResult.Yes == MessageBox.Show("Có chắc chắn xóa thông tin không..?",
145             "Thông báo", MessageBoxButtons.YesNo, MessageBoxIcon.Warning))
146         {
147             var obj = db.Sinhviens.Where(x => x.MaSV == txtMasv.Text).FirstOrDefault();
148             db.Sinhviens.Remove(obj);
149             db.SaveChanges();
150             string malop = Convert.ToString(txtMaLop.SelectedValue);
151             loadDataGridView(malop); //Đọc lại dữ liệu sinh viên của lớp và Gridview
152             MessageBox.Show("Xóa thông tin thành công...!", "Kết quả");
153         }
154     }
155     catch (Exception ex)
156     {
157         MessageBox.Show(ex.ToString());
158     }
159 }
```

```
159 // Nút bấm Xuất Excel
160 -references
161 private void cmdExportExcel_Click(object sender, EventArgs e)
162 {
163     try
164     {
165         Service.exportExcel(myGridView);//Gọi phương thức xuất Excel từ lớp Service
166     }
167     catch (Exception ex)
168     {
169         MessageBox.Show(ex.Message.ToString());
170     }
171 }
172 // Nút bấm Tìm lớp: Tìm lớp nhập từ bàn phím để hiển thị danh sách sinh viên
173 -references
174 private void cmdFindLop_Click(object sender, EventArgs e)
175 {
176     string malop = txtMaLopf.Text;
177     loadDataGridView1(null, malop);
178 }
179 // Tìm danh sách các lớp thuộc khoa khi được chọn
180 -references
181 private void txtMakhoaf_DropDownClosed(object sender, EventArgs e)
182 {
183     string makhoa = Convert.ToString(txtMakhoaf.SelectedValue);
184     loadDataGridView1(makhoa, null);
185 }
186 // Nút bấm Vào lớp: Liệt kê chi tiết danh sách sinh viên thuộc lớp được chọn
187 -references
188 private void cmdGoInLop_Click(object sender, EventArgs e)
189 {
190     string malop = txtMaLopf.Text;
191     if (!string.IsNullOrEmpty(malop))
192     {
193         loadDataGridView(malop);
194     }
195 }
196 // Hiển thị lớp được chọn vào txtMaLopf.Text khi chọn lớp trong Gridview
197 -references
198 private void myGridView1_RowEnter(object sender, DataGridViewCellEventArgs e)
199 {
200     try
201     {
202         txtMaLopf.Text = Convert.ToString(myGridView1.Rows[e.RowIndex].Cells[1].Value);
203     }
204     catch { }
```

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

■ Các phương thức trong form

```
202 // Xuất Excel danh sách lớp
203 -references
204 private void cmdExportExcel_Click(object sender, EventArgs e)
205 {
206     try
207     {
208         Service.exportExcel(myGridView1); //Gọi phương thức xuất Excel từ lớp Service
209     }
210     catch (Exception ex)
211     {
212         MessageBox.Show(ex.Message.ToString());
213     }
214 }
215 //Đọc dữ liệu sinh viên của một lớp và GridView
216 -references
217 private void loadDataGridView(string malop)
218 {
219     using (thuchanhEntities context = new thuchanhEntities())
220     {
221         //Gọi thủ tục cùng với tham số
222         IEnumerable<sinhvienModel> list = context.Database.SqlQuery<sinhvienModel>("spDsSinhvien @malop = {0}", malop).ToList();
223         var source = new BindingSource();
224         source.DataSource = list;
225         myGridView.DataSource = source;
226         myGridView.AutoGenerateColumns = false;
227     }
228 }
229 //Đọc dữ liệu lớp và GridView: Tìm kiếm theo khoa và lớp
230 -references
231 private void loadDataGridView1(string makhoa, string malop)
232 {
233     using (thuchanhEntities context = new thuchanhEntities())
234     {
235         IEnumerable<lopModel> list = context.Database.SqlQuery<lopModel>("spDmLop").ToList();
236         var source = new BindingSource();
237         //Tìm kiếm theo khoa
238         if (!string.IsNullOrEmpty(makhoa))
239             source.DataSource = list.Where(x => x.MaKhoa == makhoa);
240         else
241             //Tìm kiếm theo lớp
242             source.DataSource = list.Where(x => x.Malop == malop);
243         myGridView1.DataSource = source;
244         myGridView1.AutoGenerateColumns = false;
245     }
246 }
247 }
248 }
249 }
```

Goi thủ tục và truyền tham số

Cần phải định nghĩa dữ liệu trung gian này

```
sinhvienModel.cs
WinFormsApp1
using System.Threading.Tasks;
namespace WinFormsApp2.model
{
    -references
    public class sinhvienModel
    {
        -references
        public int id { get; set; }
        -references
        public string MaSV { get; set; }
        -references
        public string HoTen { get; set; }
        -references
        public string Ngaysinh { get; set; }
        -references
        public string GioiTinh { get; set; }
        -references
        public string Dantoc { get; set; }
        -references
        public string HoKhuu { get; set; }
        -references
        public string Malop { get; set; }
        -references
        public string TenLop { get; set; }
        -references
        public string NiemKhoa { get; set; }
        -references
        public string MaKhoa { get; set; }
        -references
        public string TenKhoa { get; set; }
        -references
        public string MaNganh { get; set; }
        -references
        public string TenNganh { get; set; }
    }
}
```

Solution Explorer

Search Solution Explorer (Ctrl+)

Solution 'Thuchanh' (3 of 3 projects)

- ▶ C# ConsoleApp1
- ▶ C# WinFormsApp1
- ▶ C# WinFormsApp2
 - ▶ Properties
 - ▶ References
 - ▶ ADO.NET.Entity
 - ▶ icon
 - ▶ model
 - ▶ C# lopModel.cs
 - ▶ C# sinhvienModel.cs
 - ▶ sinhvienModel
 - id : int
 - MaSV : string
 - HoTen : string
 - Ngaysinh : string
 - GioiTinh : string
 - Dantoc : string
 - HoKhuu : string
 - Malop : string
 - TenLop : string
 - NiemKhoa : string
 - MaKhoa : string
 - TenKhoa : string
 - MaNganh : string
 - TenNganh : string

App.config

frmDantoc.cs

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

■ Các phương thức trong form

```
250
251 //Lưu dữ liệu chỉnh sửa: Thông tin sinh viên
252 -references
253 private string EditSave()
254 {
255     string str = "";
256     if(idRow != -1) //Kiểm tra đã bấm vào dòng trên GridView chưa
257     {
258         if(!string.IsNullOrEmpty(txtMasv.Text)) //Yêu cầu phải có mã số
259         {
260             var obj = db.Sinhviens.Where(x => x.MaSV==txtMasv.Text).FirstOrDefault();
261             obj.HoTen = txtHoten.Text;
262             obj.Ngaysinh = txtNgaysinh.Text;
263             obj.GioiTinh = txtGioitinh.Text;
264             obj.Dantoc = txtDantoc.Text;
265             obj.HoKhau = txtHokhau.Text;
266             obj.MaLop = Convert.ToString(txtMaLop.SelectedValue);
267             db.Entry(obj).State = EntityState.Modified;
268             db.SaveChanges();
269             string malop = Convert.ToString(txtMaLop.SelectedValue);
270             loadDataGridView(malop); //Đọc lại dữ liệu và GridView
271             str = "Ghi dữ liệu sửa thành công...!";
272         }
273     }
274     else
275     {
276         str = "Vui lòng chọn nội dung cần sửa...!";
277     }
278     return str;
279 }
280 //Thiết lập trạng thái Form
281 -references
282 private void statusWindow(string click)
283 {
284     switch(click)
285     {
286         //Khi mở Form
287         case "Load":
288             cmdAddSave.Enabled = false;
289             cmdCancel.Enabled = false;
290             myGridView.ReadOnly = true; //Chỉ cho phép đọc
291             myGridView.AllowUserToAddRows = false; //Không cho thêm dòng trực tiếp
292             myGridView.AllowUserToDeleteRows = false; //Không cho xóa dòng trực tiếp
293             myGridView.AllowUserToOrderColumns = true; //Cho phép sắp xếp
294             //myGridView.AutoGenerateColumns = false;
295             break;
296         //Khi bấm nút Thêm
297         case "Add":
298             txtMasv.Text = "";
299             txtHoten.Text = "";
300             txtNgaysinh.Text = "";
301             txtGioitinh.Text = "";
302             txtDantoc.Text = "";
303             txtHokhau.Text = "";
304             txtMaLop.Enabled = false;
305             txtMakhoa.Enabled = false;
306             txtMang.Enabled = false;
307             txtMasv.Focus();
```

```
306             cmdAdd.Enabled = false;
307             cmdAddSave.Enabled = true;
308             cmdEditSave.Enabled = true;
309             cmdCancel.Enabled = true;
310             cmdEditSave.Enabled = false;
311             cmdDelete.Enabled = false;
312             myGridView.Enabled = false;
313             break;
314         //Khi bấm nút Lưu thêm
315         case "AddSave":
316             txtMaLop.Enabled = true;
317             txtMakhoa.Enabled = true;
318             txtMang.Enabled = true;
319             cmdEditSave.Enabled = true;
320             cmdCancel.Enabled = false;
321             cmdAdd.Enabled = true;
322             cmdAddSave.Enabled = false;
323             cmdDelete.Enabled = true;
324             myGridView.Enabled = true;
325             myGridView.Refresh();
326             break;
327         //Khi bấm nút Lưu sửa
328         case "EditSave":
329             cmdAdd.Enabled = true;
330             cmdCancel.Enabled = false;
331             myGridView.Refresh();
332             break;
333         //Khi bấm nút Bỏ sửa
334         case "Cancel":
335             txtMasv.Text = "";
336             txtHoten.Text = "";
337             txtNgaysinh.Text = "";
338             txtGioitinh.Text = "";
339             txtDantoc.Text = "";
340             txtHokhau.Text = "";
341             txtMaLop.Enabled = true;
342             txtMakhoa.Enabled = true;
343             txtMang.Enabled = true;
344             cmdCancel.Enabled = false;
345             cmdAdd.Enabled = true;
346             cmdAddSave.Enabled = false;
347             cmdEditSave.Enabled = true;
348             cmdDelete.Enabled = true;
349             //txtMaLop.ReadOnly = true;
350             myGridView.Enabled = true;
351             break;
352     }
353 }
```

❑ BÀI 4: LẬP TRÌNH CSDL VỚI ADO.NET

■ Các phương thức trong form

```
354
355 //Hiển thị dòng thông tin sinh viên GridView lên các Textbox, Combobox khi click chọn dòng trong GridView
356 -references
357 private void selectRowGridview(int idRow)
358 {
359     try
360     {
361         txtMasv.Text = Convert.ToString(myGridView.Rows[idRow].Cells[1].Value);
362         txtHoten.Text = Convert.ToString(myGridView.Rows[idRow].Cells[2].Value);
363         txtNgaysinh.Text = Convert.ToString(myGridView.Rows[idRow].Cells[3].Value);
364         txtHokhau.Text = Convert.ToString(myGridView.Rows[idRow].Cells[6].Value);
365         //Lấy tên khoa, ngành ở dòng tương ứng trong GridView
366         string gioitinh = Convert.ToString(myGridView.Rows[idRow].Cells[4].Value);
367         string tendantoc = Convert.ToString(myGridView.Rows[idRow].Cells[5].Value);
368         string tenlop = Convert.ToString(myGridView.Rows[idRow].Cells[8].Value);
369         string tenkhoa = Convert.ToString(myGridView.Rows[idRow].Cells[11].Value);
370         string tenng = Convert.ToString(myGridView.Rows[idRow].Cells[13].Value);
371         //Gọi phương thức để hiển thị lựa chọn khoa, ngành trong Combobox txtGioitinh, txtDantoc, txtMaLop, txtMakhoa và txtMang
372         Service.selectedCombox(gioitinh, txtGioitinh);
373         Service.selectedCombox(tendantoc, txtDantoc);
374         Service.selectedCombox(tenlop, txtMaLop);
375         Service.selectedCombox(tenkhoa, txtMakhoa);
376         Service.selectedCombox(tenng, txtMang);
377     }
378     catch (Exception ex)
379     {
380         MessageBox.Show(ex.ToString());
381     }
382 }
383 //Khi click chọn dòng trên dòng của GridView
384 -references
385 private void myGridView_RowEnter(object sender, DataGridViewCellEventArgs e)
386 {
387     idRow = e.RowIndex; //Lấy chỉ số của dòng
388     selectRowGridview(idRow); //Hiển thị dòng thông tin sinh viên GridView lên các Textbox, ComboBox
389 }
390
391
392
```

```
Service.cs  X frmMain.cs [Design]  dataModel.edmx [Diagram1]  frmSinhvien.cs*
WinFormsApp2
83
84
85 //Hiển thị item tương ứng khi được chọn ở GridView
86 -references
87 public static void selectedCombox(string textDisplay, ComboBox myCombobox)
88 {
89     for (int i = 0; i < myCombobox.Items.Count; i++)
90     {
91         object a = myCombobox.Items[i];
92         object ae = myCombobox.GetItemText(a);
93         if (ae.ToString() == textDisplay)
94         {
95             myCombobox.SelectedIndex = i;
96             break;
97         }
98     }
99
100 //Tải dữ liệu vào Combobox
101 -references
102 public static void loadCombobox(ComboBox myCombobox, object obj, string DisplayMember, string ValueMember)
103 {
104     myCombobox.DataSource = new BindingSource(obj, null);
105     myCombobox.DisplayMember = DisplayMember;
106     myCombobox.ValueMember = ValueMember;
107 }
108
109
```

▪ Mục đích, yêu cầu

- Tìm hiểu và thao tác với ngôn ngữ LINQ (Language Integrated Query) trong C#
- Xây dựng ứng dụng Window form kết nối dữ liệu XML, Sql Server sử dụng LINQ

▪ Các bài tập vận dụng

