

BÁO CÁO THỰC HÀNH

Môn học: Lập trình mạng căn bản

Buổi báo cáo: Lab 02

Tên chủ đề: Lập trình C# & Windows Forms cơ bản

GVHD: Đỗ Thị Hương Lan

Ngày thực hiện: 30/09/2024

Ngày nộp báo cáo: 06/10/2024

THÔNG TIN CHUNG:

Lớp: NT106.P11.1

STT	Họ và tên	MSSV	Email
1	Nguyễn Lê Nhật Đăng	23520231	23520231@gm.uit.edu.vn
2	Huỳnh Minh Đạt	23520249	23520249@gm.uit.edu.vn

ĐÁNH GIÁ KHÁC:

Nội dung	Kết quả
Tổng thời gian thực hiện bài thực hành trung bình	1 tuần
Link Video thực hiện (nếu có)	
Ý kiến (nếu có) + Khó khăn + Đề xuất ...	

Phần bên dưới của báo cáo này là báo cáo chi tiết của nhóm thực hiện

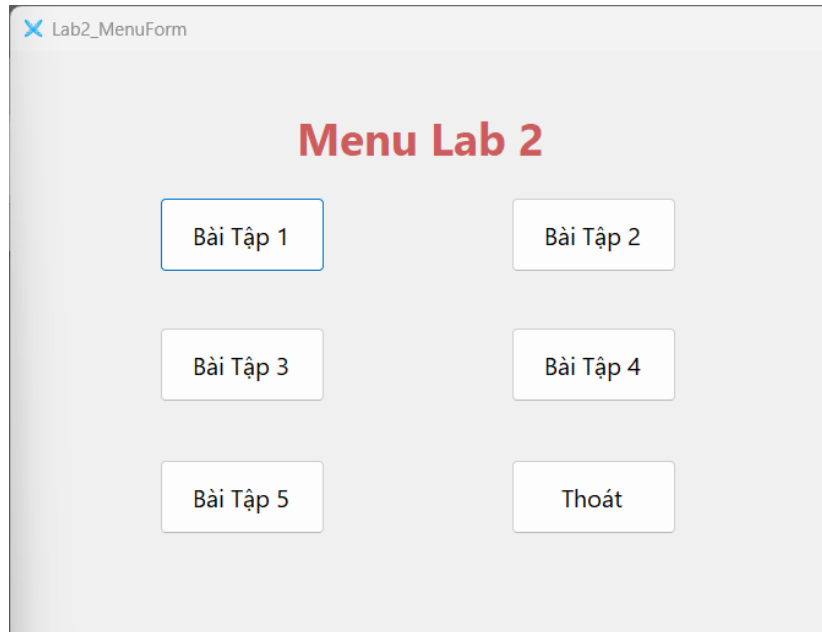
MỤC LỤC

1. Form “Menu” của Lab2:.....	3
a) Tổng quan:	3
b) Chi tiết:	3
2. Bài tập 1 – Đọc và ghi file cơ bản (.txt):.....	4
a) Tổng quan:	4
b) Chi tiết:	4
3. Bài tập 2 – Đọc thông tin tập tin:	7
a) Tổng quan:	7
b) Chi tiết:	7
4. Bài tập 3 – Đọc và ghi file (mở rộng):	10
a) Tổng quan:	10
b) Chi tiết:	10
5. Bài tập 4 – Làm việc với file/CSDL:.....	13
a) Tổng quan:	13
b) Chi tiết:	13
6. Bài tập 5 –Duyệt thư mục:.....	21
a) Tổng quan:	21
b) Chi tiết:	21

BÁO CÁO CHI TIẾT

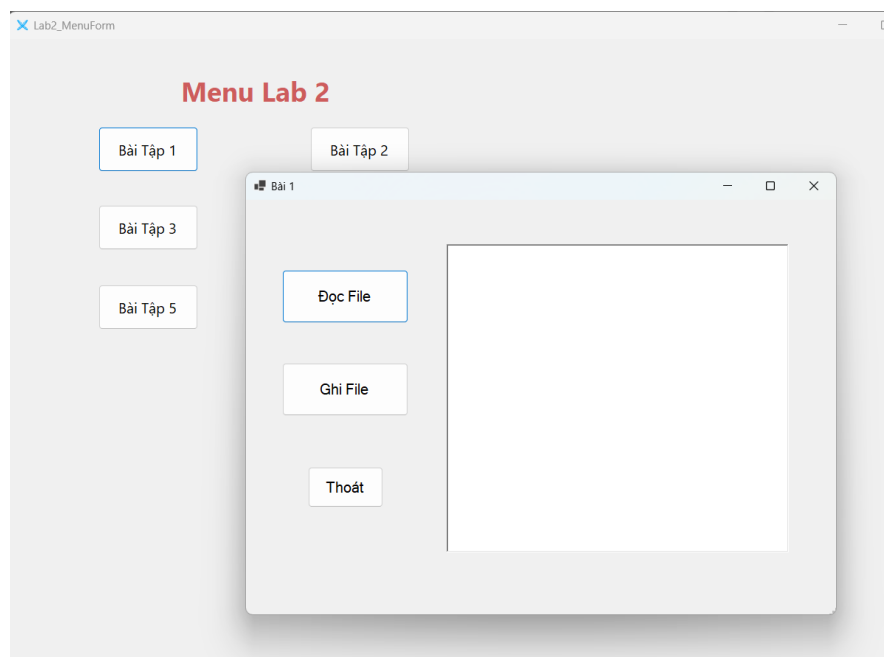
1. Form “Menu” của Lab02:

a) Tổng quan:



b) Chi tiết:

- Sử dụng label làm tiêu đề cho form và các button để thực hiện chức năng mở form bài tập mong muốn.
- Sử dụng button để mở form bài tập tương ứng:
- Sử dụng button “Thoát” để thoát chương trình:



```
private void btn_Exit_Click(object sender, EventArgs e)
{
    Application.Exit();
}

1 reference
private void btn_BaiTap1_Click(object sender, EventArgs e)
{
    Lab2_BaiTap1 BaiTap1 = new Lab2_BaiTap1();
    BaiTap1.ShowDialog();
}

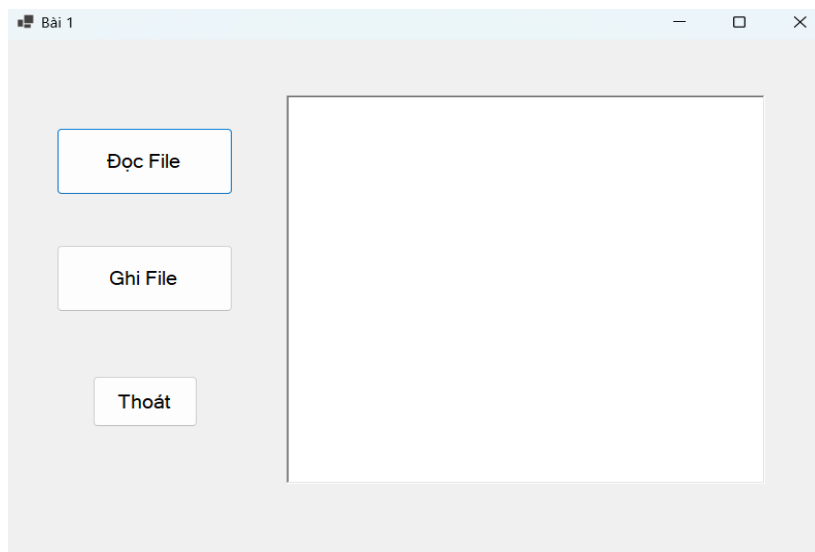
1 reference
private void btn_BaiTap2_Click(object sender, EventArgs e)
{
    Lab2_BaiTap2 BaiTap2 = new Lab2_BaiTap2();
    BaiTap2.ShowDialog();
}

1 reference
private void btn_BaiTap4_Click(object sender, EventArgs e)
{
    Lab2_BaiTap4 BaiTap4 = new Lab2_BaiTap4();
    BaiTap4.ShowDialog();
}

1 reference
private void btn_BaiTap5_Click(object sender, EventArgs e)
{
    Lab2_BaiTap5 BaiTap5 = new Lab2_BaiTap5();
    BaiTap5.ShowDialog();
}
```

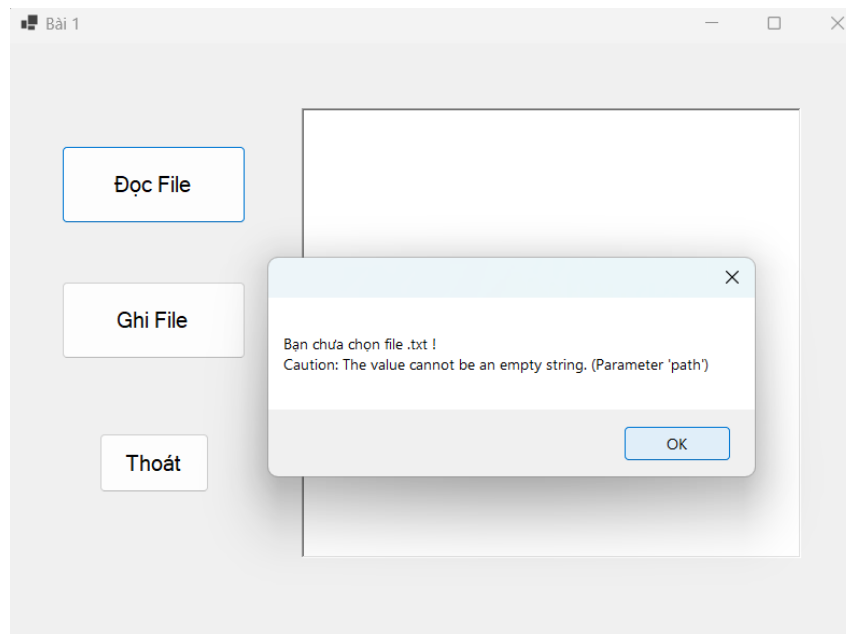
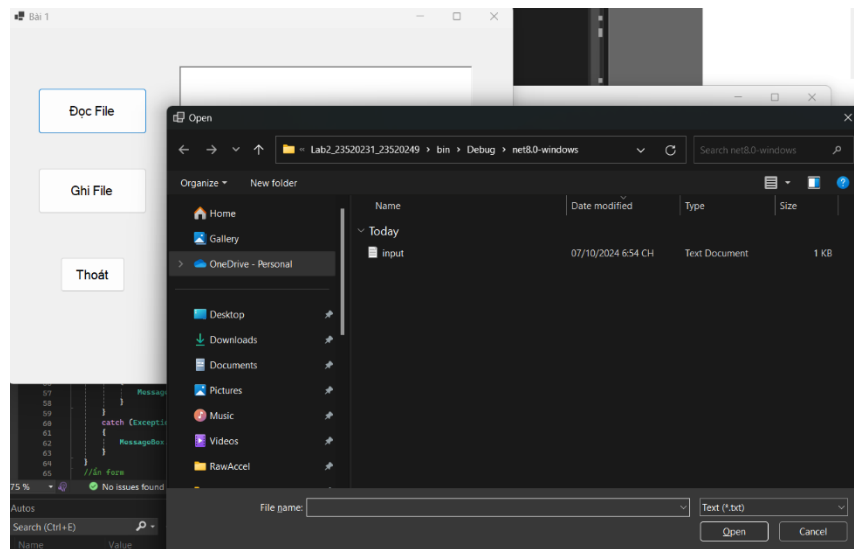
2. Bài tập 1 – Đọc và ghi file cơ bản (.txt)

a) Tổng quan:



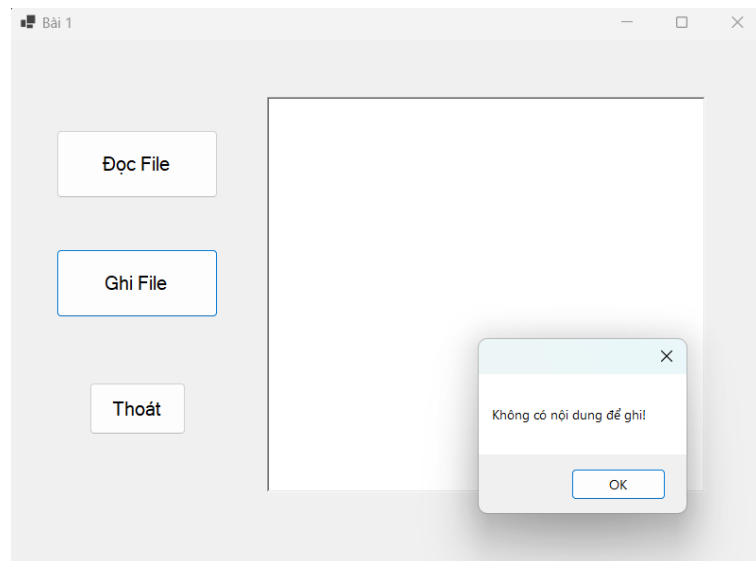
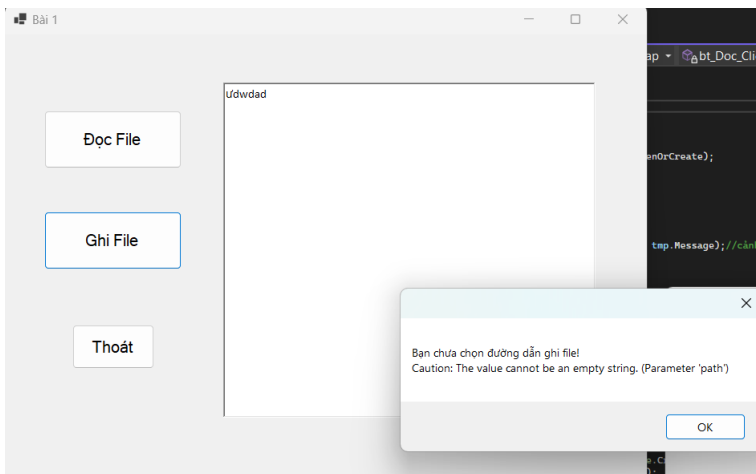
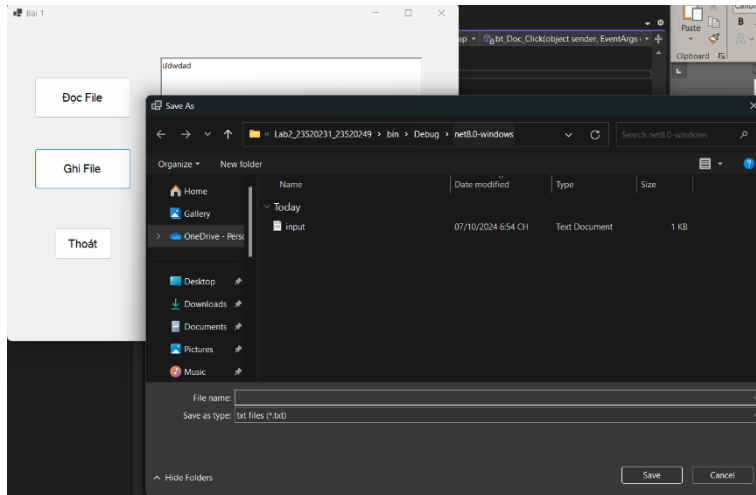
b) Chi tiết:

- Sử dụng Filter để lọc file văn bản “.txt” trong quá trình chọn file và chọn đường dẫn lưu file.
- Đọc file được chọn và sử dụng cấu trúc try-catch kèm hộp thoại để thông báo khi người dùng không chọn file.



```
try
{
    OpenFileDialog ofd = new OpenFileDialog();
    ofd.Filter = "Text (*.txt)|*.txt";//lọc file .txt
    ofd.ShowDialog();
    FileStream fs = new FileStream(ofd.FileName, FileMode.OpenOrCreate);
    StreamReader sr = new StreamReader(fs);
    rtbx_Show.Text = sr.ReadToEnd();
    fs.Close();
}
catch (Exception tmp)
{
    MessageBox.Show("Bạn chưa chọn file .txt !\nCaution: " + tmp.Message);//cảnh báo khi xảy ra lỗi
}
```

- Sử dụng phương thức ToUpper để in hoa nội dung hiển thị và ghi thành file .txt với tên và đường dẫn mong muốn, cấu trúc try-catch kèm hộp thoại tiếp tục được sử dụng để thông báo khi người dùng không chọn đường dẫn để lưu file.
- Khi nội dung ghi là rỗng sẽ có hộp thoại thông báo và không tiến hành ghi.



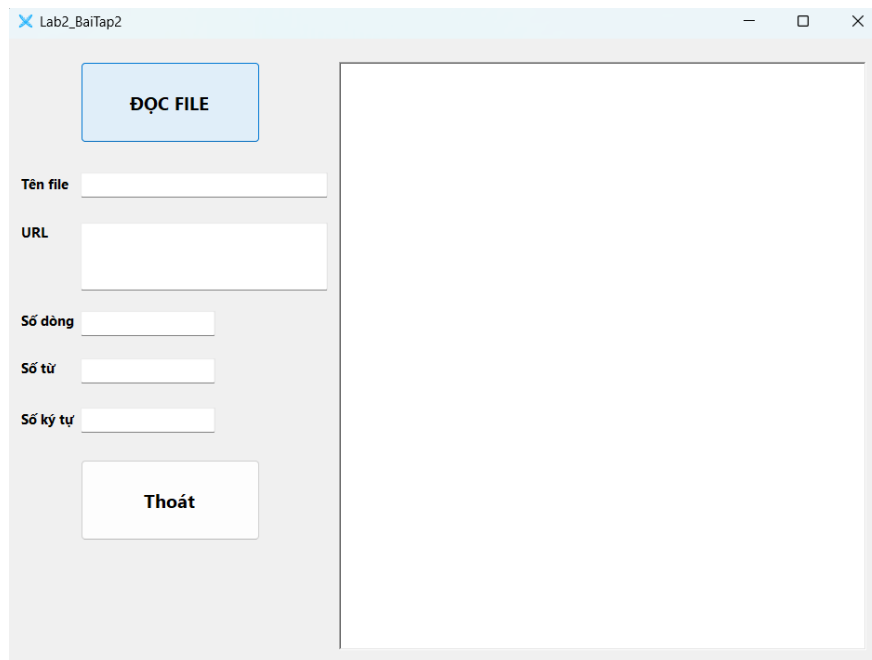
```

try
{
    if (rtbx_Show.Text.Length != 0)
    {
        string str = rtbx_Show.Text;
        str = str.ToUpper(); //Viết hoa toàn bộ ký tự
        SaveFileDialog sfd = new SaveFileDialog();
        sfd.Filter = "txt files (*.txt)|*.txt";
        sfd.ShowDialog();
        FileStream fs = new FileStream(sfd.FileName, FileMode.Create);
        StreamWriter sw = new StreamWriter(fs, Encoding.UTF8);
        sw.WriteLine(str);
        sw.Close();
        fs.Close();
    }
    else
    {
        MessageBox.Show("Không có nội dung để ghi!");
    }
}
catch (Exception tmp)
{
    MessageBox.Show("Bạn chưa chọn đường dẫn ghi file!\nCaution: " + tmp.Message);
}

```

3. Bài tập 2 – Đọc thông tin tập tin:

a) Tổng quan:



b) Chi tiết:

- Sử dụng Filter để lọc file văn bản “.txt” trong quá trình chọn file và chọn đường dẫn lưu file.
- Lấy tên File bằng phương thức SafeFileName.
- Lấy đường dẫn URL bằng phương thức Name.
- Tính kích thước của File, số dòng, số từ, số ký tự trong File bằng các hàm có sẵn trong C#
- Hộp thoại được sử dụng để thông báo khi người dùng không chọn file.

ĐỌC FILE

Tên file BaiTap2.txt

URL C:\Users\Tofu\Downloads\BaiTap2.txt

Số dòng 4

Số từ 9

Số ký tự 34

Note này
Dùng để
Kiểm tra
Bài số 2

ĐỌC FILE

Tên file

URL

Số dòng

Số từ

Số ký tự

Thoát

Chỉ tiết:
Sử dụng Filter để lọc file văn bản

Open

Documents

Home

Gallery

OneDrive - Personal

Desktop

Downloads

Documents

Pictures

Music

UIT

Visual Studio 2017

Visual Studio 2022

Zalo Received Files

asaSasaSas

dasdasdasd

sdoiahsghdhoiahdpouashdopu

sdsd

sdsdsd

29/09/2024 6:04 CH

04/10/2024 10:43 CH

25/07/2024 10:48 SA

04/10/2024 10:49 CH

04/10/2024 11:00 CH

04/10/2024 10:46 CH

05/10/2024 2:41 SA

05/10/2024 3:07 SA

File folder

File folder

File folder

Text D

Text D

Text D

Text D

File name: Text Files

Open Cancel

ĐỌC FILE

Tên file

URL

Số dòng

Số từ

Số ký tự

Thoát

Lỗi

Lỗi khi không chọn file, vui lòng đưa đúng file .txt

OK

```

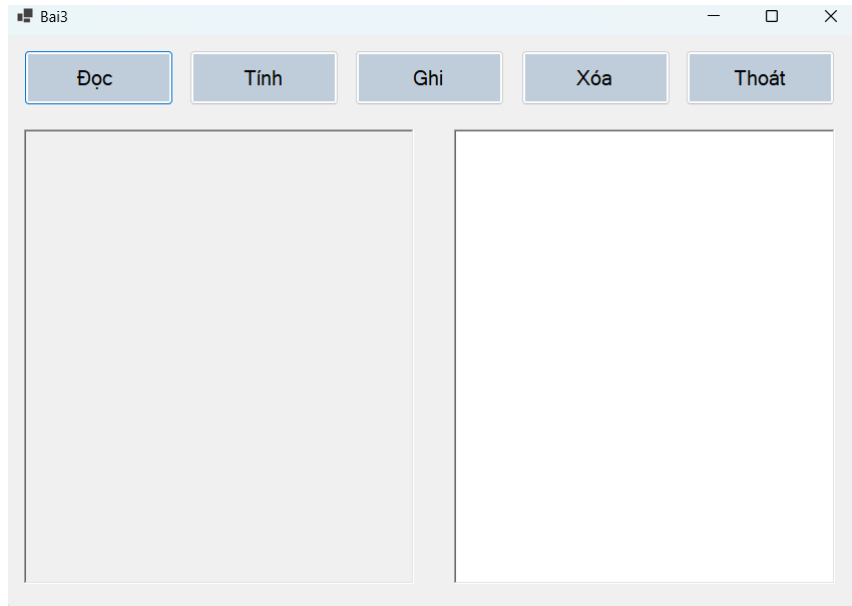
private void btn_ReadFile_Click(object sender, EventArgs e)
{
    OpenFileDialog openFileDialog = new OpenFileDialog();
    openFileDialog.Filter = "Text Files|*.txt";
    if (openFileDialog.ShowDialog() == DialogResult.OK)
    {
        StreamReader streamReader = new StreamReader(openFileDialog.FileName);
        string content = streamReader.ReadToEnd();
        rtbx_Output.Text = content;
        tbx_FileName.Text = openFileDialog.SafeFileName;
        tbx_URL.Text = openFileDialog.FileName;
        // Count Line
        int lineCount = 0;
        using (StreamReader streamReaderLine = new StreamReader(openFileDialog.FileName))
        {
            while (!streamReaderLine.EndOfStream)
            {
                streamReaderLine.ReadLine();
                lineCount++;
            }
        }
        tbx_CountLine.Text = lineCount.ToString();
        content = content.Replace("\r\n", "\r"); // Chuyển đổi xuống dòng
        content = content.Replace('\r', ' '); // Thay thế ký tự xuống dòng bằng khoảng trắng
        string[] source = content.Split(new char[] { ' ', '?', '!', ',', '.', ':', ';', '-', '_', '+', '=', '&', '*', '&#34;', '&#39;', '&#124;', '&#92;', '&#99;', '&#97;', '&#98;', '&#99;', '&#108;', '&#105;', '&#110;', '&#104;', '&#105;', '&#106;', '&#107;', '&#108;', '&#109;', '&#110;', '&#111;', '&#112;', '&#113;', '&#114;', '&#115;', '&#116;', '&#117;', '&#118;', '&#119;', '&#120;', '&#121;', '&#122;', '&#123;', '&#124;', '&#125;', '&#126;', '&#127;', '&#128;', '&#129;', '&#130;', '&#131;', '&#132;', '&#133;', '&#134;', '&#135;', '&#136;', '&#137;', '&#138;', '&#139;', '&#140;', '&#141;', '&#142;', '&#143;', '&#144;', '&#145;', '&#146;', '&#147;', '&#148;', '&#149;', '&#150;', '&#151;', '&#152;', '&#153;', '&#154;', '&#155;', '&#156;', '&#157;', '&#158;', '&#159;', '&#160;', '&#161;', '&#162;', '&#163;', '&#164;', '&#165;', '&#166;', '&#167;', '&#168;', '&#169;', '&#170;', '&#171;', '&#172;', '&#173;', '&#174;', '&#175;', '&#176;', '&#177;', '&#178;', '&#179;', '&#180;', '&#181;', '&#182;', '&#183;', '&#184;', '&#185;', '&#186;', '&#187;', '&#188;', '&#189;', '&#190;', '&#191;', '&#192;', '&#193;', '&#194;', '&#195;', '&#196;', '&#197;', '&#198;', '&#199;', '&#200;', '&#201;', '&#202;', '&#203;', '&#204;', '&#205;', '&#206;', '&#207;', '&#208;', '&#209;', '&#210;', '&#211;', '&#212;', '&#213;', '&#214;', '&#215;', '&#216;', '&#217;', '&#218;', '&#219;', '&#220;', '&#221;', '&#222;', '&#223;', '&#224;', '&#225;', '&#226;', '&#227;', '&#228;', '&#229;', '&#230;', '&#231;', '&#232;', '&#233;', '&#234;', '&#235;', '&#236;', '&#237;', '&#238;', '&#239;', '&#240;', '&#241;', '&#242;', '&#243;', '&#244;', '&#245;', '&#246;', '&#247;', '&#248;', '&#249;', '&#250;', '&#251;', '&#252;', '&#253;', '&#254;', '&#255;' }, StringSplitOptions.RemoveEmptyEntries);
        int wordCount = source.Length; // Sử dụng Length để đếm số từ
        int charCount = content.Length;
        tbx_CountWord.Text = wordCount.ToString();
        tbx_CountCharacter.Text = charCount.ToString();
        streamReader.Close();
    }
}
    
```



```
,
    tbx_CountLine.Text = lineCount.ToString();
    content = content.Replace("\r\n", "\r"); // Chuyển đổi xuống dòng
    content = content.Replace('\r', ' '); // Thay thế ký tự xuống dòng bằng khoảng trắng
    string[] source = content.Split(new char[] { '.', '?', '!', ',', ';', ':', ' ' }, StringSplitOptions.RemoveEmptyEntries);
    int wordCount = source.Length; // Sử dụng Length để đếm số từ
    int charCount = content.Length;
    tbx_CountWord.Text = wordCount.ToString();
    tbx_CountCharacter.Text = charCount.ToString();
    streamReader.Close();
}
else // Khi lỗi
{
    MessageBox.Show("Lỗi khi không chọn file, vui lòng đưa đúng file .txt", "Lỗi", MessageBoxButtons.OK, MessageBoxIcon.Error);
}
}
```

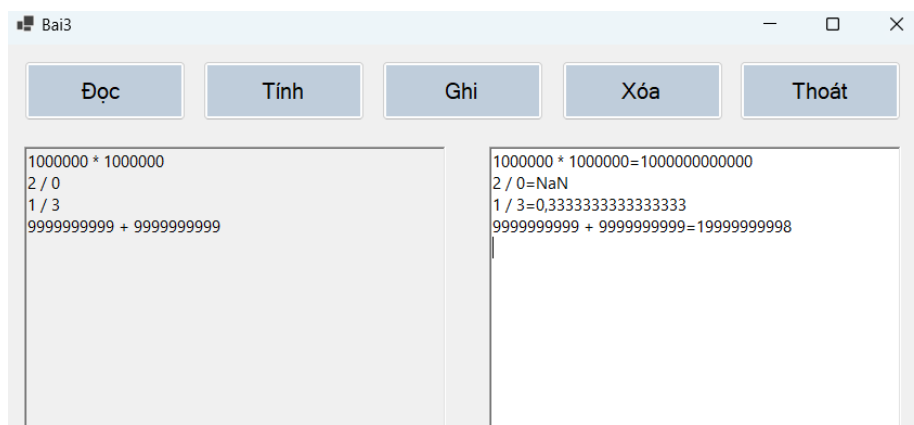
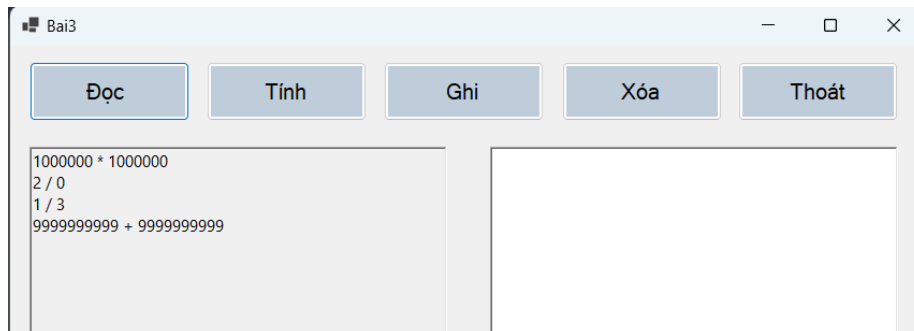
4. Bài tập 3 – Đọc và ghi file (mở rộng):

a) Tổng quan:

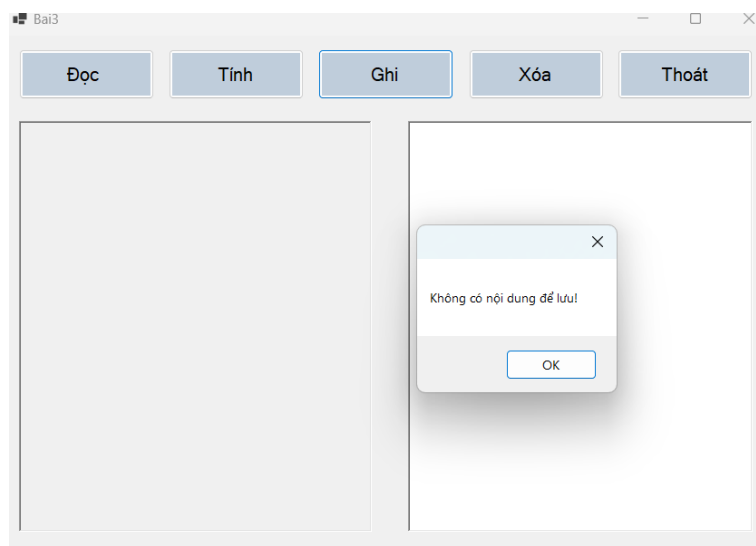
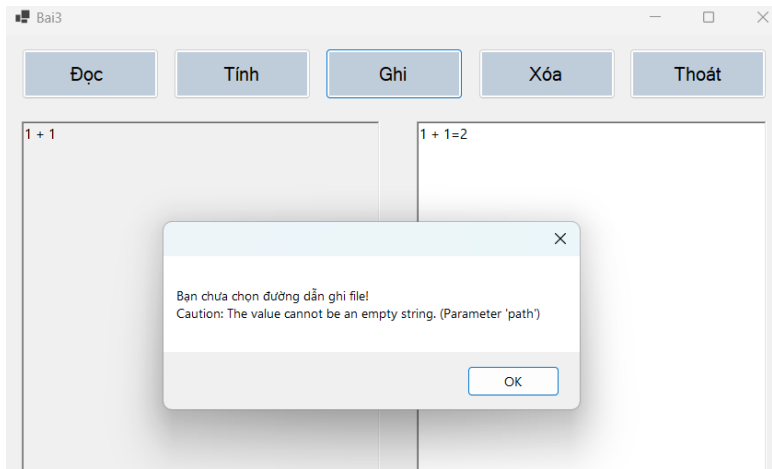


b) Chi tiết:

- Nút đọc mở giao diện chọn file và nội dung file được chọn được hiển thị ở richTextBox bên trái.
- Nút Tính sẽ thực hiện tính qua hàm Cal và xuất kết quả ở richTextBox trên phải.
- Nút Ghi sẽ ghi kết quả vào file mới với tên và đường dẫn mong muốn.



- Hộp thoại thông báo khi người dùng không chọn file hoặc không chọn đường dẫn để ghi file mới và khi không có nội dung để ghi.



```
static double Cal(string s)
{
    double result = 0;

    string pattern = @"(?!(\d+)\s*(\+|-|/))\s*(\d+)\s*"; // Tách thành 3 nhóm đối tượng
    Match match = Regex.Match(s, pattern); // Tách chuỗi
    if (match.Success) // Nếu chuỗi đầu vào đủ 3 nhóm
    {
        double num1 = double.Parse(match.Groups[1].Value); // vị trí các số
        double num2 = double.Parse(match.Groups[3].Value);
        string op = match.Groups[2].Value;

        switch (op)
        {
            case "+":
                result = (num1 + num2);
                break;
            case "-":
                result = (num1 - num2);
                break;
            case "*":
                result = (num1 * num2);
                break;
            case "/":
                if (num2 == 0 && op == "/")
                {
                    MessageBox.Show("Không thể chia cho 0");
                    return double.NaN;
                }
                else result = (num1 / num2);
                break;
        }
    }
    else
    {
        MessageBox.Show("Chuỗi không hợp lệ");
    }
    return result;
}
```

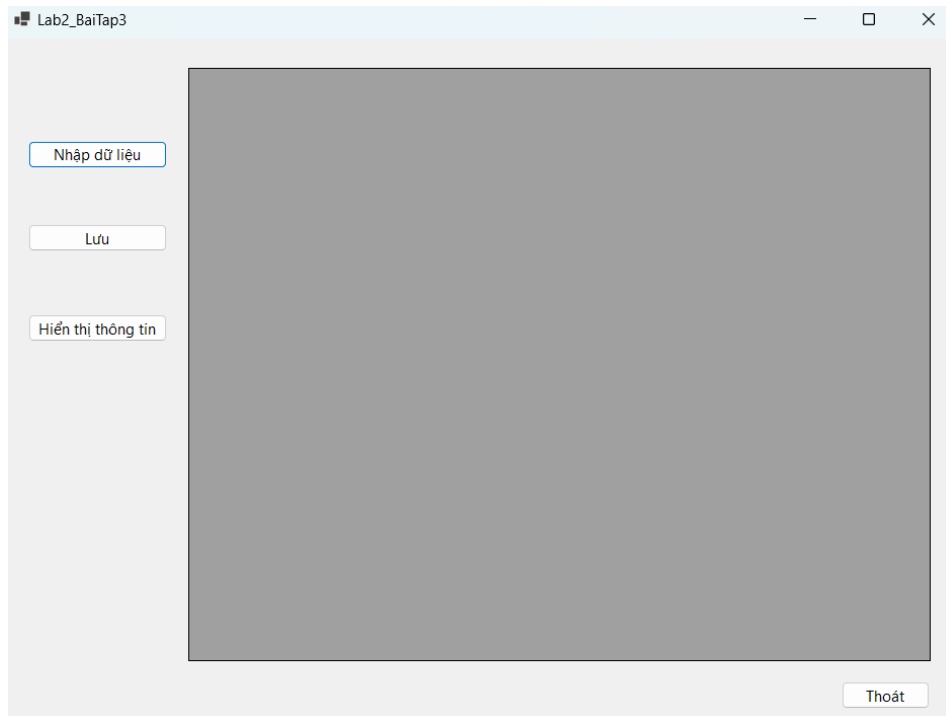
```
public void bt_Doc_Click(object sender, EventArgs e)
{
    try
    {
        OpenFileDialog ofd = new OpenFileDialog();
        ofd.Filter = "txt files (*.txt)|*.txt";
        if (ofd.ShowDialog() == DialogResult.OK) // Kiểm tra nếu người dùng chọn file
        {
            selectedFilePath = ofd.FileName; // Lưu đường dẫn file vào biến toàn cục
            using (FileStream fs = new FileStream(selectedFilePath, FileMode.Open, FileAccess.Read))
            {
                StreamReader sr = new StreamReader(fs);
                var str = sr.ReadToEnd();
                rtbx_input.AppendText(str);
            }
        }
    }
    catch (Exception tmp)
    {
        MessageBox.Show("Bạn chưa chọn file .txt !\n Caution: " + tmp.Message); // Cảnh báo khi xảy ra lỗi
    }
}
```

```
private void bt_Ghi_Click(object sender, EventArgs e)
{
    using (StreamWriter writefileinput = new StreamWriter("DocFile.txt")) //Ghi đè file input bằng dữ liệu mới
    {
        writefileinput.WriteLine(rtbx_output.Text);
    }
    try
    {
        if (rtbx_output.Text.Length != 0)
        {
            string str = rtbx_output.Text;
            SaveFileDialog sfd = new SaveFileDialog();
            sfd.Filter = "txt files (*.txt)|*.txt";
            sfd.ShowDialog();
            FileStream fs = new FileStream(sfd.FileName, FileMode.Create);
            StreamWriter sw = new StreamWriter(fs, Encoding.UTF8);
            sw.WriteLine(str);
            sw.Close();
            fs.Close();
            MessageBox.Show("Đã lưu file!");
        }
        else
        {
            MessageBox.Show("Không có nội dung để lưu!");
        }
    }
    catch (Exception tmp)
    {
        MessageBox.Show("Bạn chưa chọn đường dẫn ghi file!\nCaution: " + tmp.Message);
    }
}
```

```
private void bt_Tinh_Click(object sender, EventArgs e)
{
    if (string.IsNullOrEmpty(selectedFilePath))
    {
        MessageBox.Show("Bạn chưa chọn file để tính toán!");
        return;
    }
    using (FileStream fs = new FileStream(selectedFilePath, FileMode.Open, FileAccess.Read))
    {
        StreamReader sr = new StreamReader(fs);
        string input;
        while ((input = sr.ReadLine()) != null)
        {
            rtbx_output.AppendText(input + "=" + Convert.ToString(Cal(input)) + Environment.NewLine);
        }
    }
}
```

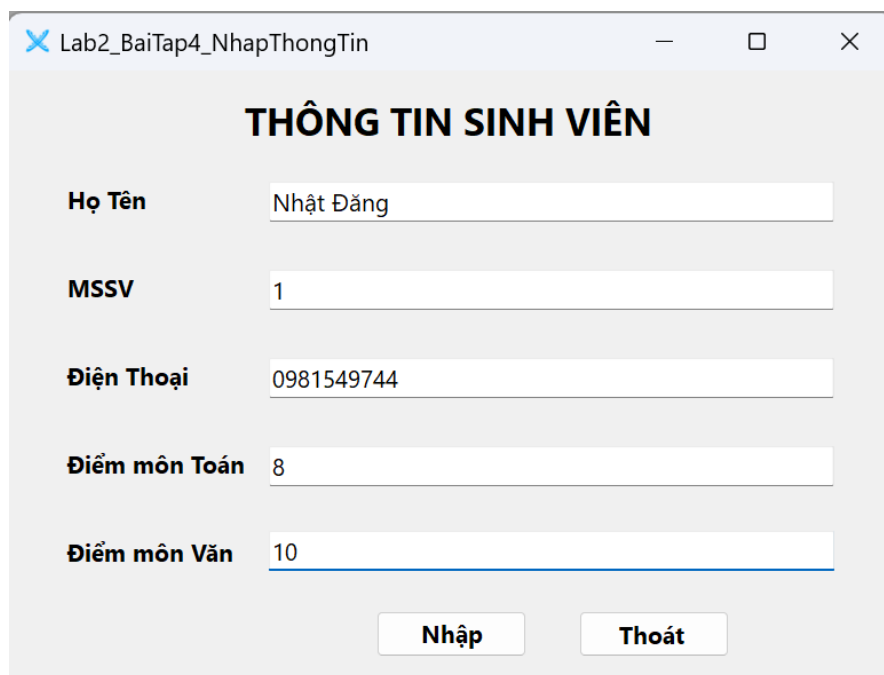
5. Bài tập 4 – Làm việc với file/CSDL:

a) Tổng quan:



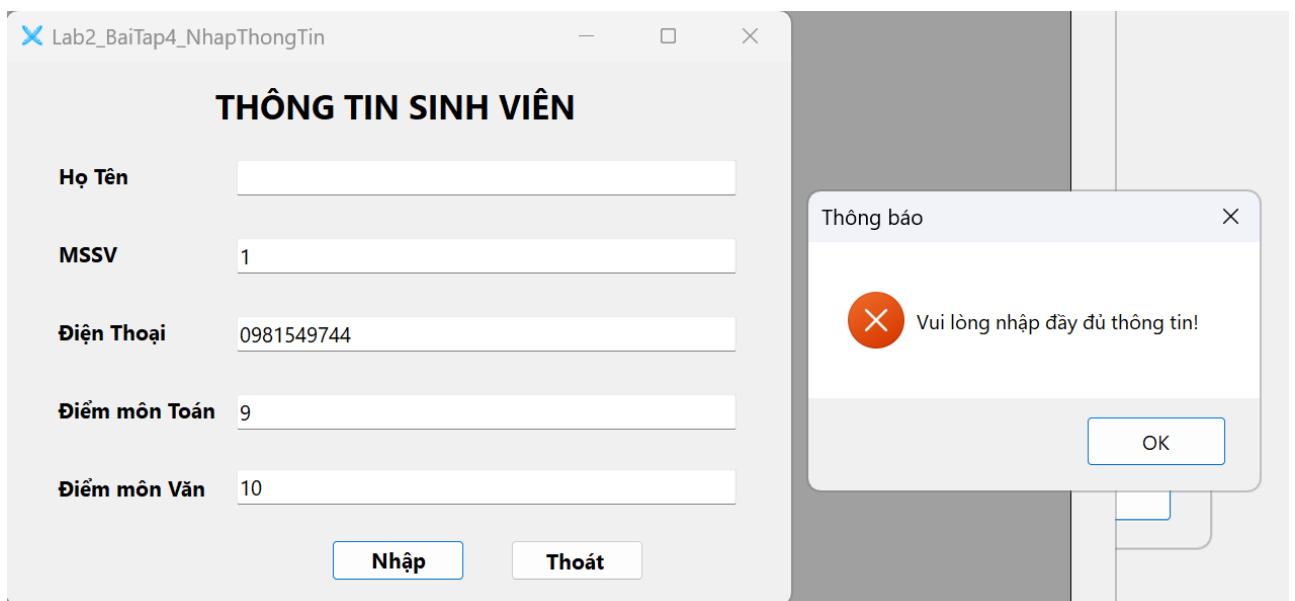
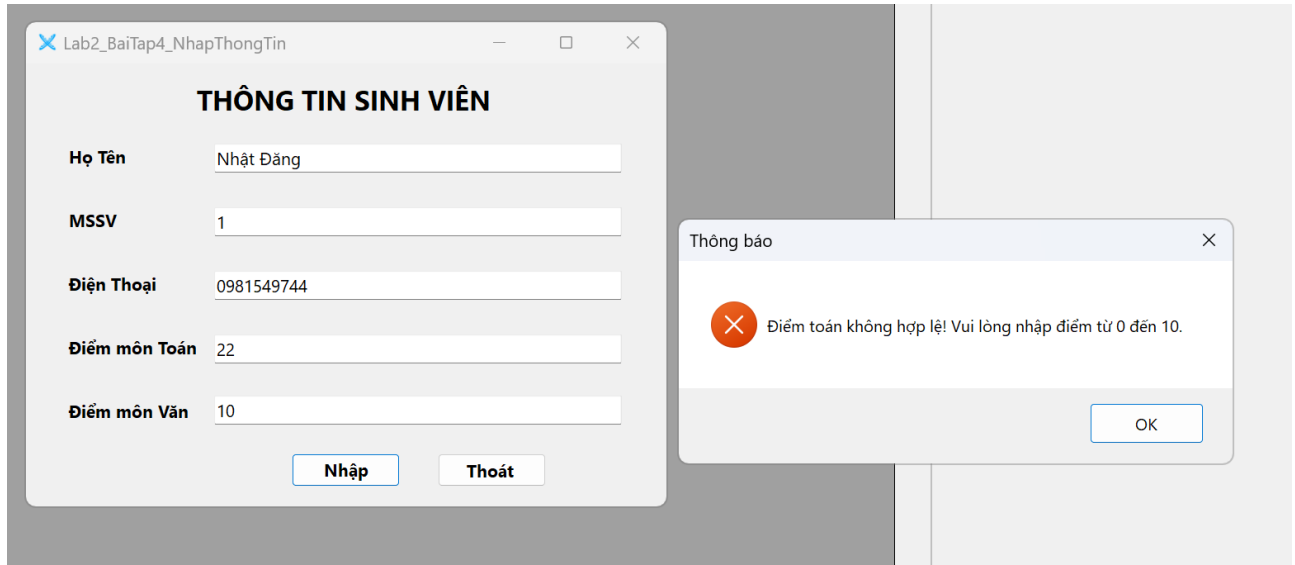
b) Chi tiết:

- Tiến hành nhập dữ liệu bằng Button “Nhập dữ liệu”
- Cửa sổ nhập thông tin sau đó sẽ hiện ra



THÔNG TIN SINH VIÊN	
Họ Tên	Nhật Đăng
MSSV	1
Điện Thoại	0981549744
Điểm môn Toán	8
Điểm môn Văn	10
<div>Nhập Thoát</div>	

- Kiểm tra dữ liệu đầu vào khi không hợp lý hoặc người dùng không nhập một trường dữ liệu nào



- Khi nhập liệu thành công, sẽ hiện ra hộp thoại thông báo thành công

THÔNG TIN SINH VIÊN

Họ Tên: Minh Đạt

MSSV: 1

Điện Thoại: 0981549744

Điểm môn Toán: 9

Điểm môn Văn: 10

Nhập Thoát

Đã nhập thành công sinh viên: Minh Đạt

OK

- Ta có thể nhập số lượng sinh viên (không giới hạn) và có thể ấn thoát khi đã nhập xong, dữ liệu sẽ tạm thời được lưu vào “input.txt”
- Ấn nút lưu, sẽ hiển thị ra hộp thoại thông báo thành công và dữ liệu được xử lý từ file “input.txt” sau đó được lưu vào file “output.xlsx”

Nhập dữ liệu

Lưu

Hiển thị thông tin

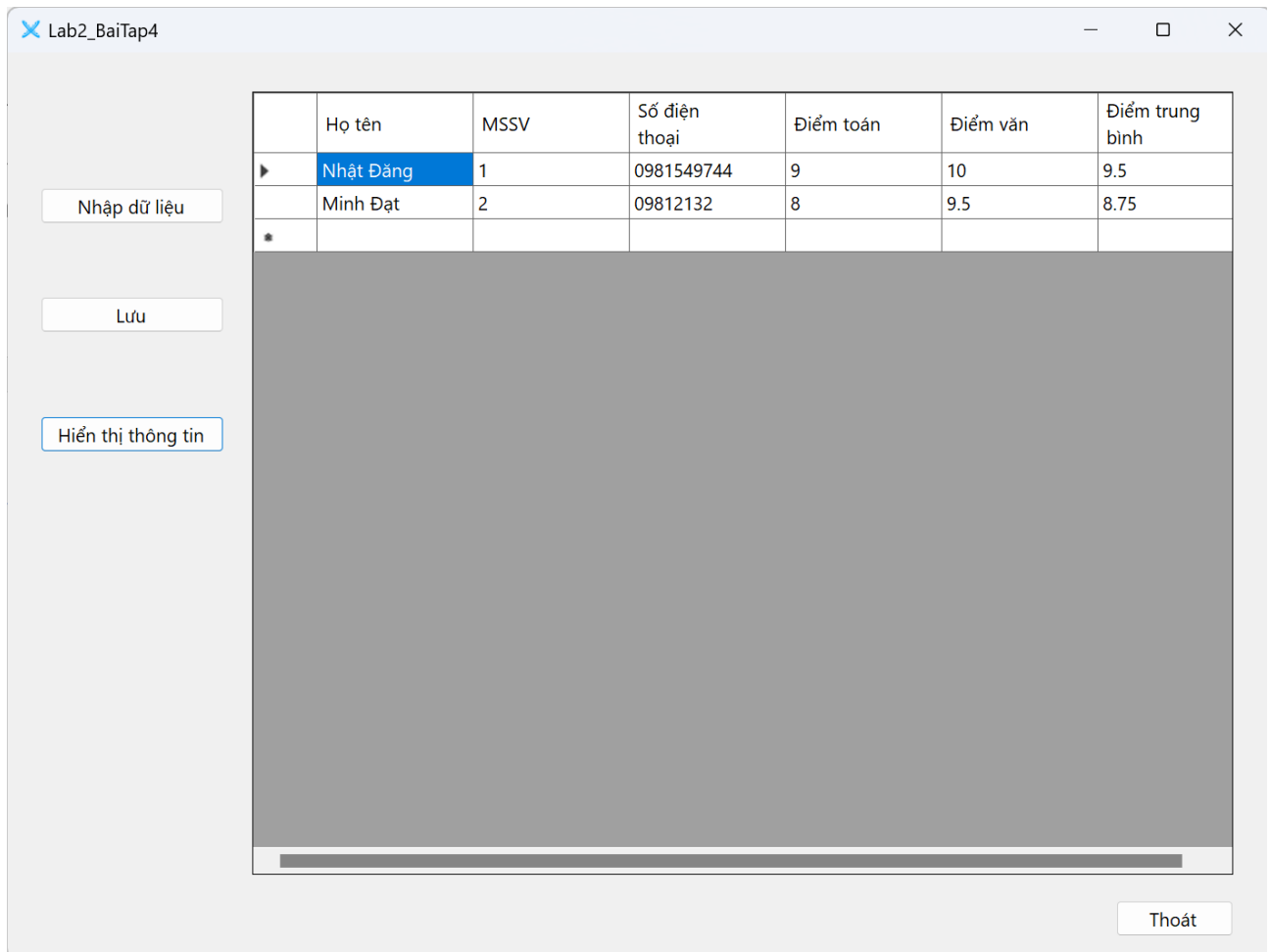
Thông báo

Lưu dữ liệu thành công!

OK

Thoát

- Ấn nút hiển thị thông tin, sẽ hiện ra danh sách khi được đọc từ file “output.xlsx”



	Họ tên	MSSV	Số điện thoại	Điểm toán	Điểm văn	Điểm trung bình
▶	Nhật Đăng	1	0981549744	9	10	9.5
	Minh Đạt	2	09812132	8	9.5	8.75
*						

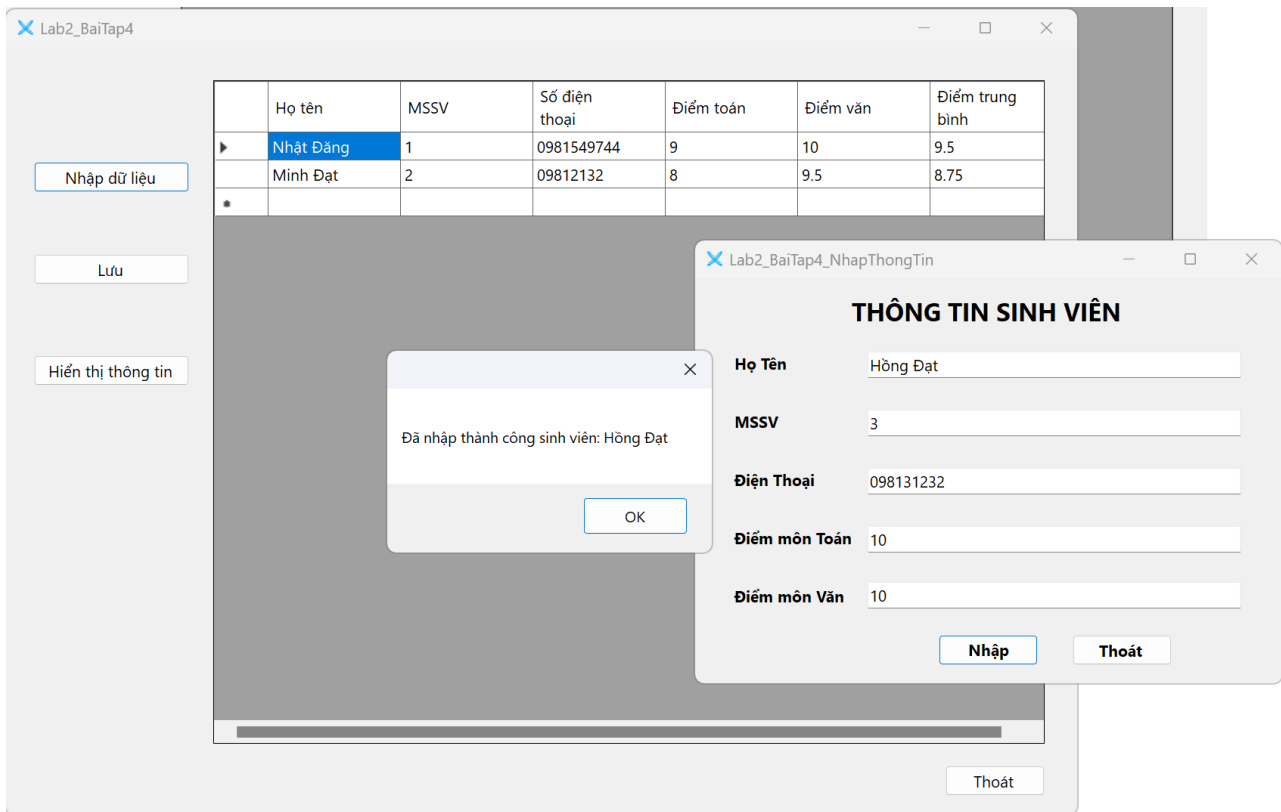
Nhập dữ liệu

Lưu

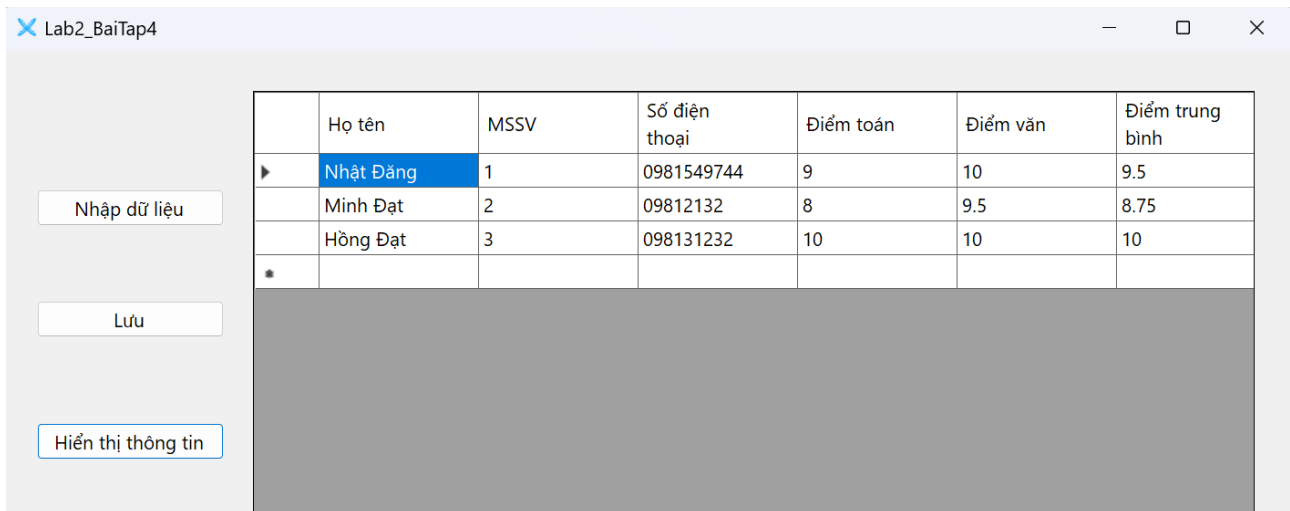
Hiển thị thông tin

Thoát

- Ngoài ra, khi người dùng muốn nhập thêm thông tin vào trong dữ liệu, có thể ấn nút nhập dữ liệu



- Sau đó người dùng chỉ cần nhấn nút “Lưu” và “Hiển thị thông tin” dữ liệu sẽ được thêm vào



- Sử dụng Button “Thoát” để về Menu Lab 2 để tiếp tục sử dụng

- File 1: Lab2_BaiTap4_NhapThongTin.cs

```

11
12 namespace Lab2_23520231_23520249
13 {
14     public partial class Lab2_BaiTap4_NhapThongTin : Form
15     {
16         private List<SinhVien> danhSachSinhVien;
17         public Lab2_BaiTap4_NhapThongTin()
18         {
19             InitializeComponent();
20             danhSachSinhVien = new List<SinhVien>();
21         }
22         public class SinhVien
23         {
24             public required string HoTen;
25             public required string MSSV;
26             public required string SDT;
27             public required double DiemToan;
28             public required double DiemVan;
29             public required double DiemTB;
30         }
31     }
32 }
33
34 private void btn_Nhap_Click(object sender, EventArgs e)
35 {
36     // Kiểm tra đầu vào hợp lệ
37     if (string.IsNullOrWhiteSpace(tb_HoTen.Text) ||
38         string.IsNullOrWhiteSpace(tb_MSSV.Text) ||
39         string.IsNullOrWhiteSpace(tb_SDT.Text) ||
40         string.IsNullOrWhiteSpace(tb_DiemToan.Text) ||
41         string.IsNullOrWhiteSpace(tb_DiemVan.Text))
42     {
43         MessageBox.Show("Vui lòng nhập đầy đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);
44         return;
45     }
46
47     if (!double.TryParse(tb_DiemToan.Text, out double diemToan) || diemToan < 0 || diemToan > 10)
48     {
49         MessageBox.Show("Điểm toán không hợp lệ! Vui lòng nhập điểm từ 0 đến 10.", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);
50         return;
51     }
52
53     if (!double.TryParse(tb_DiemVan.Text, out double diemVan) || diemVan < 0 || diemVan > 10)
54     {
55         MessageBox.Show("Điểm văn không hợp lệ! Vui lòng nhập điểm từ 0 đến 10.", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);
56         return;
57     }
58
59     SinhVien sv = new SinhVien()
60     {
61         HoTen = tb_HoTen.Text,
62         MSSV = tb_MSSV.Text,
63         SDT = tb_SDT.Text,
64         DiemToan = double.Parse(tb_DiemToan.Text),
65         DiemVan = double.Parse(tb_DiemVan.Text),
66         DiemTB = 0
67     };
68     MessageBox.Show("Đã nhập thành công sinh viên: " + sv.HoTen);
69     danhSachSinhVien.Add(sv);
70     tb_HoTen.Text = "";
71     tb_MSSV.Text = "";
72     tb_SDT.Text = "";
73     tb_DiemToan.Text = "";
74     tb_DiemVan.Text = "";
75 }

```

```

1 reference
private void btn_Exit_Click(object sender, EventArgs e)
{
    // Làm mới nội dung input.txt
    File.WriteAllText("input.txt", string.Empty);
    using StreamWriter writer = new StreamWriter("input.txt");
    try
    {
        foreach (SinhVien sv in danhSachSinhVien)
        {
            writer.Write(sv.HoTen + ";");
            writer.Write(sv.MSSV + ";");
            writer.Write(sv.SDT + ";");
            writer.Write(sv.DiemToan + ";");
            writer.Write(sv.DiemVan);
            writer.WriteLine();
        }
        Close();
    }
    catch (Exception ex)
    {
        MessageBox.Show("Lỗi: " + ex.Message, "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
}

```

- File 2: Lab2_BaiTap4.cs

```

namespace Lab2_23520231_23520249
{
    5 references
    public partial class Lab2_BaiTap4 : Form
    {
        private List<Lab2_BaiTap4_NhapThongTin.SinhVien> danhSachSinhVien = new List<Lab2_BaiTap4_NhapThongTin.SinhVien>();

        1 reference
        public Lab2_BaiTap4()
        {
            InitializeComponent();
        }

        1 reference
        private void btn_Nhap_Click(object sender, EventArgs e)
        {
            Lab2_BaiTap4_NhapThongTin NhapThongTin = new Lab2_BaiTap4_NhapThongTin();
            NhapThongTin.ShowDialog();
        }
    }
}

```

```

1 reference
private void btn_Luu_Click(object sender, EventArgs e)
{
    using (StreamReader reader = new StreamReader("input.txt"))
    {
        string? line;
        while ((line = reader.ReadLine()) != null)
        {
            var parts = line.Split(new[] { ';' }, StringSplitOptions.RemoveEmptyEntries);
            if (parts.Length >= 5)
            {
                var sv = new Lab2_BaiTap4_NhapThongTin.SinhVien
                {
                    HoTen = parts[0],
                    MSSV = parts[1],
                    SDT = parts[2],
                    DiemToan = double.Parse(parts[3]),
                    DiemVan = double.Parse(parts[4]),
                    DiemTB = (double.Parse(parts[3]) + double.Parse(parts[4])) / 2;
                };
                danhSachSinhVien.Add(sv);
            }
        }
        MessageBox.Show("Lưu dữ liệu thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}

```

```

using (var workbook = new XSSFWorkbook())
{
    var sheet = workbook.CreateSheet("Danh sách sinh viên");

    // Tạo tiêu đề
    var headerRow = sheet.CreateRow(0);
    headerRow.CreateCell(0).SetCellValue("Họ tên");
    headerRow.CreateCell(1).SetCellValue("MSSV");
    headerRow.CreateCell(2).SetCellValue("Số điện thoại");
    headerRow.CreateCell(3).SetCellValue("Điểm toán");
    headerRow.CreateCell(4).SetCellValue("Điểm văn");
    headerRow.CreateCell(5).SetCellValue("Điểm trung bình");

    // Thêm dữ liệu sinh viên
    int rowIndex = 1;
    foreach (var sv in danhSachSinhVien)
    {
        var row = sheet.CreateRow(rowIndex++);
        row.CreateCell(0).SetCellValue(sv.HoTen);
        row.CreateCell(1).SetCellValue(sv.MSSV);
        row.CreateCell(2).SetCellValue(sv.SDT);
        row.CreateCell(3).SetCellValue(sv.DiemToan);
        row.CreateCell(4).SetCellValue(sv.DiemVan);
        row.CreateCell(5).SetCellValue(sv.DiemTB);
    }

    // Lưu workbook vào file
    using (var fileStream = new FileStream("output.xlsx", FileMode.Create, FileAccess.Write))
    {
        workbook.Write(fileStream);
    }
}

private void btn_HienThiThongTin_Click(object sender, EventArgs e)
{
    using (FileStream fs = new FileStream("output.xlsx", FileMode.Open, FileAccess.Read))
    {
        IWorkbook workbook = new XSSFWorkbook(fs);
        ISheet sheet = workbook.GetSheetAt(0);

        // Tạo một DataTable để lưu trữ dữ liệu
        DataTable dt = new DataTable();

        // Khởi tạo các hàng, cột cho dataTable
        IRow headerRow = sheet.GetRow(0);
        for (int i = 0; i < headerRow.LastCellNum; i++)
        {
            dt.Columns.Add(headerRow.GetCell(i).StringCellValue);
        }

        // Đọc toàn bộ file vào dataTable
        for (int i = 1; i <= sheet.LastRowNum; i++)
        {
            IRow row = sheet.GetRow(i);
            DataRow dataRow = dt.NewRow();
            for (int j = 0; j < row.LastCellNum; j++)
            {
                if (row.GetCell(j) != null)
                {
                    dataRow[j] = row.GetCell(j).ToString();
                }
            }
            dt.Rows.Add(dataRow);
        }

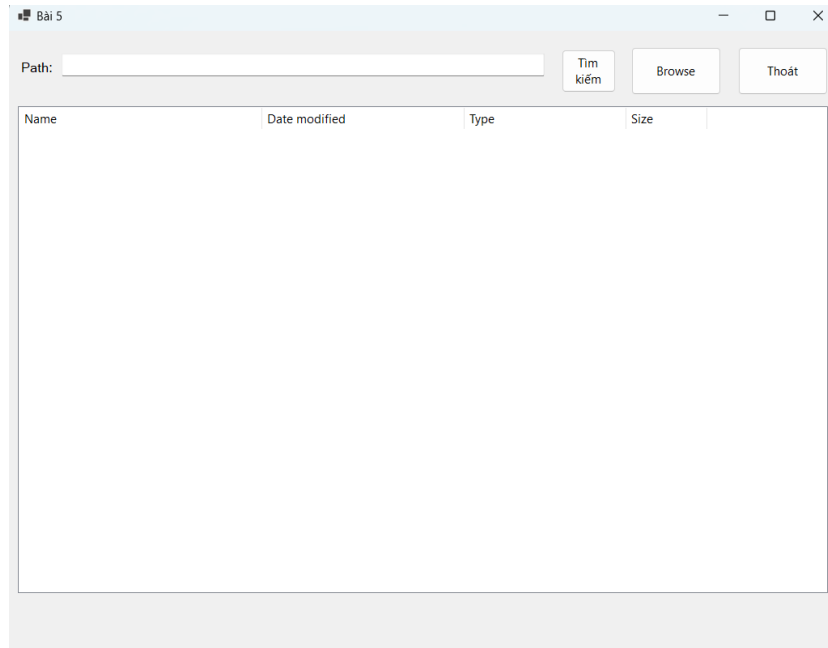
        // Đẩy dữ liệu từ dataTable cho DataGridView
        dataGridView_SinhVien.DataSource = dt;
    }
}

1 reference
private void btn_Exit_Click(object sender, EventArgs e)
{
    Close();
}
}

```

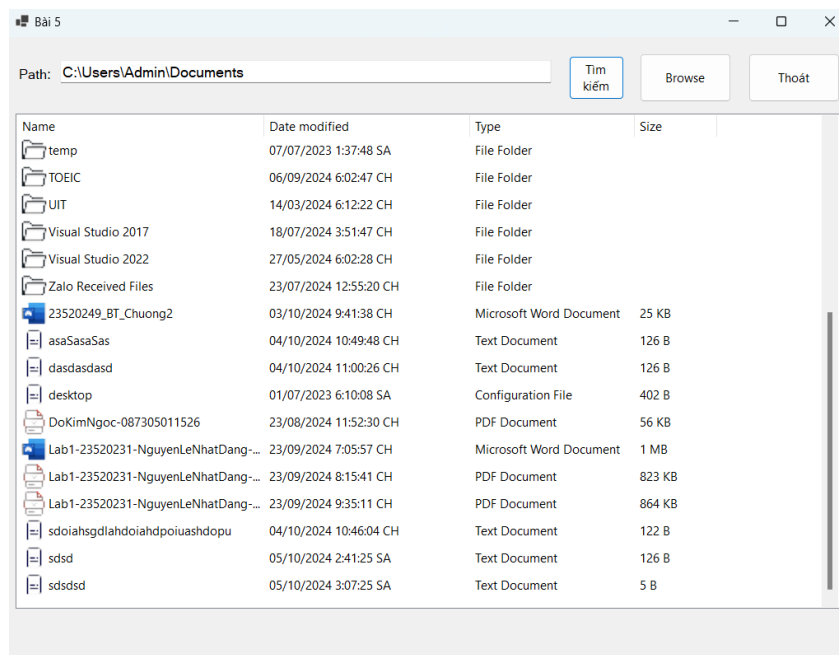
6. Bài tập 5 – Duyệt thư mục:

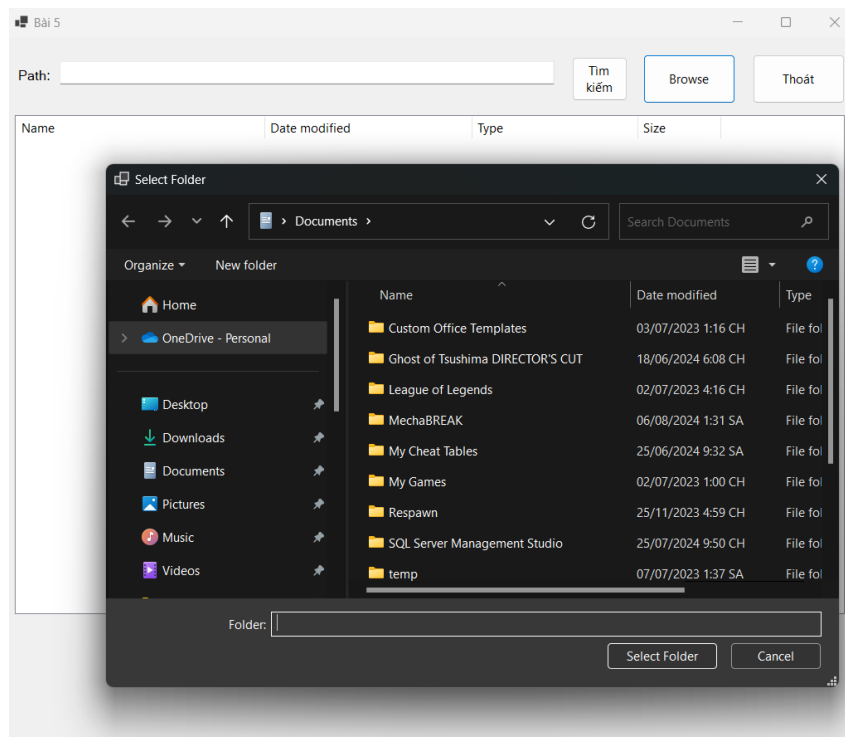
a) Tổng quan:



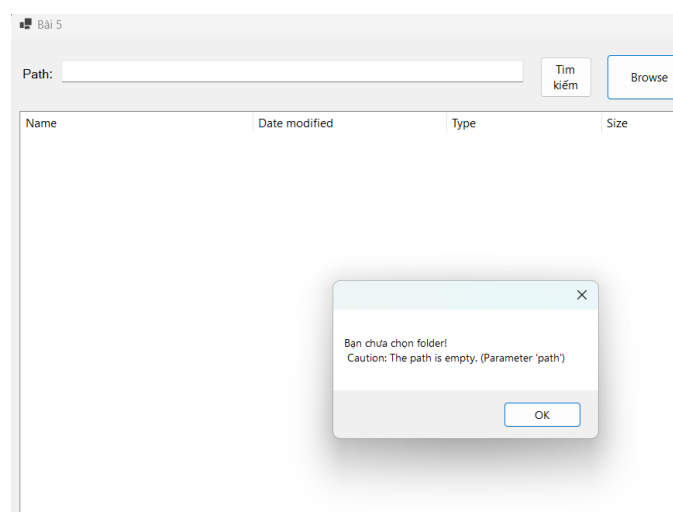
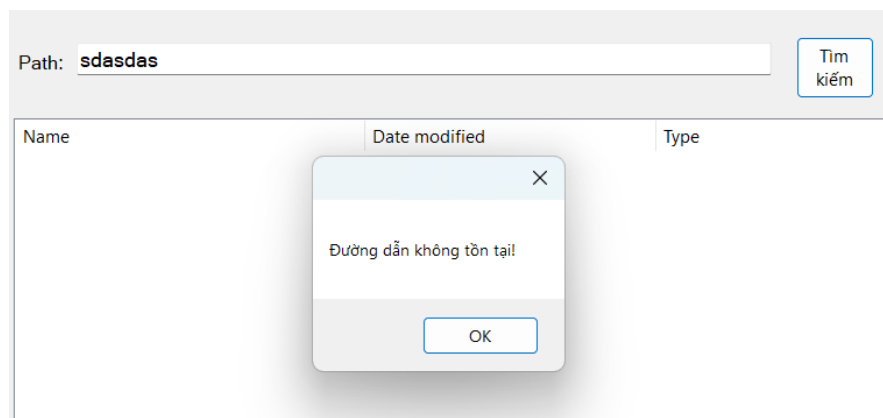
b) Chi tiết:

- Sử dụng nút Tìm kiếm để tìm thư mục bằng đường dẫn đã nhập bằng Directory hoặc nút Browse để chọn thư mục bằng ShowDialog.
- Sử dụng imageList để hiển thị icon và các phương thức đã biết cho thông tin khác của file.
- Các file trong thư mục được chọn được hiển thị các thông tin như tên, ngày mở cuối cùng, loại file và kích thước.





- Hộp thoại được sử dụng để thông báo khi người dùng nhập đường dẫn sai hoặc không chọn thư mục.



```

string folderPath = tbx_Path.Text; // Lấy đường dẫn folder ở textbox
if (Directory.Exists(folderPath))
{
    listView_Show.Items.Clear(); // Clear listView
    string[] files = Directory.GetFiles(folderPath);
    string[] directories = Directory.GetDirectories(folderPath);

    foreach (string dir in directories) // Hiển thị thư mục
    {
        DirectoryInfo di = new DirectoryInfo(dir);
        ListViewItem item = new ListViewItem(RemoveExtension(di.Name));
        listView_Show.Items.Add(item);
        listView_Show.SmallImageList = imageList;
        listView_Show.LargeImageList = imageList; // Sử dụng imageList để hiển thị icon
        item.ImageIndex = 10; // Chỉ số icon cho thư mục
        item.SubItems.Add(di.CreationTime.ToString());
        item.SubItems.Add("File Folder");
        item.SubItems.Add("");
    }

    foreach (string file in files) // Hiển thị file
    {
        FileInfo fi = new FileInfo(file);
        ListViewItem item = new ListViewItem(RemoveExtension(fi.Name));
        listView_Show.Items.Add(item);
        listView_Show.SmallImageList = imageList;
        listView_Show.LargeImageList = imageList; // Sử dụng imageList để hiển thị icon
        item.ImageIndex = GetIconIndex(fi.Name); // Chỉ số icon cho file
        item.SubItems.Add(fi.CreationTime.ToString());
        item.SubItems.Add(GetFileType(fi.Name));
        item.SubItems.Add(FormatFileSize(fi.Length));
    }
}
else
{
    MessageBox.Show("Đường dẫn không tồn tại!"); // Cảnh báo khi đường dẫn không hợp lệ
}

```

```

FolderBrowserDialog fbd = new FolderBrowserDialog();
fbd.ShowDialog();
try
{
    listView_Show.Items.Clear();
    tbx_Path.Text = fbd.SelectedPath; //đổi đường dẫn của textbox thành đường dẫn của thư mục được chọn
    string[] files = Directory.GetFiles(fbd.SelectedPath);
    string[] directories = Directory.GetDirectories(fbd.SelectedPath);
    foreach (string dir in directories) // Hiển thị thư mục
    {
        DirectoryInfo directoryInfo = new DirectoryInfo(dir);
        ListViewItem item = new ListViewItem(directoryInfo.Name);
        listView_Show.Items.Add(item);
        listView_Show.SmallImageList = imageList;
        listView_Show.LargeImageList = imageList; // Sử dụng imageList để hiển thị icon
        item.ImageIndex = 10; // Chỉ số của icon thư mục
        item.SubItems.Add(directoryInfo.CreationTime.ToString());
        item.SubItems.Add("File Folder"); // Loại file
        item.SubItems.Add(""); // Không có phần mở rộng
        item.SubItems.Add(""); // Không có dung lượng
    }

    foreach (string file in files) // Hiển thị file
    {
        FileInfo fi = new FileInfo(file);
        ListViewItem item = new ListViewItem(RemoveExtension(fi.Name));
        listView_Show.Items.Add(item);
        listView_Show.SmallImageList = imageList;
        listView_Show.LargeImageList = imageList; // Sử dụng imageList để hiển thị icon
        item.ImageIndex = GetIconIndex(fi.Name); // Chỉ số của icon file
        item.SubItems.Add(fi.CreationTime.ToString());
        item.SubItems.Add(GetFileType(fi.Name)); // Loại file
        item.SubItems.Add(FormatFileSize(fi.Length).ToString());
    }
}
catch (Exception tmp)
{
    MessageBox.Show("Đã xảy ra lỗi khi chọn folder!\n Caution: " + tmp.Message);
}

```

- Sử dụng GetFileNameWithoutExtension để lấy tên file không có phần mở rộng.

```

private string RemoveExtension(string fileName)
{
    return Path.GetFileNameWithoutExtension(fileName); // Phương thức này sẽ trả về tên file không kèm theo phần mở rộng
}

```

- Quy định đơn vị kích thước file.

```

private string FormatFileSize(Long size) // KB, MB, GB, TB
{
    string[] suffixes = { "B", "KB", "MB", "GB", "TB" };
    int suffixIndex = 0;
    while (size > 1024)
    {
        size /= 1024;
        suffixIndex++;
    }
    return size + " " + suffixes[suffixIndex];
}

```