

THỰC HÀNH LẬP TRÌNH GIAO DIỆN

BUỔI 1 – 2

Bài tập căn bản

Câu 1. Viết chương trình thực hiện tìm số lớn nhất trong hai số a, b.

Câu 2. Viết chương trình thực hiện tính tổng S theo công thức : $S = 1+2+3+...+n$ (n là số nguyên, $n > 0$).

HƯỚNG DẪN

CÂU 1:

Câu 1: Yêu cầu:

Tạo form sau kéo thả 3 Label, 2 Button, 3 Textbox. Sau đó đặt tên và giá trị cho các công cụ như sau:

- **Form:** thay đổi thuộc tính **Name** thay đổi giá trị thành **frmMain** và thuộc tính **Text** thành **Tìm số lớn nhất.**
- **Label:** thay đổi thuộc tính **Name** Label 1 thành **lbNhapSoA** và thuộc tính **Text** thành **Nhập số A.** Thay đổi thuộc tính **Name** Label 2 thành **lbNhapSoB** và thuộc tính **Text** thành **Nhập số B.** Thay đổi thuộc tính **Name** Label 3 thành **lbKetQua** và thuộc tính **Text** thành **Kết quả.**
- **TextBox:** Thay đổi thuộc tính **Name** TextBox1 thành **txtSoA**, thuộc tính **Name** TextBox2 thành **txtSoB** và thuộc tính **Name** TextBox3 thành **txtKetQua**
- **Button:** Thay đổi thuộc tính **Name** Button 1 thành **btnTinh** và thuộc tính **Text** thành **Tính tổng.** Thay đổi thuộc tính **Name** Button 2 thành **btnHuy** và thuộc tính **Text** thành **Hủy**
- Khi nhấn vào Button **So sánh** thì chương trình sẽ thực hiện công việc so sánh hai số nhập vào, số nào lớn hơn sẽ được gán vào TextBox **Kết quả**
- Khi nhấn vào Button **Hủy** dữ liệu của 3 TextBox được làm rỗng.

Thiết kế và xử lý:

Bước 1: Tạo Project (theo hướng dẫn như trên)

Bước 2: Thay đổi các thuộc tính của Form (theo hướng dẫn như trên)

Bước 3: Kéo thả các công cụ tại cửa sổ **Toolbox** (theo hướng dẫn như trên).

Bước 4: Thay đổi các thuộc tính của các công cụ mới vừa kéo thả lên form và sắp xếp lại theo bố cục hợp lý (tham khảo giao diện sau).

Bước 5: Viết mã nguồn để xử lý các sự kiện.

Để lấy giá trị của Textbox chúng ta sử dụng lệnh sau:

`string giatri = TextBox.Text` (Giá trị của TextBox là kiểu chuỗi)

Để tính toán ta cần chuyển đổi giá trị về kiểu số nguyên. Ta sử dụng hàm sau:

`Convert.ToInt32(tham_so)`

Ví dụ: `int songuyen_n = Convert.ToInt32(giatri)`

Ngược lại, để gán giá trị cho Textbox ta sử dụng lệnh sau:

`TextBox.Text = ten_bien.ToString();`

Ví dụ: `txtGiaTriTong.Text = songuyen_n.ToString();`

Sử dụng cấu trúc if else để so sánh hai số

Cấu trúc if else:

if(thỏa điều kiện)

```
{  
    //thực hiện công việc } else {    //Thực hiện công việc  
}
```

Ví dụ: Đề so sánh hai số a và b

```
    if(a>b)  
{  
        //Kết luận a lớn hơn b  
        MessageBox.Show("Số a lớn hơn số b", "Kết quả"); }  
else {  
    //Kết luận b lớn hơn a MessageBox.Show("Số b lớn hơn số a", "Kết quả");  
}
```

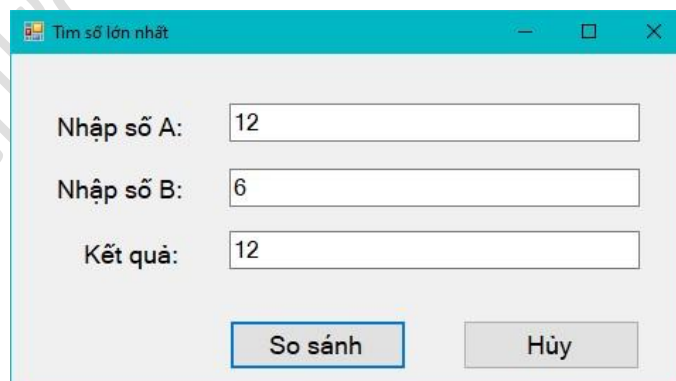
Để so sánh ta viết mã nguồn để xử lý sự kiện Click vào button So sánh. Chúng ta Double click vào button So sánh và tiến hành viết mã nguồn xử lý cho sự kiện. Chúng ta lấy giá trị của hai số a và b sau đó chuyển thành kiểu số. Sử dụng cấu trúc if else để thực hiện so sánh.

Mã nguồn chương trình tham khảo:

```
1 reference
private void btnTinh_Click(object sender, EventArgs e)
{
    //khai báo biến "giatriA, giatriB" kiểu string(chuỗi)
    string giatriA = "";
    string giatriB = "";
    //gán trị trị số A vào biến "giatriA"
    giatriA = txtSoA.Text;
    //gán trị trị số B vào biến "giatriB"
    giatriB = txtSoB.Text;
    //Chuyển đổi kiểu dữ liệu kiểu string sang kiểu int(số nguyên)
    int soA = Convert.ToInt32(giatriA);
    int soB = Convert.ToInt32(giatriB);

    //Sử dụng câu trúc if else để so sánh hai số
    //Kiểm tra số A có lớn hơn số B hay không?
    if(soA>soB)
    {
        //Số A lớn hơn số B. Gán giá trị số A vào Textbox kết quả
        txtKetQua.Text = soA.ToString();
    }
    else
    {
        //Ngược lại thì số B lớn hơn số A. Gán giá trị của số B vào Textbox kết quả
        txtKetQua.Text = soB.ToString();
    }
}
```

Kết quả chương trình:



1 reference

Tìm số lớn nhất

Nhập số A: 12

Nhập số B: 6

Kết quả: 12

So sánh Hủy

Xử lý sự kiện cho **button Hủy**

– Làm rỗng làm dữ liệu của các **TextBox**. Tương tự như **Button So sánh**, Double click vào **button Hủy**, sau đó tiến hành viết mã nguồn để xử lý. Mã nguồn tham khảo:

```
0 references
private void btnHuy_Click(object sender, EventArgs e)
{
    //Gán giá trị rỗng cho Textbox
    txtSoA.Text = "";
    //Gán giá trị rỗng cho Textbox
    txtSoB.Text = "";
    //Gán giá trị rỗng cho Textbox
    txtKetQua.Text = "";
}
```

Kết quả:



CÂU 2

Câu 2. Yêu cầu:

Tạo form sau kéo thả 2 Label, 2 Button, 2 TextBox. Sau đó đặt tên và giá trị cho các công cụ như sau:

- **Form:** thay đổi thuộc tính **Name** thay đổi giá trị thành **frmMain** và thuộc tính **Text** thành **Tính tổng S.**
- **Label:** thay đổi thuộc tính **Name** Label 1 thành **lbTextNhapN** và thuộc tính **Text** thành **Nhập N**. Thay đổi thuộc tính **Name** Label 2 thành **lbTong** và thuộc tính **Text** thành **Tổng**
- **TextBox:** Thay đổi thuộc tính **Name** TextBox1 thành **txtGiaTriN** và thuộc tính **Name** TextBox2 thành **txtGiaTriTong**

- **Button:** Thay đổi thuộc tính **Name** Button 1 thành btnTinh và thuộc tính **Text** thành **Tính tổng**. Thay đổi thuộc tính **Name** Button 2 thành btnHuy và thuộc tính **Text** thành **Hủy**
- Khi nhấn vào Button **Tính tổng** thì chương trình sẽ thực hiện tính tổng: $S = 1 + 2 + 3 + \dots + n$ và tổng được gán vào TextBox **Tổng**.
- Khi nhấn vào Button **Hủy** dữ liệu của 2 TextBox (**txtGiaTriN**, **txtGiaTriTong**) được làm rỗng.

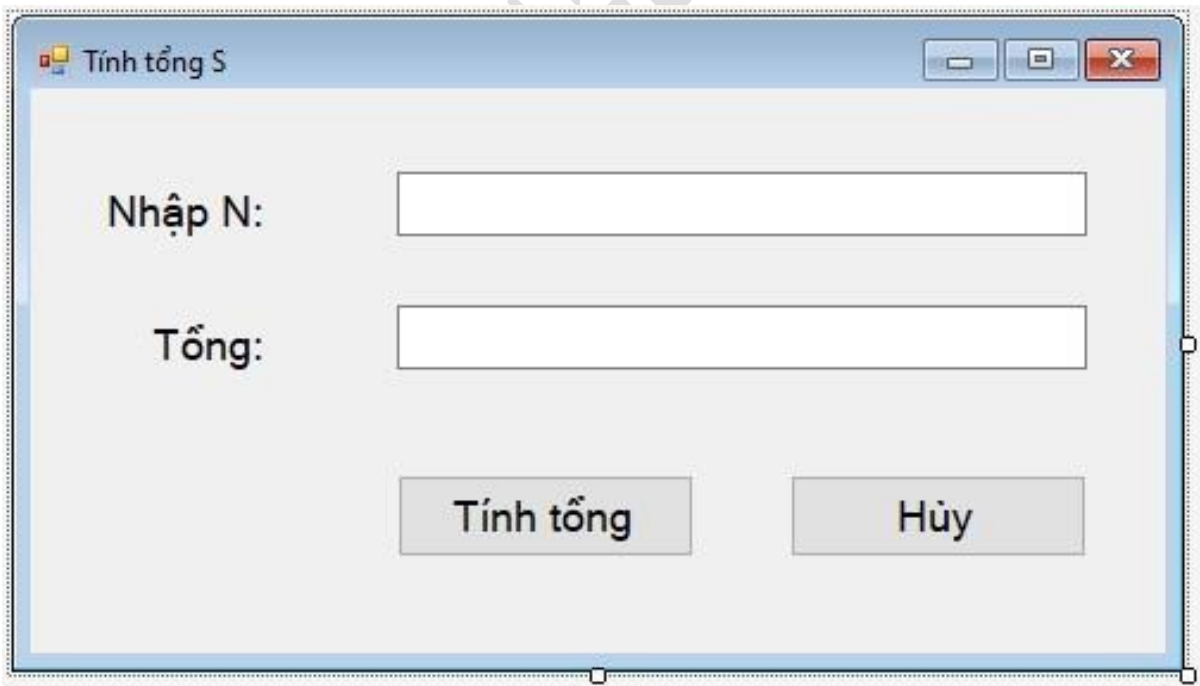
Thiết kế và xử lý:

Bước 1: Tạo Project (theo hướng dẫn như trên)

Bước 2: Thay đổi các thuộc tính của Form (theo hướng dẫn như trên)

Bước 3: Kéo thả các công cụ tại cửa sổ **Toolbox** (theo hướng dẫn như trên). **Bước**

4: Thay đổi các thuộc tính của các công cụ mới vừa kéo thả lên form và sắp xếp lại theo bố cục hợp lý (tham khảo giao diện sau).



The screenshot shows a Windows application window with the title bar 'Tính tổng S'. Inside the window, there are two text input fields. The first field is preceded by the label 'Nhập N:' and the second field is preceded by the label 'Tổng:'. Below these fields, there are two buttons: 'Tính tổng' on the left and 'Hủy' on the right. The window has a standard Windows XP-style title bar with minimize, maximize, and close buttons.

Bước 5: Viết mã nguồn để xử lý các sự kiện.

Lưu ý: Để lấy giá trị của Textbox chúng ta sử dụng lệnh sau:

`string giatri = TextBox.Text` (Giá trị của TextBox là kiểu chuỗi)

Để tính toán ta cần chuyển đổi giá trị về kiểu số nguyên. Ta sử dụng hàm sau:

`Convert.ToInt32(tham_so)` Ví dụ: `int songuyen_n = Convert.ToInt32(giatri)`

Ngược lại, để gán giá trị cho Textbox ta sử dụng lệnh sau:

`TextBox.Text = ten_bien.ToString();`

Ví dụ: `txtGiaTriTong.Text = songuyen_n.ToString();`

Để tính tổng ta viết mã nguồn để xử lý sự kiện Click vào button Tính tổng.

Chúng ta Double click vào button Tính tổng và tiến hành viết mã nguồn xử lý cho sự kiện. Mã nguồn tham khảo:

```
private void btnTinh_Click(object sender, EventArgs e)
{
    //Lấy giá trị của textbox
    string giatri = txtNhapGiaTriN.Text;

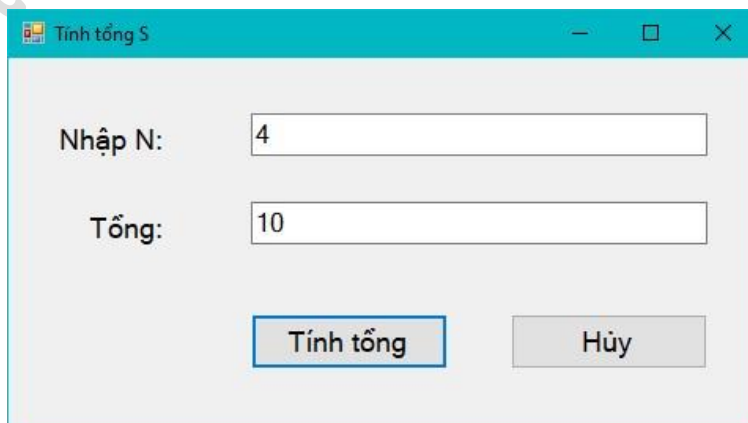
    //Chuyển giá trị kiểu chuỗi mới lấy sang kiểu số nguyên
    int songuyen_n = Convert.ToInt32(giatri);

    //Thực hiện phép tính tổng theo công thức: S = 1+2+3+...+n;
    //Để thực hiện phép tính tổng ta cần có biến tổng để lưu. các số từ 1 đến n là số nguyên nên biến tổng
    ta khai báo kiểu số nguyên và khởi tạo giá trị mặc định là 0.
    int tongS = 0;

    //Sử dụng vòng lặp FOR. giá trị bắt đầu từ 1 và dừng lại khi biến i có giá trị lớn hơn biến
    songuyen_n

    for(int i=1; i<=songuyen_n; i++)
    {
        //Lấy từng giá trị cộng lại và gán vào biến tongS.
        tongS = tongS + i;
    }
    //Gán lại giá trị sau khi đã tính tổng vào TextBox Tổng
    txtGiaTriTong.Text = tongS.ToString();
}
```

Kết quả sau khi chạy chương trình:

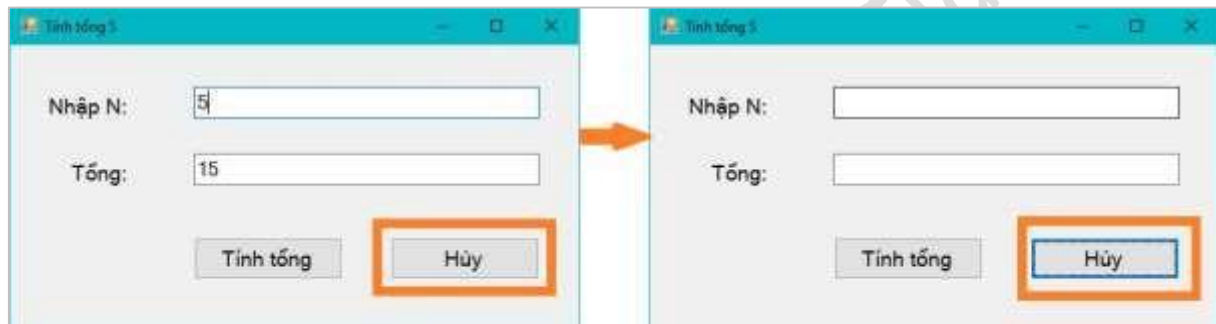


The screenshot shows a simple Windows application window titled "Tính tổng S". It has a light gray background. At the top, there's a title bar with standard Windows window controls (minimize, maximize, close). Below the title bar, there are two text boxes. The first one is labeled "Nhap N:" and contains the number "4". The second one is labeled "Tổng:" and contains the number "10". Below these text boxes, there are two buttons. The first button is labeled "Tính tổng" and has a blue border, indicating it's the active or default button. The second button is labeled "Hủy" and has a gray border.

Xử lý sự kiện cho Button **Hủy** – Làm rỗng làm dữ liệu của các TextBox. Tương tự như Button **Tính tổng**, Double click vào button **Hủy**, sau đó tiến hành viết mã nguồn để xử lý. Mã nguồn tham khảo:

```
1 reference
private void btnHuy_Click(object sender, EventArgs e)
{
    //Gán giá trị rỗng cho Textbox
    txtNhapGiaTriN.Text = "";
    //Gán giá trị rỗng cho Textbox
    txtGiaTriTong.Text = "";
}
```

Kết quả khi thực hiện:



Bài tập nâng cao

CÂU 3.

Xây dựng ứng dụng các phép tính đơn giản với giao diện như sau:

The screenshot shows a Windows-style application window titled "Các phép tính". The background is blue. At the top, the text "CÁC PHÉP TÍNH ĐƠN GIẢN" is displayed in white. Below this, there are two input fields, each preceded by the label "Số thứ 1:". The first input field contains the number "3", and the second contains "5". Between these two input fields are four buttons with red symbols: a plus sign (+), a minus sign (-), a multiplication sign (x), and a division sign (/). Below these buttons is a larger white rectangular box containing the number "8". At the bottom center is a button labeled "Đóng" in blue text.

Hướng dẫn:

- Thiết kế giao diện như sau:

This diagram illustrates the design of the calculator application interface, showing the layout and naming of its components. Red dashed arrows point from text labels to the corresponding UI elements. The components are as follows:

- Form:** The main window, titled "Các phép tính", with the text property set to "Các phép tính".
- Label:** The title text "CÁC PHÉP TÍNH ĐƠN GIẢN" at the top of the form.
- TextBox (Name = txtSo1):** The first input field, preceded by the label "Số thứ 1:".
- TextBox (Name = txtSo2):** The second input field, also preceded by the label "Số thứ 1:".
- Button (Name = btCong, btTru, btNhan, btChia):** A group of four buttons containing the symbols +, -, x, and /.
- Button (Name = btDong):** A button at the bottom labeled "Đóng".
- Label (Name = lbKetqua):** A large rectangular box intended for the calculation result.

- Double click lên button btCong (+), chuyển sang phần code behind, viết code xử lý trong hàm btCong_Click như sau:

```

namespace Cacpheptinh
{
1   public partial class Form1 : Form
    {
1       public Form1()
        {
            InitializeComponent();
        }

1       private void btCong_Click(object sender, EventArgs e)
        {
            //lấy giá trị nhập trong hai textbox
            //chuyển về số và gán vào hai biến a,b
            int a = int.Parse(txtSo1.Text);
            int b = int.Parse(txtSo2.Text);
            int c = a + b;
            //chuyển c thành chuỗi, gán vào thuộc tính Text
            //của label để hiển thị kết quả
            lbKetqua.Text = c.ToString();
        }
    }
}

```

- Thao tác tương tự cho hai button trừ và nhân
- Riêng với code tính phép chia, kết quả phải là số thực:

```

private void btChia_Click(object sender, EventArgs e)
{
    int a = Int32.Parse(txtSo1.Text);
    int b = Int32.Parse(txtSo2.Text);
    double c = (double) a / b; //ép kiểu sang số thực
    lbKetqua.Text = c.ToString();
}

```

- Double click lên button btDong (Đóng), viết code như sau:

```

private void btDong_Click(object sender, EventArgs e)
{
    Close(); //lệnh đóng form
}

```

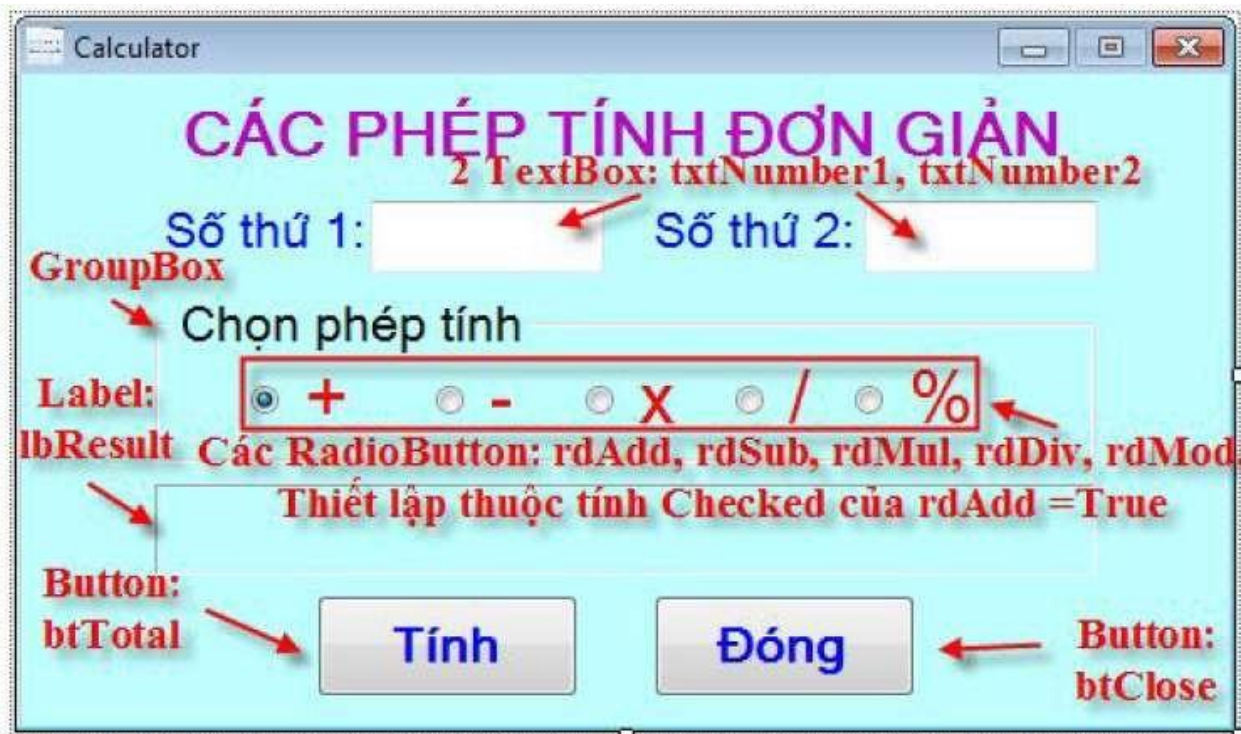
- Biên dịch và chạy chương trình, nhập các giá trị và click lần lượt lên các button xem kết quả.

CÂU 4

Tương tự Câu 3, nhưng thay đổi giao diện như hình bên dưới và có xử lý lỗi nhập sai giá trị:



Hướng dẫn: Thiết kế giao diện như sau:



- Sử dụng try .. catch .. finally, throw, ... để xử lý các lỗi: để trống ô, nhập sai giá trị, chia cho 0, ...
 - Double click lên button Đóng, viết hàm xử lý sự kiện btClose_Click như sau:
- ```
private void btClose_Click(object sender, EventArgs e)
{
 this.Close();
}
```
- Biên dịch, chạy chương trình, lần lượt nhập các giá trị đúng, sai để kiểm tra kết quả.

### CÂU 5:

Viết ứng dụng RandomNumber (chương trình game quay số: người chơi chọn một đáp án (3-10 hoặc 11-18)), sau đó nhấn nút quay số, các số sẽ phát sinh ngẫu nhiên. Nếu tổng ba số đúng trong khoản người chơi đã chọn, điểm được +10, ngược lại điểm bị -10. (điểm ban đầu bằng 0)

Random Number

1

3

1

Chọn:

3-10

11-18

Quay số

ĐIỂM: 10

Hướng dẫn:

Tạo form với giao diện sau:





- Double Click lên vùng trống Form tạo sự kiện FormLoad, chuyển sang phần code behind.
- Sử dụng đối tượng Random để sinh số ngẫu nhiên.
- Double click vào nút quay số, viết code xử lý sự kiện click vào nút này.

#### CÂU 6:

Viết chương trình giúp bé học toán có giao diện như sau:



Yêu cầu:

- Giao diện đầu tiên khi chương trình chạy lên hiển thị phép toán  $0+0$ .
- Click vào nút Tiếp tục, sinh ngẫu nhiên một biểu thức trong các phép cộng, trừ, nhân, chia.
- Người chơi nhập kết quả, click nút Xem, chương trình hiển thị kết quả đúng hoặc sai.
- Có thể click vào các button để nhập số vào ô Trả lời thay vì nhập bằng bàn phím.
- Chương trình chỉ thực hiện các phép toán trên số nguyên nhỏ hơn 100.
- Xử lý ngoại lệ như: không nhập hoặc nhập sai giá trị.

### Câu 7:

- Tạo một ứng dụng Windows Form cơ bản tính tiền công dịch vụ tại phòng nha.
- Với mỗi khách hàng, các dịch vụ cung cấp gồm: tẩy răng, cạo vôi, chụp hình răng và trám răng. Mỗi loại sẽ có chi phí riêng. Cuối cùng tính tổng các chi phí mà người khách phải trả. Lưu ý: chỉ tính tiền khi phần thông tin tên khách hàng đã được nhập (nếu thông tin này chưa có thì chương trình phát sinh MessageBox cảnh báo).
- Ứng dụng có giao diện đơn giản như hình 1 bên dưới.

Dental Payment Application

### Dental Payment Form

Tên khách hàng

|                                          |              |
|------------------------------------------|--------------|
| <input type="checkbox"/> Cạo vôi         | \$100.000    |
| <input type="checkbox"/> Tẩy trắng       | \$1.200.000  |
| <input type="checkbox"/> Chụp hình răng  | \$200.000    |
| Trám răng <input type="text" value="0"/> | \$80.000/cái |

Total

### Câu 8:

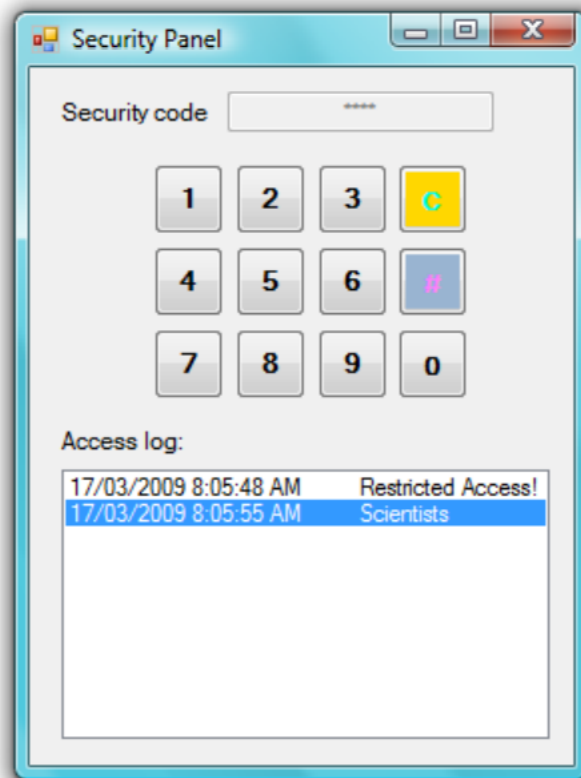
Xây dựng ứng dụng mô phỏng thiết bị Security Panel theo yêu cầu sau:

Một phòng Lab muốn thiết lập một Security Panel đặt bên ngoài cửa. Chỉ cho phép những các cá nhân có trách nhiệm mới được vào và mỗi lần đăng nhập họ phải nhập các security code được cung cấp. Những security code (hay còn gọi là access code) sau được xem là hợp lệ và cung cấp cho các nhóm nhân viên như bảng sau:



| Value             | Group       |
|-------------------|-------------|
| 1645 or 1689      | Technicians |
| 8345              | Custodians  |
| 9998, 1006 - 1008 | Scientist   |

Một khi nhập access code thì sẽ có hai trạng thái: granted hoặc denied. Tất cả thông tin truy cập sẽ được hiển thị trong một khung thông tin bên dưới keypad. Nếu access là granted thì ngày, thời gian, group (technician, custodian, scientist) sẽ được hiển thị ở khung thông tin. Trường hợp access là denied thì ngày, giờ và thông tin “Access denied” sẽ hiển thị ở khung bên dưới. Ngoài ra user nếu chỉ nhấn một con số security code thì sẽ hiển thị ra thông báo là ngày, giờ và “Restricted Access”



### Câu 9:

Từ Câu 8 bổ sung các chức năng:

- Chức năng Log file: Tất cả thông tin login dù access granted hay denied đều được ghi nhận vào file dạng text. File này được lưu trữ cùng với thư mục của ứng dụng.
- Chức năng: Cho phép user dùng keyboard gõ các phím số tương ứng. Ví dụ gõ phím 0 thì tương ứng với việc nhấn button 0...

Bài thực hành Lập trình giao diện

### BUỔI 3 – 4 - 5

#### Câu 10:

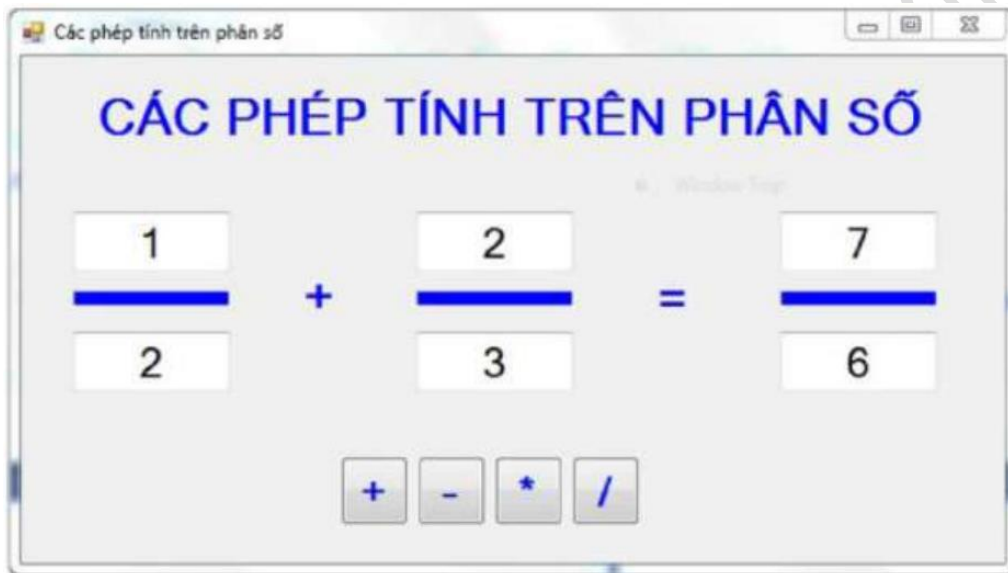
Cài đặt lớp phân số với các yêu cầu sau:

Dữ liệu thành phần là hai số nguyên tuSo và mauSo

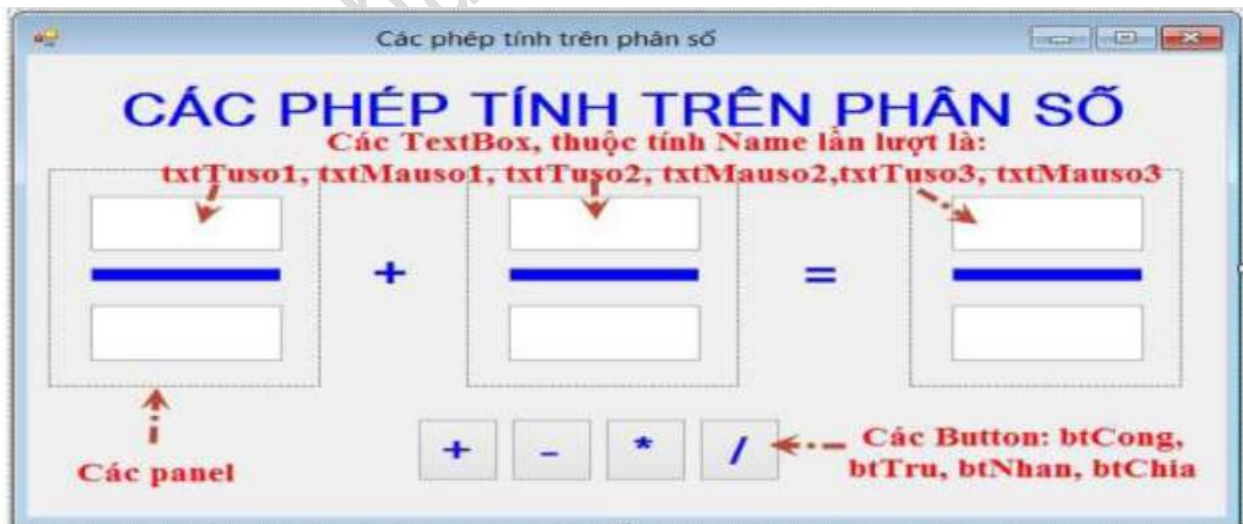
Xây dựng các thuộc tính (property) cho phép truy xuất dữ liệu của lớp.

Cài đặt các phương thức cộng, trừ, nhân, chia, tối giản phân số.

Áp dụng vào chương trình để thực hiện các phép tính trên phân số với giao diện như sau:



Hướng dẫn: - Thiết kế giao diện như sau



### Câu 11:

Phát triển Câu 11 thành game sinh ngẫu nhiên 2 phân số và một phép toán. Người dùng nhập vào đáp án, chương trình cho biết đáp án đúng hay sai và tính điểm cho người chơi. (Cần xây dựng thêm phương thức so sánh bằng giữa phân số trong lớp phân số)

### Câu 12:

Viết chương trình tăng bóng với giao diện sau:



Yêu cầu: Quả bóng tăng tự do, khi gặp các biên thì dội ngược lại.

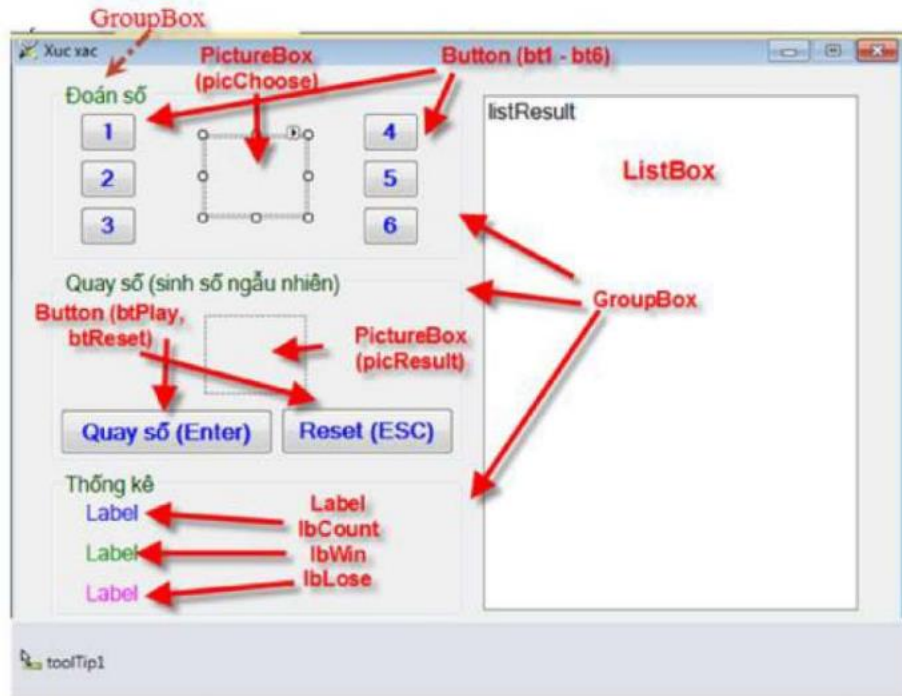
### Câu 13:

Viết ứng dụng đoán XucXac với giao diện như sau:



Yêu cầu: Người chơi đoán một số, click nút Quay số thì sẽ hiển thị thống kê số lượt quay đúng/ sai trên các label và listbox như hình.

Hướng dẫn: Thiết kế giao diện như sau:



| Tooltip      |         |
|--------------|---------|
| ToolTipIcon  | Info    |
| ToolTipTitle | Đoán số |

**Câu 14:**

Tạo Form frmSinhvien thực hiện thao tác nhập danh sách sinh viên:

Yêu cầu:

- Nút > dùng để di chuyển các mục (còn gọi là item) được chọn từ listbox trái qua listbox phải và ngược lại cho <
- Nút >> dùng để di chuyển tất cả các mục (kể cả không được chọn) từ listbox trái qua listbox phải và ngược lại cho <<
- Nút Xóa dùng để xóa những item được chọn trên textbox trái

### **Câu 15:**

Từ Câu 7 bổ sung các chức năng:

- Bổ sung một ListBox vào form tính tiền, ListBox này dùng để lưu trữ các thông tin tính tiền của khách hàng. Mỗi thông tin tính tiền sẽ được lưu trên một dòng trong ListBox (một item của listbox). Một item gồm hai thông tin:
- Bổ sung chức năng Lưu: cho phép lưu trữ các thông tin tính tiền của khách hàng trong một file text. File text này có định dạng mỗi dòng là một thông tin tính tiền: gồm tên khách hàng + tổng số tiền.
- Bổ sung chức năng Đọc file: Cho phép load thông tin tính tiền khách hàng từ một file lưu trữ .
- Tạo chức năng tùy chọn: Cho phép người tính tiền phòng nha có thể chỉnh lại đơn giá cho từng dịch vụ. Hiện tại ứng dụng trên các đơn giá là fix, ta sẽ cải tiến lại chức năng này. Sinh viên nên tạo file chứa đơn giá từng dịch vụ, tạo form cho phép người quản lý phòng nha có thể hiệu chỉnh lại giá tiền này, lưu lại file đó, và mỗi lần ứng dụng chạy thì đọc file đó để lấy đơn giá.

### **Câu 16 (nâng cao):**

Các chức năng trong Câu 10 là dùng file dạng text. Phần nâng cao này sinh viên tìm hiểu file XML và xây dựng lại các file theo định dạng XML phục vụ các yêu cầu đã mô tả (Sinh viên tìm hiểu phần đọc và lưu trữ file dạng XML trong C#)

### **Câu 17:**

Tạo Form frmChuongTrinhXemPhim với thanh menu có control Multimedia như sau



Yêu cầu: Tạo ứng dụng chứa Windows Media Player control cho phép Play các file video/sound theo nhiều dạng format:

MPEG (Motion Pictures Expert Group): video

AVI (Audio-video Interleave): video

WAV (Windows Wave-file Format): audio

MIDI (Musical Instrument Digital Interface): audio

Ứng dụng gồm :

- Menu bar có 1 Menu File và 2 Sub Menu Open và Exit, có đường kẻ ngang giữa 2 Sub Menu
- Khi chọn Sub Menu sẽ thoát ứng dụng
- Khi chọn Sub Menu Open sẽ mở hộp thoại Open File để chọn mở file Media - Control Multi Media sẽ phát file Media đã chọn.
- StatusStrip hiển thị: Ngày giờ hệ thống và thay đổi giờ theo mỗi giây.



## BUỔI 6 -7 -8

### Câu 18:

Viết chương trình di chuyển ảnh với giao diện sau:



- Yêu cầu:
  - Dùng chuột hoặc các phím mũi tên di chuyển ảnh vào trong khung bên phải.
  - Khi ảnh được di chuyển vào trong khung (FlowLayoutPanel), ảnh được thêm vào panel và góc trái form sẽ xuất hiện ngẫu nhiên một hình khác.
  - Các hình được lấy ngẫu nhiên từ một thư mục hình đã có.

### Câu 19:

Viết chương trình StringProcess để xử lý chuỗi với giao diện và chức năng như sau:



Xử lý chuỗi

## CHƯƠNG TRÌNH MINH HỌA XỬ LÝ CHUỖI

Chuỗi s1:

Chuỗi s2:  Chuỗi s3:

Chèn chuỗi s2 vào s1

Vị trí:

- Yêu cầu:
  - Chuỗi “Chương trình minh họa xử lý chuỗi” di chuyển (cuộn từ trái sang phải).
  - Nút Chèn: Chèn chuỗi s2 vào chuỗi s1 tại vị trí được nhập trong ô vị trí.
  - Nút Thay thế s2 bằng s3: Thay thế chuỗi con s2 trong s1 (nếu có) bằng chuỗi s3.
  - Đảo từ trong s1: Đảo các từ trong chuỗi s1.
  - Xóa s2 trong s1: Xóa chuỗi con s2 trong s1 (nếu có).
  - Chuẩn hóa chuỗi: Chuẩn hóa chuỗi s1 theo nguyên tắc:
    - Xóa các khoảng trắng thừa, đầu, giữa và cuối chuỗi. Chỉ giữ lại giữa các từ một khoảng trắng.
    - Chuyển thành chữ hoa các ký tự đầu mỗi từ.

### Câu 20:

Tạo Form frmSinhvien thực hiện thao tác nhập danh sách sinh viên:

Tại một quán ăn nhanh, người ta muốn toàn bộ công việc order các món ăn được diễn ra một cách nhanh chóng và chuẩn hóa. Nên họ xây dựng một hệ thống e-order, hệ thống này được thực hiện thông qua một chương trình order cài đặt trên máy PDA, mỗi người phục vụ sẽ được cung cấp một PDA, khi khách hàng gọi món thì người phục vụ này sẽ đến tận bàn, và sử dụng chương trình e-order đó trên PDA để order món ăn. Khi việc order xong thì người phục vụ sẽ chọn chức năng send order và thông tin này sẽ được gửi xuống nhà bếp thông qua hệ thống wireless được cài đặt...

Sinh viên hãy viết lại chương trình order trên theo dạng Windows Form. Giao diện chương trình Order được thể hiện như hình dưới.

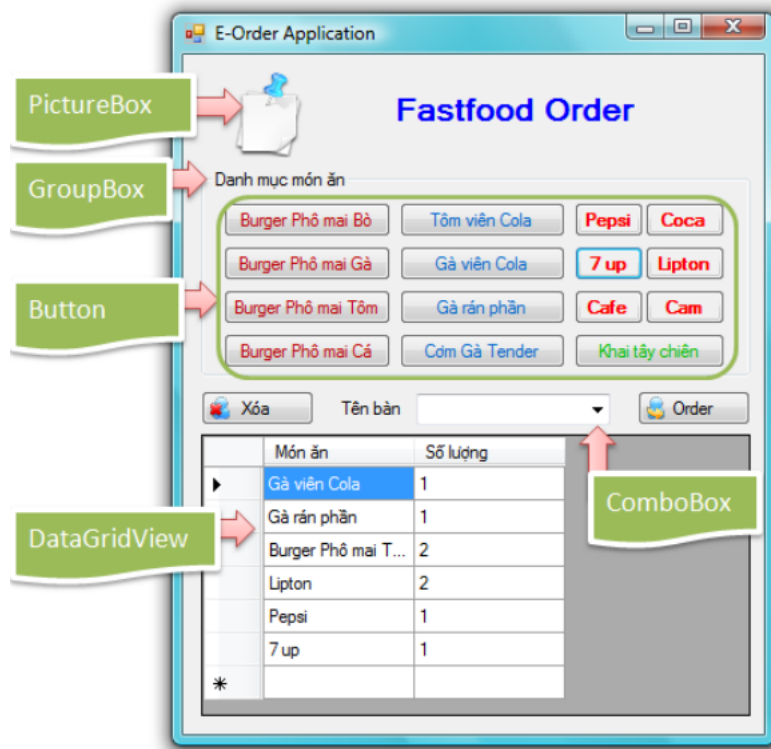
| Món ăn              | Số lượng |
|---------------------|----------|
| Gà viên Cola        | 1        |
| Gà rán phần         | 1        |
| Burger Phô mai T... | 2        |
| Lipton              | 2        |
| Pepsi               | 1        |
| 7 up                | 1        |
| *                   |          |

Thao tác sử dụng:

- Người phục vụ sẽ chọn tên bàn được list trong ComboBox, sau đó tùy theo yêu cầu gọi món của client mà người phục vụ sẽ chọn món ăn, thức uống thông qua danh mục món ăn được thể hiện bởi danh sách các button. Mỗi lần chọn món ăn sẽ bổ sung thêm số lượng gọi món là 1, ví dụ 2 lần chọn Bugar Phô mai Bò thì số lượng là 2 và danh mục gọi món của bàn đó sẽ hiển thị trong danh sách bên dưới.

- Kết thúc quá trình gọi món ăn thì người phục vụ sẽ chọn chức năng “Order”, thông tin này sẽ được gửi cho đầu bếp... Mô tả các control trên Form E-Order:

Mô tả các control trên Form E-Order:



Hướng dẫn:

- Trong chương trình sử dụng lớp DataTable để chứa thông tin order, bao gồm có 2 cột: {FoodName} chứa tên món ăn và {Quantity} số lượng.
- Cách tạo bảng này như sau:

VD: Biến DataTable trong Form1 là tbOrder thì code tạo bảng chứa dữ liệu order là:

```
tbOrder.Columns.Add("FoodName");//thêm cột FoodName
```

```
tbOrder.Columns.Add("Quantity");// thêm cột Quantity
```

Lưu ý: Phần tạo đối tượng DataTable và khởi tạo cấu trúc (Column) của DataTable được đặt trong hàm xử lý sự kiện Load của Form.

Mỗi khi click vào món ăn thì chương trình sẽ tìm trong DataTable này xem có món ăn đó chưa, nếu chưa có thì thêm dòng mới vào với tên món ăn và số lượng là 1. Ngược lại đã có chọn món này thì số lượng của nó tăng 1.

Cách thêm một dòng (món ăn) mới vào DataTable tbOrder

```

DataRow r = tbOrder.NewRow();//tạo dòng mới
// thiết lập cột FoodName với món ăn được chọn
r["FoodName"] = <tên món ăn được chọn>;
r["Quantity"] = 1; // thiết lập cột Quantity
tbOrder.Rows.Add(r); // thêm vào bảng

```

Sử dụng thuộc tính DataSource của DataGridView để kết buộc với dữ liệu trong DataTable, phần code này khai báo sau khi khởi tạo DataTable (trong hàm xử lý sự kiện load của form). VD: tên của DataGridView trong chương trình là dataGridView1 và biến DataTable là tbOrder thì code kết buộc như sau:

```

// binding nội dung trong DataTable cho DataGridView
dataGridView1.DataSource = tbOrder;

```

### **Câu 21 (nâng cao):**

Việc tìm kiếm một món ăn có trong bảng tbOrder đơn giản là duyệt qua từng dòng và kiểm tra tên của món ăn. Tuy nhiên, ta có thể làm cách khác bằng việc dùng hàm Find của thuộc tính Rows trong DataTable. Nếu muốn sử dụng hàm Find được thì bảng Order phải được thiết lập khoá chính. Do đó trong bảng tbOrder trên ta có thể tạo khoá chính là thuộc tính “FoodName” và sau đó tìm kiếm theo thuộc tính này.

Chức năng Clear ở phần trên là xoá toàn bộ thông tin order của một bàn, chức năng này đôi khi bất tiện, vì nếu người phục vụ nhập sai một món ăn, hoặc muốn xoá một món ăn khi order thì phải xoá hết toàn bộ thông tin và phải nhập lại. Sinh viên hãy mở rộng bằng cách cho user chọn một dòng trong DataGridView và xoá dòng đó (lưu ý, xoá trong DataTable, DataGridView chỉ là phần hiển thị). Sinh viên đọc thêm chi tiết lớp DataTable và control DataGridView trên MSDN Online.

### **Câu 22:**

Thêm vào project form frmNhanvien với listview như sau:

Yêu cầu :

Nhập thông tin vào phần “thông tin chi tiết”, khi nhấn các nút lệnh sẽ thực hiện như sau:

- Nút Thêm: Thêm các thông tin vào listview. Chú ý : Kiểm tra thông tin họ tên nhân viên không được rỗng, nếu rỗng thì không cho thêm.

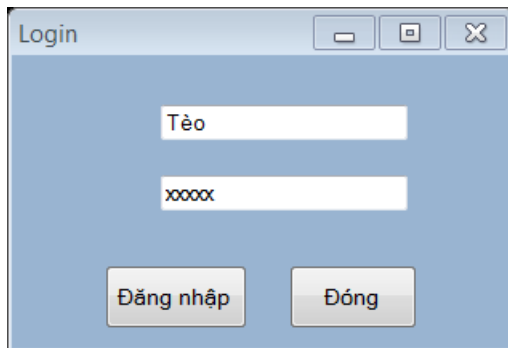
- Nút Xóa: Xóa 1 dòng trên listview (chú ý nếu người dùng chưa chọn 1 dòng trên listview thì yêu cầu người dùng phải chọn rồi mới xóa).
- Nút Sửa: Cập nhật thông tin đã điều chỉnh 1 dòng trên listview.
- Thoát: thoát khỏi chương trình.
- Khi chọn 1 dòng trên listview thì nội dung của dòng được chọn sẽ được thể hiện trên các ô “Thông tin chi tiết”

Hướng dẫn:

| STT | Tên Control  | Loại           | Tên hàm xử lý                       |
|-----|--------------|----------------|-------------------------------------|
| 1   | lsvNhanVien  | ListView       | lsvNhanVien_SelectedIndexChanged    |
| 2   | txtHoTen     | TextBox        |                                     |
| 3   | dtpNgaySinh  | DateTimePicker |                                     |
| 4   | txtDiaChi    | TextBox        |                                     |
| 5   | txtDienThoai | TextBox        |                                     |
| 6   | cboBangCap   | ComboBox       | cboBangCap.SelectedValue.ToString() |
| 7   | btnThem      | Button         | btnThem_Click                       |
| 8   | btnXoa       | Button         | btnXoa_Click                        |
| 9   | btnCapNhat   | Button         | btnCapNhat_Click                    |
| 10  | btnSua       | Button         | btnSua_Click                        |
| 11  | btnHuy       | Button         | btnHuy_Click                        |
| 12  | btnThoat     | Form           | btnThoat_Click                      |
| 13  | frmNhanVien  | Form           | frmNhanVien_Load                    |

### Câu 23:

Xây dựng ứng dụng Quản lý nhân viên như sau:



A login window titled "Login" with a blue background. It contains two text input fields: the first is labeled "Tên" (Name) and the second is masked with "xxxxx". Below the fields are two buttons: "Đăng nhập" (Login) and "Đóng" (Close).



An application window titled "Quản lý nhân viên" (Employee Management) with a teal header and a white body. The header contains the title "QUẢN LÝ NHÂN VIÊN". The body is divided into two main sections. On the left, there are form fields for "Họ tên:" (Last Name) and "Ngày sinh:" (Date of Birth) with a calendar icon. Below these are radio buttons for "Nam" (Male) and "Nữ" (Female), and two buttons: "Thêm" (Add) and "Xóa" (Delete). At the bottom left is a text input field labeled "Tèo". On the right, there is a table with three columns: "Họ tên", "Ngày sinh", and "Giới tính". The table contains two rows of data, each with a small person icon next to the name.

| Họ tên                                                                                       | Ngày sinh  | Giới tính |
|----------------------------------------------------------------------------------------------|------------|-----------|
|  Nguyễn Thị | 01/12/2000 | Nữ        |
|  Trần Tốt   | 09/08/2000 | Nam       |

## **Câu 24 [Sinh viên tự tìm hiểu]:**

Sử dụng công cụ thiết kế giao diện ribbon Devexpress

Bộ thiết kế Devexpress là 1 công cụ của hãng thứ 3, cho phép người dùng tạo ra các giao diện đẹp, bóng bẩy hơn. Khắc phục những hạn chế của các công cụ toolbox trên Visual Studio. Cho phép lập trình viên làm được rất nhiều việc phức tạp. Giúp tạo ứng dụng báo cáo, in ấn, Xuất File, Tạo menu giống Office, tạo Group Narbar Control đẹp mắt...

DevExpress là một Framework được viết cho nền tảng .NET Framework. Nó cung cấp các control và công nghệ để phục vụ cho quá trình phát triển phần mềm.

Hiện nay có nhiều công ty lập trình sử dụng Devexpress để làm việc cho khách hàng, nhiều sinh viên dùng để làm đồ án của trường. Hỗ trợ C#, VB, ASP.NET,... Đặc biệt hỗ trợ từ tìm kiếm nâng cao, hiện hình ảnh ... Đối với Devexpress thì rất dễ dàng học , phần trợ giúp của cực kỳ tốt, có demo và hướng dẫn chi tiết

Các trang web tham khảo:

[1] <https://www.devexpress.com/>

[2] <https://www.devexpress.com/Home/Training/>

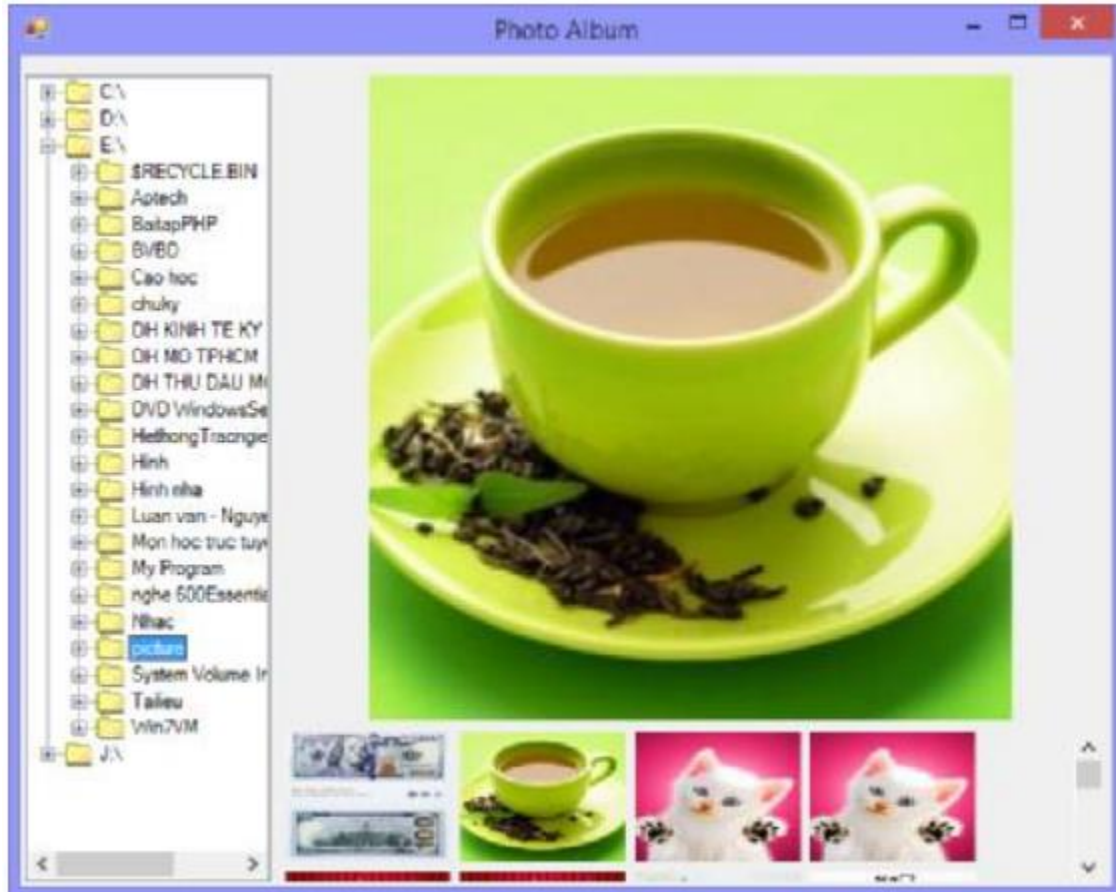
[3] <http://www.youtube.com/user/DeveloperExpress>

Các công cụ khác tạo Ribbon: DotNetBar, Component One...

## BUỔI 9-10

### Câu 25:

Xây dựng ứng dụng PhotoAlbum với giao diện và chức năng sau:



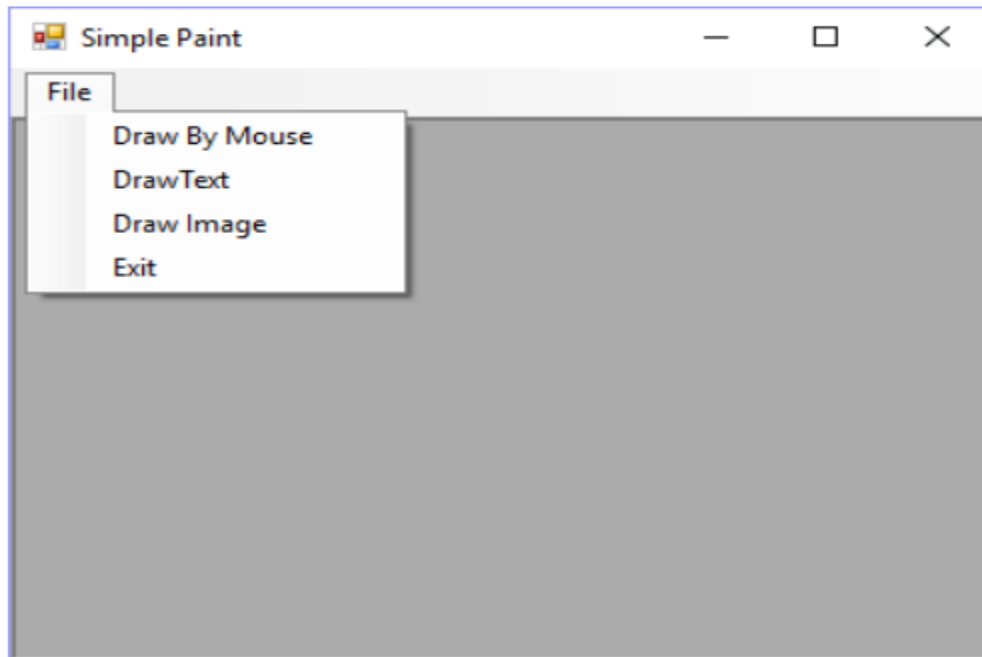
- Yêu cầu:
  - Khi chạy ứng dụng, danh sách các ổ đĩa sẽ hiển thị trên treeview
  - Click lên một thư mục chứa hình ảnh, các hình ảnh trong thư mục đó sẽ hiển thị trên FlowLayoutPanel
  - Chọn 1 hình trên panel, hình sẽ hiển thị trên pictureBox phía trên.

### Câu 26:

Viết ứng dụng SimplePaint với giao diện như sau

Form chính:





Menu Draw Text: mở form con vẽ các chuỗi sau:



Menu Draw by Mouse: vẽ tự do bằng chuột



Yêu cầu:

- Giữ chuột trái để rê vẽ
- Nhấn phím R,G,B: để chuyển màu
- Nhấn phím ↑↓ để tăng và giảm độ dày nét vẽ
- Menu Draw Image: vẽ hình.
- Menu Exit: Đóng ứng dụng

-----Kết thúc bài thực hành-----

Sinh viên báo cáo đồ án theo nhóm.