

Trường ĐH CT TP. HCM Khoa: CNTT Bộ môn: CNPM Môn: Lập trình .NET	BÀI 6 GAO DIỆN (tiếp theo)	
---------------------------------------------------------------------------	-------------------------------------------------------	--

A. MỤC TIÊU:

- Thiết kế được giao diện Windows Form từ các control trong C#.
- Sử dụng được các Controls Textbox, Label, Button, ListView, Menu, ContextMenuStrip, Timer, Common Dialog.

B. DỤNG CỤ - THIẾT BỊ THỰC HÀNH CHO MỘT SV:

STT	Chủng loại – Quy cách vật tư	Số lượng	Đơn vị	Ghi chú
1	Computer	1	1	

C. NỘI DUNG THỰC HÀNH

1. Cơ sở lý thuyết

1.1 Kiến thức cần nhớ

– **ListView:**

- + Thuộc tính: Columns, Items, View, FullRowSelect, MultiSelect, SelectedItems
- + Sự kiện: SelectedIndexChanged, Click
- + Phương thức: Add, AddRange, RemoveAt, Remove, IndexOf, Contains
 - **Remove()**: Bỏ mục chọn.
 - **RemoveAt()**: Bỏ mục chọn tại vị trí i.
 - **Contains()**: Trả về True nếu có mục chọn trong danh sách và trả về False nếu không có mục chọn trong danh sách.
 - **Clear()**: Xóa tất cả các mục chọn.
 - **IndexOf()**: Trả về vị trí mục chọn trong danh sách, nếu không tìm thấy sẽ trả về -1.

Một số thuộc tính thường dùng của ListView:

Thuộc tính	Mô tả
View	Thuộc tính View quy định cách hiển thị các Item trong ListView. Thuộc tính View có 5 giá trị: Detail, LargeIcons, List, SmallIcons, Titles
MultiSelect	True/ False: Cho phép hoặc không cho phép chọn một lúc nhiều Item trong ListView

Thuộc tính	Mô tả
FullRowSelect	FullRowSelect khi chọn dòng dữ liệu highlighted cả dòng hay chỉ ô được chọn
GridLines	GridLines nếu thiết lập True sẽ hiển thị các dòng và cột dạng lưới, thiết lập False không hiển thị dạng lưới
SelectedItems	Trả về tập các Items được chọn trong ListView
LargeImageIcon	Gán đối tượng ImageList cho ListView
SmallImageIcon	Gán đối tượng ImageList cho ListView
FocusedItem.Index	Trả về chỉ số dòng được chọn trong ListView
SelectedIndices.Count	Trả về số lượng Item được chọn trong ListView
SelectedIndices	Trả về danh sách chỉ mục các Item được chọn
Items	<p>Trả về các Item chứa trong ListView. Một số phương thức và thuộc tính thường dùng của ListView.</p> <p>Count: Đếm số lượng Item trong L</p> <p>Insert(<i>i</i>, <Item mới>): Chèn thêm Item mới vào vị trí <i>i</i> trong ListView</p> <p>Add(<Item mới>): Thêm Item vào cuối ListView</p> <p>Remove(<Item cần xóa>): Xóa Item khỏi ListView</p> <p>RemoveAt(<i>i</i>): Xóa Item có chỉ số <i>i</i> khỏi ListView</p> <p>Contains(<Item cần tìm>): Trả về True nếu tìm thấy trong ListView, trả về False nếu không có trong ListView</p> <p>IndexOf(<Item cần tìm>): Nếu có trong ListView thì trả về chỉ số của item tìm thấy trong ListView, nếu không tìm thấy sẽ trả về -1</p>

Một số phương thức thường dùng của ListView:

Phương thức	Mô tả
Clear()	Xóa tất cả các Item và Column trong ListView

Phương thức	Mô tả
Sort()	Sắp xếp các Item trong ListView
GetItemAt(x, y)	Lấy Item tại vị trí tọa độ x và y (<i>x và y có thể lấy được thông qua sự kiện Click chuột</i>)

Một số sự kiện thường dùng của ListView:

Sự kiện	Mô tả
SelectedIndexChanged	Sự kiện phát sinh khi có sự thay đổi về chỉ mục được chọn của Item trên ListView
ItemSelectionChanged	Sự kiện phát sinh khi có sự thay đổi lựa chọn một Item trên ListView
ItemCheck	Xảy ra khi trạng thái chọn của Item thay đổi
ColumnClick	Sự kiện phát sinh khi có sự thay đổi lựa chọn một Item trên ListView
MouseClick	Sự kiện phát sinh khi nhấp chuột chọn một Item trong ListView

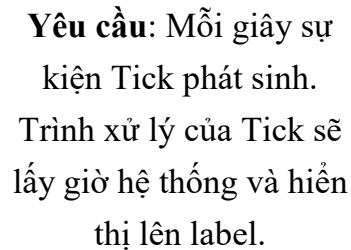
- **MenuStrip:** cung cấp nhóm lệnh có quan hệ với nhau cho các ứng dụng Windows Form
 - + Khai báo trình xử lý sự kiện Click cho MenuStrip:
 - Kích đúp vào menu item của MenuStrip để tạo sự kiện.
 - Hoặc trong cửa sổ Properties → chọn Event, double click vào sự kiện Click.
- **ContextMenu:** Khi user kích chuột phải lên control thì sẽ hiển thị Context Menu đã cài đặt sẵn. Cách tạo và sử dụng tương tự MenuStrip.
- **Timer** (bộ định thời gian):
 - + Thuộc tính: Enabled, Interval.
 - + Sự kiện: Tick.
 - + Phương thức: Start, Stop.
- **Dialog:** SaveDialog, OpenFileDialog, ColorDialog, FontDialog

Các bước tạo một Dialog:

- ❑ 1. Kéo thả dialog vào form

- ## ❑ 2. Khai báo đối tượng và tạo thể hiện của lớp CD

Bài 1: Thực hiện form sau:



- Các control được sử dụng trong form gồm: một Label (label1) (dùng để hiển thị giờ lấy từ hệ thống), một Timer (timer1). Timer được khai báo với các thuộc tính sau:
 - Enabled: Enable sự kiện Tick.
 - Interval: khoảng thời gian chờ giữa 2 lần gọi Tick (đơn vị mili giây)
- **Các sự kiện được xử lý:**

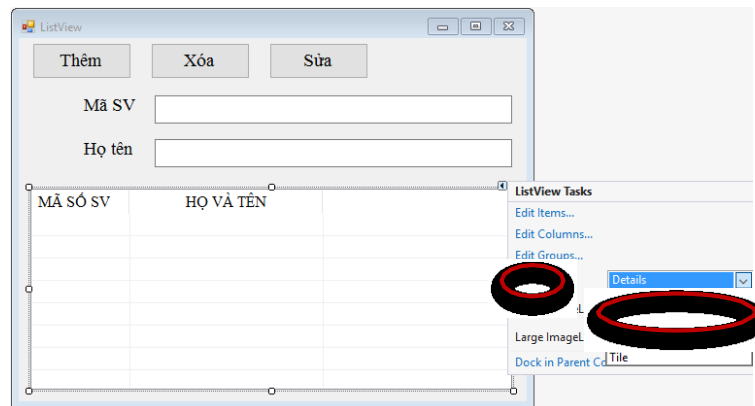
Bài 2: Thiết kế form sau:

The screenshot shows a Java Swing application window titled "ListView". It features a standard Windows-style title bar with minimize, maximize, and close buttons. The main area contains three buttons: "Thêm" (Add), "Xóa" (Delete), and "Sửa" (Edit). Below these buttons are two text input fields, one labeled "Mã SV" (Student ID) and another labeled "Họ tên" (Full Name). At the bottom of the window is a table with two columns, "MÃ SỐ SV" and "HỌ VÀ TÊN", which is currently empty.

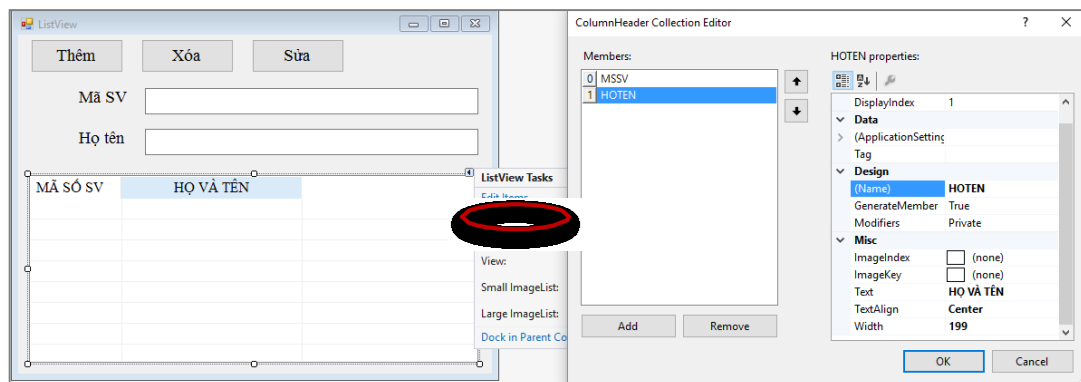
Hướng dẫn:

– Thiết kế giao diện

Bước 1: Chọn chế độ hiển thị View cho listview:



Bước 2: Thiết kế cột (Columns) cho listview:



Bước 3: Thiết lập thuộc tính GridLines, FullRowSelect (nếu cần) cho listview:

– Các sự kiện xử lý cho button:

+ Button **Thêm**:

```
private void btnthem_Click(object sender, EventArgs e)
{
    //cách 1
    //lstv1.Items.Add(txtmssv.Text).SubItems.Add(txthoten.Text);
    //cách 2
    ListViewItem item = new ListViewItem();
    ListViewItem.ListViewSubItem subitem = new
        ListViewItem.ListViewSubItem();

    item.Text = txtmssv.Text;
    subitem.Text = txthoten.Text;
    item.SubItems.Add(subitem);
    lstv1.Items.Add(item);
}
```

```
//cách 3
/*ListViewItem item1 = new ListViewItem(new[] { txtmssv.Text,
                                                    txthoten.Text });

lstv1.Items.Add(item1); */
}
```

🔗 **Lưu ý:** Sinh viên tự viết phương thức kiểm tra trùng mã số sinh viên trước khi thêm vào danh sách.

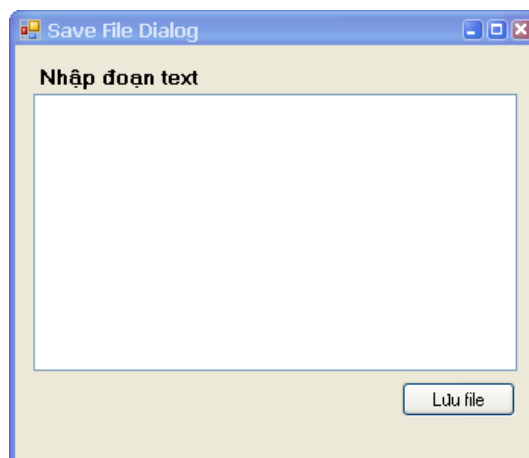
+ Button **Xóa:**

```
private void btnxoa_Click(object sender, EventArgs e)
{
    foreach (ListViewItem l in lstv1.SelectedItems)
    {
        l.Remove();
    }
}
```

+ Button **Sửa:**

```
private void btnsua_Click(object sender, EventArgs e)
{
    if (lstv1.FocusedItem != null && lstv1.FocusedItem.Index >= 0)
    {
        ListViewItem item = lstv1.FocusedItem;
        item.SubItems[0].Text = txtmssv.Text;
        item.SubItems[1].Text = txthoten.Text;
    }
}
```

Bài 3. Thực hiện Form sau:

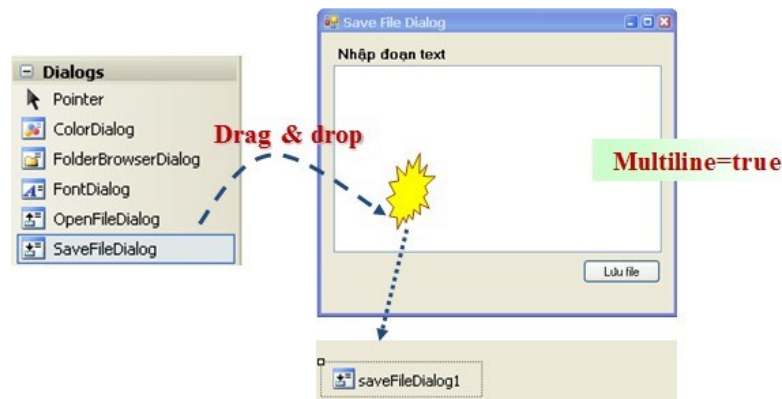


Yêu cầu:

- Nhập văn bản vào textbox, sau đó Lưu nội dung đã nhập thành file: *.txt
- Bổ sung phần đọc file text từ ví dụ trên: mở file *.txt và hiển thị nội lên TextBox.

Hướng dẫn:

- **Thiết kế giao diện**



- **Các sự kiện:**

- + **Button Lưu File:**

```
SaveFileDialog svg = new SaveFileDialog();
svg.Filter = "Text files | *.txt";
if (svg.ShowDialog() == DialogResult.OK)
{
    string filename;
    filename = svg.FileName;
    textBox1.Text = filename;
    StreamWriter sw = File.CreateText(filename);
    sw.Write(richTextBox1.Text);
    sw.Close();
}
```

- + **Button Mở File**

```
//Lọc loại đuôi file theo mong muốn
openFileDialog1.Filter = "All Text File (*.txt)|*.txt|All Document (*.pdf)|*.pdf";
//Nếu dialog openFileDialog1 chọn OK
if (openFileDialog1.ShowDialog() == DialogResult.OK)
{
    //Gán đường dẫn lên Textbox
    textBox1.Text = openFileDialog1.FileName;
}
```

```

//Đọc dữ liệu từ file txt dùng
StreamReader f = new StreamReader(openFileDialog1.FileName);
//Đọc dữ liệu text và gán vào richTextBox
richTextBox1.Text = f.ReadToEnd();
// Đóng luồng đọc
f.Close();
}

```

2. Bài tập tại lớp

Bài 1. Thiết kế giao diện sau:

The form contains the following controls:

- Họ và Tên:** Text box
- Mã sinh viên:** Text box
- Giới tính:** Radio buttons for **Nam** (selected) and **Nữ**
- Ngoại ngữ:** Checkboxes for **Anh** (checked), **Pháp**, and **Hoa**
- Dân tộc:** Dropdown menu with **Kinh** selected
- Table:**

Họ tên	Mã sinh viên	Giới tính	Ngoại ngữ	Dân tộc
- Buttons:** **Thêm** (green + icon), **Xóa** (red - icon), **Sửa** (purple circular arrow icon)

Yêu cầu:

- Khi Load form lên thực hiện thêm dữ liệu vào Combobox dân tộc.
- Thực hiện Thêm, Xóa, Sửa từ các control lên Listview. Lưu ý “Thêm không cho trùng mã sinh viên”, không cho sửa Mã sinh viên.
- Sử dụng ContextMenuStrip cho thực hiện Xóa khi kích chọn phải chuột vào Listview.
- Hỏi xác nhận trước khi đóng Form.

Bài 2. Thiết kế chương trình quản lý thông tin tài khoản cho một ngân hàng với giao diện sau:

QUẢN LÝ THÔNG TIN TÀI KHOẢN

Số tài khoản:

Tên khách hàng:

Địa chỉ khách hàng:

Số tiền trong tài khoản:

Thêm Lưu Xóa Thoát

STT	Số tài khoản	Tên khách hàng	Địa chỉ	Số tiền

Tổng tiền:

Yêu cầu:

- Form load lên có hiện trạng như hình trên.
- Khi nhấn vào button “Thêm”, text của button chuyển thành “Hủy” và Enabled button “Lưu”, các control textbox là rỗng, đặt con trỏ vào textbox Số tài khoản.
- Button “Lưu” có tác dụng Lưu dữ liệu từ các textbox vào trong listview.
- Button “Xóa” có tác dụng xóa mẫu tin hiện hành trong listview (Enabled = true khi một mẫu tin được chọn trong listview).
- Khi click một khách hàng trong listview, hiển thị thông tin lên các textbox tương ứng.
- Thực hiện tính Tổng tiền, STT được đánh tự động khi thực hiện Thêm, Xóa.

QUẢN LÝ THÔNG TIN TÀI KHOẢN

Số tài khoản:

Tên khách hàng:

Địa chỉ khách hàng:

Số tiền trong tài khoản:

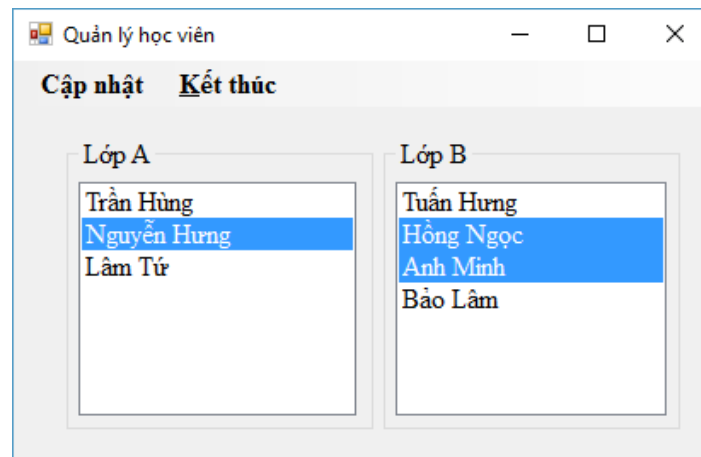
Hủy Lưu Xóa Thoát

STT	Số tài khoản	Tên khách hàng	Địa chỉ	Số tiền
1	0160901609	Nguyễn Văn An	TPHCM	5,200,000
2	0160901610	Trần Văn Anh	Tây Ninh	1,200,000

Tổng tiền:

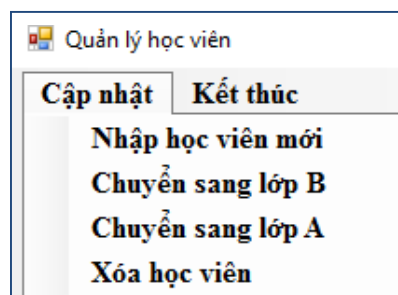
3. Bài tập nâng cao

Bài 1. Thiết kế form có giao diện sau:



Yêu cầu:

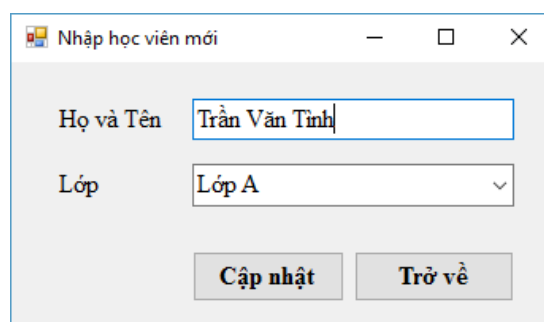
- Khi người sử dụng click vào mục kết thúc thì xuất hiện một MessageBox hỏi người sử dụng sau đó mới quyết định có dừng chương trình hay không.
- Trong menu Cập Nhật có các mục:



- + Xóa học viên: Xóa tất cả các học viên đang chọn trong Lớp A và Lớp B. (các listbox Lớp A và Lớp B cho phép chọn cùng lúc nhiều mục).
- + Chuyển sang lớp A: Chuyển các mục đang chọn trong Lớp B sang Lớp A
- + Chuyển sang lớp B: Chuyển các mục đang chọn trong Lớp A sang Lớp B

🔍 **Lưu ý:** Khi hai mục trên được chọn nếu không có học viên nào được chọn trong listbox thì xuất hiện một MessageBox yêu cầu người sử dụng phải chọn học viên trước.

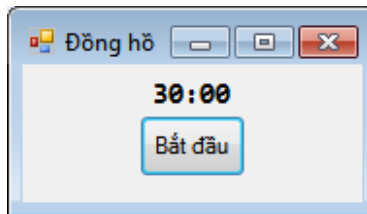
- + Nhập học viên mới: Khi mục này được chọn thì xuất hiện form cho phép nhập thông tin như sau (combobox Lớp: chứa Lớp A và Lớp B).



- Khi người sử dụng click vào button Cập Nhật:
 - Nếu Họ và Tên khác rỗng thì:
 - Phụ thuộc vào lớp đã chọn trong combobox cập nhật họ tên vào listbox tương ứng trên form chính.
 - Nếu Họ và Tên rỗng:
 - Xuất hiện một MessageBox yêu cầu phải nhập họ tên
 - Đặt con trỏ vào textbox Họ và Tên.
- Khi người sử dụng click vào button Trở về thì đóng form nhập thông tin và trở về form chính.

4. Bài tập về nhà

Bài 1. Thiết kế form có giao diện thực hiện đếm ngược thời gian:



Bài 2. Thiết kế form main có menustrip, gắn tất cả bài tập trong tuần này và hiển thị các form khi chọn menu tương ứng.

Họ tên	Mã sinh viên	Giới tính	Ngoại ngữ	Dân tộc