

MÔN : LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Bài thực hành số 3.1 : Xây dựng ứng dụng OOP đơn giản & tìm hiểu sự tương tác giữa các đối tượng

I. Mục tiêu :

- Giúp SV làm quen với qui trình thiết kế trực quan 1 ứng dụng Dialog Based.
- Giúp SV làm quen với việc dùng lại linh kiện phần mềm.
- Giúp SV thấy cụ thể cấu trúc ứng dụng cấu thành từ các đối tượng.
- Giúp SV thấy sự tương tác giữa các đối tượng.

II. Nội dung :

- Xây dựng ứng dụng Dialog Based cho phép người dùng chơi các file multimedia.
- Dịch và chạy chương trình.
- Debug chương trình để tìm lỗi nếu có.
- Tìm hiểu cấu trúc ứng dụng & sự tương tác cụ thể giữa các đối tượng.


III. Chuẩn đầu ra :

- Thành thạo việc xây dựng 1 ứng dụng theo qui trình thiết kế trực quan.
- Thành thạo việc dùng lại linh kiện phần mềm có sẵn, thấy rõ cấu trúc phần mềm và sự tương tác giữa các đối tượng trong phần mềm.
- Thành thạo việc viết code thay đổi kích thước và vị trí các đối tượng giao diện khi cửa sổ chứa chúng bị thay đổi.

IV. Phân tích :

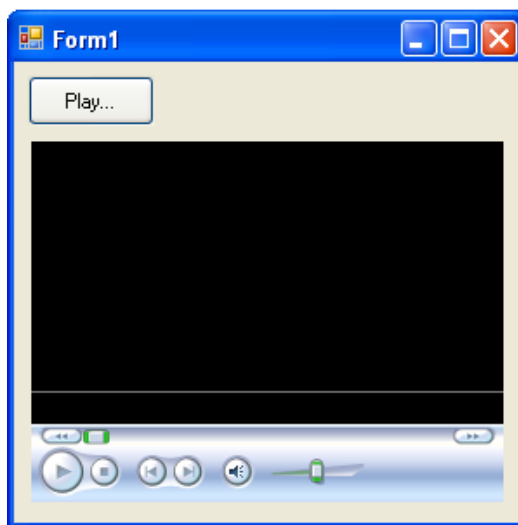
- Sau khi phân tích chức năng của chương trình, ta thấy chương trình là 1 form giao diện trực quan, nó chứa 1 button Play để người dùng ra lệnh chơi file, nó cũng cần 1 đối tượng cho phép người dùng duyệt trực quan cây thư mục để chọn file (giả sử ta biết đó là đối tượng OpenFileDialog). Cuối cùng chương trình cần 1 đối tượng có khả năng play file multimedia (giả sử ta biết đó là đối tượng Windows Media Player).
- Theo kết quả phân tích trên, cấu trúc chương trình cần viết khá đơn giản, nó chỉ chứa các đối tượng đã có sẵn, ta không cần phải bận tâm đặc tả chi tiết các đối tượng cần dùng mà chỉ cần dùng lại chúng, lắp ghép chúng lại để tạo thành chương trình. Đây là trường hợp may mắn nhất, nhưng trong thực tế lập trình hướng đối tượng, ta sẽ có được may mắn này thường xuyên.


V. Qui trình :

1. Chạy VS .Net, chọn menu File.New.Project để hiển thị cửa sổ New Project.
2. Mở rộng mục Visual C# trong TreeView "Project Types", chọn mục Window, chọn icon "Windows Application" trong listbox "Templates" bên phải, thiết lập thư mục chứa Project trong listbox "Location", nhập tên Project vào textbox "Name:" (thí dụ MicroWMP), click button OK để tạo Project theo các thông số đã khai báo.
3. Form đầu tiên của ứng dụng đã hiển thị trong cửa sổ thiết kế, việc thiết kế form là quá trình lập 4 thao tác tạo mới/xóa/hiệu chỉnh thuộc tính/tạo hàm xử lý sự kiện cho từng đối tượng cần dùng trong form.
4. Nếu cửa sổ ToolBox chưa hiển thị chi tiết, chọn menu View.Toolbox để hiển thị nó (thường nằm ở bên trái màn hình). Click chuột vào button  (Auto Hide) nằm ở góc trên phải của cửa sổ ToolBox để chuyển nó về chế độ hiển thị thường trực.
5. Duyệt tìm phần tử Button (trong nhóm Common Controls), chọn nó, dời chuột về góc trên trái của form và vẽ nó với kích thước mong muốn. Xem cửa sổ thuộc tính của Button vừa vẽ (thường ở góc dưới phải màn hình), duyệt tìm và hiệu chỉnh thuộc tính Text = "Play...", duyệt tìm và thay đổi thuộc tính (Name) = btnPlay.

6. Dời chuột vào trong cửa sổ Toolbox, duyệt tìm nhóm Components, ấn phải chuột vào mục Components (hay vào bất kỳ phần tử nào trong nhóm này) để hiển thị menu lệnh, chọn option "Choose Items". Khi cửa sổ "Choose Toolbox Items" hiển thị, click chuột vào button COM Components để hiển thị danh sách các linh kiện phần mềm theo chuẩn COM đang được Windows quản lý, hãy duyệt tìm và đánh dấu chọn vào phần tử "Windows Media Player", rồi click button OK để "add" các control này vào nhóm Components của cửa sổ Toolbox của Project ứng dụng. Bây giờ việc dùng control "Windows Media Player" giống y như các điều khiển có sẵn khác.
7. Duyệt tìm phần tử "Windows Media Player" (trong nhóm Components), chọn nó, dời chuột ngay dưới Button vừa vẽ, vẽ nó chiếm hết phần còn lại của form. Xem cửa sổ thuộc tính của control vừa vẽ (thường ở góc dưới phải màn hình), duyệt tìm và hiệu chỉnh thuộc tính thuộc tính (Name) = wmpPlay.

Sau khi thiết kế xong, Form có dạng sau :




8. Dời chuột về button btnPlay, ấn kép chuột vào nó để tạo hàm xử lý sự kiện Click chuột cho button, cửa sổ mã nguồn sẽ hiển thị để ta bắt đầu viết code cho hàm. Cách tổng quát để tạo hàm xử lý sự kiện là chọn đối tượng btnPlay, cửa sổ thuộc tính của nó sẽ hiển thị, click icon  để hiển thị danh sách các sự kiện của đối tượng, duyệt tìm sự kiện quan tâm (Click), ấn kép chuột vào comboBox bên phải sự kiện Click để máy tạo tự động hàm xử lý cho sự kiện này. Cửa sổ mã nguồn sẽ hiển thị khung sườn của hàm vừa được tạo với thân rỗng, viết thân cho hàm này như sau :


```
private void btnPlay_Click(object sender, EventArgs e) {
    //tạo form duyệt chọn file cần chơi
    OpenFileDialog dlg = new OpenFileDialog();
    //hiển thị form duyệt chọn file cần chơi
    DialogResult ret = dlg.ShowDialog();
    //kiểm tra quyết định của người dùng, nếu người dùng chọn OK thì chơi
    if (ret == DialogResult.OK)
        wmpPlay.URL = dlg.FileName;
}
```
9. Chọn menu Debug.Start Debugging để dịch và chạy thử ứng dụng. Khi Form chương trình hiển thị, hãy click chuột vào button Play, cửa sổ duyệt chọn file sẽ hiển thị, hãy duyệt và chọn 1 file multimedia nào đó (ảnh tĩnh, film, âm thanh, file playlist,...), click button Open để chơi file này. Hãy thử chọn nhiều file khác nhau xem chương trình chạy ổn định không.
10. Dừng chương trình, dời chuột về cửa sổ "Solution Explorer", duyệt tìm phần tử Form1, ấn phải chuột trên nó để hiển thị menu lệnh, chọn lệnh "View Code" để hiển thị lại mã nguồn

của Form chương trình, khảo sát thân của hàm btnPlay và tìm hiểu sự tương tác giữa đối tượng `Form1` với các đối tượng `OpenFileDialog` và `wmpPlay`.

11. Chạy lại ứng dụng, quan sát góc trên phải của Form chương trình, ta thấy 3 button chức năng thông thường là 1. thu nhỏ form về dạng icon, 2. phóng to/thu nhỏ kích thước form, 3. đóng form :



12. Thử click chuột vào button phóng to/thu nhỏ, form sẽ được phóng to để chiếm hết màn hình nhưng các đối tượng giao diện được thiết kế bên trong nó (button, WMP) không hề thay đổi kích thước theo. Làm sao thay đổi kích thước các đối tượng trong form theo kích thước của form ?
13. Nếu muốn phóng to/thu nhỏ các đối tượng bên trong theo kích thước cửa sổ, ta cần viết code thực hiện việc này 1 cách tường minh, đoạn code thay đổi kích thước và vị trí các đối tượng bên trong được để trong hàm xử lý sự kiện `SizeChanged` của đối tượng `Form`.
14. Dừng chương trình, dời chuột về cửa sổ “Solution Explorer”, duyệt tìm phần tử `Form1`, ấn kép chuột trên nó để hiển thị cửa sổ thiết kế `Form`, chọn `Form` để hiển thị cửa sổ thuộc tính của nó, click icon  để hiển thị danh sách các sự kiện của `Form`, duyệt tìm sự kiện quan tâm (`SizeChanged`), ấn kép chuột vào comboBox bên phải sự kiện `SizeChanged` để máy tạo tự động hàm xử lý cho sự kiện này. Cửa sổ mã nguồn sẽ hiển thị khung sườn của hàm vừa được tạo với thân rỗng, viết thân cho hàm này như sau :

```
private void Form1_SizeChanged(object sender, EventArgs e) {
    //xác định kích thước hiện hành của Form
    int cx = this.Size.Width;
    int cy = this.Size.Height;
    //thay đổi vị trí của đối tượng WMP về vị trí mong muốn
    wmpPlay.Location = new Point (10, 40);
    //thay đổi kích thước của đối tượng WMP theo kích thước Form
    wmpPlay.Size = new Size (cx - 30, cy - 80);
}
```

15. Chạy lại ứng dụng, thử phóng to/thu nhỏ, thay đổi kích thước của cửa sổ, ta thấy đối tượng `WMP` sẽ thay đổi kích thước theo. Riêng button thì không thay đổi vì ta không có nhu cầu và không viết code làm việc này.