

BÀI 1: Giới Thiệu Lập Trình Windows Forms Sử Dụng C#



Giúp học viên nắm rõ ý nghĩa cũng như lợi ích của việc sử dụng Windows Form trong môi trường C#.

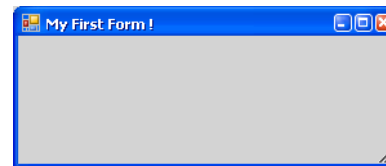
1. Tạo Form đơn giản bằng lệnh

- Mục tiêu: Làm quen với việc tạo một form đơn giản thông qua khai báo và cài đặt các lệnh xử lý.
- Công việc thực hiện:
 - Sử dụng Visual Studio, tạo một ứng dụng dạng Console với tên là My_First_Form
 - Tạo tham chiếu (references) đến 2 thư viện: System.Drawing và System.Windows.Forms
 - Xóa class Program ra khỏi ứng dụng.
 - Tạo một class mới có tên FormApp kế thừa từ class Form với các khai báo như sau:

```
using System.Windows.Forms;
using System.Drawing;

class FormApp: Form
{
    public static void Main(string[] args)
    {
        FormApp myform = new FormApp();
        myform.BackColor = Color.LightGray;
        myform.Text = "My First Form !";
        myform.SizeGripStyle = SizeGripStyle.Show;
        Application.Run(myform);
    }
}
```

- Thực thi ứng dụng và ghi nhận kết quả:



(hình 1.1)

2. Tạo form đơn giản bằng lệnh (cách 2)

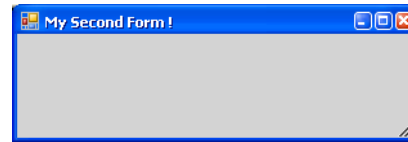
- Mục tiêu: Làm quen với việc tạo một form đơn giản thông qua khai báo và cài đặt các lệnh xử lý.
- Công việc thực hiện:
 - Sử dụng Visual Studio, tạo một ứng dụng dạng Console với tên là My_First_Form_2
 - Tạo tham chiếu (references) đến 2 thư viện: System.Drawing và System.Windows.Forms
 - Tạo một class mới có tên FormApp kế thừa từ class Form với các khai báo như sau:

```
using System.Windows.Forms;
using System.Drawing;
class FormApp: Form
{
    public FormApp()
    {
        this.BackColor = Color.LightGray;
        this.Text = "My Second Form !";
        this.SizeGripStyle = SizeGripStyle.Show;
    }
}
```

4. Trong phương thức Main của class Program, khai báo lệnh gọi Form như sau:

```
static void Main(string[] args)
{
    Application.Run(new FormApp());
}
```

5. Thực thi ứng dụng và ghi nhận kết quả:



(hình 1.2)

3. Tạo form có điều khiển bằng lệnh

- Mục tiêu: Làm quen với việc tạo một form có điều khiển thông qua khai báo và cài đặt các lệnh xử lý.
- Công việc thực hiện:
 1. Sử dụng Visual Studio, tạo một ứng dụng dạng Console với tên là Them_control_vao_form (có thể sử dụng lại ứng dụng trong bài 1.2)
 2. Tạo tham chiếu (references) đến 2 thư viện: System.Drawing và System.Windows.Forms
 3. Tạo một class mới có tên FormApp kế thừa từ class Form với các khai báo như sau:

```
using System.Windows.Forms;
using System.Drawing;
class FormApp: Form
{
    public FormApp()
    {
        this.BackColor = Color.LightGray;
        this.Text = "My Third Form !";
        this.SizeGripStyle = SizeGripStyle.Show;
        this.StartPosition = FormStartPosition.CenterScreen;

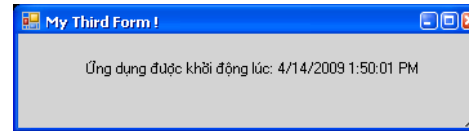
        Label myLabel = new Label();

        myLabel.Text = "Ứng dụng được khởi động lúc: " +
            DateTime.Now.ToString();
        myLabel.AutoSize = true;
        myLabel.Left = 50;
        myLabel.Top = 20;

        this.Width = myLabel.PreferredWidth + 100;
        this.Height = 100;
    }
}
```

```
        this.Controls.Add(myLabel);  
    }  
}
```

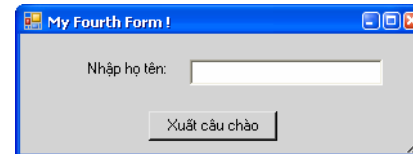
- Trong phương thức Main của class Program, khai báo lệnh gọi Form.
- Thực thi ứng dụng và ghi nhận kết quả



(hình 1.3)

4. Tạo form có điều khiển và các sự kiện

- Mục tiêu: Làm quen với việc tạo một form có điều khiển cũng như cài đặt các sự kiện cho các điều khiển thông qua khai báo và cài đặt các lệnh xử lý.
- Công việc thực hiện:
 - Sử dụng Visual Studio, tạo một ứng dụng dạng Console với tên là Form_control_event (có thể sử dụng lại ứng dụng trong bài 1.3)
 - Tạo tham chiếu (references) đến 2 thư viện: System.Drawing và System.Windows.Forms
 - Khai báo các lệnh cần thiết để thiết kế Form có dạng như sau:



(hình 1.4a)

Khai báo TextBox và nút lệnh Xuất câu chào:

```
TextBox myText = new TextBox();  
myText.Name = "txtHoTen";  
myText.Width = 150;  
myText.Top = 20;  
myText.Left = myLabel.PreferredWidth + 60;  
  
Button myButton = new Button();  
myButton.Name = "btnXuat";  
myButton.Text = "Xuất câu chào";  
myButton.Width = 100;  
myButton.Top = 60;  
myButton.Left = (int)this.Width/2 - 50;
```

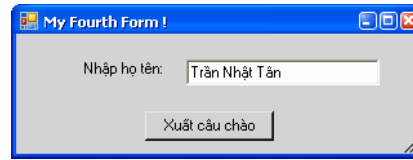
- Khai báo phương thức xử lý cho sự kiện Click của nút lệnh xuất thông báo như sau:

```
protected void btnXuat_Click(object sender, EventArgs e)  
{  
    MessageBox.Show("Xin chào " +  
        ((TextBox)this.Controls["txtHoTen"]).Text);  
}
```

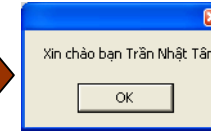
- Gắn phương thức btnXuat_Click cho sự kiện Click của nút lệnh:

```
myButton.Click += new EventHandler(this.btnXuat_Click);
```

6. Trong phương thức Main của class Program, khai báo lệnh gọi Form.
7. Thực thi ứng dụng và ghi nhận kết quả



(hình 1.4b)



(hình 14c)