

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

KHOA CÔNG NGHỆ THÔNG TIN

BỘ MÔN CÔNG NGHỆ PHẦN MỀM

-----oOo-----



BÀI TẬP THỰC HÀNH

Tên môn học: **Công nghệ Dot Net**

Số tin chỉ/ĐVHT: 3TC

Hệ đào tạo: Đại học chính qui

Ngành: KTPM

Khoa: Công nghệ thông tin

Họ tên giảng viên: **Nguyễn Thu Phương**

Đơn vị công tác: Bộ môn CNPM – Khoa CNTT

Năm học: 2014-2015

Thái Nguyên, tháng 8 năm 2014

Mục lục

BÀI TẬP THỰC HÀNH 1: NGÔN NGỮ C#.....	3
BÀI TẬP THỰC HÀNH 2: NGÔN NGỮ C#.....	9
BÀI TẬP THỰC HÀNH 3: LỚP VÀ ĐỐI TƯỢNG	10
BÀI TẬP THỰC HÀNH 4: LỚP VÀ ĐỐI TƯỢNG (tiếp)	13
BÀI TẬP THỰC HÀNH 5: LẬP TRÌNH C# WIN FORM	17
BÀI TẬP THỰC HÀNH 6: LẬP TRÌNH C# WIN FORM (tiếp)	29
BÀI TẬP THỰC HÀNH 7: TRUY CẬP DỮ LIỆU VỚI ADO.NET	37
BÀI TẬP THỰC HÀNH 8: TRUY CẬP DỮ LIỆU VỚI ADO.NET (TIẾP)	45
BÀI TẬP THỰC HÀNH 9: ỨNG DỤNG WEB VỚI WEB FORMS.....	59
BÀI TẬP THỰC HÀNH 10: ỨNG DỤNG WEB VỚI WEB FORMS (tiếp)	71

BÀI TẬP THỰC HÀNH 1: NGÔN NGỮ C#

Mục tiêu kiến thức

- Giúp sinh viên làm quen với ngôn ngữ C#: qua việc viết các ứng dụng console đơn giản, các cấu trúc điều khiển, các câu lệnh...
- Làm quen với môi trường phát triển tích hợp VS .NET 2010: các công cụ hỗ trợ soạn thảo mã nguồn, các công cụ biên dịch, debug...

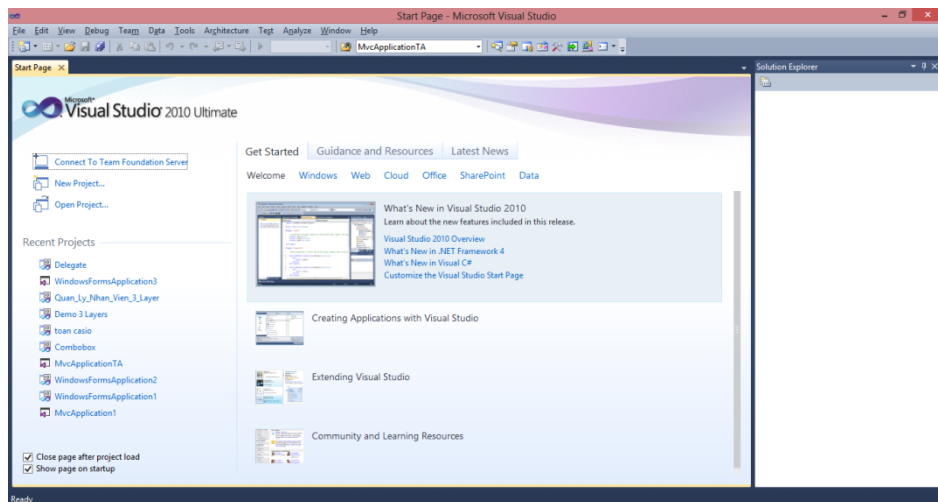
Mục tiêu kỹ năng:

- Sinh viên bắt đầu làm quen với môi trường lập trình .NET

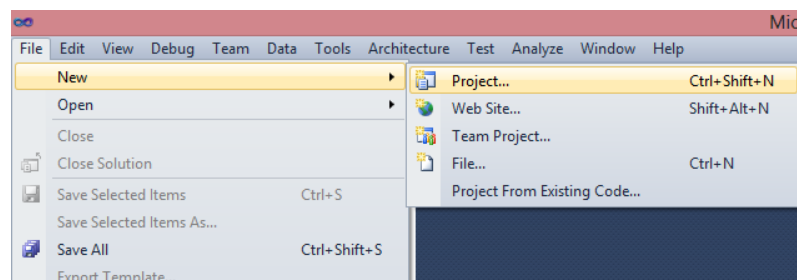
Nội dung và hướng dẫn thực hành

Bài 1: Tạo ứng dụng HelloWorld

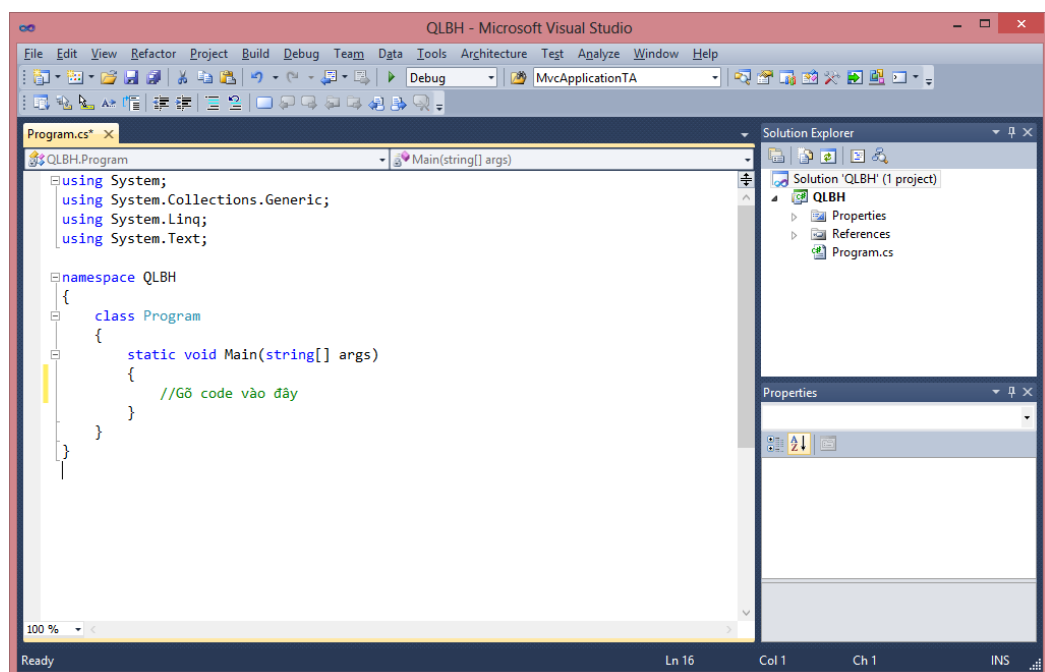
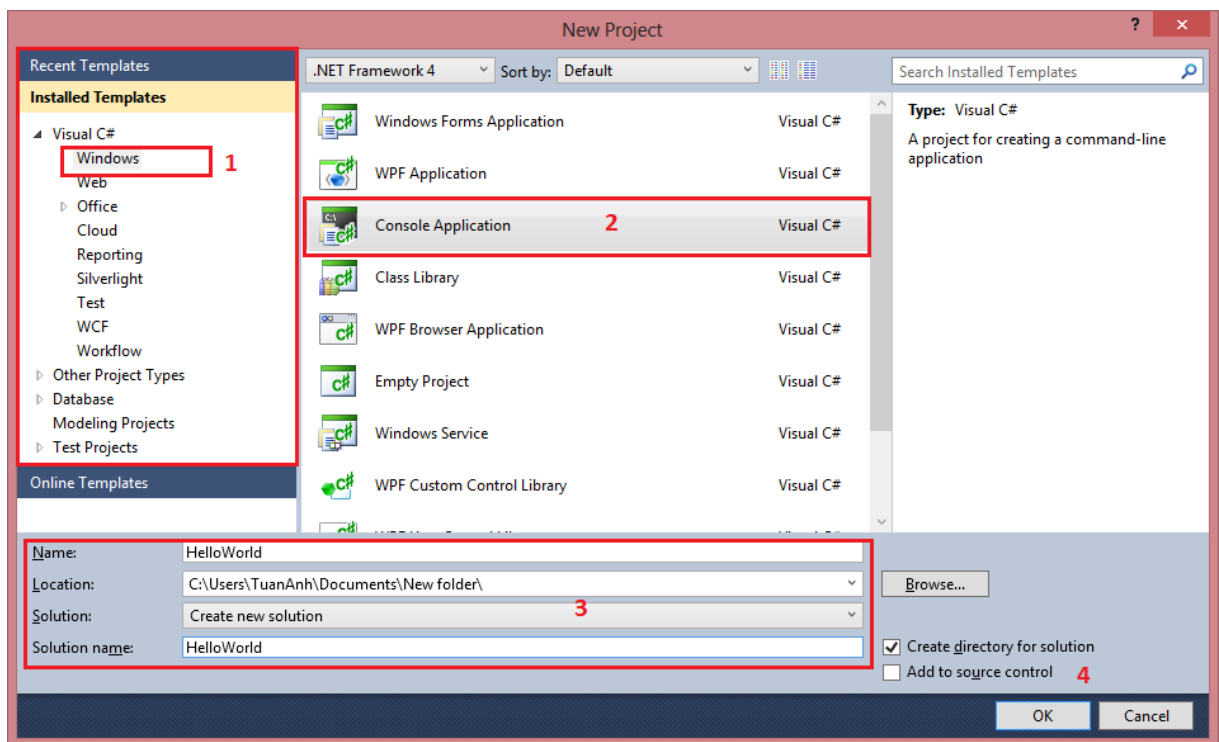
1. Để sử dụng VS. NET thực hiện như sau: **Start | Programs | Microsoft Visual Studio .NET | Microsoft Visual Studio .NET**. Start Page xuất hiện như hình dưới đây.



2. Từ menu **File | New | Project**. Cửa sổ **New project** xuất hiện như hình dưới đây.



Trong hộp thoại **New Project**, ở vùng **Recent Template**, click **Visual C#**. Ở vùng **Templates**, click **Console Application**, đặt tên của project là “HelloWorld” sau đó click **OK** để tạo một project mới.



Mã nguồn chương trình:

```
using System;
namespace HelloWorld
{
    class Program
```

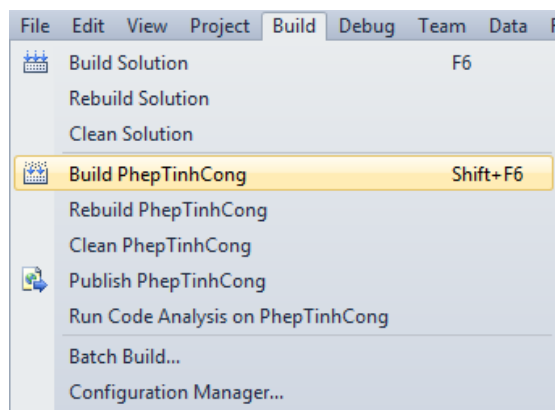
```

{
    static void Main(string[] args)
    {
        System.Console.WriteLine("Hello Character World");
        System.Console.ReadLine();
    }
}

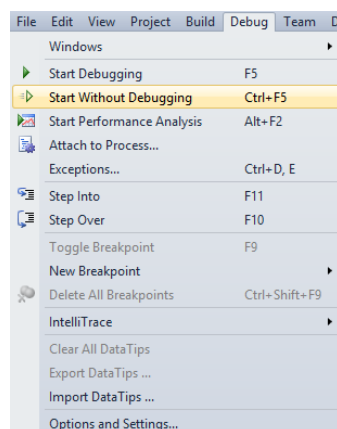
```

- Dịch và chạy chương trình

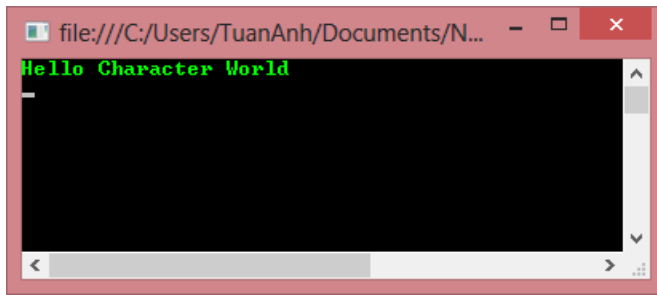
- Build chương trình bằng cách Build -> Build Solution. Hoặc sử dụng phím tắt **Ctrl+Shift+B**.



- Chạy chương trình sử dụng Debug->Start Without Debugging. Hoặc có thể sử dụng phím tắt **Ctrl+F5**.



- Kết quả hiển thị ra màn hình như sau:



Bài 2: Viết chương trình nhập vào tên của mình và xuất ra màn hình “Xin chào + Tên”.

Chú ý:

- Khai báo biến chuỗi:
`string ten;`
- Nhập chuỗi từ bàn phím:
`ten = Console.ReadLine();`
- Xuất ra màn hình:
`Console.WriteLine(“Chuỗi” + biến);`

Mã nguồn chương trình:

```
using System;
namespace HelloWorld1
{
    class Program
    {
        static void Main(string[] args)
        {
            string ten;
            ten = Console.ReadLine();
            Console.WriteLine("Xin chào " + ten);
        }
    }
}
```

Bài 3: Viết chương trình nhập 2 số nguyên, xuất tổng, hiệu, tích, thương.

Mã nguồn chương trình:

```
using System;
namespace Demo
```

```

{
    class Program
    {
        static void Main(string[] args)
        {
            int a, b;          //Toán hạng
            char ch; //Phép toán
            Console.Write("Nhập A: ");
            a = Convert.ToInt16(Console.ReadLine());
            Console.Write("Nhập B: ");
            b = Convert.ToInt16(Console.ReadLine());
            Console.Write("Nhập phép toán: ");
            ch = Console.ReadKey().KeyChar;
            Console.WriteLine("\n" + TinhToan(a, b, ch));
            Console.ReadLine();

        }
        static string TinhToan(int a, int b, char ch)
        {
            switch (ch)
            {
                case '+': return "a + b = " + (a + b);
                case '-': return "a - b = " + (a - b);
                case '*': return "a * b = " + (a * b);
                case '/':
                    if (b != 0) return "a / b = " + (a / b);
                    else return "Nhập b khác không";
                default: return "Nhập phép toán khác";
            }
        }
    }
}

```

Các bài tập tương tự:

Bài 4: Viết chương trình giải phương trình bậc nhất có dạng $ax + b = 0$, với các hệ số a, b được nhập từ bàn phím.

Bài 5: Viết chương trình nhập chiều dài, chiều rộng hình chữ nhật, xuất chu vi, diện tích của hình chữ nhật đó.

BÀI TẬP THỰC HÀNH 2: NGÔN NGỮ C#

Mục tiêu kiến thức:

- Giúp sinh viên làm quen với ngôn ngữ C#: qua việc viết các ứng dụng console đơn giản, các cấu trúc điều khiển, các câu lệnh...
- Làm quen với môi trường phát triển tích hợp VS .NET 2010: các công cụ hỗ trợ soạn thảo mã nguồn, các công cụ biên dịch, debug...

Mục tiêu kỹ năng:

- Có kỹ năng tốt trong thực hành .NET nói chung và C# nói riêng

Bài 1: Viết chương trình nhập bán kính hình tròn, xuất chu vi, diện tích của hình tròn đó.

Hướng dẫn:

- Đặt const double pi = 3.14
- Tính chu vi và diện tích hình tròn

Bài 2: Viết chương trình nhập số nguyên N, kiểm tra và xuất kết quả N là số chẵn/lẻ.

Bài 3: Viết chương trình nhập số tự nhiên N, kiểm tra và xuất kết quả N là số nguyên tố hay không.

Bài 4: Viết chương trình nhập số tự nhiên N, xuất kết quả:

- Các số tự nhiên $\leq N$ và tổng của chúng
- Các số tự nhiên chẵn $\leq N$ và tổng của chúng
- Các số tự nhiên lẻ $\leq N$ và tổng của chúng
- Các số tự nhiên là số nguyên tố $\leq N$ và tổng của chúng

Bài 5: Viết chương trình nhập thông tin của N sinh viên gồm: Mã sinh viên, Họ tên, điểm toán, điểm lý, điểm hóa ($N > 0$ nhập từ bàn phím). Hãy in thông tin của các sinh viên vừa nhập.

Hướng dẫn: Sử dụng vòng lặp để nhập N sinh viên.

BÀI TẬP THỰC HÀNH 3: LỚP VÀ ĐỐI TƯỢNG

Mục tiêu kiến thức

- Giúp sinh viên nâng cao kỹ năng xây dựng lớp đối tượng trong C#
- Xây dựng giao diện, kế thừa và thực thi giao diện.

Mục tiêu kỹ năng:

- Có kỹ năng tốt trong thực hành .NET nói chung và C# nói riêng

Bài 1. Viết lớp giải phương trình bậc hai.

Lớp này có các thuộc tính a, b, c và nghiệm x1, x2. Hãy xây dựng theo hướng đối tượng lớp trên. Lớp cho phép bên ngoài xem được các nghiệm của phương trình và cho phép thiết lập hay xem các giá trị a, b, c.

```
namespace PTB2_Lop
{
    class PTB2
    {
        // Khai bao cac thuoc tinh
        private double a, b;

        // Khai bao nghiem
        public double x1, x2;

        public void TinhNghiem(double a, double b, double c)
        {
            double delta; //khai bao bien delta
            delta = b * b - 4 * a * c;
            // Kiem tra delta de tinh nghiem
            if (delta < 0)
                this.x1 = this.x2 = double.NaN;
            else
            {
                if (delta == 0)
                    this.x1 = this.x2 = -b / (2 * a);
                else
                {
                    this.x1 = (-b + Math.Sqrt(delta)) / (2*a);
                    this.x2 = (-b - Math.Sqrt(delta)) / (2*a);
                }
            }
        }
    }
}

namespace PTB2_Lop
```

```

{
    class Program
    {
        static void Main(string[] args)
        {
            // Tao lap doi tuong
            PTB2 phuongtrinhbac2= new PTB2();
            phuongtrinhbac2.TinhNghiem(4,-2,5);
            Console.WriteLine("Nghiem cua PT bac 2 la: {0:0.00}" ,
phuongtrinhbac2.x2);
            Console.Read();
            //phuongtrinhbac2.HienthiKQ();
        }
    }
}

```

Bài 2. Viết một chương trình để cài đặt một hệ thống quản lý kho.

Hãy lưu trữ mã số, tên hàng, giá và số lượng đang có của mỗi món hàng trong một lớp. Nhập chi tiết của N (N nhập từ bàn phím) món hàng hiển thị tên từng món hàng và tổng giá trị của nó.

Bài 3. Viết một chương trình để lưu trữ các sinh viên gồm: mã sinh viên, họ và tên và điểm trung bình của N (N nhập từ bàn phím) sinh viên.

Hãy sắp xếp danh sách sinh viên này theo thứ tự điểm trung bình giảm dần. Hiển thị 3 sinh viên có điểm trung bình cao nhất.

Bài 4. Tạo một lớp CD gồm: CDName, CDType và CDPrice và các phương thức cần thiết khác.

Viết một ứng dụng có dạng menu như sau:

Menu

1. Add CD
2. Search CD
3. Display catalog
4. Exit

Your choice: _

Thêm CD: Cho phép thêm một đĩa CD vào danh mục các đĩa CD hiện có. Hãy cung cấp khả năng quản lý khoảng 1000 đĩa CD. Cần có sự kiểm soát nếu số lượng đĩa CD được nhập vượt quá 1000.

Search CD: Nhập vào tên của đĩa CD, nếu không tìm thấy thì báo lỗi, nếu tìm thấy thì in các thông tin liên quan đến đĩa CD đó.

CD No.	CD Name	CD Type	CD Price
1	Ngay khong em	Ca nhac	70K(VND)
2	Thoi xa vang	Phim	320K(VND)
3	De che	Tro choi	6K(VND)

Display catalog: Hiển thị tất cả các đĩa CD hiện có trong danh mục, danh mục đĩa CD được hiển thị ở dạng bảng, có chứa cột tiêu đề.

Bài 5.

1. Xây dựng các lớp đối tượng hình học như: điểm, đoạn thẳng, đường tròn, hình chữ nhật, hình vuông, tam giác, hình bình hành, hình thoi. Mỗi lớp có các thuộc tính riêng để xác định được hình vẽ biểu diễn của nó như đoạn thẳng thì có điểm đầu, điểm cuối....

2. Mỗi lớp thực thi một phương thức Draw() ghi đè (overriding) phương thức Draw() của lớp cơ sở gốc của các hình mà nó dẫn xuất. Hãy xây dựng lớp cơ sở của các lớp trên và thực thi đa hình với phương thức Draw(), sau đó tạo lớp Tester cùng với hàm Main() để thử nghiệm.

BÀI TẬP THỰC HÀNH 4: LỚP VÀ ĐỐI TƯỢNG (tiếp)

Mục tiêu kiến thức

- Giúp sinh viên nâng cao kỹ năng xây dựng lớp đối tượng trong C#
- Xây dựng giao diện, kế thừa và thực thi giao diện.
- Nạp chồng phương thức
- Thuộc tính và thủ tục thuộc tính
- Thành phần tĩnh và cách sử dụng

Mục tiêu kỹ năng:

- Có kỹ năng tốt trong thực hành .NET nói chung và C# nói riêng

Bài 1: Cho thiết kế lớp Employee (nhân viên) như sau:

Các thành phần dữ liệu:

- id: Định danh, kiểu int. Định danh này được sinh tự động và tăng dần bắt đầu từ 1.
- name: Họ tên nhân viên, kiểu String.
- yearOfBirth: Năm sinh nhân viên, kiểu int.
- salaryLevel: Bậc lương, kiểu double.
- basicSalary: Lương cơ bản, kiểu double. (Chú ý lương cơ bản là thuộc tính được sử dụng chung cho mọi đối tượng của lớp Employee).

Các phương thức:

- GetId(): trả lại định danh của nhân viên.
- GetName(): trả lại tên của nhân viên.
- GetYearOfBirth(): trả lại năm sinh của nhân viên.
- GetIncome(): trả lại thu nhập của nhân viên. Thu nhập được tính bằng bậc lương nhân lương cơ bản (salaryLevel * basicSalary).
- Input(): nhập thông tin nhân viên.
- Display(): hiển thị thông tin về nhân viên. Bao gồm các thông tin: định danh, tên, năm sinh, lương cơ bản, thu nhập.

- SetSalaryLevel(): thiết lập bậc lương cho nhân viên.
- SetBasicSalary(): thiết lập lương cơ bản.

Hãy viết chương trình cài đặt lớp Employee và lớp sử dụng Employee.

Bài 2: Xây dựng lớp Stack để mô phỏng một stack bao gồm:

- Phương thức khởi tạo không tham số với việc khởi tạo stack ngầm định có 20 phần tử và stack rỗng
- Phương thức khởi tạo một tham số với việc khởi tạo số phần tử của stack được truyền vào thông qua đối số của phương thức và stack rỗng
- Phương thức IsEmpty kiểm tra xem stack có rỗng không
- Phương thức IsFull kiểm tra xem stack có đầy không
- Phương thức Push và Pop để thêm vào, lấy ra một phần tử

Hãy viết chương trình cài đặt lớp Stack và lớp sử dụng Stack.

Bài 3: Xây dựng lớp Phanso và thao tác lớp với các phương thức bao gồm:

- Phương thức khởi tạo phân số có tử bằng 0 và mẫu bằng 1
- Phương thức khởi tạo (int ts, int ms) để khởi tạo phân số có tử số bằng ts và mẫu số bằng ms
- Phương thức nhập thông tin cho phân số
- Phương thức cộng hai phân số
- Phương thức cộng phân số với một số
- Phương thức trừ hai phân số
- Phương thức trừ 1 phân số cho một số
- Phương thức in kết quả ra màn hình

Hướng dẫn:

- Thuộc tính của lớp: int ts,ms: tử số và mẫu số của phân số

Các phương thức của lớp:

- Khởi tạo không tham số: public phanso() và gán giá trị của tử =0, mẫu số=1
- Khởi tạo hai tham số: public phanso(int tu,int mau)
- Cộng phân số với phân số
- Cộng phân số với một số
- Trừ phân số cho phân số

- Trừ phân số cho một số
- Phương thức hiện kết quả lên màn hình

Vì trong Bài tập này có nhiều phương thức chồng nhau (sự chồng phương thức) nên các bạn chú ý cách gọi các phương thức.

Bài 4: Xây dựng chương trình quản lý lương cho công ty ABC. Thông tin để tính lương cho mỗi Nhân viên trong công ty bao gồm: Họ tên, quê quán, hệ số lương, lương cơ bản. Hãy nhập vào một danh sách các Nhân viên của công ty sau đó thực hiện các yêu cầu sau:

- Tính và hiển thị lương của các Nhân viên có trong danh sách
- Liệt kê những nhân viên có hệ số lương cao nhất
- Sắp xếp danh sách theo thứ tự tăng dần của trường hệ số lương

Hướng dẫn:

- Xây dựng lớp có tên NhanVien với các thành phần
 - Dữ liệu: Họ tên, quê quán, hệ số lương, lương cơ bản(là thành phần dữ liệu tĩnh)
 - Phương thức: Các thuộc tính để truy xuất tới các thành phần dữ liệu họ tên, hệ số lương, phương thức nhập, hiển thị, tính lương,...
- Xây dựng lớp QuanLy nhân viên bao gồm
 - Dữ liệu: ds là một mảng các nhân viên

Phương thức: Phương thức nhập, hiển thị, sắp xếp,...

Bài 5: Xây dựng lớp có tên là TienDien với các thông tin bao gồm:

+ Dữ liệu:

- Họ tên chủ hộ
- Địa chỉ
- Số công tơ tháng trước
- Số công tơ tháng này

+ Phương thức

- Phương thức thiết lập không tham số và 4 tham số

- Phương thức nhập dữ liệu
- Phương thức hiển thị dữ liệu
- Thuộc tính tính số công tơ điện đã dùng(=Số công tơ tháng này- Số công tơ tháng trước)
- Phương thức tính tiền điện được tính theo công thức: Số điện đã dùng*1240
Sau đó xây dựng lớp TienDienMoi bằng việc kế thừa lớp TienDien để tính tiền điện theo một quy định mới. Việc tính tiền điện lúc này căn cứ vào định mức quy định. Nếu trong định mức là 1240, ngoài định mức là 1600

Hướng dẫn:

- Xây dựng lớp TienDien theo như đã mô tả

BÀI TẬP THỰC HÀNH 5: LẬP TRÌNH C# WIN FORM

Mục tiêu kiến thức:

- + Tạo ứng dụng trên Windows Form.
- + Sử dụng các thuộc tính, phương thức và sự kiện của các control: Label, Textbox, Button, ErrorProvider, RadioButton, CheckBox, GroupBox, Panel, PictureBox, ListBox, CheckedListBox, ComboBox.

Mục tiêu kỹ năng:

- Có kỹ năng tốt trong thực hành xây dựng ứng dụng trên Winform với các điều khiển cơ bản.

Bài 1: Thiết kế Form sau:

The image shows a Windows Form titled "My Name Project". Inside the form, there are two text boxes: "Your Name" and "Year of birth". Below these text boxes are three buttons: "Show", "Clear", and "Exit". Arrows point from the text boxes to labels "txtYourName" and "txtYear" on the right. Arrows point from the buttons to labels "btnExit", "btnClear", and "btnExit" on the right. Below the form, there is an "ErrorProvider" control with a label "errorProvider1" and an arrow pointing to it.

Yêu cầu:

- Chương trình cho phép nhập tên, năm sinh vào Textbox YourName và Year of birth tương ứng. Nếu YourName không nhập dữ liệu, Year of birth không phải là số thì phải thông báo lỗi (dùng ErrorProvider). Người dùng nhấn nút Show sẽ hiển thị thông tin nhập vào MessageBox bao gồm: tên, tuổi (năm hiện tại – năm sinh).
- Người dùng nhấn nút Clear sẽ xóa hết thông tin đã nhập trên các Textbox, đồng thời đặt con trỏ văn bản vào Textbox YourName.

- Nút Exit xác nhận người dùng có thực sự muốn thoát khỏi chương trình không?
(Yes: thoát, No: không).

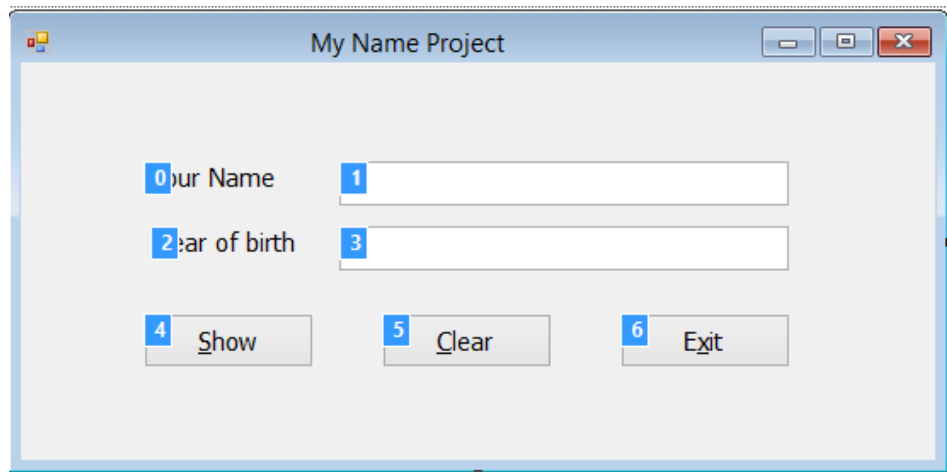
Hướng dẫn:

Danh sách các thuộc tính của các object:

Object	Properties	Events
frmMain	Name: frmMain Text: My name Project FontName: Tahoma FontSize: 11 AcceptButton: btnShow (nhận sự kiện click chuột khi nhấn Enter) CancelButton: btnExit (nhận sự kiện click chuột khi nhấn Esc)	FormClosing
txtYourName	Name: txtYourName BorderStyle: FixSingle	Leave (mất tiêu điểm)
TxtYear	Name: txtYear BorderStyle: FixSingle	TextChanged
btnShow	Name: btnShow Text: &Show	Click
btnClear	Name: btnClear Text: &Clear	Click
BtnExit	Name: btnExit Text: E&xit	Click
errorProvider	Name: errorProvider1	

Thứ tự nhận tiêu điểm trên Form: chọn menu View → Tab Order

Lần lượt thực hiện click chọn từng phần tử trên Form theo thứ tự nhận tiêu điểm:



Các sự kiện:

```
private void btnClear_Click(object sender, EventArgs e)
{
    txtYourName.Clear();
    txtYear.Clear();
    txtYourName.Focus();
}

private void btnShow_Click(object sender, EventArgs e)
{
    int age = DateTime.Now.Year -
Convert.ToInt32(txtYear.Text);
    string s = "My name is: " + txtYourName.Text + "\n" +
age.ToString();
    MessageBox.Show(s);
}

private void txtYourName_Leave(object sender, EventArgs e)
{
    Control ctr = (Control)sender;
    if (ctr.Text.Trim().Length == 0)
        this.errorProvider1.SetError(txtYourName, "You must
enter Your name");
    else
        this.errorProvider1.Clear();
}
```

```

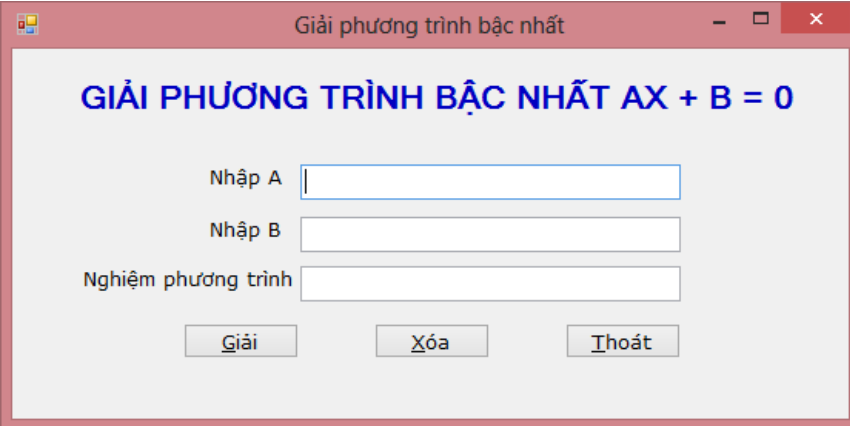
private void txtYear_TextChanged(object sender, EventArgs e)
{
    Control ctr = (Control)sender;
    if (ctr.Text.Trim().Length > 0 &&
!char.IsDigit(ctr.Text, ctr.Text.Length - 1))
        this.errorProvider1.SetError(txtYear, "This is not
invalid number");
    else
        this.errorProvider1.Clear();
}

private void frmMain_FormClosing(object sender,
FormClosingEventArgs e)
{
    DialogResult r;
    r = MessageBox.Show("Do you want to close?", "Exit",
        MessageBoxButtons.YesNo,
        MessageBoxIcon.Question,
        MessageBoxDefaultButton.Button1);
    if (r == DialogResult.No)
        e.Cancel = true;
}

private void btnExit_Click(object sender, EventArgs e)
{
    this.Close();
}

```

Bài 2: Thiết kế chương trình hiện thực bài toán giải phương trình bậc 1:



The screenshot shows a Windows application window with the title bar "Giải phương trình bậc nhất". The main content area has a light gray background. At the top, the text "GIẢI PHƯƠNG TRÌNH BẬC NHẤT AX + B = 0" is displayed in blue. Below this, there are three input fields: "Nhập A", "Nhập B", and "Nghiệm phương trình". At the bottom, there are three buttons: "Giải", "Xóa", and "Thoát".

Yêu cầu:

- Khi form hiện lên thì nút Giải và nút Xóa bị mờ (Enabled=false). Nếu dữ liệu nhập không hợp lệ thì thông báo lỗi (dùng errorProvider). Sau khi nhập dữ liệu hợp lệ và đầy đủ thì nút Giải có tác dụng (Enabled=true).

- Khi nhấn nút Tính: tính nghiệm phương trình (xét tất cả các trường hợp xảy ra: PT có 1 nghiệm, vô nghiệm, vô số nghiệm) và hiện kết quả vào Nghiệm PT. Khi đó nút Xóa có tác dụng, nút Tính bị mờ.

- Khi nhấn nút Xóa: xóa các Textbox và Label, đặt con trỏ vào Textbox A, nút Xóa bị mờ

- Khi nhấn nút Thoát: xác nhận người dùng có chắc chắn thoát khỏi ứng dụng hay không?

Bài 3: Thiết kế giao diện như hình. Khi nhấn chọn vào phép tính nào thì sẽ hiện kết quả của phép tính đó vào ô Kết quả.

Trước khi tính cần kiểm tra dữ liệu nhập phải là số.

Bài 4: Thiết kế giao diện như hình sau:

Yêu cầu:

- Khi chương trình hiện lên:
 - Radiobutton Red được chọn mặc định (đổi màu chữ ô lblLapTrinh và ô txtNhapTen)
 - Con trỏ văn bản xuất hiện ngay tại ô txtNhapTen.
- Khi gõ vào ô txtNhapTen thì Label lblLapTrinh chạy song song cùng nội dung.
- Nhấn nút "Thoát" hoặc Esc thì thoát chương trình.

- Nhấn Radiobutton Red, Green, Blue, Black thì đổi màu chữ tương ứng trong ô lblLapTrinh và ô txtNhapten.

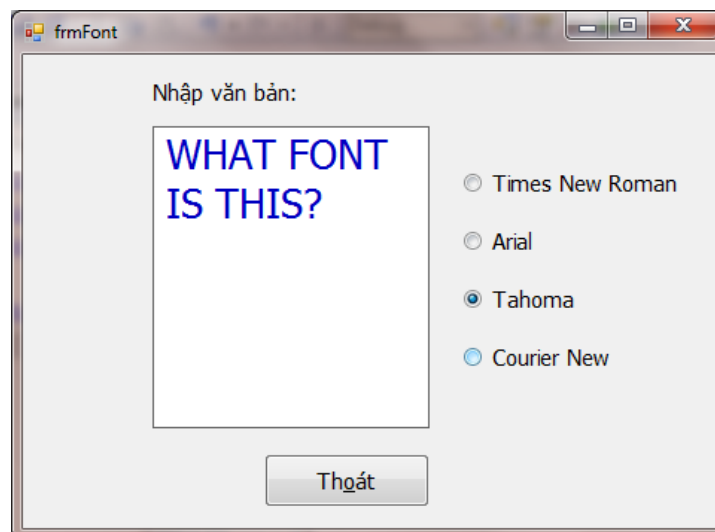
- Nhấn các checkbox chữ đậm, nghiêng, gạch chân thì đổi style chữ trong ô lblLapTrinh và ô txtNhapten tương ứng.

Hướng dẫn: đổi style chữ trong ô lblLapTrinh

Viết trong sự kiện CheckedChanged của từng checkbox:

```
private void chkdam_CheckedChanged(object sender, EventArgs e)
{
    lblLapTrinh.Font = new Font(lblLapTrinh.Font.Name,
                                lblLapTrinh.Font.Size,
                                lblLapTrinh.Font.Style ^ FontStyle.Bold);
}
```

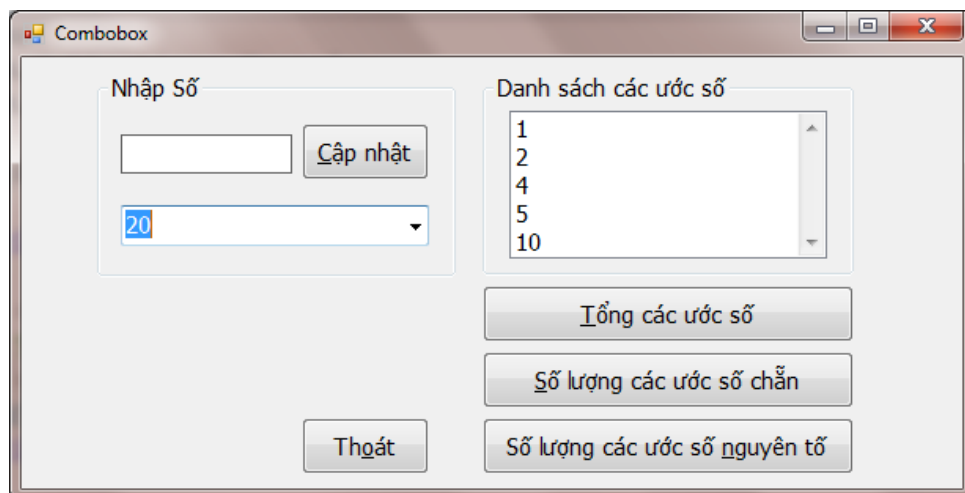
Bài 5: Viết chương trình định dạng Textbox theo các font chữ tương ứng với từng Radiobutton



Bài 6: Viết chương trình hiển thị vào PictureBox lá cờ tương ứng với nước được chọn trên Radio Button tương ứng.



Bài 7. Thiết kế giao diện như sau:



Yêu cầu:

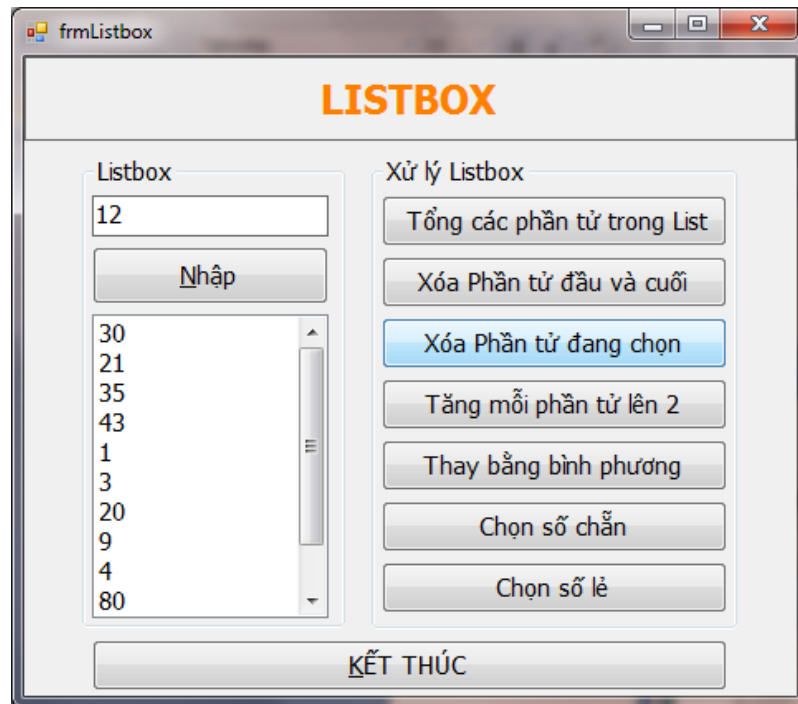
Khi Form vừa hiện lên, các Textbox, Combobox, Listbox chưa có dữ liệu, con trỏ đặt tại Textbox (thiết lập Tab Order hợp lý).

Nhấn nút “Cập nhật” hoặc Enter: thêm số vừa nhập ở Textbox vào Combobox (nhớ kiểm tra dữ liệu nhập), đồng thời xóa nội dung Textbox và đặt con trỏ lại Textbox.

Khi chọn 1 số trên Combobox thì danh sách các ước số của số này sẽ hiển thị vào Listbox bên phải tương ứng.

Khi nhấn các nút: “Tổng các ước số”, “Số lượng các ước số chẵn”, “Số lượng các ước số nguyên tố” thì sẽ hiển thị thông tin tương ứng vào MessageBox dựa vào các ước số trên Listbox.

Bài 8. Thiết kế giao diện như sau:



Yêu cầu:

Khi Form vừa hiện lên, các Textbox, Listbox để trống, con trỏ đặt tại Textbox (thiết lập Tab Order hợp lý).

Khi người sử dụng nhập một số vào Textbox rồi Enter hoặc nhấn nút "Nhập" thì số đó được thêm vào Listbox, đồng thời nội dung trong Textbox bị xóa và con trỏ được chuyển về Textbox.

Người dùng nhấn vào nút nào thì thực hiện chức năng tương ứng của nút đó. Hiện kết quả ra MessageBox (nếu có).

Thiết lập thuộc tính Anchor hợp lý cho các control.

Thiết lập MinimumSize cho form.



Bài 9. Viết chương trình nhập danh sách sinh viên theo yêu cầu sau: (xem hình bên dưới).



Quy định Form hiển thị giữa màn hình. Không cho người sử dụng thay đổi kích thước Form.

Quy định việc di chuyển tab hợp lý.

Các Listbox được phép chọn nhiều mục (kết hợp giữa phím Shift, Ctrl và chuột)

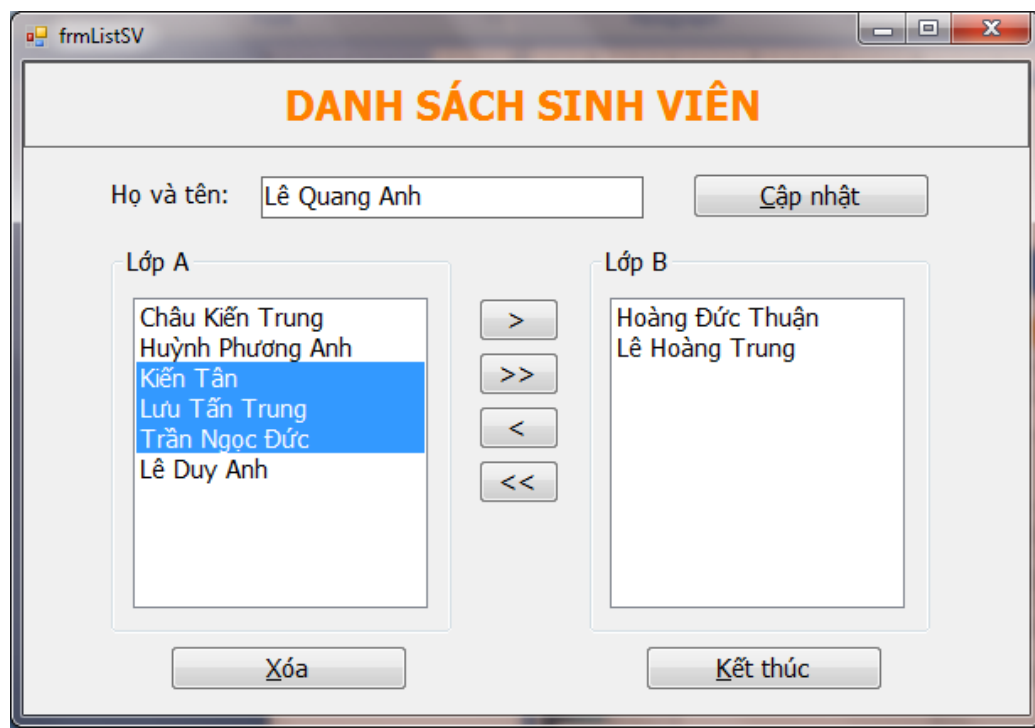
Khi người dùng nhập Họ và tên của sinh viên vào Textbox, click nút Cập Nhật (hoặc Enter) thì tên sinh viên đó sẽ được đưa vào danh sách lớp A (không chấp nhận dữ liệu rỗng).

,  chuyển các tên đang chọn từ Listbox trái sang Listbox phải và ngược lại.

,  chuyển hết toàn bộ các tên từ Listbox trái sang Listbox phải và ngược lại.

Nút Xóa: cho phép xóa các tên đang chọn trong danh sách lớp A.

Thêm vào giao diện 1 combobox Lớp, trong đó có 2 lớp: Lớp A, Lớp B, theo đó người sử dụng có thể chọn lớp để cập nhật sinh viên vào lớp mong muốn.



Hướng dẫn:

```
// copy selected strings in the source list to the destination list
```

```
for (int i = 0; i < SourceListbox.SelectedItems.Count; i++)
```

```
{
```

```
    DestinationListbox.Items.Add(SourceListbox.SelectedItems[i]);
```

```
}
```

```
// remove selected strings from the source list
for(int j = SourceListbox.SelectedItems.Count - 1; j >= 0; j --)
{
    SourceListbox.Items.Remove(SourceListbox.SelectedItems[j]);
}
}
```

Bài 10. Viết chương trình cho phép sinh viên đăng ký học các môn học trong học kỳ:

Yêu cầu:

Khi Form hiện lên, các ô nhập đều để trống (thiết lập tab hợp lý).

Nút Đăng ký: Hiện thị các thông tin mà sinh viên đã đăng ký lên MessageBox như hình:

The screenshot shows a Windows form titled 'frmDangkyMonhoc' with the main heading 'ĐĂNG KÝ MÔN HỌC'. It includes the following fields and controls:

- MSSV: Text box containing 'SV0013'.
- Họ và tên: Text box containing 'Lê Quang Anh'.
- Niên Khóa: Dropdown menu showing '2007'.
- Lớp: Dropdown menu showing 'TH02'.
- Học kỳ: Radio buttons for I, II, III, and IV, with IV selected.
- Môn học: A list box with checkboxes for 'LT Windows', 'LT Internet', 'Mạng máy tính', and 'UML'. 'UML' is currently selected.
- Buttons: 'Đăng ký', 'Hủy', and 'Thoát' at the bottom.

The screenshot shows a standard Windows MessageBox dialog box with the following text:

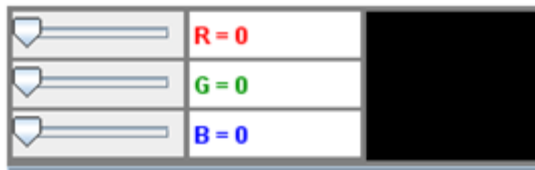
Sinh viên: Lê Quang Anh
 Lớp: TH02
 Niên khóa: 2007
 Đã đăng ký học Học kỳ 4 Các môn học sau:
 1. LT Windows
 2. LT Internet
 3. UML

An 'OK' button is located at the bottom right of the dialog box.

Nút Hủy: trả lại trạng thái ban đầu của Form.

Nút Thoát: thoát khỏi ứng dụng.

Bài 11. Thiết kế giao diện cho phép đổi màu Panel như sau:



Để đổi màu Panel, người dùng có thể kéo các TrackBar để thay đổi các giá trị màu red(R), green(G) và blue(B). Yêu cầu khi TrackBar nào kéo đến đâu thì giá trị của nó phải được hiện ra tương ứng trên các Label. Các TrackBar có giá trị trong khoảng 0 đến 255.

BÀI TẬP THỰC HÀNH 6: LẬP TRÌNH C# WIN FORM (tiếp)

Mục tiêu kiến thức:

+ Sử dụng các thuộc tính, phương thức và sự kiện của các control: TrackBar, NumericUpDown, MaskedTextBox, DateTimePicker, MonthCalendar, Timer, ProgressBar, ToolTip, MenuStrip.

+ Viết ứng dụng dạng MDI

+ ListView, ImageList, TreeView

Mục tiêu kỹ năng:

- Có kỹ năng tốt trong lập trình Winform với các điều khiển nâng cao.

Bài 1. Thiết kế giao diện như sau:

The screenshot shows a Windows Form titled "Employee Details". It contains the following fields and controls:

- Employee Name: A standard text box.
- Date of Birth: (mm/dd/yyyy): A date picker control.
- Address: A text box.
- City: A list box containing "Ho Chi Minh", "Nha Trang", and "Ha Noi".
- Country: A dropdown menu with "(Select)" as the current selection.
- Qualification: A list box containing "University", "Master", and "Ph D".
- Phone: A text box with a hyphen as a placeholder for a masked digit.
- EMail: A text box.
- Date of Joining: (mm/dd/yyyy): A date picker control.
- Buttons: "Submit" and "Exit".
- Link: "Link To VnExpress".

Arrows from the diagram point to the following controls:

- TextBox: Points to the "Employee Name" field.
- MaskedTextBox: Points to the "Phone" field.
- TextBox: Points to the "Address" field.
- MaskedTextBox: Points to the "Date of Birth" field.
- MaskedTextBox: Points to the "Date of Joining" field.
- MaskedTextBox: Points to the "E-Mail" field.

Yêu cầu:

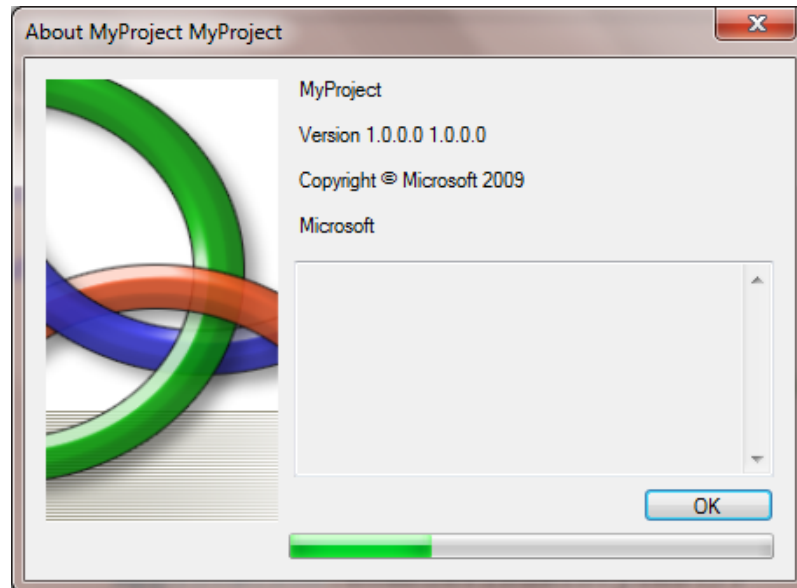
Quy định của Masktextbox Phone là 000-0000000.

ComboBox Country chỉ chứa 2 nước VietNam và Thailan (chứa 3 thành phố Pattaya, Chiang Mai và Bangkok).

Khi đang nhập 1 ô mà bỏ trống và focus đến ô khác thì sẽ có thông báo lỗi và cho focus về ô cần nhập.

Khi nhấn Submit sẽ có một MessageBox hiển thị đầy đủ thông tin vừa nhập.

Bài 2. Tạo giao diện Form Splash như hình: chứa Progressbar và một nút OK.



Khi khởi động chương trình thì Form Splash xuất hiện. Form này dừng trong thời gian là 15s.

Thanh Progressbar sẽ thể hiện tiến trình load form này, sau 15s Form Splash sẽ tự động tắt và khởi động Form trong bài 1.

Trong khi Form Splash đang hoạt động, nếu người dùng nhấn vào nút OK thì Form này ngưng hoạt động, đồng thời Form bài 1 được gọi hoạt động.

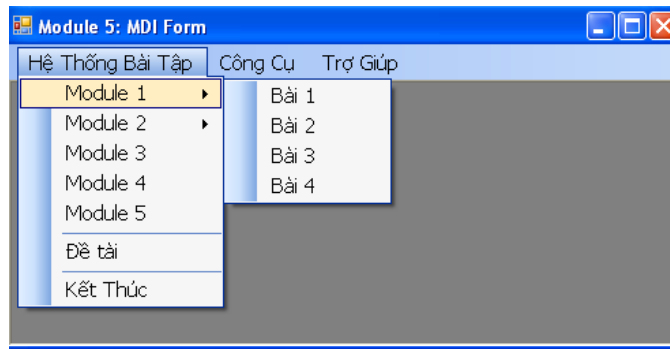
Hướng dẫn: kéo thả thanh Progressbar vào Form. Bắt sự kiện cho nút OK

Bài 3.

- + Tạo một ứng dụng liên kết các ứng dụng đã tạo ra từ các project trước.
- + Tạo form Splash
- + Tạo form About

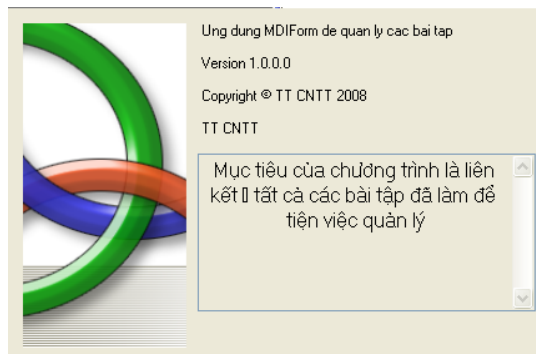
Yêu cầu:

Tổ chức Form chính như mẫu sau:



Yêu cầu liên kết các project đã có sẵn vào menu.

Thiết kế Form Splash cho chương trình (tùy ý):



Thiết kế Form About cho chương trình, tùy ý nhưng phải mang thông tin về chương trình như: tên chương trình, phiên bản, tác giả,...

Hướng dẫn:

Thiết lập một số thuộc tính của Form About:

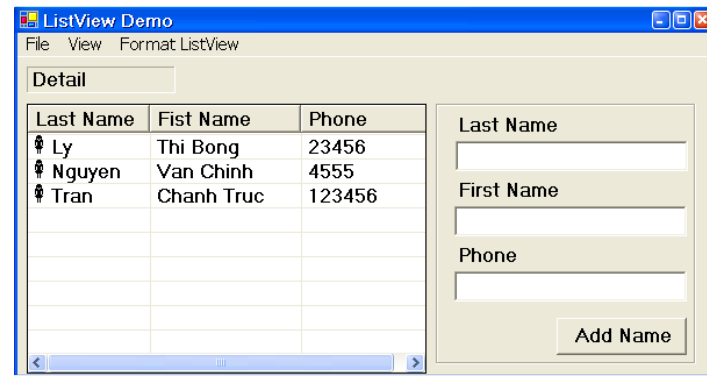
ControlBox → False

FormBorderStyle → FixedDialog

ShowInTaskbar → False

Bài 5. Thêm vào bài 1 chức năng cho phép người dùng mở dialog chọn màu để chọn màu cho Panel.

Viết chương trình nhập dữ liệu vào Listview như hình:



Yêu cầu:

Người sử dụng nhập thông tin Last name, First name, Phone và sử dụng nút Add Name để nhập vào Listview.

Các dòng trong Listview có biểu tượng (icon) hiển thị như hình.

Người sử dụng có thể thay đổi chế độ view của Listview bằng menu View.

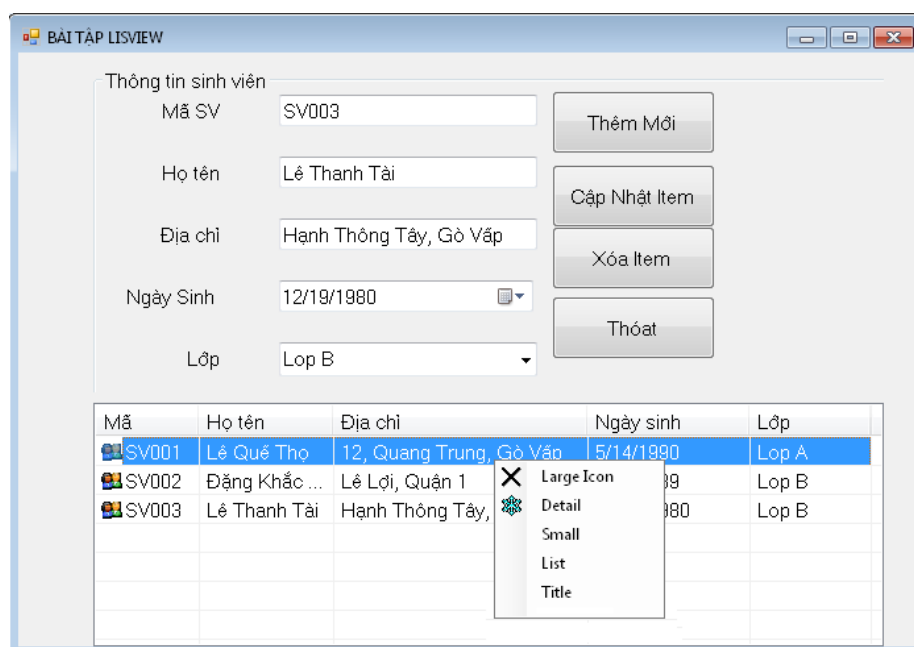
Menu FormatListView hiển thị hộp thoại chọn màu dùng để thay đổi dạng grid của Listview.

Gợi ý:

Sử dụng ListViewItem để thêm một dòng mới cho Listview.

Sử dụng ImageList để chứa thư viện icon cho Listview. Kết nối Listview với ImageList.

Bài 6. Thiết kế chương trình quản lý SV, cho phép nhập thông tin SV vào các Textbox như hình:



Yêu cầu:

Thêm vào Form hai Imagelist là *ilsNho* có kích thước mặc định 16 x 16, *ilsLon* có kích thước mặc 48 x 48 phục vụ cho ListView.

Nhấn nút Cập Nhật Item thì đưa thông tin sinh viên vào Listview theo các cột như hình.

Nhấn nút Xóa Item là xóa item đang chọn trên Listview (có thể chọn nhiều). Trước khi xóa cần xác nhận đã chọn Item nào chưa, xác nhận có chắc xóa không.

Nhấn nút Thêm Mới thì xóa thông tin sinh viên đang nhập và cho phép nhập thông tin sinh viên mới.

Nếu chọn một sinh viên nào trong Listview thì hiện lại thông tin Sinh Viên đó lên các Textbox tương ứng.

Click phải vào Listview cho phép hiện menu ngữ cảnh để chọn chức năng view.

Bài 7. Thiết kế giao diện như sau:

The screenshot shows a Windows application window titled "Tính tiền điện". The window is divided into two main sections. On the left is a form titled "BÁO CÁO TIÊU THU ĐIỆN" (Electricity Billing Report). It contains several input fields: "HỌ TÊN KH:" (Customer Name), "KHU VỰC:" (Area) with a dropdown arrow, "ĐỊNH MỨC:" (Rate), "SỐ CŨ:" (Old Meter Reading), "SỐ MỚI:" (New Meter Reading), "TIÊU THU:" (Usage), and "THÀNH TIỀN:" (Amount). Below these fields are three buttons: "TÍNH TIỀN" (Calculate), "NHẬP MỚI" (New Entry), and "THOÁT" (Exit). On the right is a list view with a header row containing five columns: "Họ tên", "Khu vực", "Định mức", "Tiêu thụ", and "Thành tiền". Below the header is a large empty area for the list items. At the bottom of the window, there is a label "TỔNG TIỀN:" (Total Amount) followed by a text box showing the value "0", and a button labeled "XÓA" (Delete).

Thực hiện các yêu cầu sau:

Thiết lập thuộc tính cho phép chọn nhiều dòng trên Listview.

Combobox có 3 khu vực: Khu vực 1 (định mức là 50), khu vực 2 (định mức là 100), khu vực 3 (định mức là 150). Khi chọn khu vực nào thì hiện định mức tương ứng.

Nút TÍNH TIỀN (hoặc Enter trên các textbox): kiểm tra dữ liệu nhập, nếu hợp lệ thì tính và xuất kết quả ra ô *Tiêu thụ* và *Thành tiền*, đồng thời thêm một dòng tương ứng vào Listview và cập nhật ô tổng tiền.

Đơn giá điện: trong định mức là 500, ngoài định mức là 1000.

Nút NHẬP MỚI: Xóa nội dung các textbox và label, đồng thời đặt con trỏ vào textbox đầu tiên

Nút XÓA: cho phép xóa 1 dòng đang chọn trong Listview, phải xác nhận lại trước khi xóa và cập nhật lại ô tổng tiền.

Nút THOÁT (hoặc nhấn Esc): thoát chương trình

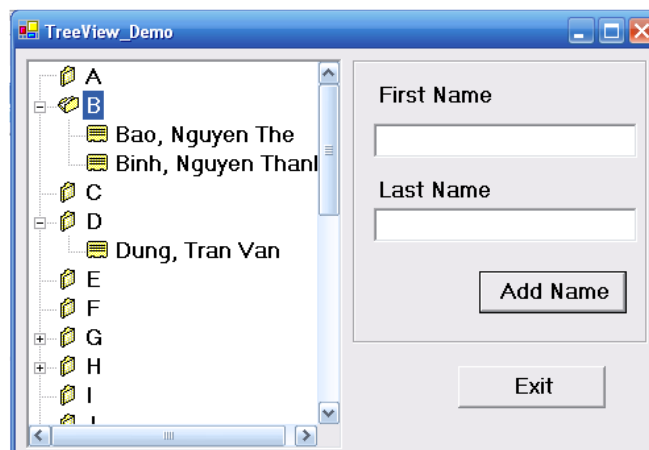
Quy định Form hiển thị giữa màn hình.

Quy định việc di chuyển tab hợp lý.

Thiết lập thuộc tính Anchor hợp lý cho các control.

Thiết lập MinimumSize cho form.

Bài 8. Viết chương trình nhập danh bạ với yêu cầu giao diện như hình dưới.



Yêu cầu:

Khi chương trình vừa hiển thị, Treeview chứa tất cả các chữ cái từ A->Z.

Nhằm mục đích tiện lợi cho người sử dụng khi tìm tên, khi người sử dụng nhập tên của một người nàp đó, chương trình sẽ đưa tên người này vào Treeview ở vị trí node có tương ứng với chữ các đầu của tên (xem hình).

Bài 9. Viết chương trình xem danh sách SV của Khoa Tin học như hình:

[illegible]

Yêu cầu:

Khi Form hiện lên, Treeview hiển thị danh sách các lớp – sinh viên như hình, chưa có nút nào được chọn. Con trỏ đặt tại ô Nhập tên.

Khi người dùng chọn nút cấp Khoa, chương trình hiện toàn bộ danh sách SV thuộc Khoa vào Listview.

Khi người dùng chọn một lớp bất kỳ thì chương trình hiện toàn bộ danh sách SV thuộc lớp đang chọn vào Listview.

Khi chọn 1 SV bất kỳ thì chỉ hiện thị SV đó vào Listview.

Nút Tìm: cho phép tìm SV (trong cấp đang chọn trên Treeview) có họ tên chứa chuỗi nhập trong Textbox. Hiện kết quả ra Listview.

Bài 10. Thiết kế giao diện như sau:

Thực hiện các yêu cầu sau:

Thiết lập HideSelection = False.

Khi Form hiện lên, đã có sẵn 1 số lớp trong danh sách lớp ở Treeview.

Nút Cập Nhật: Thêm 1 SV vào lớp đang chọn trên Treeview với nội dung các nút như hình. Trước khi thêm phải kiểm tra thông tin nhập gồm: các ô nhập không được để trống, không được trùng mã SV. Ngoài ra còn phải kiểm tra nút chọn trên Treeview có phải là nút lớp không (chỉ được thêm vào nút lớp).

Nút Xóa: cho phép xóa nút đang chọn trong Treeview, phải xác nhận lại trước khi xóa và chỉ được xóa khi chọn nút chứa mã SV.

Khi click chọn nút mã SV hoặc địa chỉ thì hiện thông tin sv đó qua các Textbox.

Quy định Form hiển thị giữa màn hình.

Quy định việc di chuyển tab hợp lý.

Thiết lập thuộc tính Dock hợp lý cho Treeview.

Thiết lập MinimumSize cho form.

BÀI TẬP THỰC HÀNH 7: TRUY CẬP DỮ LIỆU VỚI ADO.NET

Nội dung kiến thức thực hành:

- Tìm hiểu một số khái niệm liên quan đến ADO.NET: Data Provider, Kiến trúc của ADO.NET.
- Sử dụng các đối tượng ADO.NET: SqlConnection, SqlCommand, SqlDataReader, SqlDataAdapter, DataSet, DataTable

Bài 1: Lập trình phần mềm Quản lý thông tin sách có thêm các chức năng thêm mới, xóa thông tin sách từ CSDL

Mã sách	Tiêu đề	Giá	Số lượng
CSH8	Murach's C# 2008	54.5000	5136
DB1R	DB2 for the COBOL Programmer, Part 1 (2n...	42.0000	4825
DB2R	DB2 for the COBOL Programmer, Part 2 (2n...	45.0000	621
JSE6	Murach's JAVA SE 6	52.5000	3455
JSP2	Murach's JAVA Servlets and JSP (2nd Editi...	52.5000	4999

Chức năng của ứng dụng được mô tả như sau:

- Khi load ứng dụng
 - Hiển thị thông tin về tất cả các loại sách trong ứng dụng trên ListView
 - Các TextBox đều bị vô hiệu hóa
 - Button Thêm mới được kích hoạt cho phép thêm thông tin sách mới
 - Button Lưu, Xóa, Bỏ qua bị vô hiệu hóa
 - Khi một cuốn sách trên ListView được chọn
 - Thông tin về cuốn sách được hiển thị trên các TextBox vẫn đang bị vô hiệu hóa
 - Button Thêm mới, Xóa được kích hoạt
 - Button Lưu, Bỏ qua bị vô hiệu hóa

Mã sách	Tiêu đề	Giá	Số lượng
CSH8	Murach's C# 2008	54.5000	5136
DB1R	DB2 for the COBOL Programmer, Part 1 (2n...	42.0000	4825
DB2R	DB2 for the COBOL Programmer, Part 2 (2n...	45.0000	621
JSE6	Murach's JAVA SE 6	52.5000	3455
JSP2	Murach's JAVA Servlets and JSP (2nd Editi...	52.5000	4999

Hình 2. Form khi một item trên ListView được chọn .

- Khi button Thêm mới được nhấn
- Các TextBox được kích hoạt cho phép nhập thông tin
- Button Lưu được kích hoạt cho phép lưu thông tin
- Button Bỏ qua được kích hoạt cho phép bỏ qua thao tác Thêm mới
- Button Thêm mới, Xóa, bị vô hiệu hóa
- Khi button Xóa được nhấn
- Hiện thị thông báo hỏi người dùng có chắc chắn xóa không
- Xóa thông tin

Yêu cầu: Thực hiện thao tác thêm hàng, xóa hàng trên DataSet, sau đó cập nhật lại CSDL

1. DATABASE

```

use master
go
drop database sachdb
go

CREATE DATABASE SachDB
go
use SachDB
go
create table categories
(
categoryid int identity(1,1) primary key,
categoryname varchar(50)

```

```

)
go

insert into categories(categoryname) values ('Sách .net')
go
insert into categories(categoryname) values ('Sách java')
go
CREATE TABLE [dbo].[Products](

[ProductCode] [char](10) NOT NULL PRIMARY KEY,

[Description] [varchar](50) NOT NULL,

[UnitPrice] [money] NOT NULL,

[OnHandQuantity] [int] NOT NULL,

categoryid int

constraint fk_c_p foreign key(categoryid) references
categories(categoryid)

)

GO

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity],categoryid) VALUES (N'A3CS
', N'Murach"s ASP.NET 3.5 Web Programming with C# 2008',
54.5000, 4637,1)

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity],categoryid) VALUES (N'A3VB
', N'Murach"s ASP.NET 3.5 Web Programming with VB 2008',
54.5000, 3974,1)

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity],categoryid) VALUES (N'ADC3
', N'Murach"s ADO.NET 3.5, LINQ, and EF with C# 2008', 54.5000,
5244,2)

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity],categoryid) VALUES (N'ADV3
', N'Murach"s ADO.NET 3.5, LINQ, and EF with VB 2008', 54.5000,
4538,2)

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity]) VALUES (N'CRFC
',
N'Murach"s CICS Desk Reference', 50.0000, 1865)

```

```

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity]) VALUES (N'CSH8      ',
N'Murach"s C# 2008', 54.5000, 5136)

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity]) VALUES (N'DB1R      ', N'DB2 for
the COBOL Programmer, Part 1 (2nd Edition)', 42.0000, 4825)

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity]) VALUES (N'DB2R      ', N'DB2 for
the COBOL Programmer, Part 2 (2nd Edition)', 45.0000, 621)

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity]) VALUES (N'JSE6      ',
N'Murach"s JAVA SE 6', 52.5000, 3455)

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity]) VALUES (N'JSP2      ',
N'Murach"s JAVA Servlets and JSP (2nd Edition)', 52.5000, 4999)

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity]) VALUES (N'MCBL      ',
N'Murach"s Structured COBOL', 62.5000, 2386)

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity]) VALUES (N'MCCP      ',
N'Murach"s CICS for the COBOL Programmer', 54.0000, 2368)

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity]) VALUES (N'MDOM      ',
N'Murach"s JavaScript and DOM Scripting', 54.5000, 6937)

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity]) VALUES (N'SQL8      ',
N'Murach"s SQL Server 2008', 52.5000, 2465)

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity]) VALUES (N'VB08      ',
N'Murach"s Visual Basic 2008', 54.5000, 2193)

INSERT [dbo].[Products] ([ProductCode], [Description],
[UnitPrice], [OnHandQuantity]) VALUES (N'ZJLR      ',
N'Murach"s OS/390 and z/os JCL', 62.5000, 677)

go
select * from categories
go
select * from products
go

```

2. CODING


```

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

private void load_data()
{
    SqlConnection con = new
    SqlConnection("server=.;database=sachdb;integrated
    security=true;");
    SqlDataAdapter da = new SqlDataAdapter("select * from
    products",con);
    DataTable tb = new DataTable();
    da.Fill(tb);
    dataGridView1.DataSource = tb;

    //databinding
    textBox1.DataBindings.Clear();
    textBox2.DataBindings.Clear();
    textBox3.DataBindings.Clear();
    textBox4.DataBindings.Clear();

    textBox1.DataBindings.Add("Text",
    dataGridView1.DataSource, "productcode");
    textBox2.DataBindings.Add("Text",
    dataGridView1.DataSource, "Description");
    textBox3.DataBindings.Add("Text",
    dataGridView1.DataSource, "UnitPrice");
    textBox4.DataBindings.Add("Text",
    dataGridView1.DataSource, "OnHandQuantity");

}

private void Form1_Load(object sender, EventArgs e)
{
    load_data();
    textBox1.Enabled = false;
    textBox2.Enabled = false;
    textBox3.Enabled = false;
    textBox4.Enabled = false;

    button2.Enabled = false;
    button3.Enabled = false;
    button4.Enabled = false;
}

private void button1_Click(object sender, EventArgs e)
{
    textBox1.Enabled = true;

```

```

        textBox2.Enabled = true;
        textBox3.Enabled = true;
        textBox4.Enabled = true;
        textBox1.Text = "";
        textBox2.Text = "";
        textBox3.Text = "";
        textBox4.Text = "";
        textBox1.Focus();

        button1.Enabled = false;
        button3.Enabled = false;
        button2.Enabled = true;
        button4.Enabled = true;
    }

    private void button2_Click(object sender, EventArgs e)
    {
        SqlConnection con = new
        SqlConnection("server=.;database=sachdb;integrated
        security=true;");
        SqlCommand cmd = new SqlCommand("insert into products
        values('" + textBox1.Text + "\",'" + textBox2.Text + "\",'"
        + textBox3.Text + "\",'" + textBox4.Text + "\"'", con);
        con.Open();
        cmd.ExecuteNonQuery();
        con.Close();
        load_data();
    }

    private void button3_Click(object sender, EventArgs e)
    {
        DialogResult kq = MessageBox.Show("ban muon xoa khong
        ?", "tieu de", MessageBoxButtons.YesNo);

        if (kq == System.Windows.Forms.DialogResult.Yes)
        {
            SqlConnection con = new
            SqlConnection("server=.;database=sachdb;integrated
            security=true;");
            SqlCommand cmd = new SqlCommand("delete from products
            where productcode = '" + textBox1.Text + "'", con);
            con.Open();
            cmd.ExecuteNonQuery();
            con.Close();
            load_data();
        }
    }

    private void dataGridView1_CellClick(object sender,
    DataGridViewCellEventArgs e)
    {

```

```

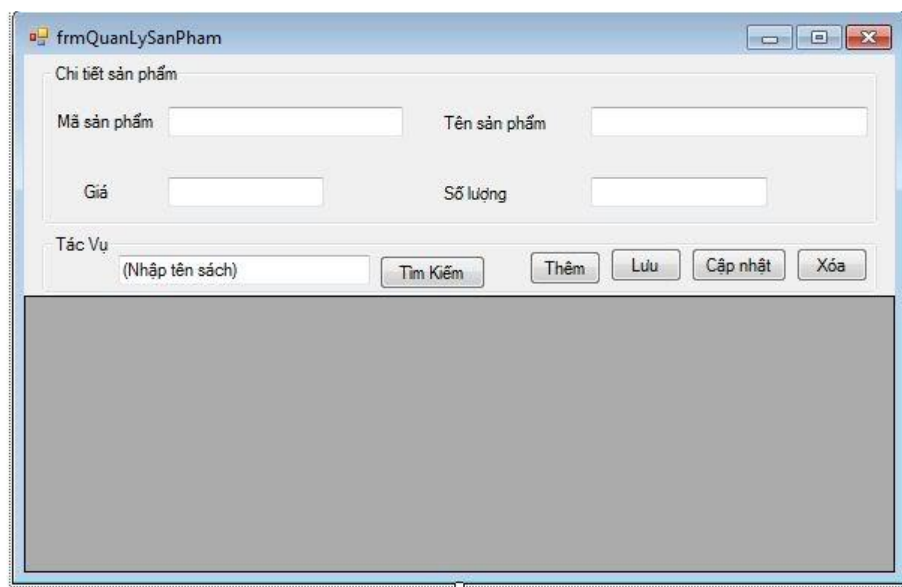
        button3.Enabled = true;
        button4.Enabled = true;
    }

    private void dataGridView1_CellContentClick(object sender,
    DataGridViewCellEventArgs e)
    {
        dataGridView1_CellClick(sender, e);
    }
}

```

Bài 2. Thực hiện các thao tác với DataGridView: hiển thị dữ liệu từ CSDL (SQL), thêm dữ liệu, xóa dữ liệu, cập nhật dữ liệu, tìm kiếm dữ liệu.

Thực hiện việc bắt lỗi nếu số lượng không phải là số nguyên và >100. Demo chương trình bằng hình ảnh.



Bài 3: Viết chương trình quản lý Sinh Viên C# với databse SQL trong đó

- Load dữ liệu từ file SQL vào Visual Studio
- Click hiện thông tin từ bảng DataGridView lên các TextBox
- Sử dụng được các nút Thêm, Xóa, Sửa, Lưu, Hủy, Thoát
- Có sử dụng việc bắt lỗi

Form1 www.tamga.tk

QUẢN LÝ THÔNG TIN SINH VIÊN

Cách 2

Mã SV Họ SV Tên SV

Mã Khoa Ngày sinh Giới tính

[www.c10mt.tk](#)
[www.c10maytinh.tk](#)
[www.tamga.tk](#)
[tamgaalbum@yahoo.com](#)
 Phone: 01283.98.69.98

	Mã SV	Họ SV	Tên SV	Ngày Sinh	Giới tính	Mã Khoa
	C04	Nguyễn Hoàng ...	Hưng ...	03/19/1990	Nam	CN
	T00	Lê	Tuấn	02/15/1991	Nam	TO
	T01	Bùi Minh	Khánh ...	04/09/1990	Nam	TO
	T02	Trần Thị	Lan ...	03/04/1990	Nữ	TO
	T03	Lê	Thiện	05/18/1990	Nam	TO
	T04	Lê Thị	Thảo ...	03/27/1990	Nữ	TO
▶	T05	Tam	Ga	09/23/1985	Nam	CN

BÀI TẬP THỰC HÀNH 8: TRUY CẬP DỮ LIỆU VỚI ADO.NET (tiếp)

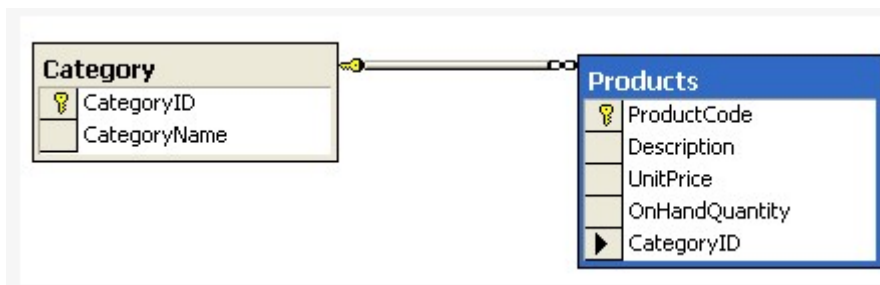
Mục tiêu kiến thức:

Sinh viên vẫn tiếp tục làm các bài thực hành liên quan tới cơ sở dữ liệu nhưng tập trung vào phần thống kê đưa ra danh sách dựa vào các điều kiện có trước. Ngoài ra, hướng dẫn một số cách làm khác để sinh viên học tập.

Mục tiêu kỹ năng:

- Có kỹ năng tốt trong thực hành kết nối cơ sở dữ liệu ADO.NET như hiển thị dữ liệu, thêm dữ liệu,...

Bài 1:



Thông tin sách

Loại sách:

Mã sách:

Tiêu đề:

Giá:

Số lượng:

Mã sách	Tiêu đề	Giá	Số lượng
A3CS	Murach's ASP.NET 3.5 Web Programming ...	54.5000	4637
A3VB	Murach's ASP.NET 3.5 Web Programming ...	54.5000	3974
ADC3	Murach's ADO.NET 3.5, LINQ, and EF with...	54.5000	5244
ADV3	Murach's ADO.NET 3.5, LINQ, and EF with...	54.5000	4538
CRFC	Murach's CICS Desk Reference	50.0000	1865
CSH8	Murach's C# 2008	54.5000	5136

Hãy lập trình để thực hiện sau

- Đưa dữ liệu từ bảng Category vào Combobox tên là loại sách

b. Thực hiện chức năng nhập mới sách theo Category.

c. Hiển thị dữ liệu của bảng Product ra DataGridView

Hướng dẫn:

```
public partial class Form2 : Form
{
    public Form2()
    {
        InitializeComponent();
    }

    private void load_data()
    {
        SqlConnection con = new
        SqlConnection("server=.;database=sachdb;integrated
        security=true;");

        SqlDataAdapter da = new SqlDataAdapter("select * from
        products", con);

        DataTable tb = new DataTable();

        da.Fill(tb);

        dataGridView1.DataSource = tb;

        //databinding

        textBox1.DataBindings.Clear();

        textBox2.DataBindings.Clear();

        textBox3.DataBindings.Clear();

        textBox4.DataBindings.Clear();
    }
}
```

```

        textBox1.DataBindings.Add("Text",
        dataGridView1.DataSource, "productcode");

        textBox2.DataBindings.Add("Text",
        dataGridView1.DataSource, "Description");

        textBox3.DataBindings.Add("Text",
        dataGridView1.DataSource, "UnitPrice");

        textBox4.DataBindings.Add("Text",
        dataGridView1.DataSource, "OnHandQuantity");
    }

    private void Form2_Load(object sender, EventArgs e)
    {
        load_data();

        textBox1.Enabled = false;

        textBox2.Enabled = false;

        textBox3.Enabled = false;

        textBox4.Enabled = false;

        button2.Enabled = false;

        button3.Enabled = false;

        button4.Enabled = false;

        // load dữ liệu lên comboBox

        SqlConnection con = new
        SqlConnection("server=.;database=sachdb;integrated
        security=true;");

        SqlDataAdapter da = new SqlDataAdapter("select * from
        categories", con);

        DataTable tb = new DataTable();

        da.Fill(tb);

        comboBox1.DataSource = tb;

        comboBox1.DisplayMember = "categoryname";

        comboBox1.ValueMember = "categoryid";
    }

```

```
}
```

```
private void button1_Click(object sender, EventArgs e)
```

```
{
```

```
    textBox1.Enabled = true;
```

```
    textBox2.Enabled = true;
```

```
    textBox3.Enabled = true;
```

```
    textBox4.Enabled = true;
```

```
    textBox1.Text = "";
```

```
    textBox2.Text = "";
```

```
    textBox3.Text = "";
```

```
    textBox4.Text = "";
```

```
    textBox1.Focus();
```

```
    button1.Enabled = false;
```

```
    button3.Enabled = false;
```

```
    button2.Enabled = true;
```

```
    button4.Enabled = true;
```

```
}
```

```
private void button2_Click(object sender, EventArgs e)
```

```
{
```

```
    SqlConnection con = new  
    SqlConnection("server=.;database=sachdb;integrated  
    security=true;");
```

```
    SqlCommand cmd = new SqlCommand("insert into products  
    values(\'" + textBox1.Text + "\",'" + textBox2.Text +  
    "\",'" + textBox3.Text + "\",'" + textBox4.Text +  
    "\",'" + comboBox1.SelectedValue.ToString() + "\')", con);
```



```

        con.Open();

        cmd.ExecuteNonQuery();

        con.Close();

        load_data();
    }

private void button3_Click(object sender, EventArgs e)
{
    DialogResult kq = MessageBox.Show("ban muon xoa khong ?", "tieu de", MessageBoxButtons.YesNo);

    if (kq == System.Windows.Forms.DialogResult.Yes)
    {
        SqlConnection con = new
        SqlConnection("server=.;database=sachdb;integrated security=true;");

        SqlCommand cmd = new SqlCommand("delete from
        products where productcode = '" + textBox1.Text
        + "'", con);

        con.Open();

        cmd.ExecuteNonQuery();

        con.Close();

        load_data();
    }
}

private void dataGridView1_CellClick(object sender,
DataGridViewCellEventArgs e)
{
    button3.Enabled = true;
}

```

```

        button4.Enabled = true;
    }

    private void dataGridView1_CellContentClick(object sender,
DataGridViewCellEventArgs e)
    {
        dataGridView1_CellClick(sender, e);
    }

    private void button5_Click(object sender, EventArgs e)
    {
        SqlConnection con = new
        SqlConnection("server=.;database=sachdb;integrated
        security=true;");

        //MessageBox.Show(textBox1.Text);

        //MessageBox.Show(textBox2.Text);

        //MessageBox.Show(textBox3.Text);

        //MessageBox.Show(textBox4.Text);

        //MessageBox.Show(comboBox1.SelectedValue.ToString())
        ;

        //string sql = "update products set description='" +
        textBox2.Text + "',unitprice='" + textBox3.Text +
        "',OnHandQuantity='" + textBox4.Text + "',categoryid='" + c
        omboBox1.SelectedValue.ToString() + "' where
        productcode='" + textBox1.Text + "'";

        //MessageBox.Show(sql);

        SqlCommand cmd = new SqlCommand("update products set
        description=@description,unitprice=@unitprice,OnHandQ
        uantity=@onhandquantity,categoryid=@categoryid where
        productcode=@productcode", con);
    }

```

```

cmd.Parameters.AddWithValue("@description",
textBox2.Text);

cmd.Parameters.AddWithValue("@unitprice",
textBox3.Text);

cmd.Parameters.AddWithValue("@OnHandQuantity",
textBox4.Text);

cmd.Parameters.AddWithValue("@categoryid",
comboBox1.SelectedValue.ToString());

cmd.Parameters.AddWithValue("@productcode",
textBox1.Text);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

load_data();

}

private void comboBox1_SelectedIndexChanged(object sender,
EventArgs e)

{

//SqlConnection con = new
SqlConnection("server=.;database=sachdb;integrated
security=true;");

//SqlDataAdapter da = new SqlDataAdapter("select *
from products where categoryid=@categoryid", con);

//da.SelectCommand.Parameters.AddWithValue("@category
id", comboBox1.SelectedValue);

//DataSet ds = new DataSet();

//da.Fill(ds);

//dataGridView1.DataSource = ds.Tables[0];

}

}

```

Bài 2: Viết chương trình thống kê sinh viên theo Môn học bằng C# với database SQL trong đó:

- Load dữ liệu từ file SQL vào Visual Studio
- Click hiển thị thông tin từ ComboBox sẽ hiện lên Textbox và DataGridView
- Có sử dụng Class để kết nối

The screenshot shows a Windows application window with a light gray background. At the top, there is a form with the following elements:

- A label "Mã Môn Học" followed by a dropdown menu showing "CNTT".
- A label "Số tiết" followed by a text box containing "45".
- A button labeled "Thoát" (Exit).
- A label "Tên Môn Học" followed by a text box containing "Cong nghe thong tin".

Below the form is a DataGridView table with the following data:

	Mã SV	Họ SV	Tên SV	Ngày sinh	Điểm
▶	s001	Tran Minh	Son	01/05/1985	4
	s001	Tran Minh	Son	01/05/1985	6
	s007	Phan Thi	Ha	03/07/1988	2
	s007	Phan Thi	Ha	03/07/1988	9
	s008	Tran The	Dung	21/10/1985	7

Hướng dẫn:

- Xây dựng lớp kết nối

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.SqlClient;
using System.Data;

namespace _39_SQLTamGa_ThongKeTheoMonHoc
{
    public class KetNoiDuLieu
    {
        public SqlConnection cnn = new SqlConnection
            ("Data Source=(local);Initial Catalog=QLSV;Integrated Security=True");

        public void myconnect()
        { cnn.Open(); }

        public void myclose()
        { cnn.Close(); }

        // www.tamga.tk www.c10mt.tk www.c10maytinh.tk
        public DataTable taobang(string sql)
        {
            DataTable dt = new DataTable();
            SqlDataAdapter ds = new SqlDataAdapter(sql, cnn);
            ds.Fill(dt);
            return (dt);
        }
    }
}

```

Các hàm trong Form

```
public Form1()
{
    InitializeComponent();
}

KetNoiDulieu kn = new KetNoiDulieu();

private void Form1_Load_1(object sender, EventArgs e)
{
    kn.myconnect();
    string sql = "SELECT * FROM MONHOC";
    cbMaMon.DataSource = kn.taobang(sql);
    cbMaMon.DisplayMember = "MAMH";
}

private void btThoat_Click(object sender, EventArgs e)
{
    this.Close();
    kn.myclose();
}

// www.tamga.tk www.c10mt.tk www.c10maytinh.tk
private void cbMaMon_SelectedIndexChanged_1(object sender, EventArgs e)
{
    // load dữ liệu từ combobox xuống text
    string s = "select * from monhoc where mamh='" + cbMaMon.Text + "'";
    DataTable d = kn.taobang(s);
    foreach (DataRow hang in d.Rows)
        KhungTenMon.Text = hang["TENMH"].ToString();
    foreach (DataRow hang in d.Rows)
        KhungSoTiet.Text = hang["SOTIET"].ToString();

    // load dữ liệu lên DataGridView
    string s2 = "select sv.masv, hosv, tensv, ngaysinh, diem " +
        "from SinhVien SV, KetQua KQ " +
        "where (SV.masv = KQ.masv) and (KQ.mamh= '" + cbMaMon.Text + "')";
    dataGridViewMH.DataSource = kn.taobang(s2);
}
}
```

Bài 3: Viết chương trình thống kê sinh viên theo Khoa bằng C# với database SQL trong đó:

- Load dữ liệu từ file SQL vào Visual Studio
- Click hiện thông tin từ ComboBox sẽ hiện lên Textbox và DataGridView
- Có sử dụng lớp để kết nối dữ liệu

Form1

THỐNG KÊ SỐ SINH VIÊN THEO KHOA

Mã Khoa: Tên Khoa:

www.tamga.tk
 www.c10mt.tk
 www.c10maytinh.tk

	Mã SV	Họ SV	Tên SV	Ngày Sinh
▶	s001	Tran Minh	Son	01/05/1985
	s002	Nguyen Quoc	Bao	16/05/1986
	s006	Nguyen Thi	Lam	11/11/1984
	s007	Phan Thi	Ha	03/07/1988
	s008	Tran The	Dung	21/10/1985
	s001	Tam	Ca	24/10/2012

Tổng số Sinh Viên:

```

/*
TamGa
www.tamga.tk www.c10mt.tk www.c10maytinh.tk
www.tamga85.multiply.com
Phone: 01283.98.69.98 Email : tamgaalbum@yahoo.com
*/
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.SqlClient;
using System.Data;

namespace _40_SQLTamGa_ThongKeTheoKhoa
{
    public class KetNoiDuLieu
    {
        public SqlConnection cnn = new SqlConnection
            ("Data Source=(local);Initial Catalog=QLSV;Integrated Security=True");

        public void myconnect()
        { cnn.Open(); }
    }
}

```

```

Form1.cs  KetNoiDuLieu.cs  Form1.cs [Design]
_40_SQLTamGa_ThongKeTheoKhoa.KetNoiDuLieu  taobang(string sql)

public void myclose()
{ cnn.Close(); }

// www.tamga.tk www.c10mt.tk www.c10maytinhtk
public DataTable taobang(string sql)
{
    DataTable dt = new DataTable();
    SqlDataAdapter ds = new SqlDataAdapter(sql, cnn);
    ds.Fill(dt);
    return (dt);
}
}

```

```

namespace _40_SQLTamGa_ThongKeTheoKhoa
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            KetNoiDuLieu kn = new KetNoiDuLieu();

            private void btThoat_Click(object sender, EventArgs e)
            {
                this.Close();
                kn.myclose();
            }

            // www.tamga.tk www.c10mt.tk www.c10maytinhtk
            private void Form1_Load(object sender, EventArgs e)
            {
                kn.myconnect();
                string sql = "select * from Khoa";
                cbMaKhoa.DataSource = kn.taobang(sql);
                cbMaKhoa.DisplayMember = "MAKHOA";
            }

            private void cbMaKhoa_SelectedIndexChanged(object sender, EventArgs e)
            {
                // load dữ liệu từ combobox xuống text
                string s = "select * from khoa where makhoa='" + cbMaKhoa.Text + "'";
                DataTable d = kn.taobang(s);
                foreach (DataRow hang in d.Rows)
                    KhungTenKhoa.Text = hang["TENKHOA"].ToString();
            }
        }
    }
}

```



```

// load dữ liệu lên DataGridView
string s1 = "select masv, hosv, tensv, ngaysinh " +
            "from SinhVien SV, Khoa KH " +
            "where (SV.makhoa = KH.makhoa) " +
            "and (KH.makhoa= '" + cbMaKhoa.Text + "')";
dataGridViewKhoa.DataSource = kn.taobang(s1);

string s2 = "select count(*) from SinhVien " +
            "where makhoa='" + cbMaKhoa.Text + "' group by makhoa";
KhungTong.Text = (dataGridViewKhoa.Rows.Count).ToString();
}

```

Bài 4. Tạo 1 ứng dụng đơn giản để quản lý sinh viên. Giao diện được thiết kế như hình

Student Name	Birth Day	Address
Hoang Tuan Dat	03/10/1985	Quang Nam
Phuong Thi Nha	23/06/1988	Dong Anh
Ngo Xuan Phu	09/10/1987	Ha Noi
Dinh Duc Thanh	05/09/1988	Ha Tay
Manh Hoai Thu	22/08/1987	Nhon

Các bạn thiết kế 1 ComboBox để hiển thị tên lớp, 1 textBox hiển thị tên giáo viên chủ nhiệm, 1 textBox hiển thị số sinh viên có trong lớp.. 1 listBox hiển thị thông tin chi tiết về sinh viên bao gồm tên, ngày sinh, địa chỉ. Thiết kế cơ sở dữ liệu như sau:

Bảng Student

Column Name	Data Type	Allow Nulls
stuNo	varchar(5)	<input type="checkbox"/>
stuName	varchar(50)	<input checked="" type="checkbox"/>
classNo	varchar(50)	<input type="checkbox"/>
stuYear	datetime	<input checked="" type="checkbox"/>
address	varchar(100)	<input checked="" type="checkbox"/>

Bảng StudentClass

THAOTRINH.Stude...bo.StudentClass		THAOTRINH.Stude...nt - dbo.Student	
	Column Name	Data Type	Allow Nulls
▶	classNo	varchar(50)	<input type="checkbox"/>
	className	varchar(30)	<input type="checkbox"/>
	totalStudent	int	<input checked="" type="checkbox"/>
	homeroomTeacher	varchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

BÀI TẬP THỰC HÀNH 9: ỨNG DỤNG WEB VỚI WEB FORMS

Mục tiêu kiến thức:

Sinh viên sẽ thực hành làm một số bài tập cơ bản để làm quen với ASP.NET như tạo trang đầu tiên với HelloWorld...

Mục tiêu kỹ năng:

- Có kỹ năng tốt ban đầu trong lập trình với Webform, tạo các ứng dụng đơn giản.

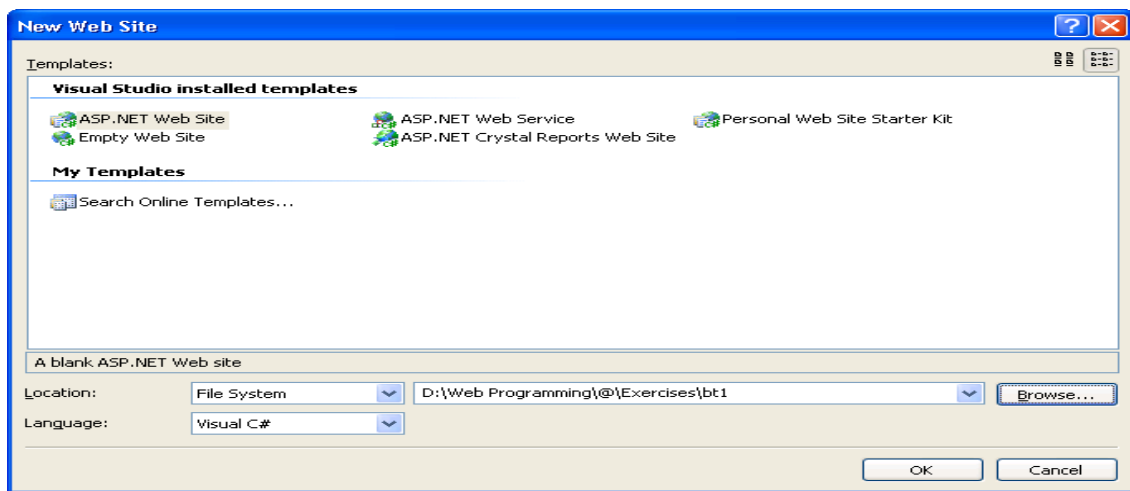
Bài 1:(Thiết lập Webserver)

Tạo 1 trang **Helloworld.aspx** như sau”

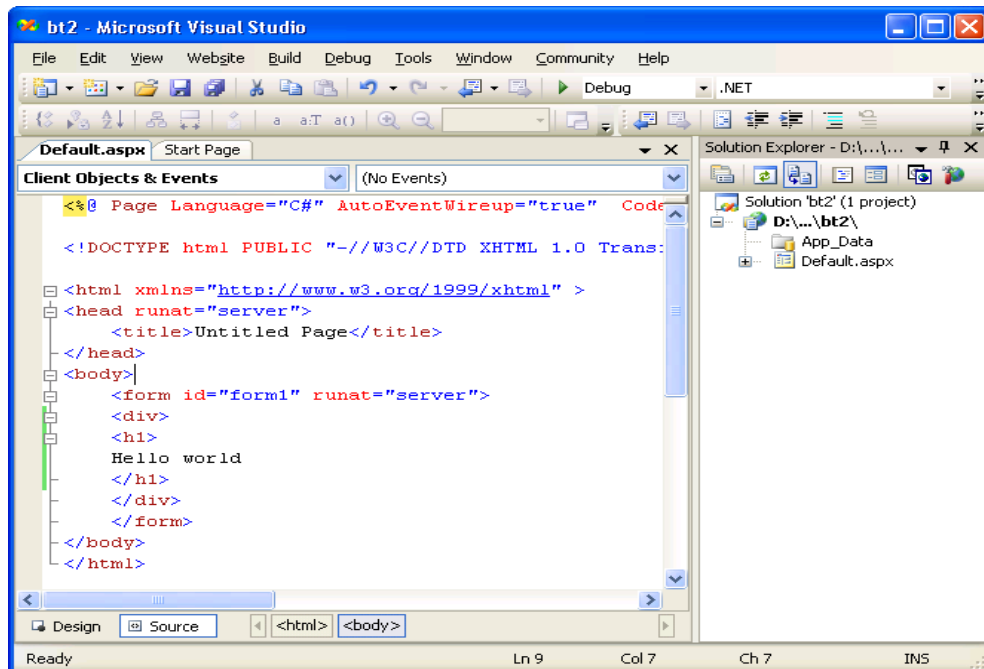
Bước 1:

Mở VS.Net tạo trang ASP.Net Website: **bt1**

Chọn Location: **File System**



Bước 2:

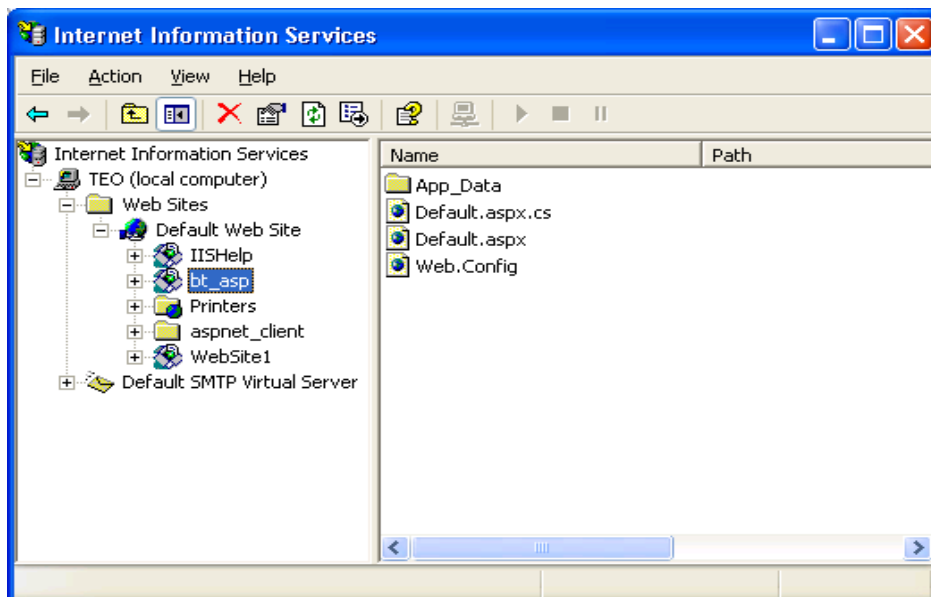


Bước 3:

Thực thi trang **Default.aspx**

Cách 1: Biên dịch trực tiếp trên Visual .Net 2005 (F5).

Cách 2: Hãy thiết lập WebServer cho Website(đặt tên ứng dụng tùy ý ví dụ **bt_asp**) sau: (Tạo thư mục ảo trên Web Server: IIS)



Thiết lập website **bt_asp** tham chiếu đến folder **bt1**. Thiết lập trang mặc định là **Default.aspx**. Sau đó mở trình duyệt IE nhập vào địa chỉ sau và xem kết quả:

Trường hợp 1: http://localhost/bt_asp sau đó nhấn Enter

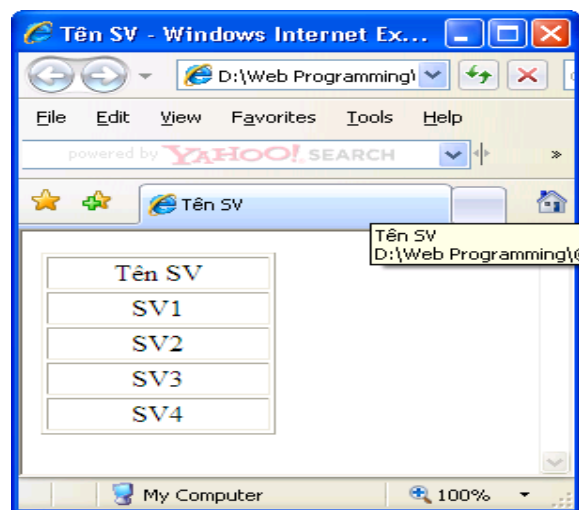
Trường hợp 2: `http://tenserver/bt_asp` sau đó nhấn Enter (tenserver chính là computerName của máy chứa website)

Trường hợp 3: `http://tenserver/bt_asp/Default.aspx` sau đó nhấn Enter (tenserver chính là computerName của máy chứa website)

Bài 2: Tạo trang như hình dưới đây

Bước 1: Yêu cầu tạo 1 tập tin CSS tên stylesheet.css (Màu nền, màu chữ, font chữ).

Bước 2: Mở lại Web site bt1, add thêm 1 trang bt11.html như sau, sau đó nhúng tập tin stylesheet.css vừa tạo vào trang bt11.html



Bước 3: Mở bt11.html đổi tên bt11.aspx. Thêm dòng code sau vào đầu trang

```
<%@ Page Language="C#" %>
```

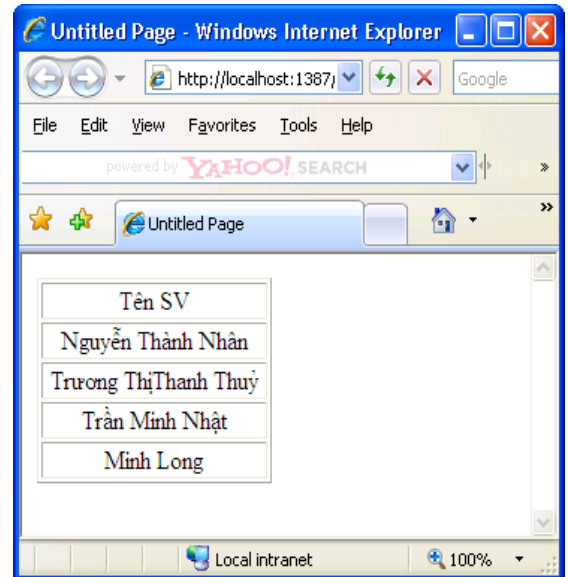
Nhập đoạn code inline sau:

```
<head>
    <title>Untitled Page</title>
    <link href="stylesheet.css" rel="stylesheet" type="text/css" />
</head>
<body>
<%
string name1 = "Nguyễn Thành Nhân", name2 = "Trương Thị Thanh Thủy", name3 =
"Trần Minh Nhật", name4="Minh Long";
%>
```

```

<table border="1" width="49%" id="table1">
    <tr>
        <td align="center">
            <p align="center">Tên SV</p>
        <td>
            <%=name1 %></td>
        </tr>
        <tr>
            <td align="center">
                <%=name2 %></td>
            <td>
                <%=name3 %></td>
            </tr>
            <tr>
                <td align="center">
                    <%=name4 %></td>
                <td>
                    <%=name5 %></td>
                </tr>
        </table>
</body>

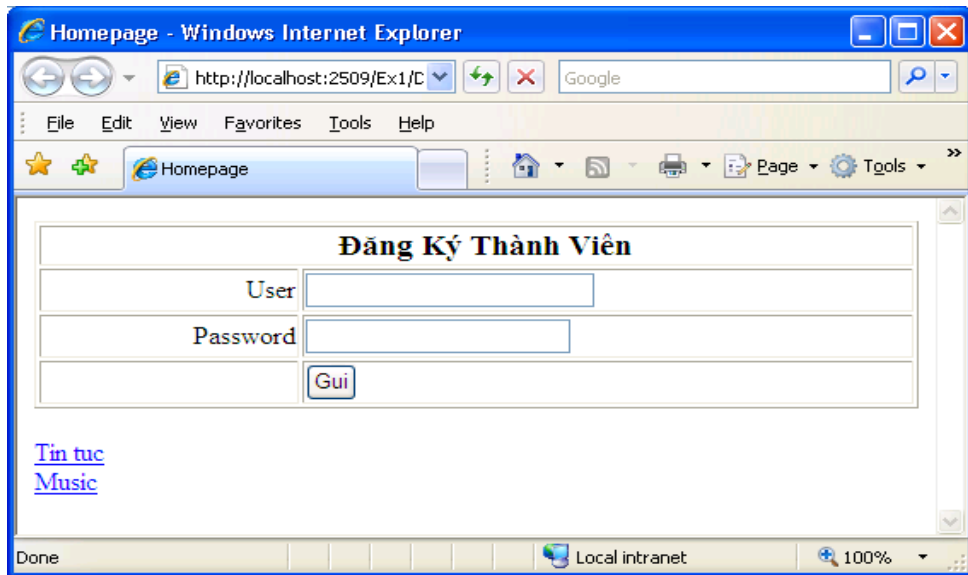
```



Bài 3:

Tạo Web site Baitap3 gồm các file: Default.aspx, CheckUser.aspx, News.aspx.

Trang Default.aspx có giao diện:



```
<body>
```

```
<form method="Get" action="CheckUser.aspx">
```

```
<!--webbot bot="SaveResults" U-
File="fpweb:///private/form_results.csv" S-Format="TEXT/CSV" S-Label-
Fields="TRUE" -->
```

```
<table border="1" width="100%" id="table1">
```

```
<tr>
```

```
<td style="text-align: center" colspan="2">
```

```
<strong><span style="font-size: 14pt">&nbsp;    Đăng Ký Thành
Viên</span></strong></td>
```

```
</tr>
```

```
<tr>
```

```
<td style="width: 148px; text-align: right">User</td>
```

```
<td>
```

```
<input type="text" name="txtUser" size="20" style="font-
family: Tahoma; font-size: 10pt; color: #0000FF; font-weight: bold"></td>
```

```
</tr>
```

```
<tr>
```

```
<td style="width: 148px; text-align: right">Password</td>
```

```
<td><input type="password" name="txtPwd" size="20"></td>
```

```
</tr>
```

```
<tr>
```

```
<td style="width: 148px">&nbsp;   </td>
```

```
<td><input type="submit" value="Gui" name="B1"></td>
```

```
</tr>
```

```

        </table>

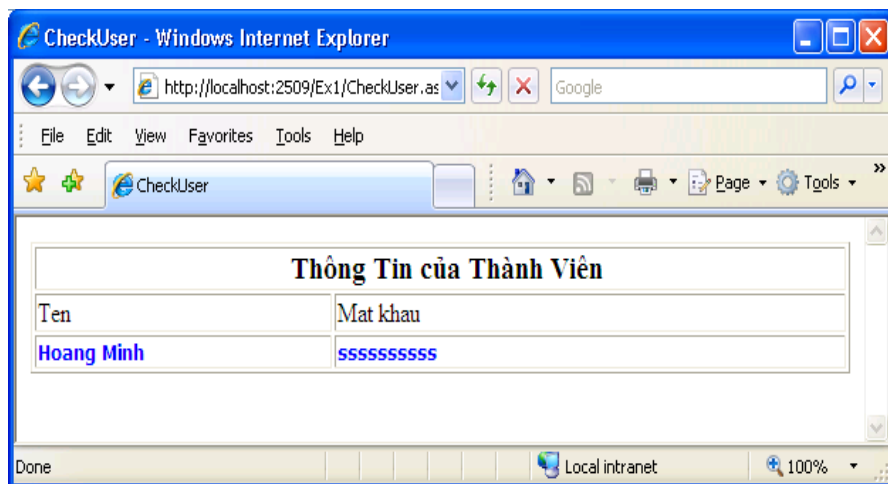
    </form>

    <a href="News.aspx">Tin tuc</a><br />
    <a href="News.aspx">Music</a><br />

</body>

```

Trang CheckUser.aspx.



```

<body>
    <%
        String strUser, strPwd;

        strUser= Request.QueryString["txtUser"];
        strPwd = Request.QueryString["txtPwd"];

        HttpCookie varCK = new HttpCookie("ckInfor");
        varCK["us"] = strUser;
        varCK["pwd"] = strPwd;
        Response.Cookies.Add(varCK);
    %>

```



```

<table border="1" width="100%" id="table1">

    <tr>

        <td style="text-align: center" colspan="2">

            <strong><span style="font-size: 14pt">&nbsp; Thông Tin của
Thành Viên</span></strong></td>

        </tr>

        <tr>

            <td style="width: 211px">Ten</td>

            <td style="width: 367px">

                Mat khau</td>

        </tr>

        <tr>

            <td style="font-family: Tahoma; font-size: 10pt; color:
#0000FF; font-weight: bold; width: 211px;">

                <%=strUser %></td>

            <td style="font-family: Tahoma; font-size: 10pt; color:
#0000FF; font-weight: bold; width: 367px;">

                <%=strPwd %></td>

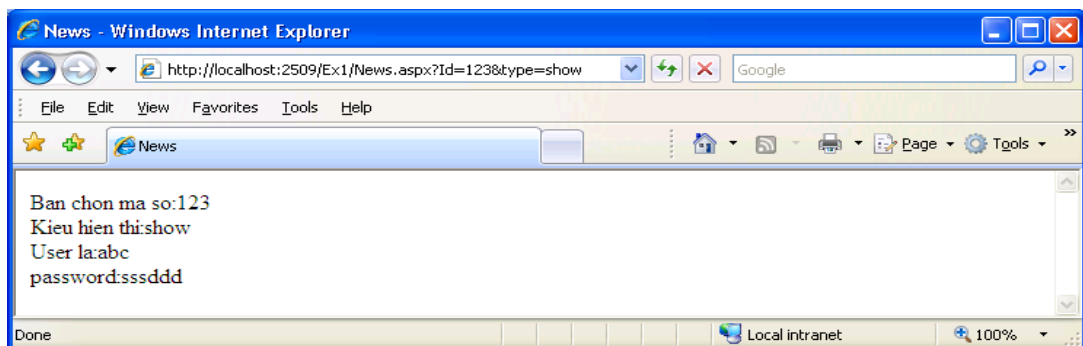
        </tr>

    </table>

</body>

Trang News.aspx

```



Truyền tham biến **Id**, **type** bằng QueryString trong trang **CheckUser.aspx**

```

<a href="News.aspx?Id=123&type=show">Tin tuc</a><br />
<a href="News.aspx?Id=36586&type=item">Music</a><br />

```

```

if (strUser == "abc")
{
    HttpCookie ckPermit = new HttpCookie("permit");
    ckPermit.Value = "true";
    Response.Cookies.Add(ckPermit);
}
else
{
    HttpCookie ckPermit = new HttpCookie("permit");
    ckPermit.Value = "false";
    Response.Cookies.Add(ckPermit);
}

```

<body>

<%

```
Boolean bl;
```

```
bl =Convert.ToBoolean( Request.Cookies["permit"].Value);
```

```
if (bl == true)
```

```
{
```

%>

```
<%String strID, strType;
```

```
strID = Request.QueryString["Id"];
```

```
strType = Request.QueryString["type"];
```

%>

```
Ban chon ma so:<%=strID%><br />
```

```
Kieu hien thi:<%=strType%><br />
```

```
<%String strckUser, strckPwd;
```

```
strckUser = Request.Cookies["ckInfor"]["us"];
```

```
strckPwd = Request.Cookies["ckInfor"]["pwd"];
```

%>

```
User la:<%=strckUser%> <br/>
```

```
password:<%=strckPwd%>
```

```

    <%>
else
    Response.Redirect("Default.aspx"); %>
</body>

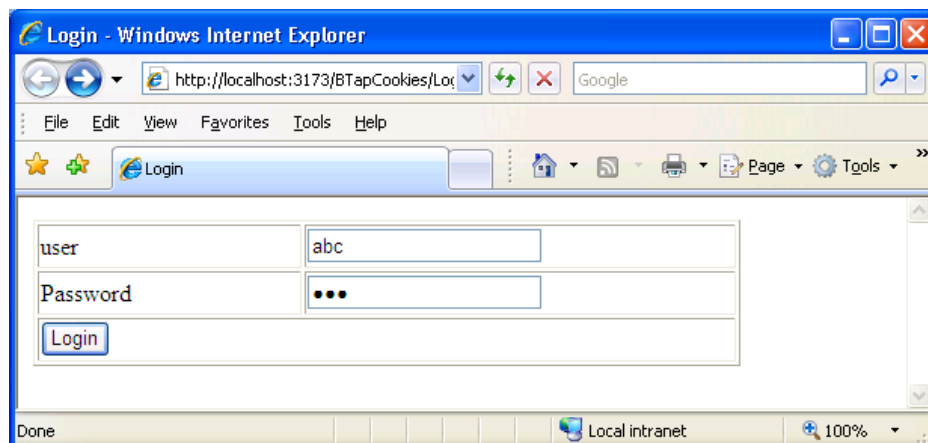
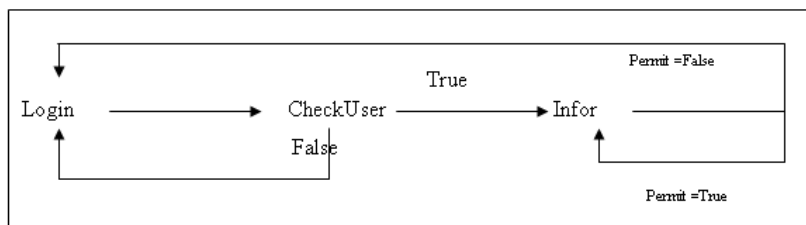
```

Bài 4:

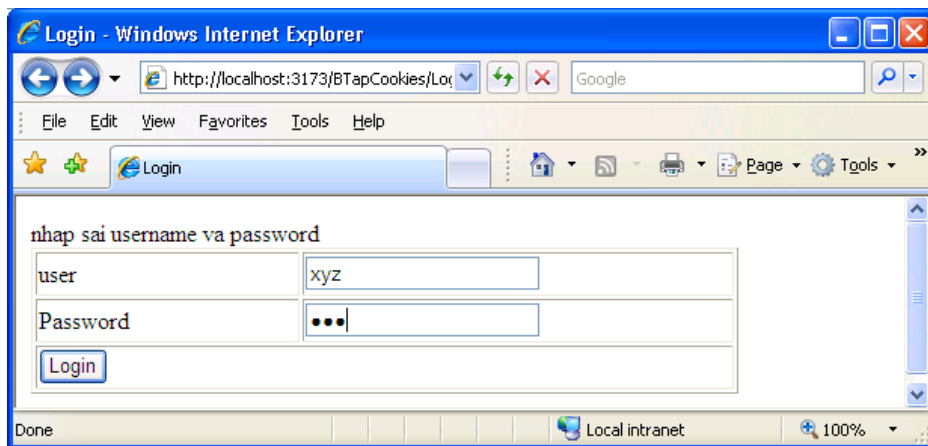
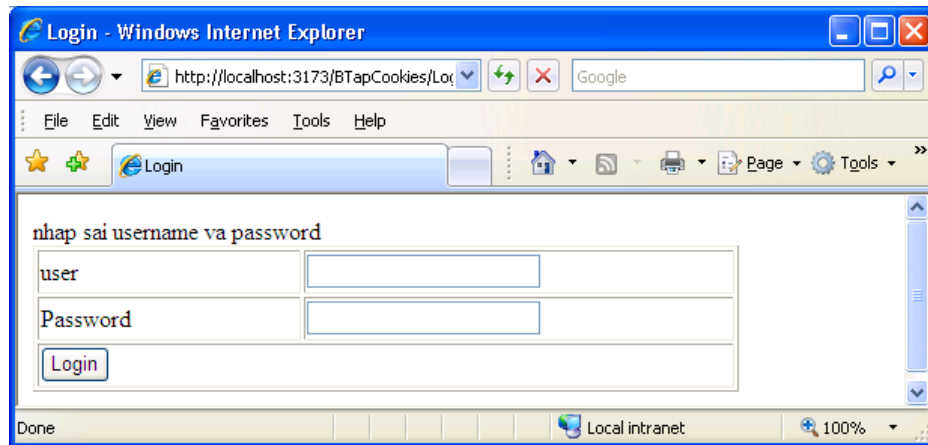
Tạo Website Baitap4 gồm các trang:

- Thiết kế 3 trang : Login.aspx , CheckUser.aspx , Infor.aspx
- Khi nhấn nút Login ở trang Login thì chuyển sang trang CheckUser để kiểm tra dữ liệu : nếu user là xyz và password là 123 thì lưu xuống 2 cookie và gọi trang Infor :
- Một cookie userCk 2 chiều : “us” = xyz và “pwd”= 123
- Một cookie permit 1 chiều chứa trị True

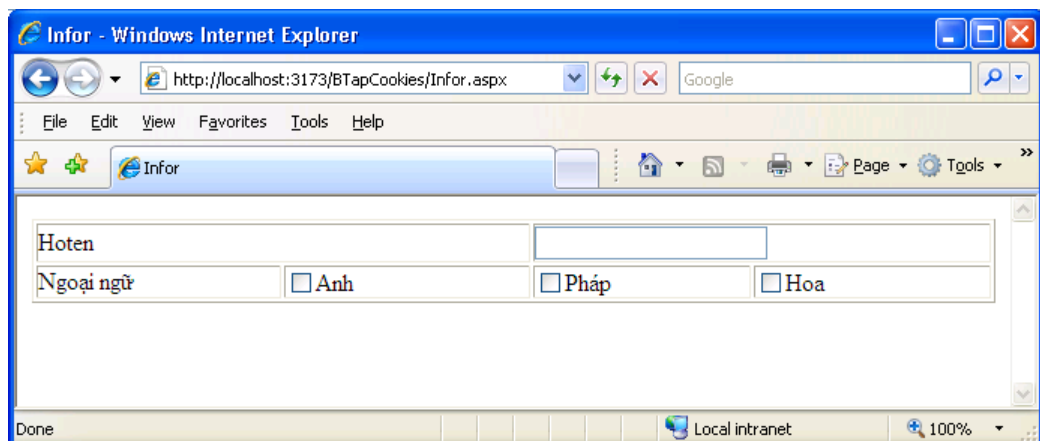
Sơ đồ thực thi của ứng dụng



Nếu nhập sai sẽ quay lại trang Login



Nếu nhập đúng userName và Password thì vào trang Infor (nội dung tùy ý)



```
// trang Login.aspx
```

```
<form method="POST" action="checkUser.aspx">
```

```
<table border="1" width="469" height="25">
```

```
<tr>
```

```

<td height="25" width="172">user</td>

<td height="25" width="281"><input type="text" name="txtUser"
size="20"></td></tr>

<tr>

    <td height="25" width="172">Password</td>

<td height="25" width="281"><input type="text" name="txtPwd"
size="20"></td></tr>

<tr>

<td height="25" width="469" colspan="2">

<input type="submit" value="Login" name="B1"></td></tr>

</table>

</form>

<%String var1;

    var1 = "1";

    var1 = Request.QueryString["th"];

    if(var1 == "59"){

        %>

            nhap sai username va password

        <%} %>

//trang CheckUser.asp

<% string strUser, strPwd;

    strUser = Request.Form["txtUser"];

    strPwd = Request.Form["txtPwd"];

HttpCookie varCk = new HttpCookie("userCK");

HttpCookie varPermit = new HttpCookie("permit");//biến này được dùng như cờ
varCk["us"] = "";

varCk["pwd"] = "";

varPermit.Value ="false";

    if ((strUser == "xyz") && (strPwd == "123"))

    {
        varPermit.Value ="true";

        varCk["us"] = strUser;

        varCk["pwd"] = strPwd;

        Response.Cookies.Add(varCk);

        Response.Cookies.Add(varPermit);

```

```

        Response.Redirect("Infor.aspx");
    }
    else
    {
        Response.Cookies.Add(varCk);
        Response.Cookies.Add(varPermit);
        Response.Redirect("Login.aspx?th = 1");
    }
%>

//trang Infor.aspx
<% try
{
    String strPermit;
    strPermit = Request.Cookies["permit"].Value;
    if (strPermit == "true")
    {
        %>
        // hienthiPage (nhấn F3 để xem)
    }
    else
        Response.Redirect("Login.aspx");
}
catch (Exception ex)
{
    Response.Redirect("Login.aspx");
}
%>

```

BÀI TẬP THỰC HÀNH 10: ỨNG DỤNG WEB VỚI WEB FORMS (tiếp)

Mục tiêu kiến thức:

Sinh viên thực hành xây dựng một số bài tập trên Server để thực hành lại phần lý thuyết đã học và củng cố kiến thức.

Mục tiêu kỹ năng:

- Sinh viên có kỹ năng trong lập trình Webform, đặc biệt trên Server

Bài 1: HTML Server Controls

Thiết kế form như sau: (Viết bằng 2 cách Code Inline & Behide)

Nhập điểm	
Mã Sv	123TCTH34A
Mã MH	Cơ Sở Dữ Liệu
Điểm	7
<input type="button" value="Nhập"/>	
Mã Sv	123TCTH34A
Mã MH	CSDL
Điểm	7

....

```
<script runat="server">
```

```
void Nhap(object sender, EventArgs e)
{
    MaSV.InnerHtml = txtTen.Value;
    MaMH.InnerHtml = drpMH.Value;
    diem.InnerHtml = txtDiem.Value;
}
```

Code Inline

```

    }
</script>

```

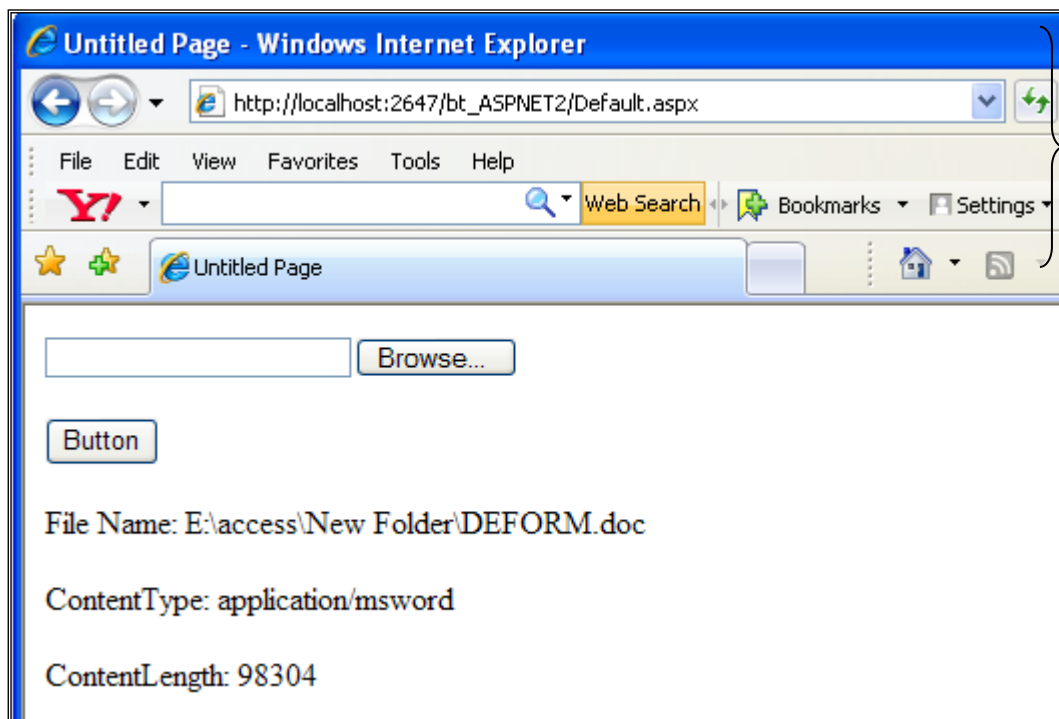
...

```

protected void Nhap_ServerClick(object sender, EventArgs e)
{
    MaSV.InnerHtml = txtTen.Value;
    MaMH.InnerHtml = drpMH.Value;
    diem.InnerHtml = txtDiem.Value;
}

```

Code Behind



Code Behind

```

protected void Button1_Click(object sender, EventArgs e)
{
    FileName.InnerHtml = FileUpload1.PostedFile.FileName;
    ContentType.InnerHtml = FileUpload1.PostedFile.ContentType;
    ContentLength.InnerHtml =
    FileUpload1.PostedFile.ContentLength.ToString();
}

```



```

string path = FileUpload1.PostedFile.FileName.ToString();
string[] filename = path.Split('\\');
string fname;

fname = filename[filename.Length - 1];

FileUpload1.PostedFile.SaveAs(Server.MapPath("/") +
    "\\UloadFiles" + fname);
}

```

Bài 2: Web Server Controls

Dùng các công cụ là Web Server Controls khi user bấm vào Thêm thì dữ liệu được chuyển sang trang Dangky.aspx

T-SPEED COMPANY

Đăng ký giấy phép lái xe ô tô

Quảng Cáo
 Giới thiệu sản phẩm
 Hợp tác
 Các đại lý
 Tuyển dụng
 Liên hệ
 Đặt hàng

Họ tên:
 Số CMND:
 Hộ khẩu: TP Hồ Chí Minh
 Hình xe: Toyota

Load hình xe

Thêm Xóa

STT	Họ tên	Loại xe
1	Minh Hoàng	Toyota
2	Ngọc Minh	Ford

Bài 7: Kiểm tra lỗi dữ liệu nhập

Yêu cầu:

Thiết kế Web Form như sau:

- Họ và tên: Không rỗng
- Số tài khoản: Là dữ liệu số, có 12 số
- Email: có dạng ten_user@ten_domain
- Thành phố: Phải được chọn
- Năm sinh: từ 1900 → 1990
- Vốn điều lệ: ≥ 20 tỉ

Nếu dữ liệu hợp lệ thì click nút Đăng ký xuất hiện thông báo “Đăng ký thành công”.

Ngược lại: Trở lại trang Dangky.aspx

The screenshot shows a Windows Internet Explorer browser window with the title 'Validation - Windows Internet Explorer'. The address bar displays 'http://localhost:3832/Validation/Default.aspx'. The page content features a registration form titled 'ĐĂNG KÝ THÀNH VIÊN VIP'. The form includes the following fields and controls:

- Họ và Tên: Text input field.
- Số tài khoản: Text input field.
- Email: Text input field.
- Thành Phố: Dropdown menu with 'Hà Nội' selected.
- Năm sinh: Text input field.
- Vốn điều lệ: Text input field.
- Buttons: 'Đăng ký' (Register) and 'Hủy' (Cancel).

The browser's status bar at the bottom shows 'Done', 'Local intranet', and '100%' zoom.