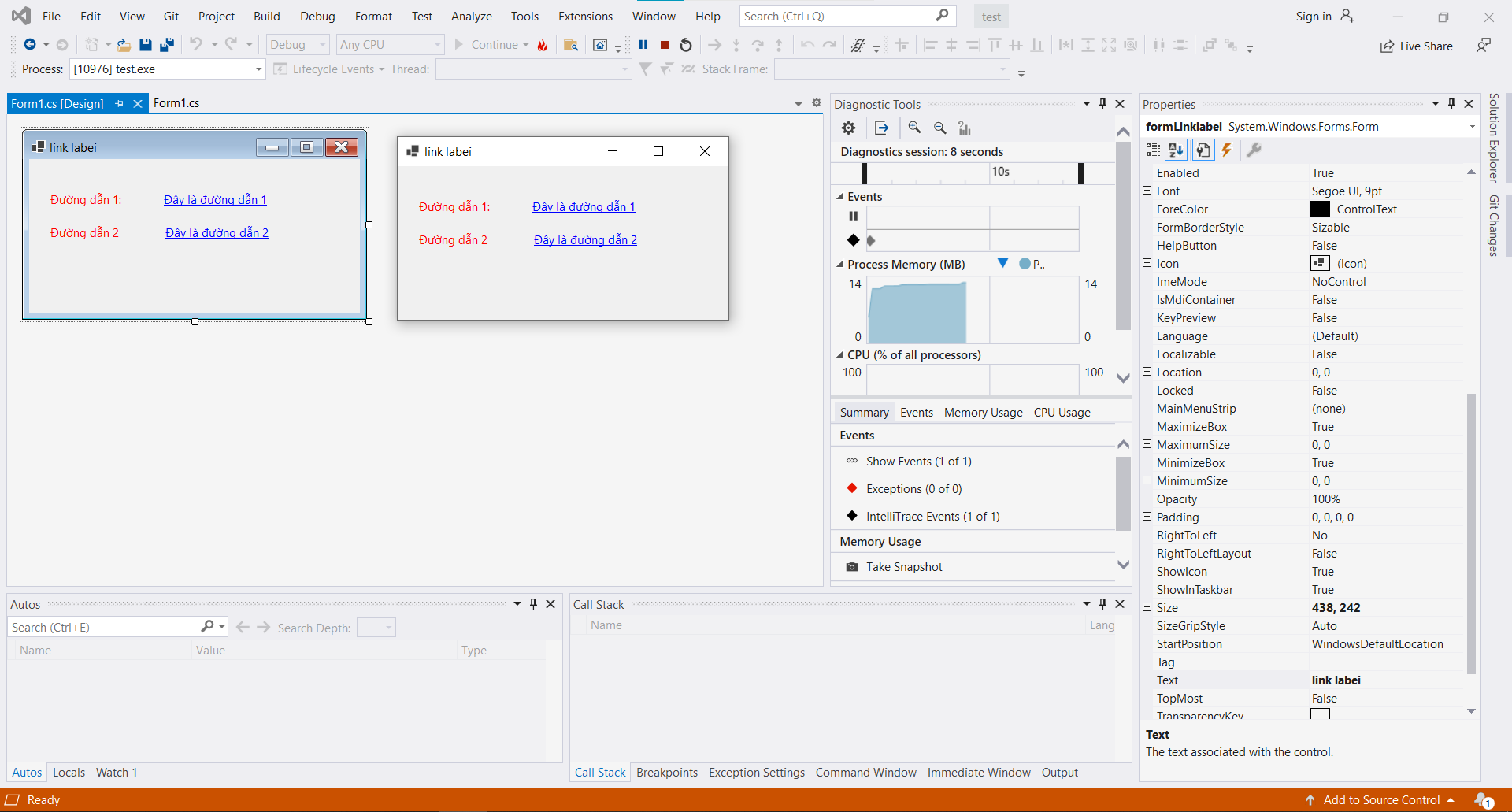
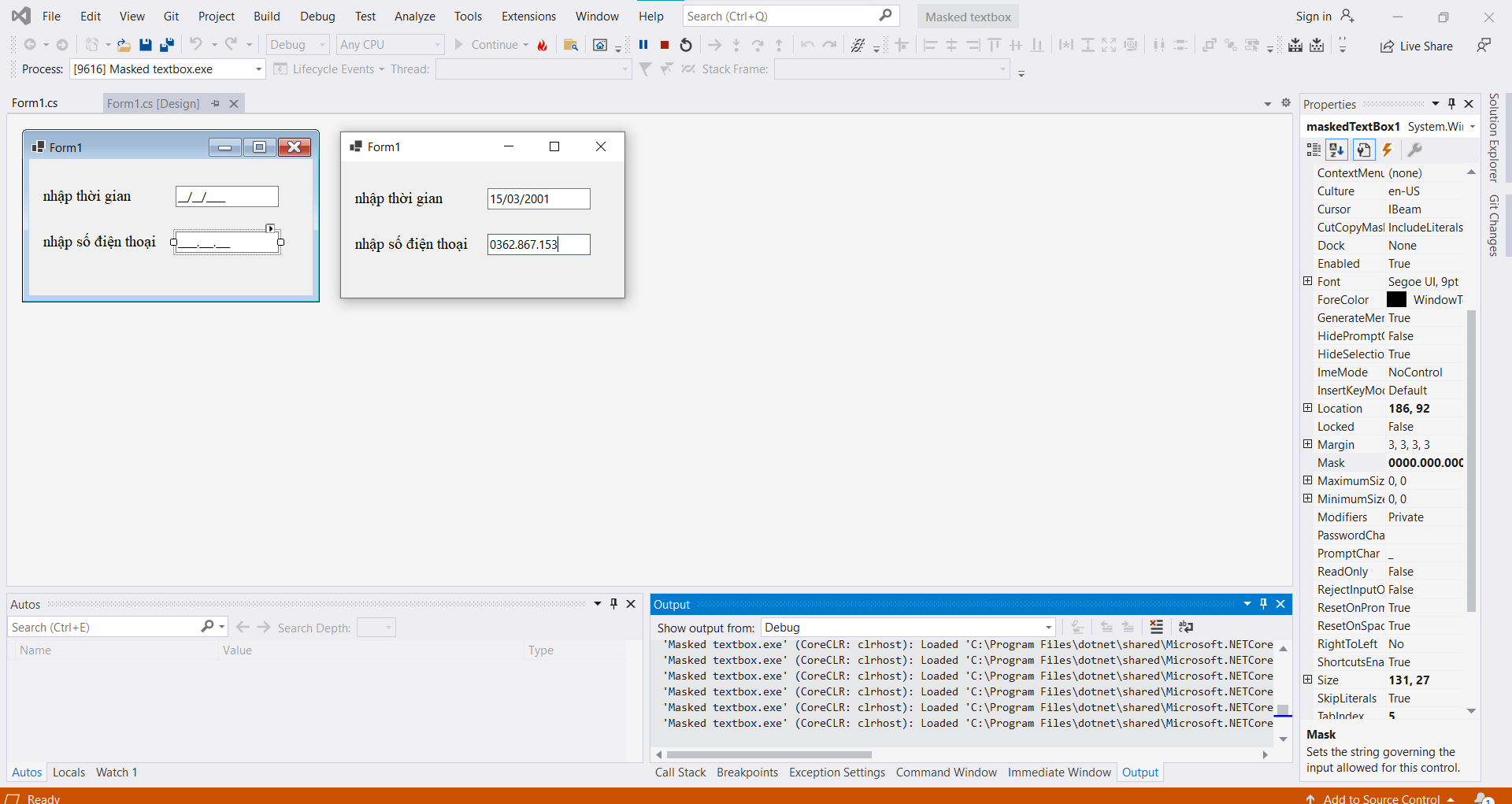
**1. Gõ và hoàn thiện các ví dụ từ slide**

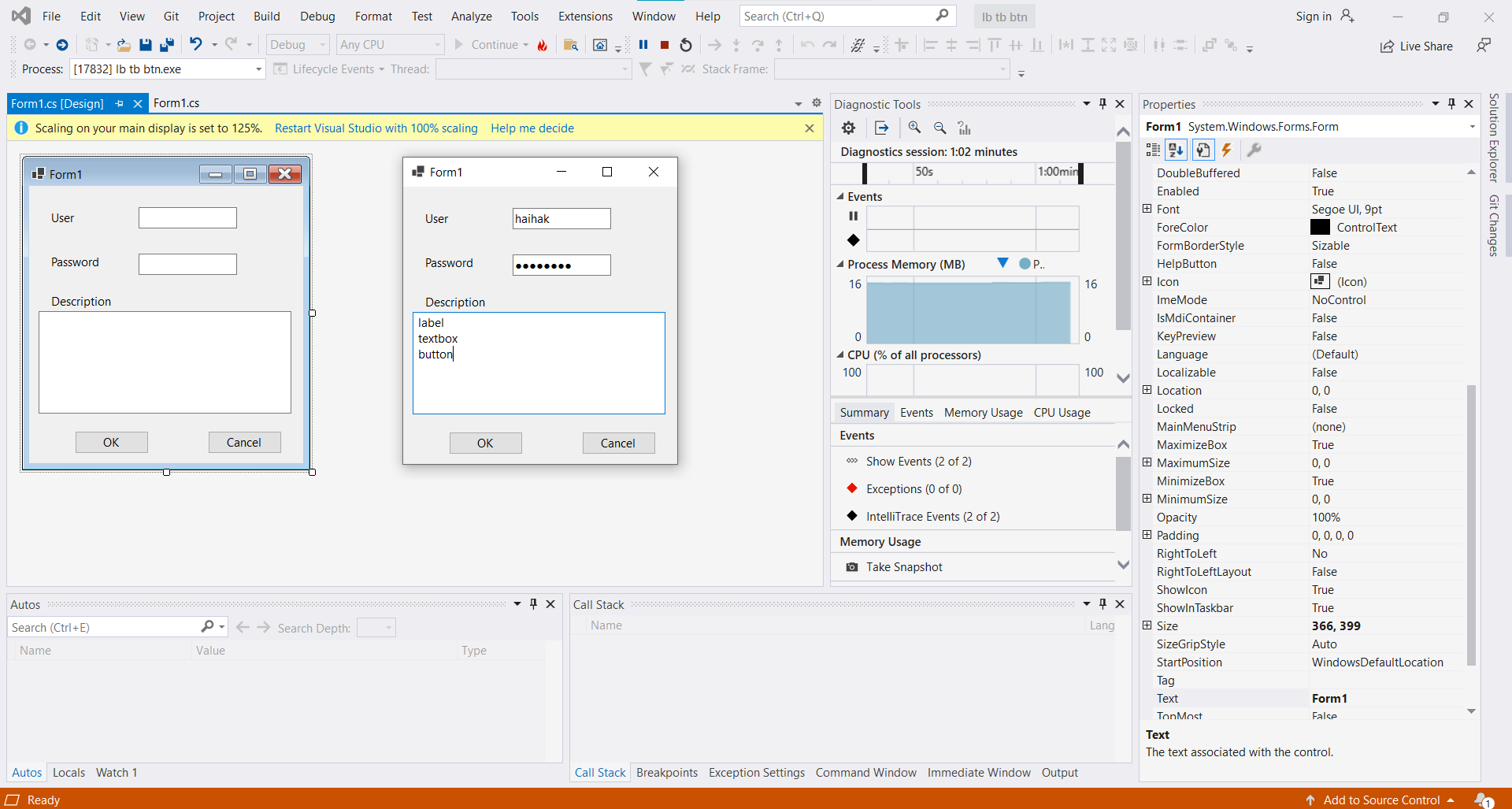
-         Label và Linklabel:



-         MaskedTextbox: yêu cầu nhập ngày sinh dạng dd/mm/yyyy và nhập số điện thoại dạng 0974.345.234.



-         Label, button, textbox:



-         MessageBox: khi xử lý sự kiện của một button bất kỳ

Code:

namespace messagebox

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void btnInformation\_Click(object sender, EventArgs e)

{

MessageBox.Show("Imformation box with an OK button","InformationIcon",MessageBoxButtons.OK, MessageBoxIcon.Information);

}

private void btnError\_Click(object sender, EventArgs e)

{

MessageBox.Show("Error box with Abort,Ertry, and Ignore buttons", "ErrorIcon",MessageBoxButtons.AbortRetryIgnore,MessageBoxIcon.Error);

}

private void btnWarning\_Click(object sender, EventArgs e)

{

MessageBox.Show("Waring box with Ok and cancel buttons","WarningIcon",MessageBoxButtons.OKCancel,MessageBoxIcon.Warning);

}

private void btnQuestion\_Click(object sender, EventArgs e)

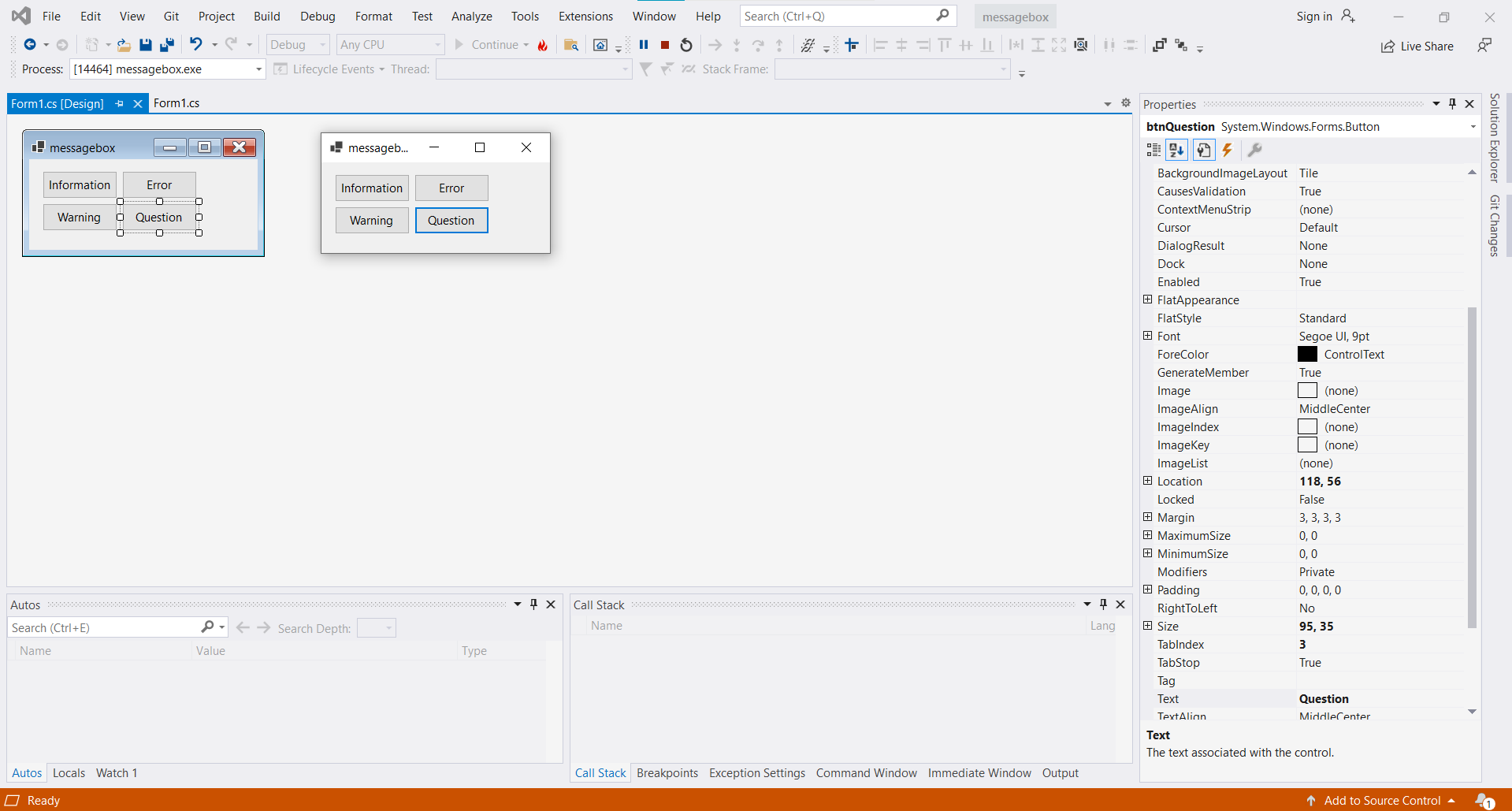
{

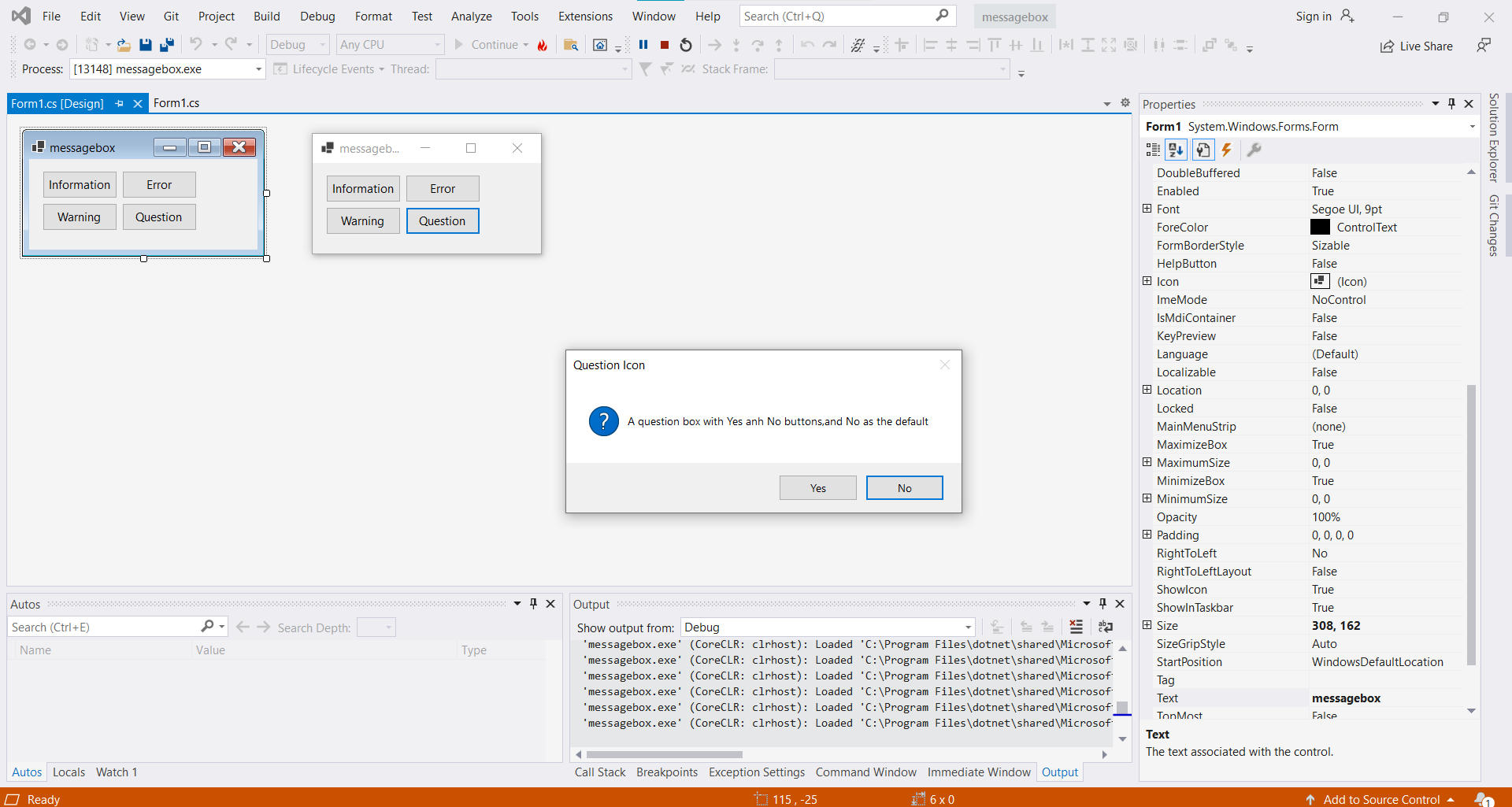
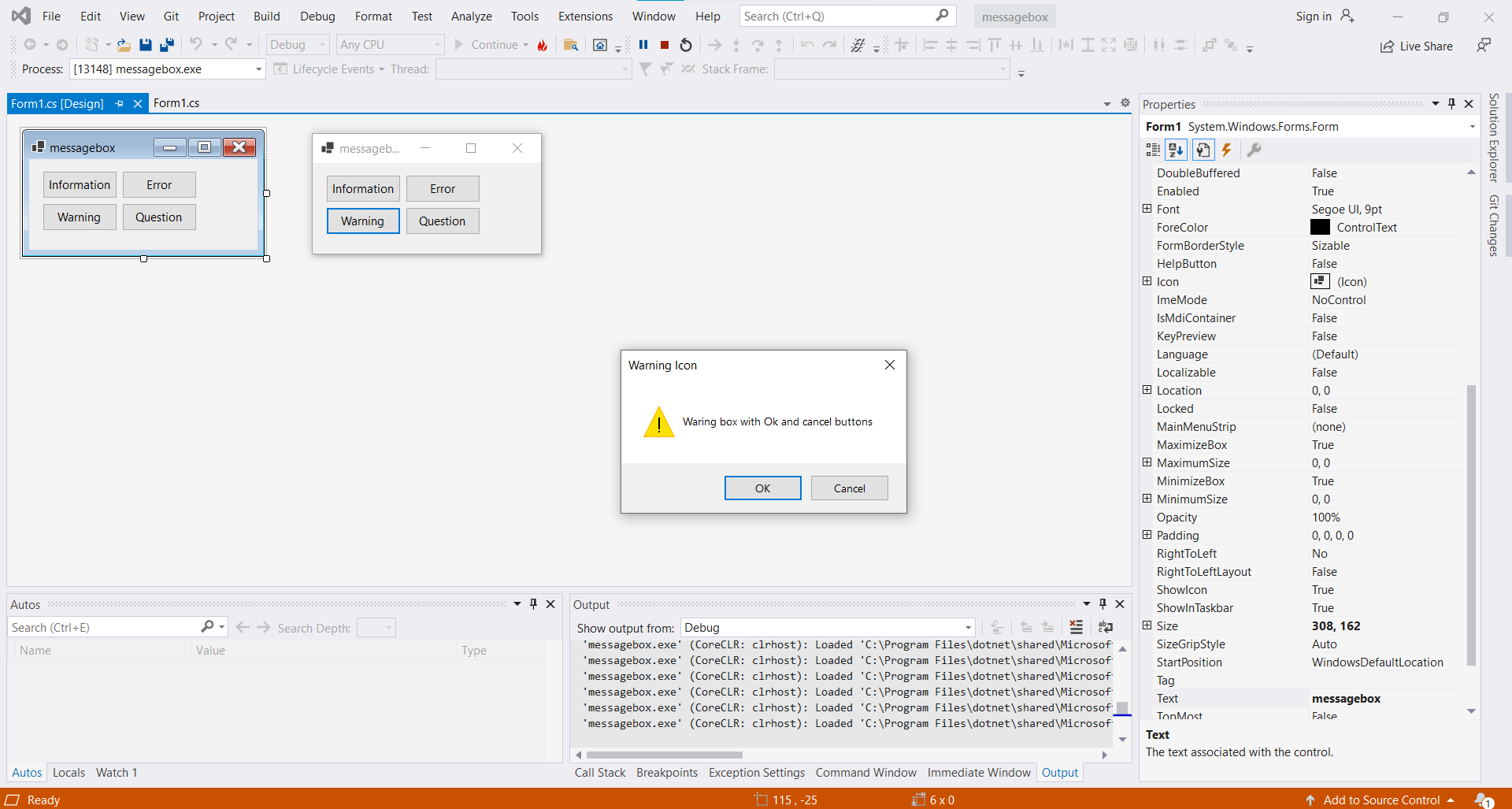
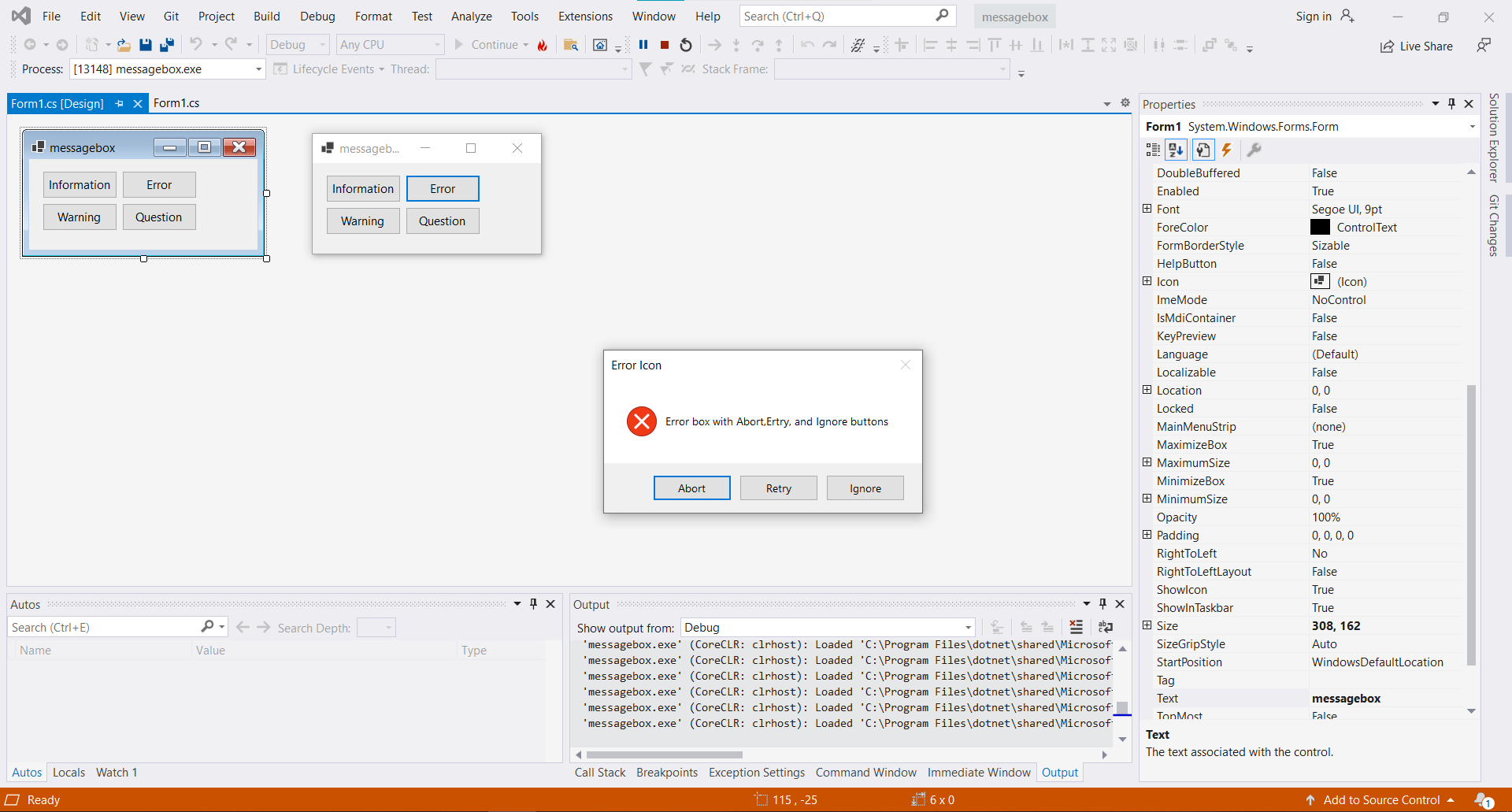
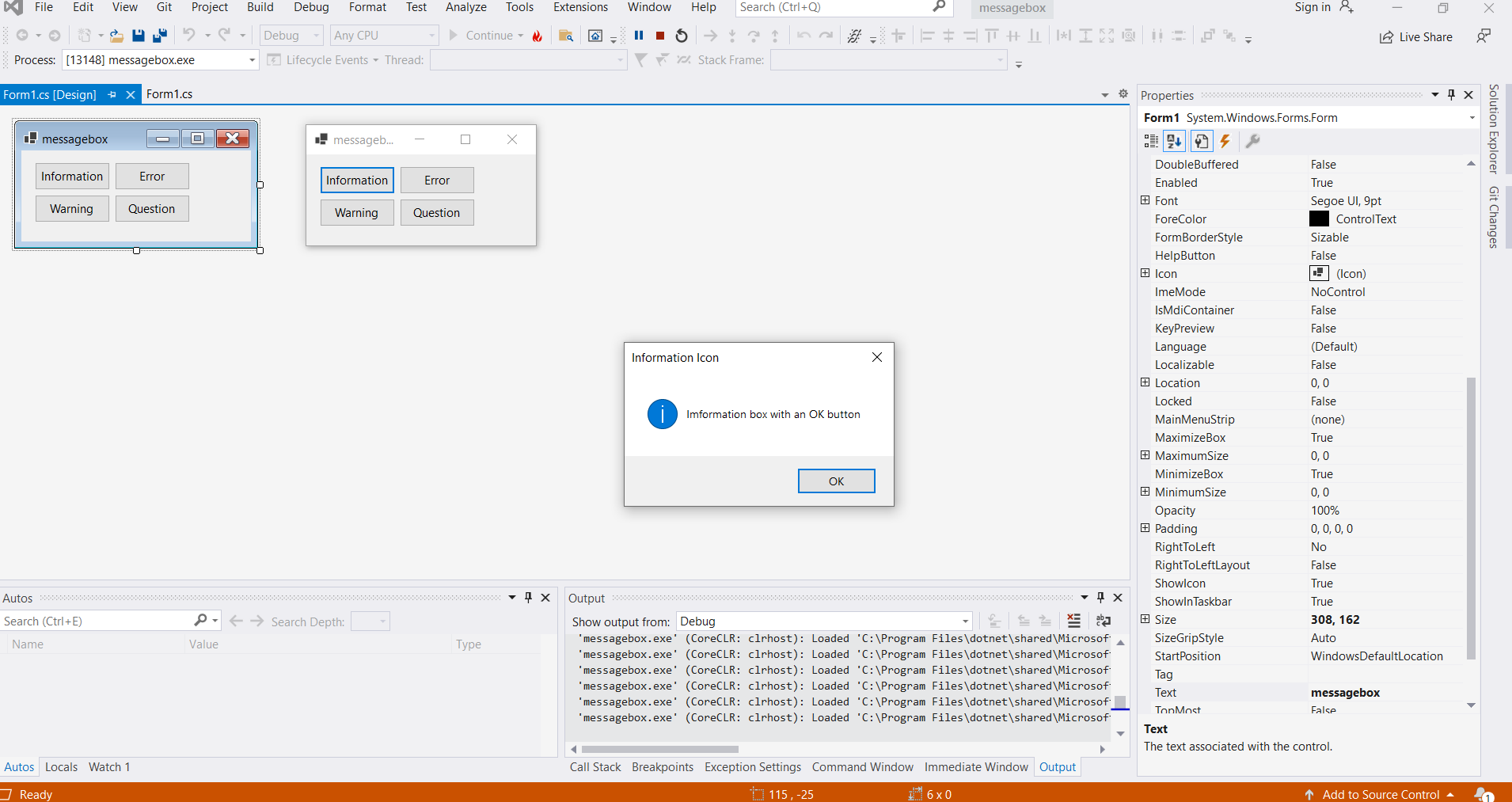
MessageBox.Show("A question box with Yes anh No buttons,and No as the default","QuestionIcon",MessageBoxButtons.YesNo,MessageBoxIcon.Question,MessageBoxDefaultButton.Button2);

}

}

}





-         Combo box:

Code:

namespace combobox

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

string[] data = { "Tiếng Anh", "Tiếng Pháp", "Tiếng Nhật" };

this.comboBox1.DataSource = data;

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

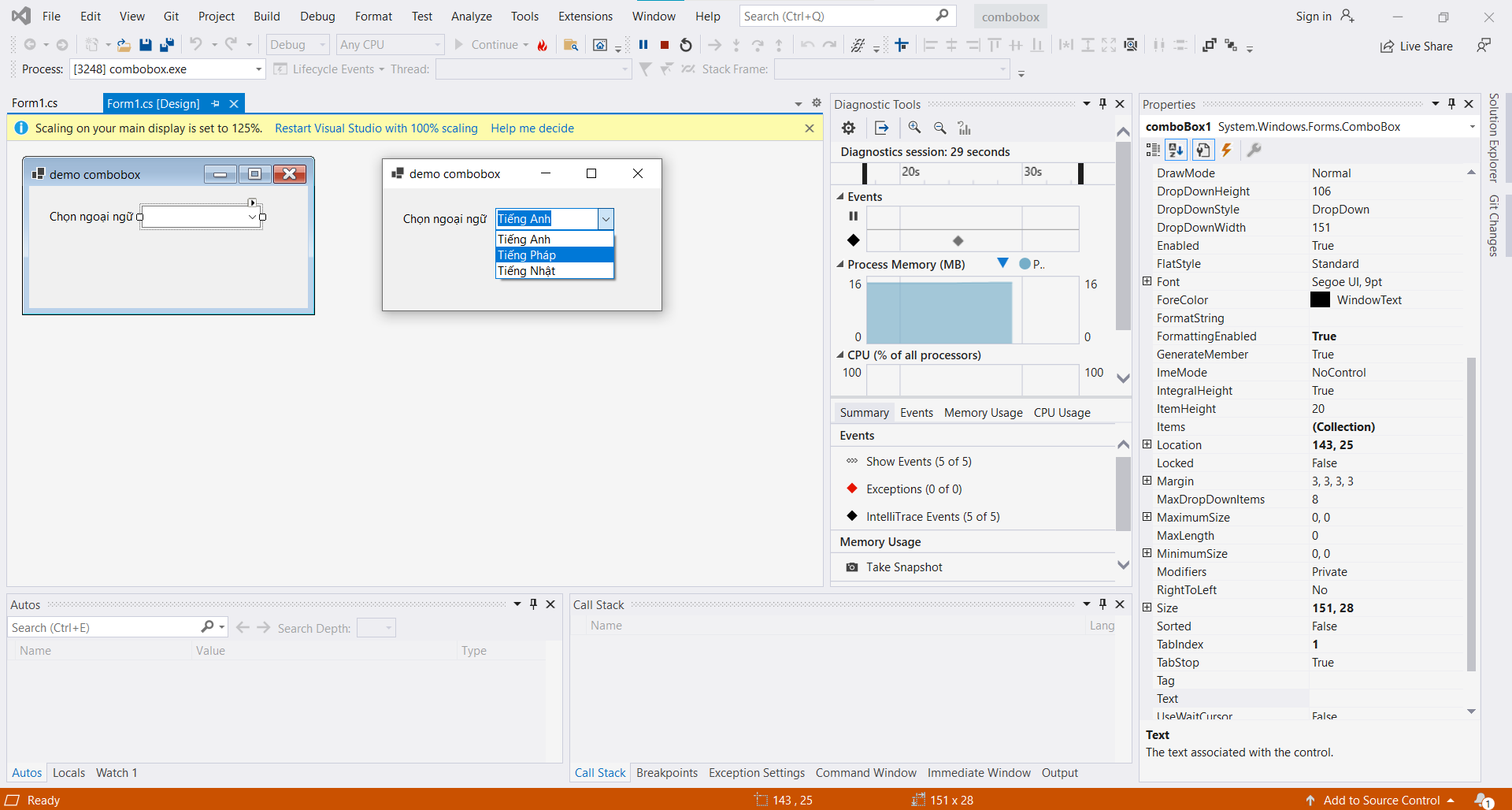
{

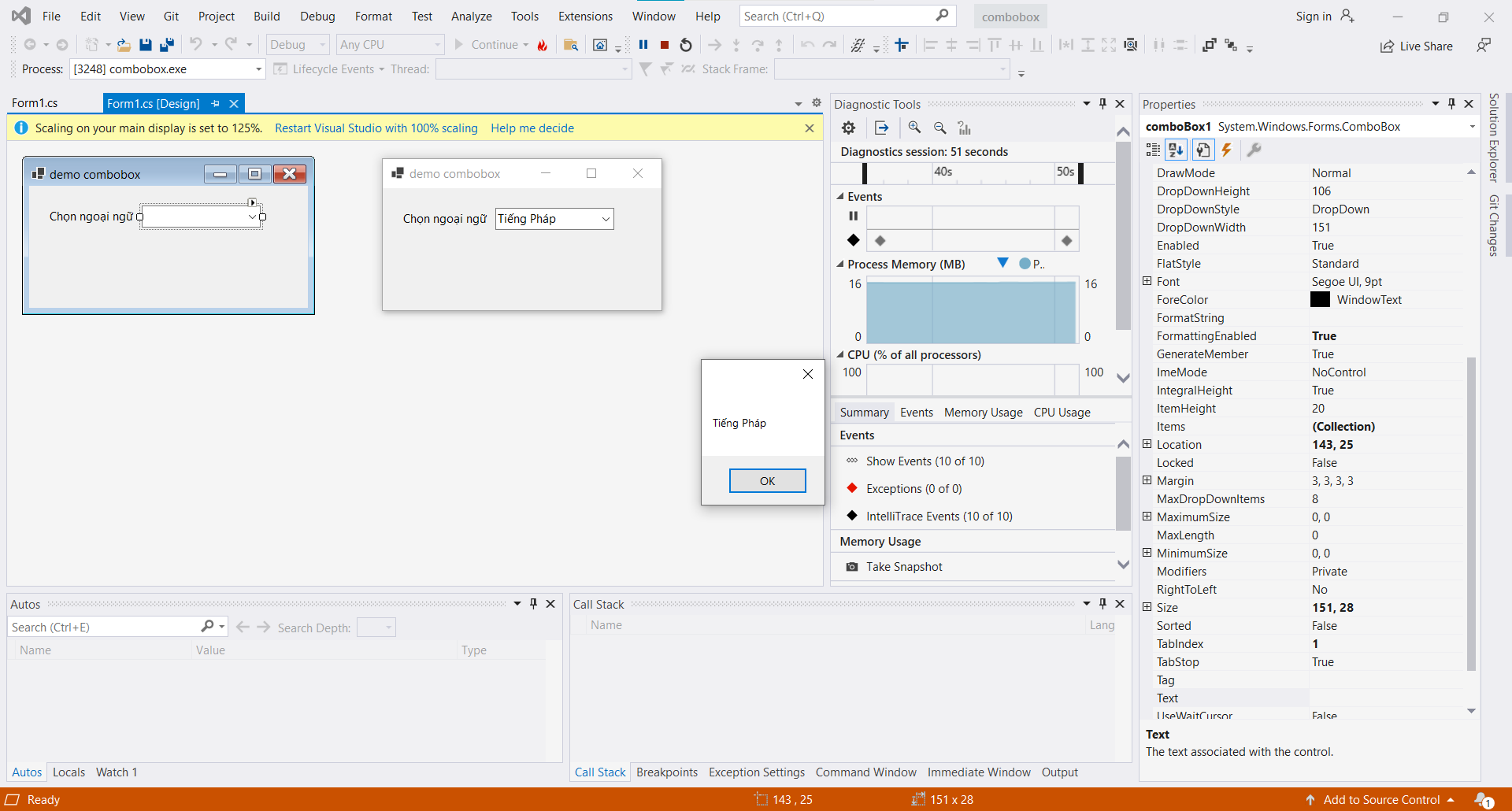
MessageBox.Show(this.comboBox1.SelectedItem.ToString());

}

}

}





-         Listbox:

Code:

namespace Listbox

{

public partial class formlistbox : Form

{

public formlistbox()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

int itemsslelect = lbDanhsach.SelectedItems.Count - 1;

for(int i= itemsslelect; i>= 0; i--)

{

lbDanhsach.Items.Add(lbDanhsachChon.SelectedItems[i]);

lbDanhsachChon.Items.Remove(lbDanhsachChon.SelectedItems[i]);

}

}

private void btnXoa\_Click(object sender, EventArgs e)

{

int i = lbDanhsach.SelectedItems.Count - 1;

while(i>=0)

{

lbDanhsach.Items.RemoveAt(lbDanhsach.SelectedIndices[i]);

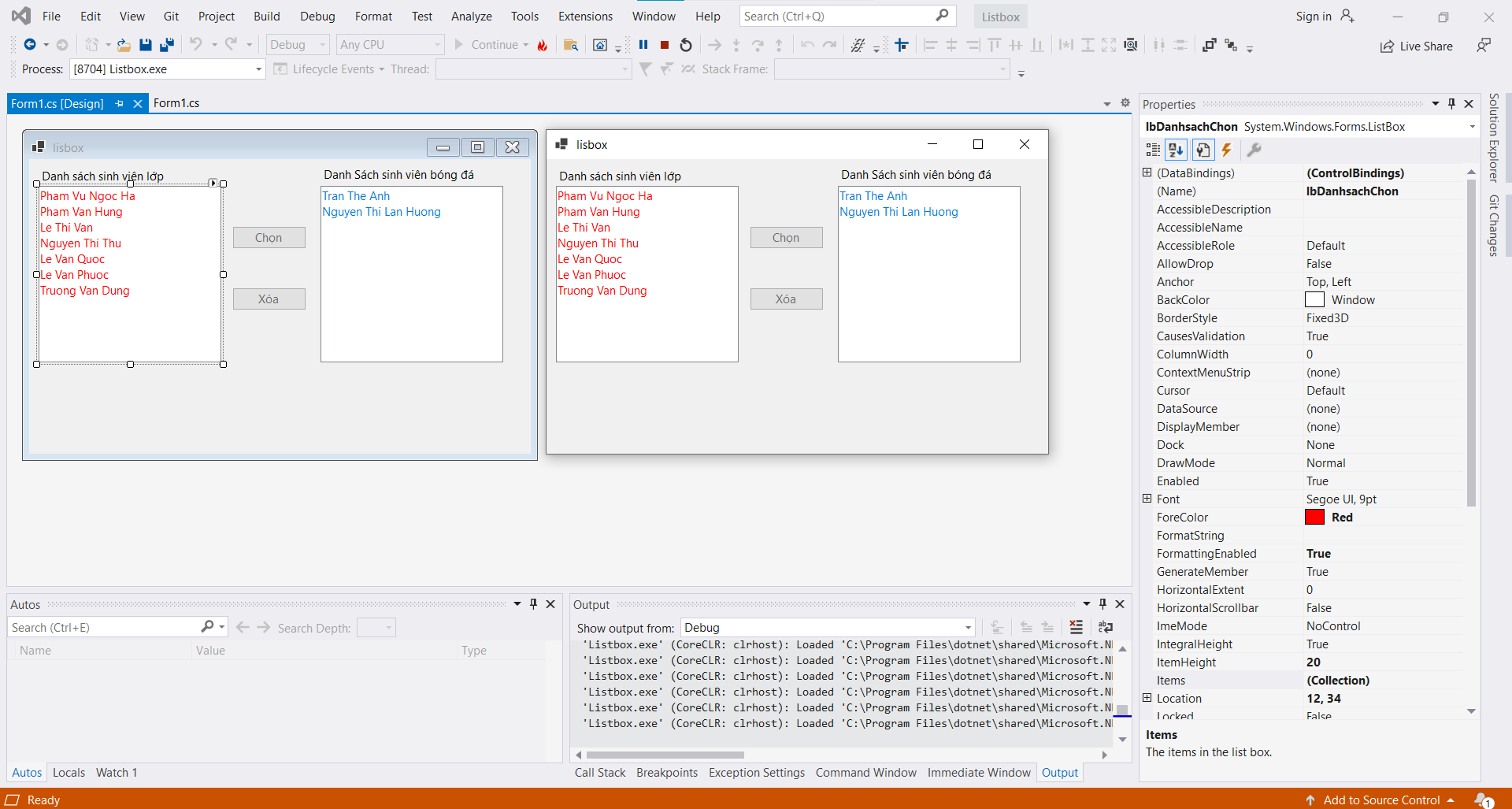
i--;

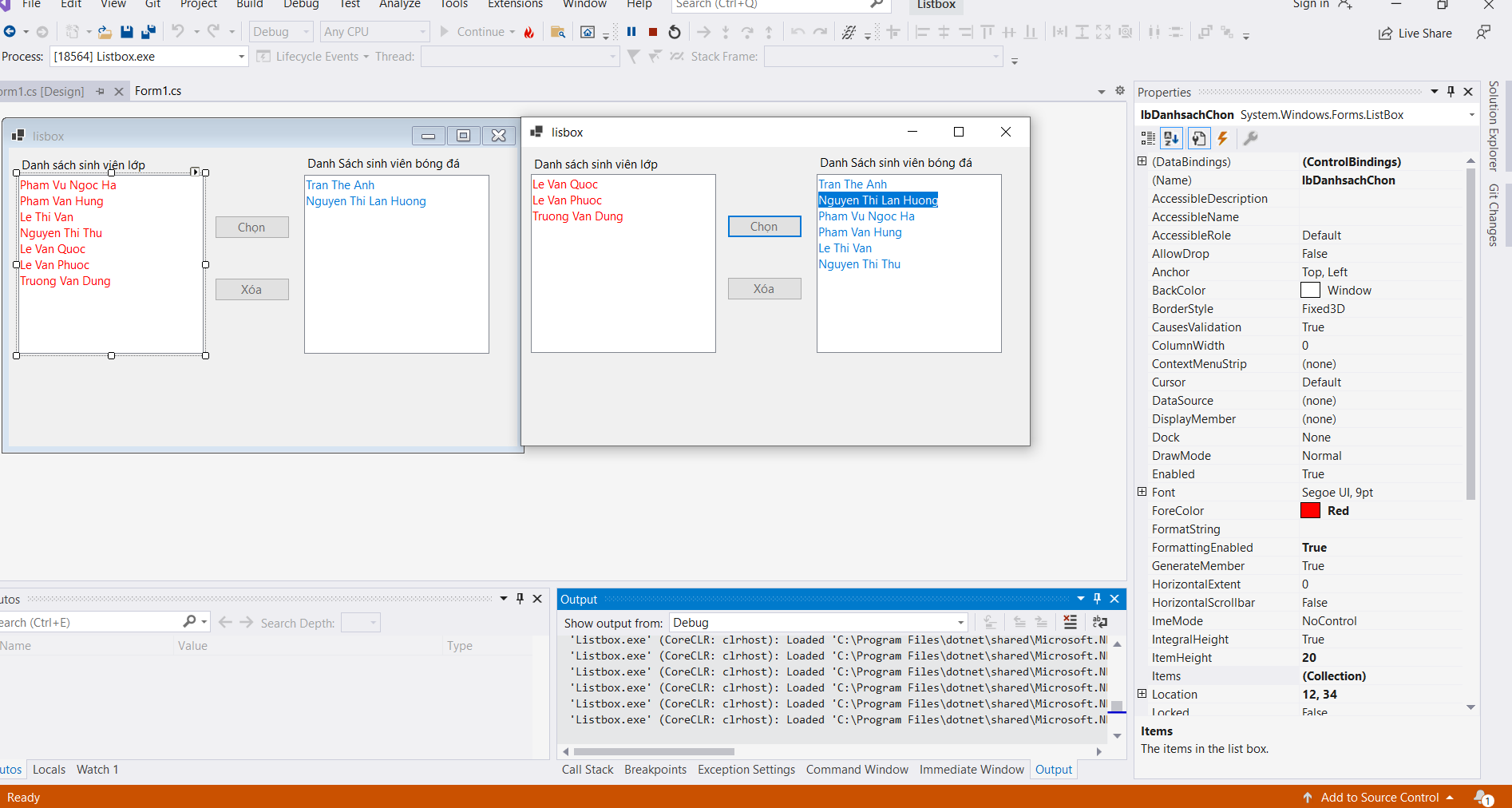
}

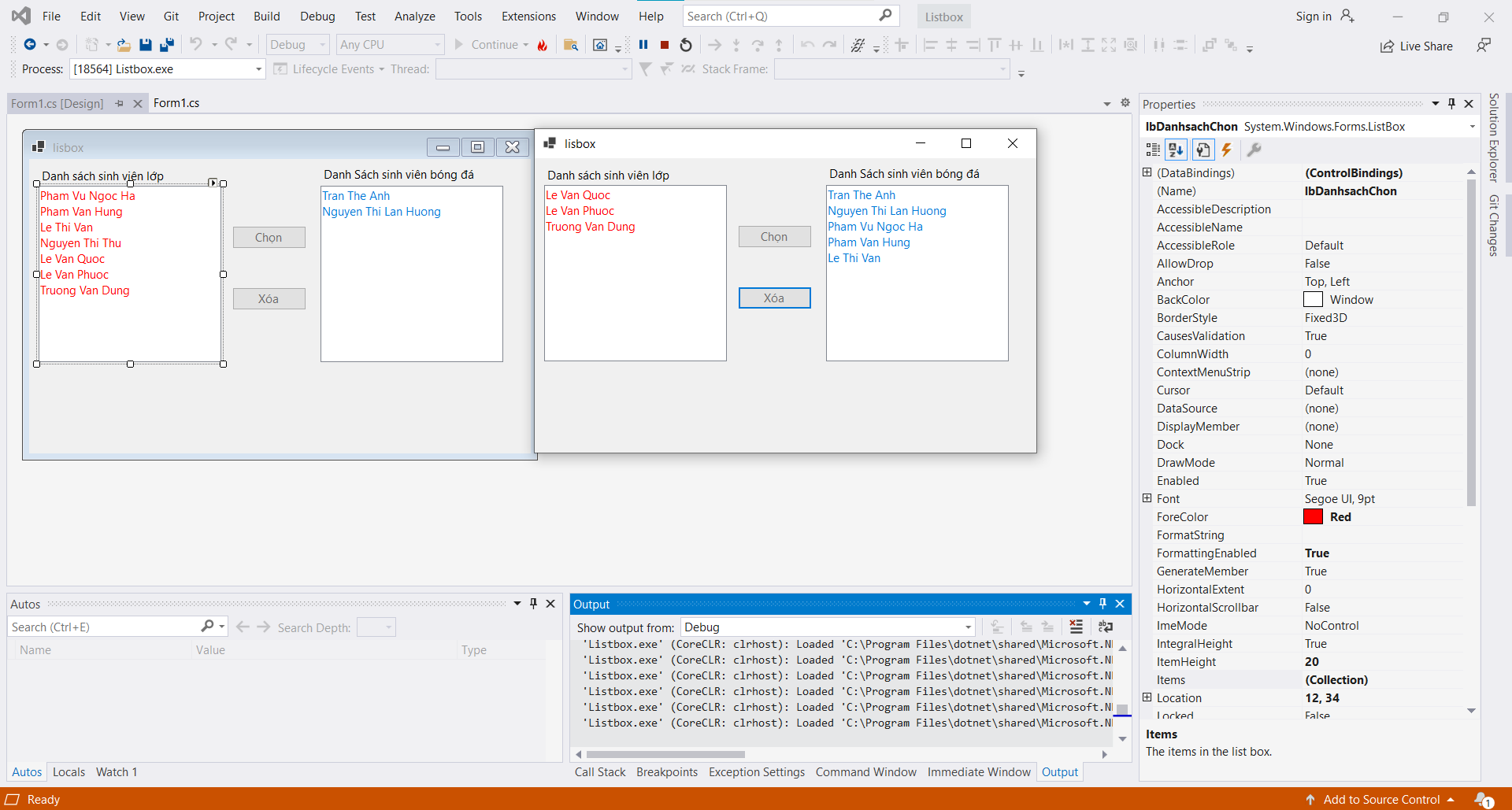
}

}

}







-         Button, checkbox, radio button:

Code:

namespace RadioButton

{

public partial class buttonss : Form

{

public buttonss()

{

InitializeComponent();

}

private void ckbMauChu\_CheckedChanged(object sender, EventArgs e)

{

if (this.ckbMauChu.Checked)

this.btnButton.ForeColor = Color.Red;

else

this.btnButton.ForeColor = Color.Black;

}

private void ckbMauNen\_CheckedChanged(object sender, EventArgs e)

{

if (this.ckbMauNen.Checked)

this.btnButton.BackColor = Color.LightCyan;

else

this.btnButton.BackColor = this.btnButtonFlat.BackColor;

}

private void rbFlat\_CheckedChanged(object sender, EventArgs e)

{

if (this.rbFlat.Checked)

this.btnButtonFlat.FlatStyle = FlatStyle.Flat;

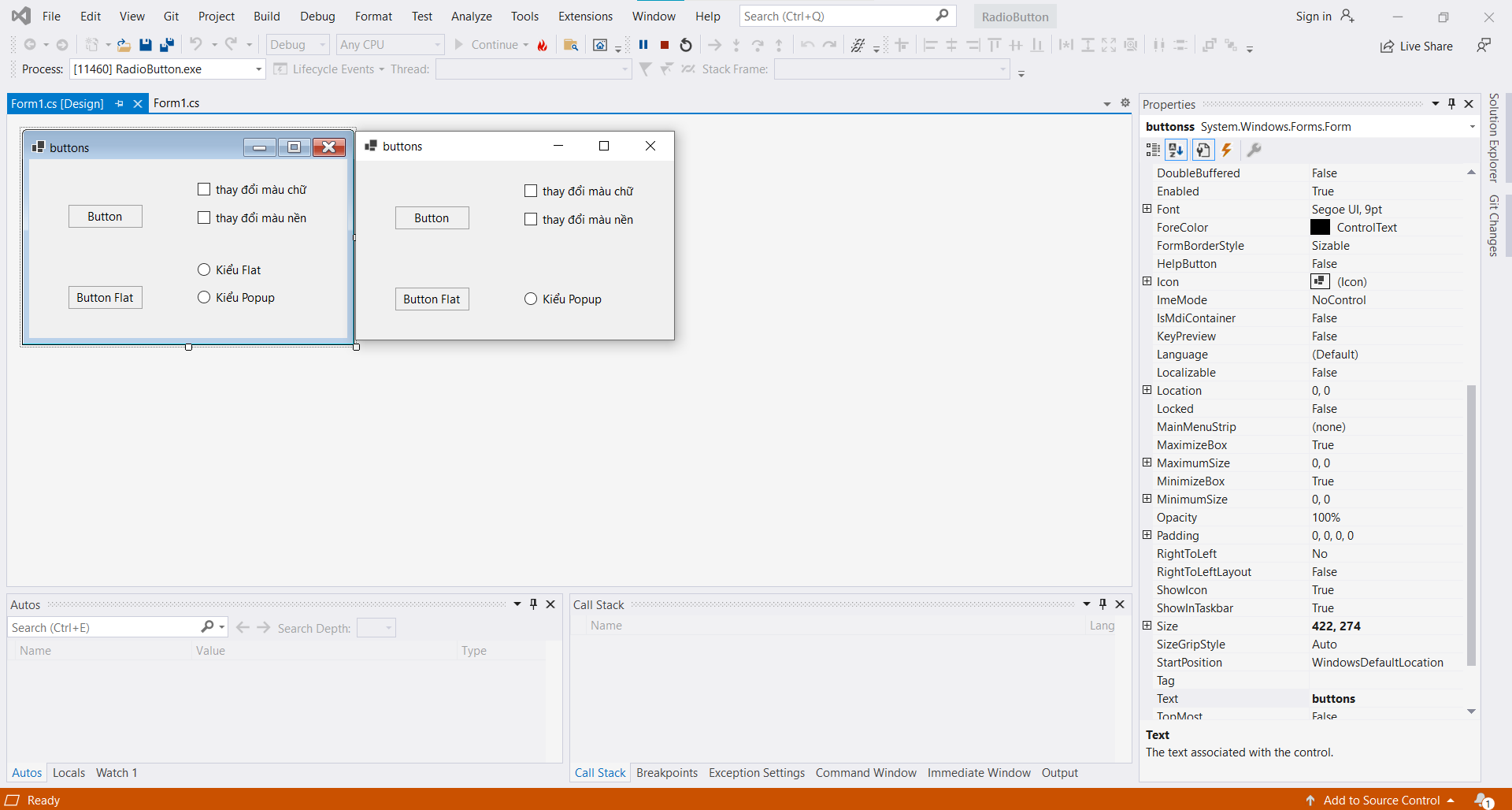
else

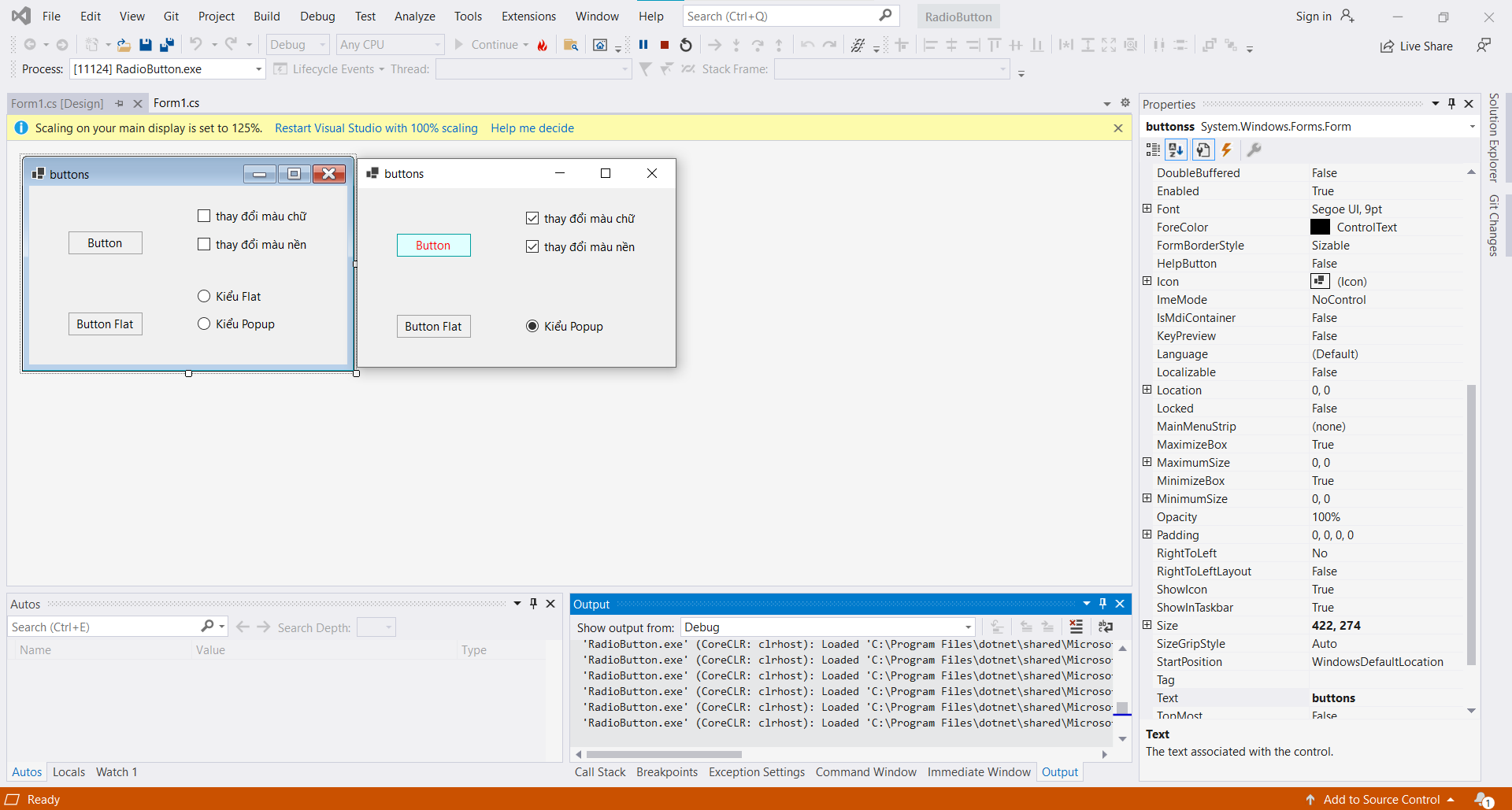
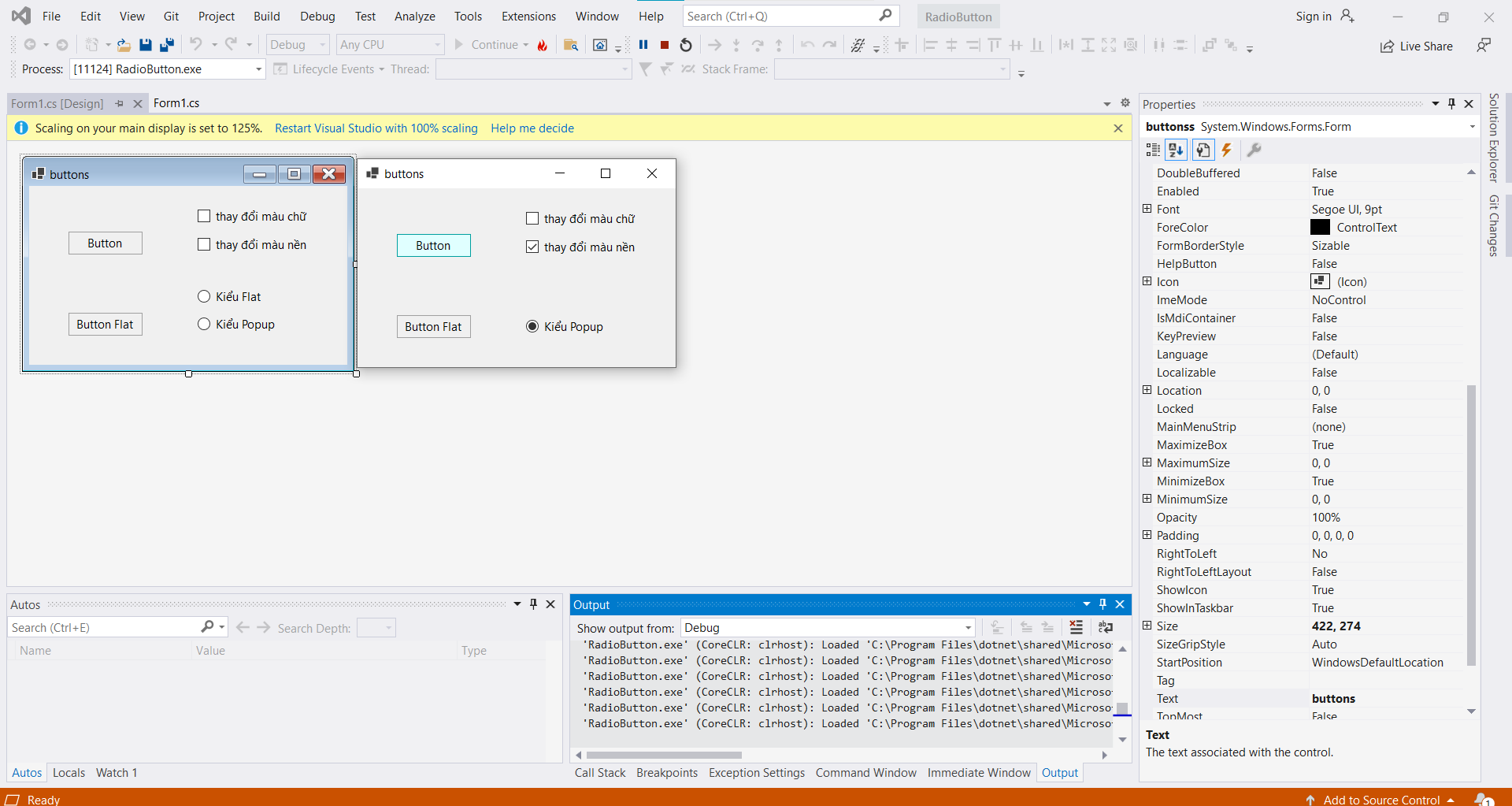
this.btnButtonFlat.FlatStyle = FlatStyle.Popup;

}

}

}





-         Checklistbox:

Code:

namespace CheckListBox

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void clbMonHoc\_SelectedIndexChanged(object sender, EventArgs e)

{

string s = "";

for (int i = 0; i < clbMonHoc.Items.Count; i++)

if (this.clbMonHoc.GetItemChecked(i) == true)

s += this.clbMonHoc.Items[i].ToString() + ",";

MessageBox.Show("Danh sach mon hoc;" + s);

/\*CheckedListBox.CheckedItemCollection items;

items = this.clbMonHoc.CheckedItems;

string s = "";

foreach(object ob in items)

{

s += ob.ToString + ",";

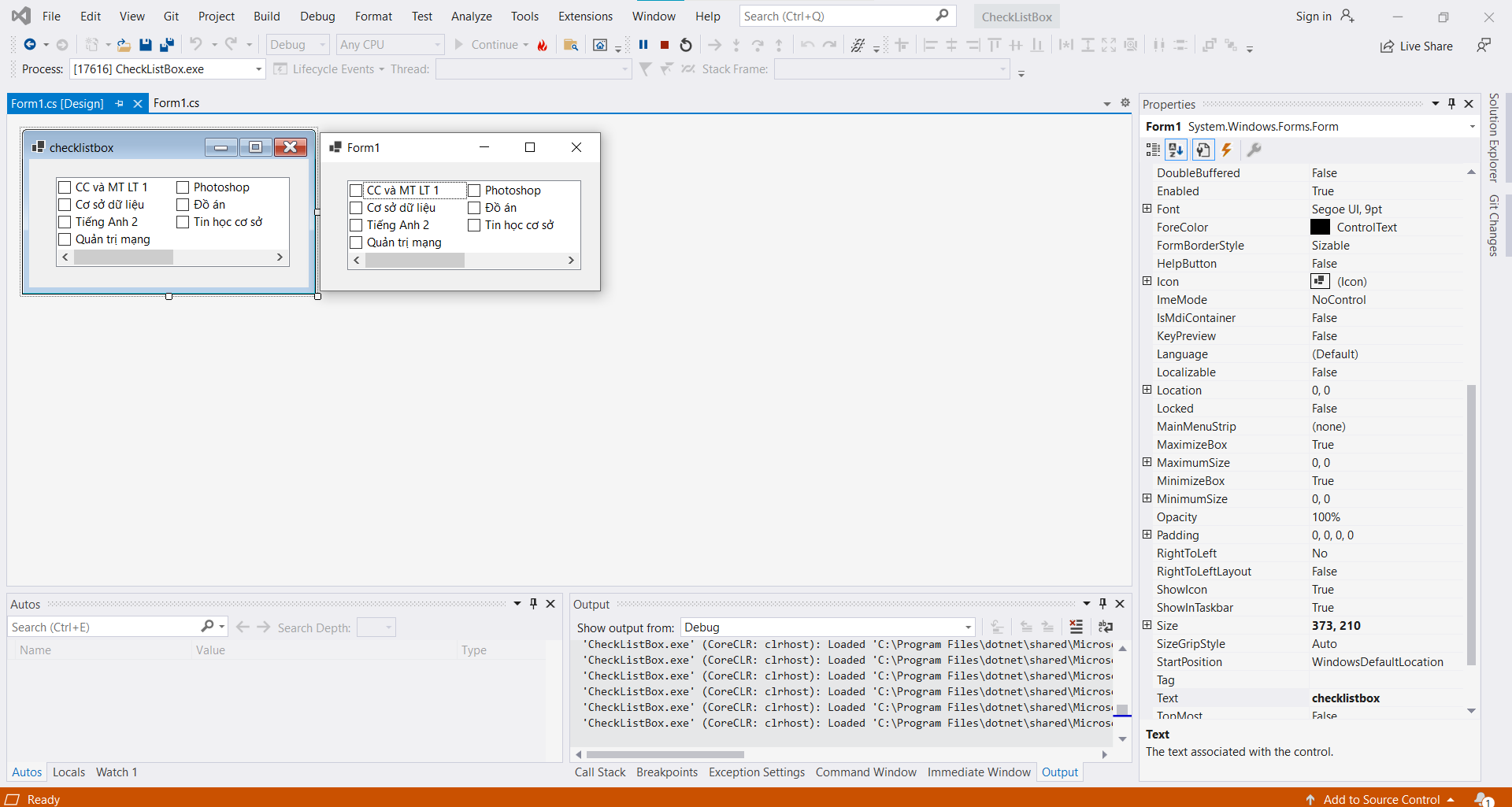
}

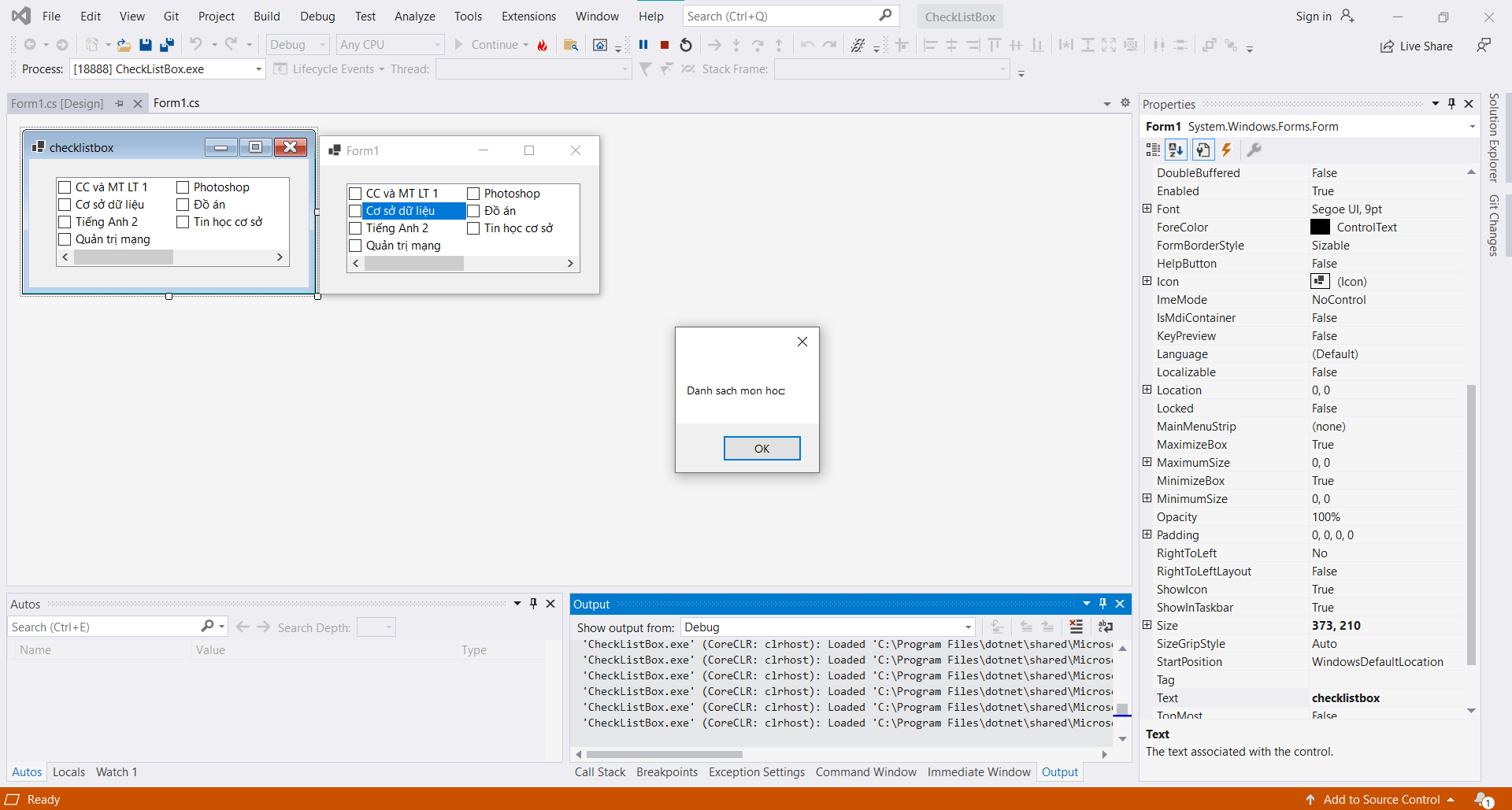
MessageBox.Show("Danh sach mon hoc:" + s);\*/

}

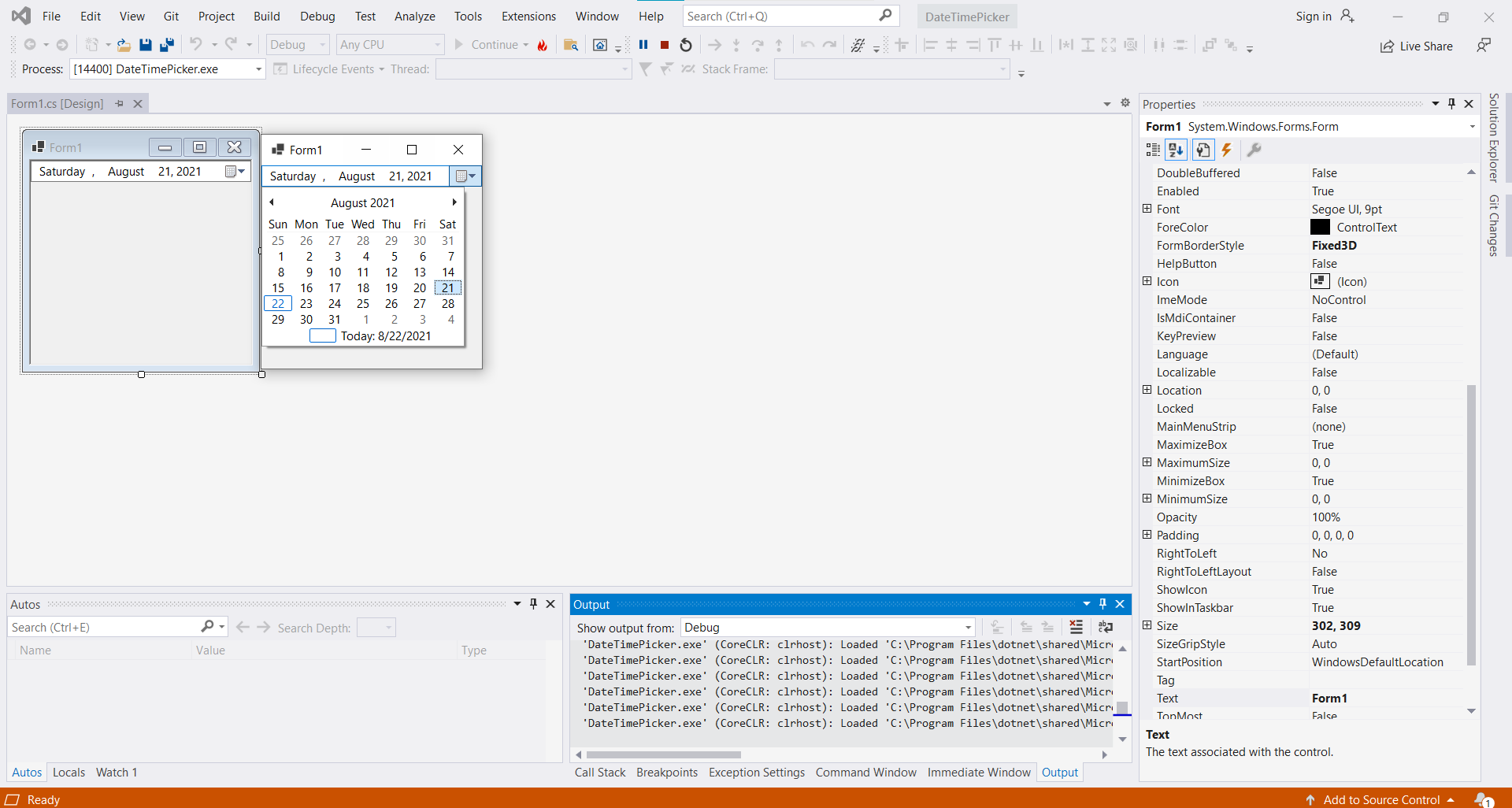
}

}





-         DatetimePicker:



-         Tab control:

Code:

namespace j.Tab\_control

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void tabPage1\_Click(object sender, EventArgs e)

{

}

private void Form1\_Load(object sender, EventArgs e)

{

this.tabControl1.TabPages.Clear();

TabPage tabSV = new TabPage();

TabPage tabGV = new TabPage();

TabPage tabMH = new TabPage();

tabSV.Text = "Sinh viên";

tabGV.Text = "Giáo Viên";

tabMH.Text = "Môn học";

this.tabControl1.TabPages.Add(tabSV);

this.tabControl1.TabPages.Add(tabGV);

this.tabControl1.TabPages.Add(tabMH);

}

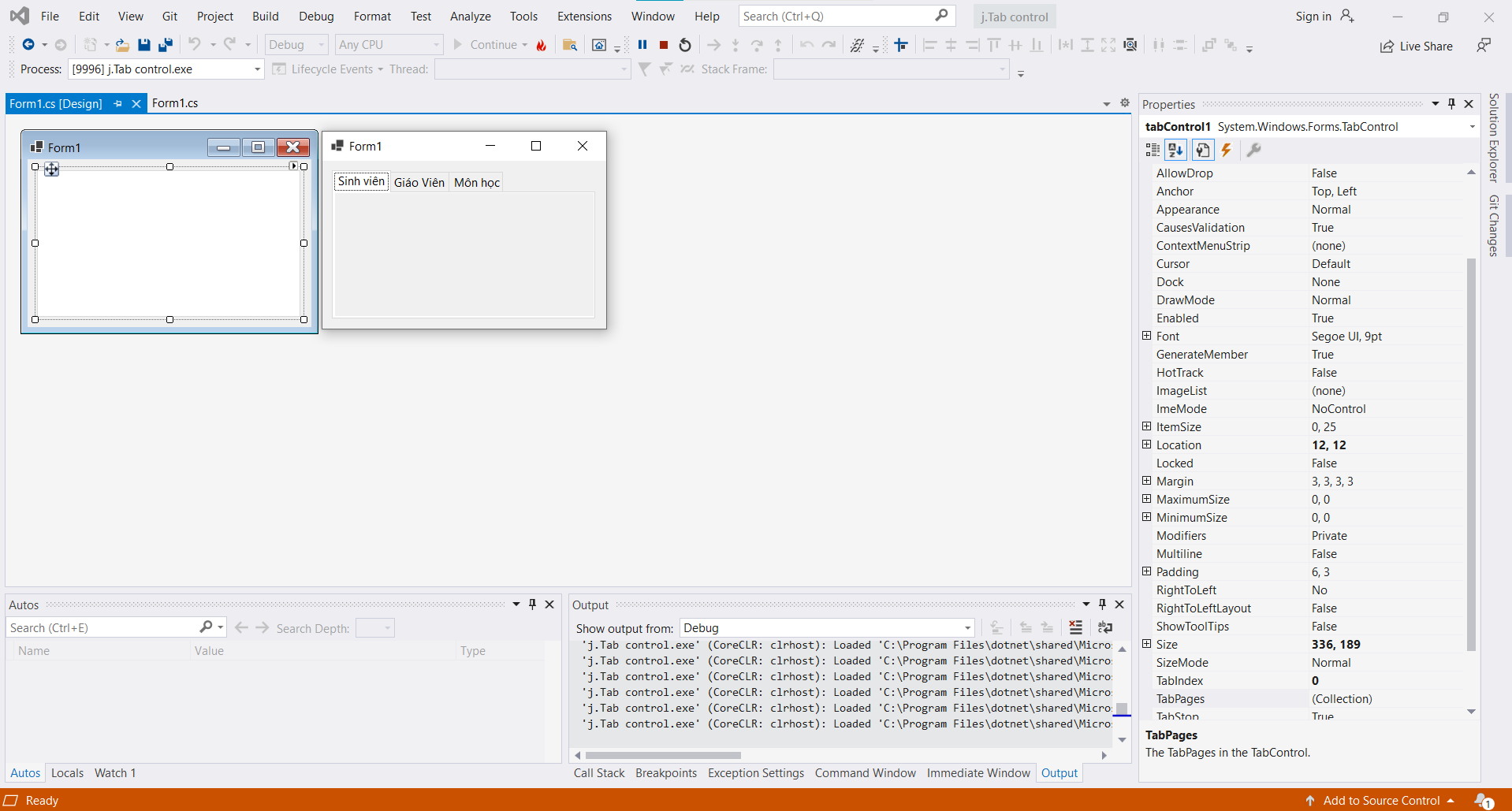
private void label1\_Click(object sender, EventArgs e)

{

}

}

}



-         Picturebox: cho chọn và hiển thị hình

Code:

namespace picturebox

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

for (int i = 1; i <= imageList1.Images.Count; i++)

{

comboBox1.Items.Add("Hình " + i) ;

}

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

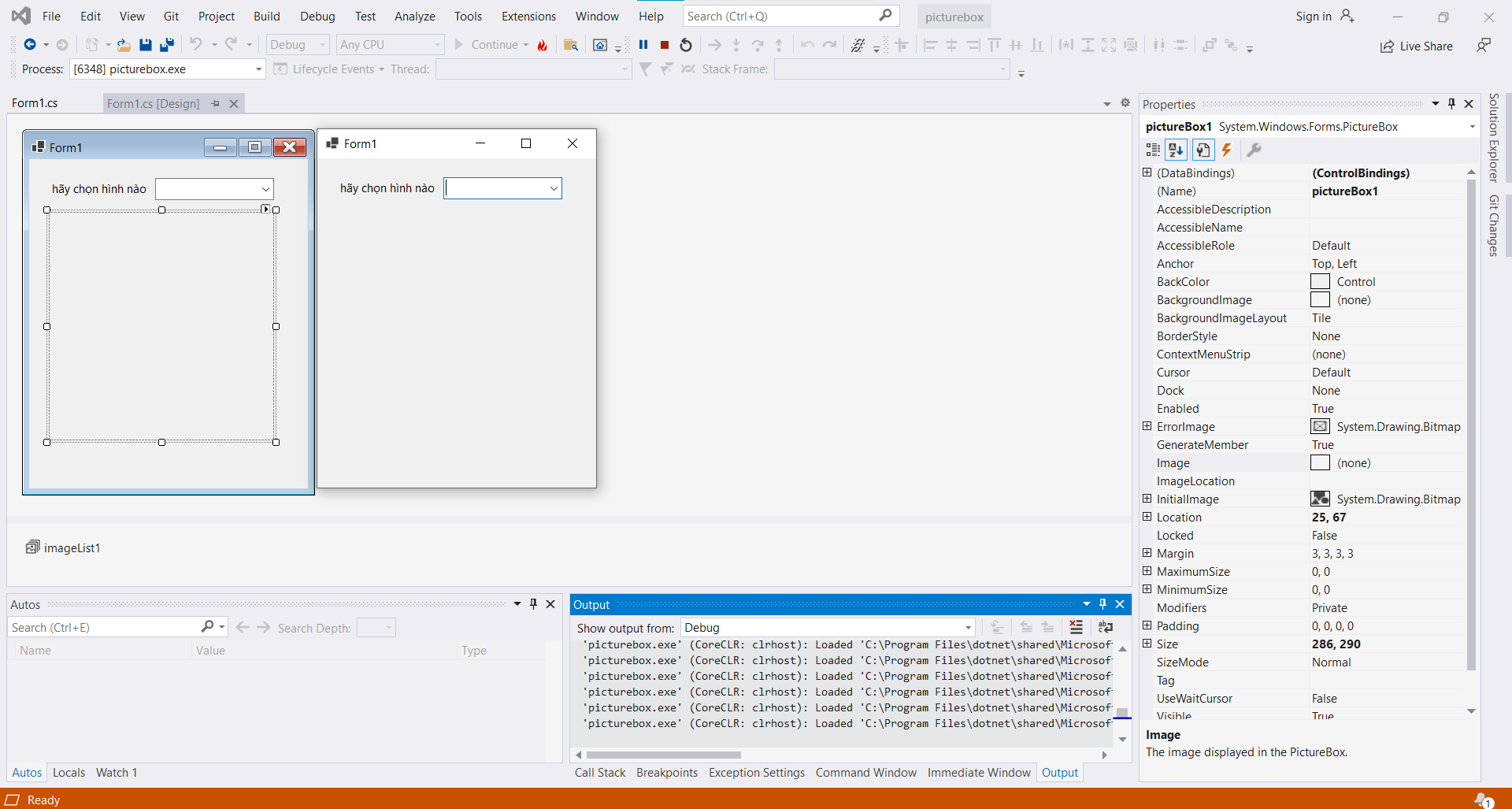
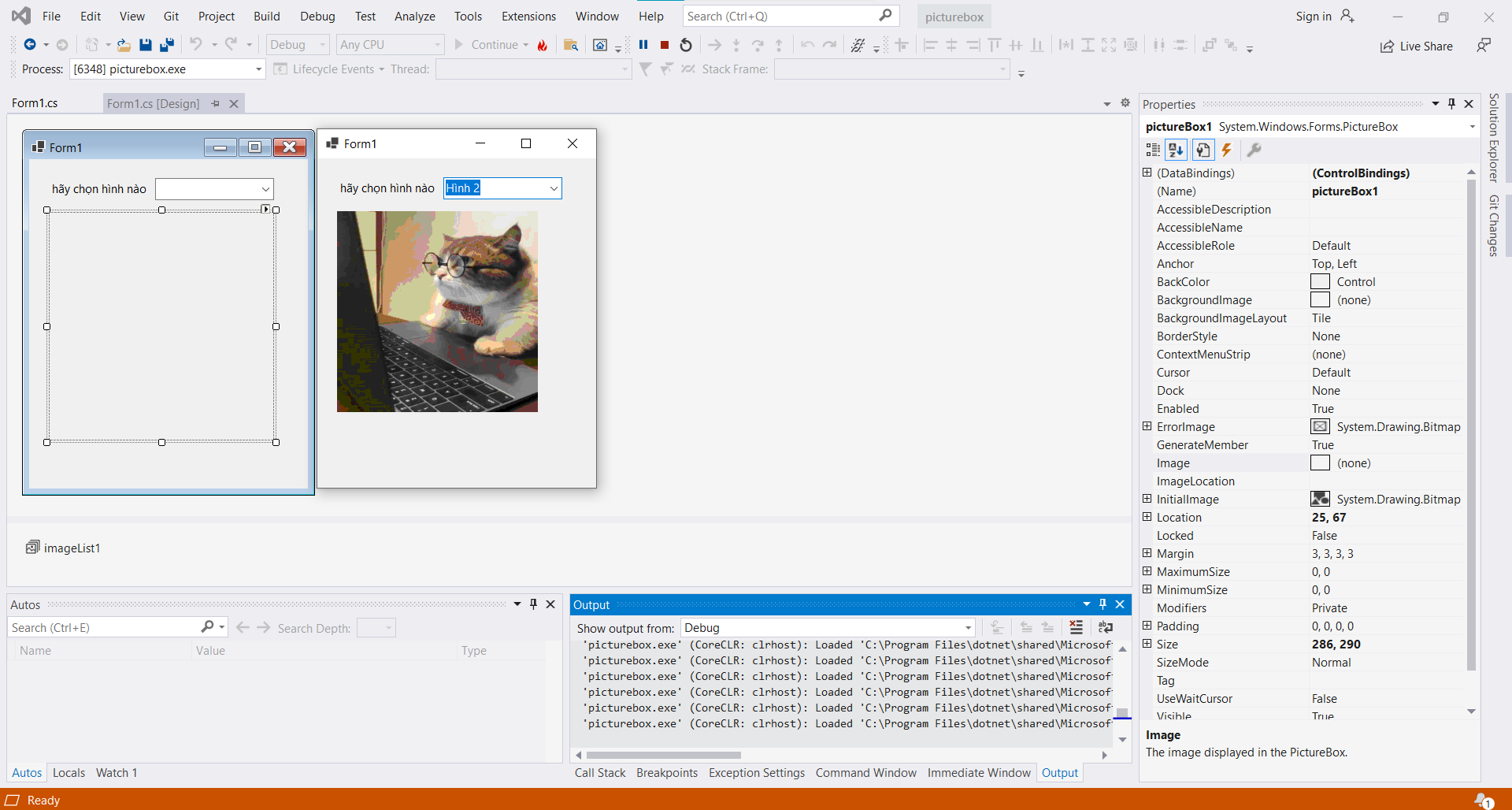
{

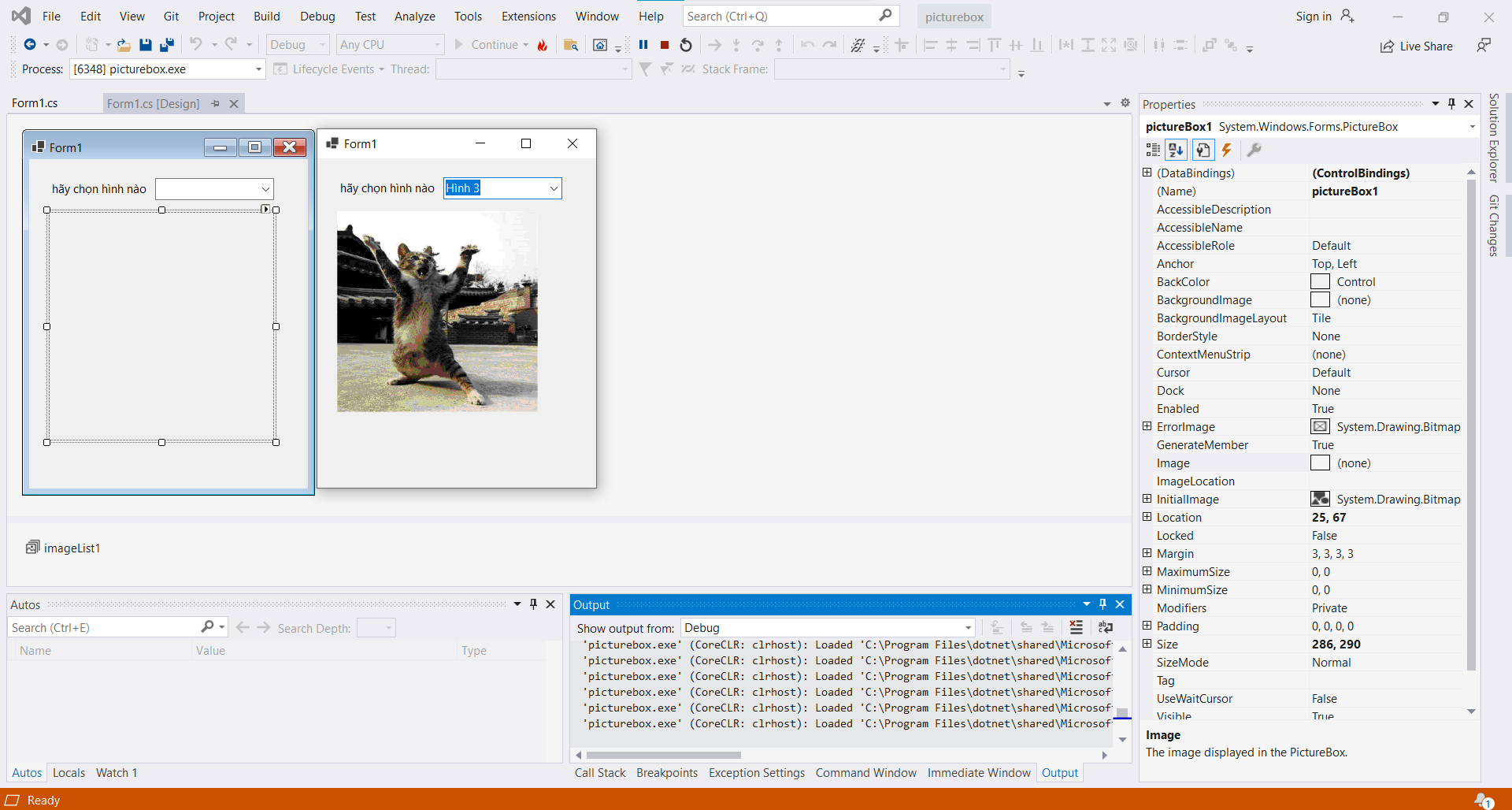
pictureBox1.Image = imageList1.Images[comboBox1.SelectedIndex];

}

}

}



-         Làm lại ví dụ cô demo: form đăng nhập, xử lý đăng nhập, hiển thị main form, xử lý các sự kiện cho main form, listbox, …

Code:

LoginForm:

namespace lam\_lai\_vi\_du\_co\_demo

{

public partial class LoginForm : Form

{

public LoginForm()

{

InitializeComponent();

}

private void label4\_Click(object sender, EventArgs e)

{

}

private void btnLogin\_Click(object sender, EventArgs e)

{

var loginName = txtLoginName.Text;

var password = txtPassword.Text;

if(loginName.CompareTo("a")==0 && password.CompareTo("1")==0)

{

this.DialogResult = DialogResult.OK;

}

else

{

lblMessage.Text = "Sai tên đăng nhập hoặc mật khẩu.";

}

}

private void btnCancel\_Click(object sender, EventArgs e)

{

Application.Exit();

}

}

}

Form1:

namespace lam\_lai\_vi\_du\_co\_demo

{

public partial class txt : Form

{

public txt()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

LoginForm login = new LoginForm();

if(login.ShowDialog() != DialogResult.OK)

{

Application.Exit();

}

label1.Text = "Đã đăng nhập";

}

private void label3\_Click(object sender, EventArgs e)

{

}

private void btnAdd\_Click(object sender, EventArgs e)

{

if (MessageBox.Show("Bạn có thật sự muốn thêm sinh viên vào danh sách hay không?", "Thông báo", MessageBoxButtons.YesNo,MessageBoxIcon.Question)==DialogResult.Yes)

{

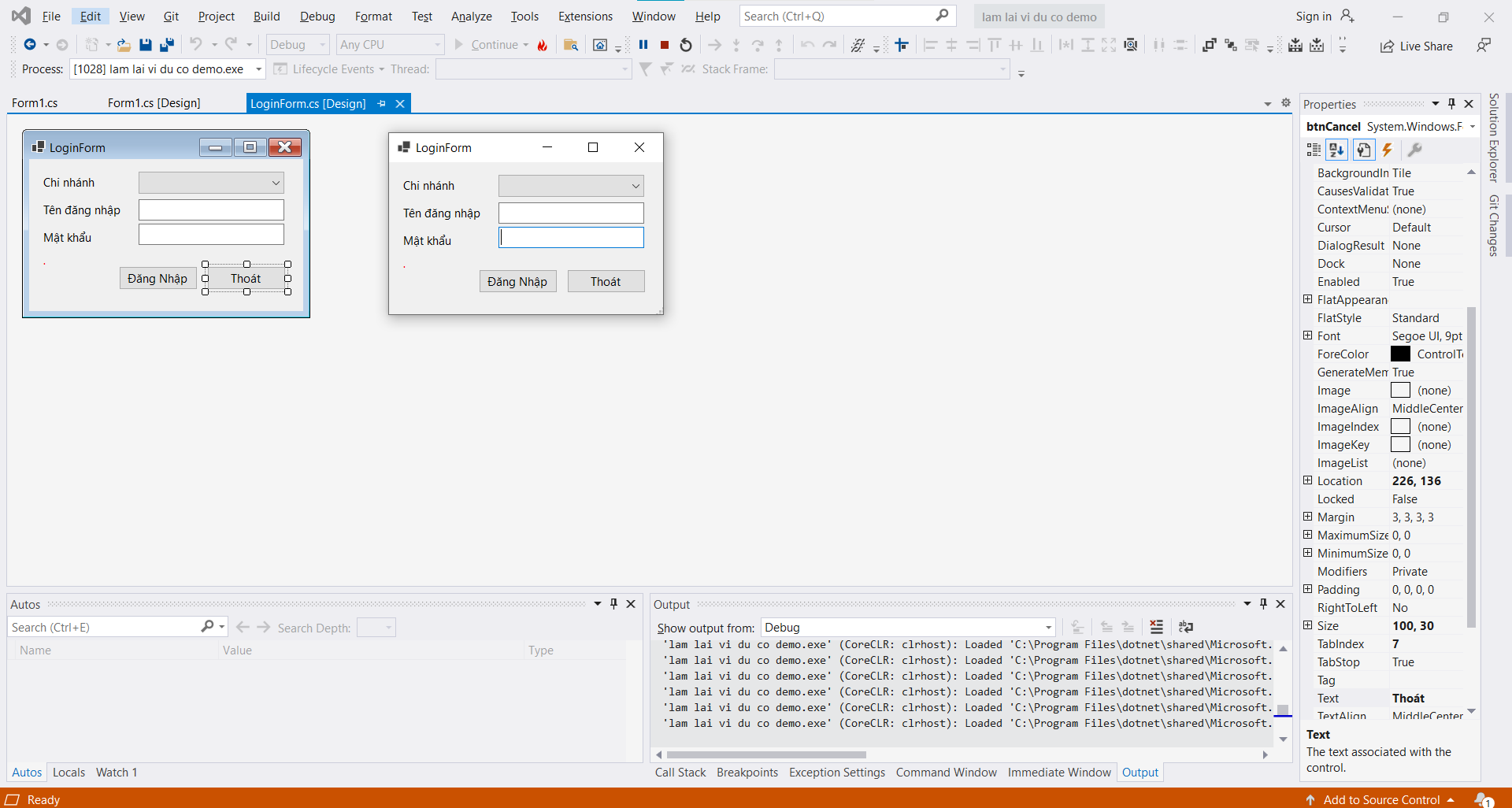
lbDanhSachSV.Items.Add(txtFullName.Text);

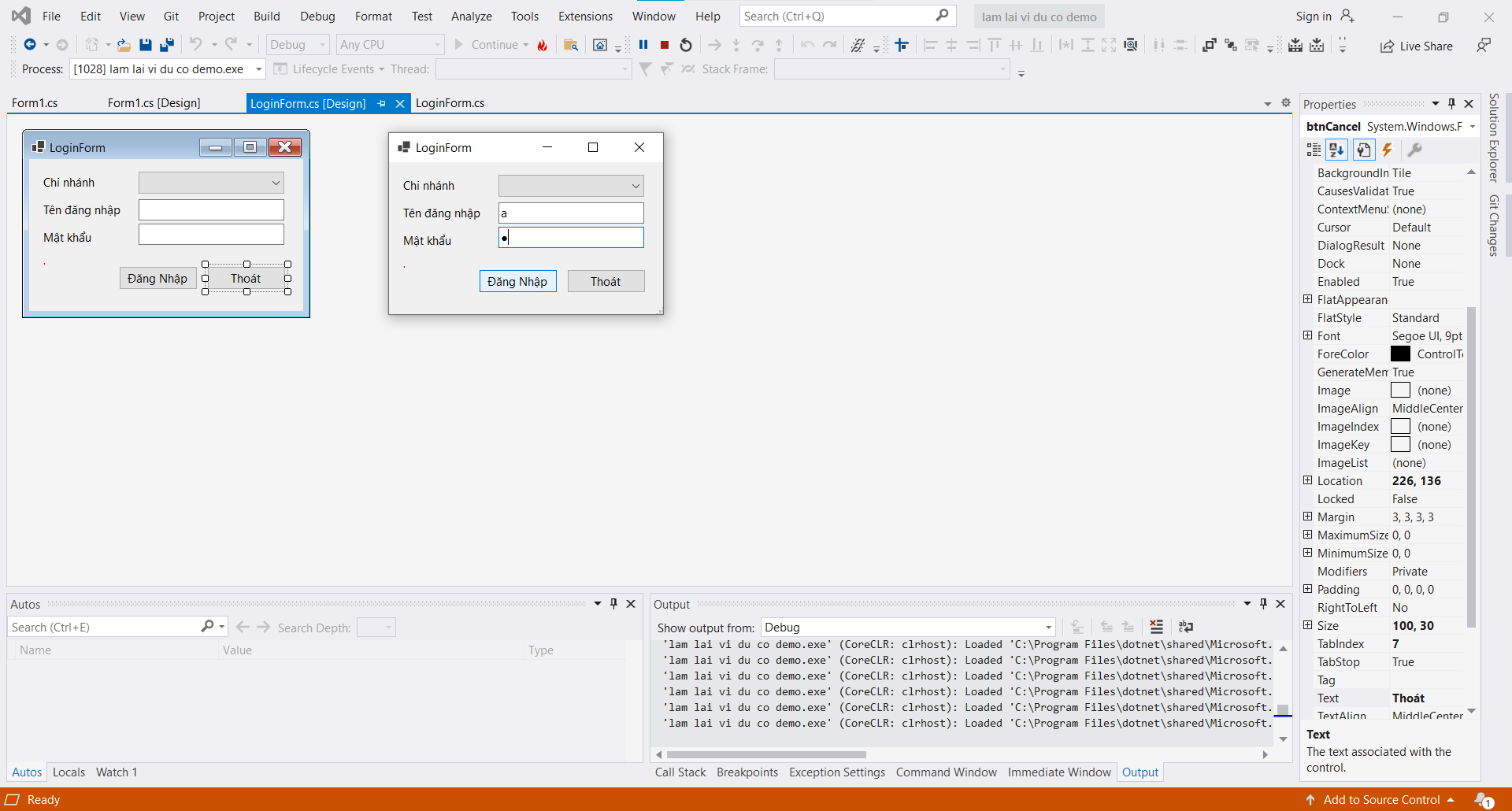
}

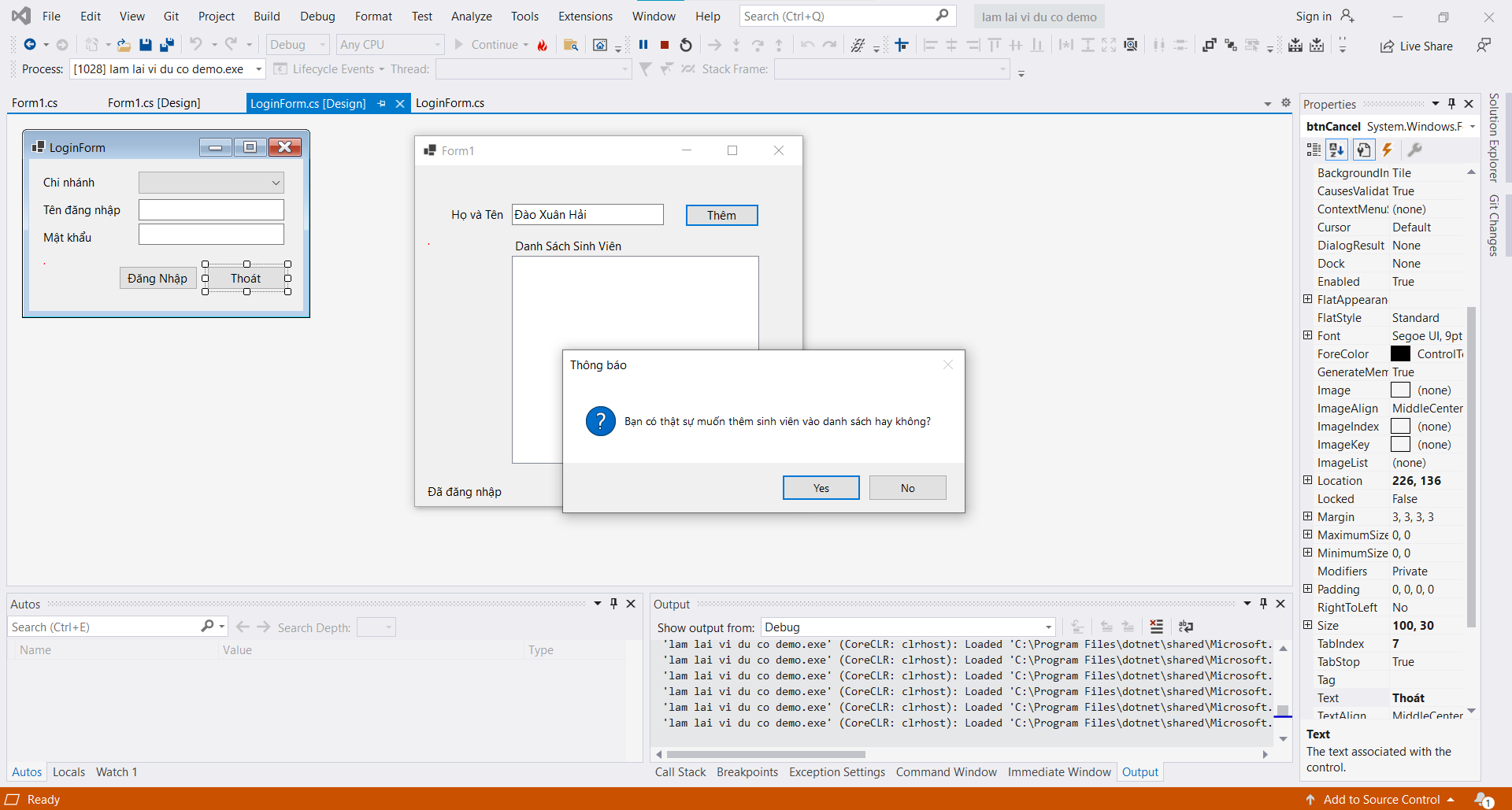
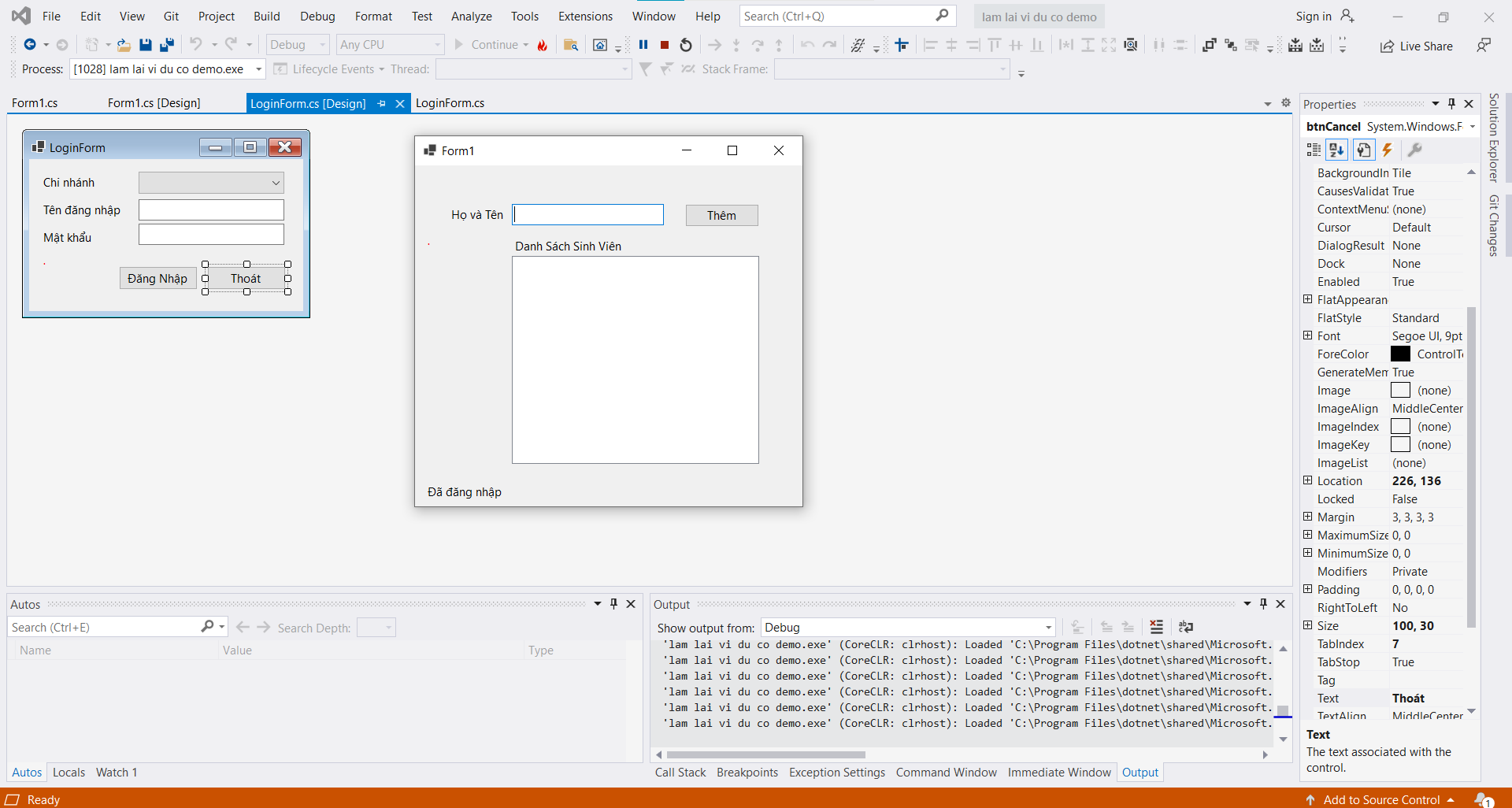
}

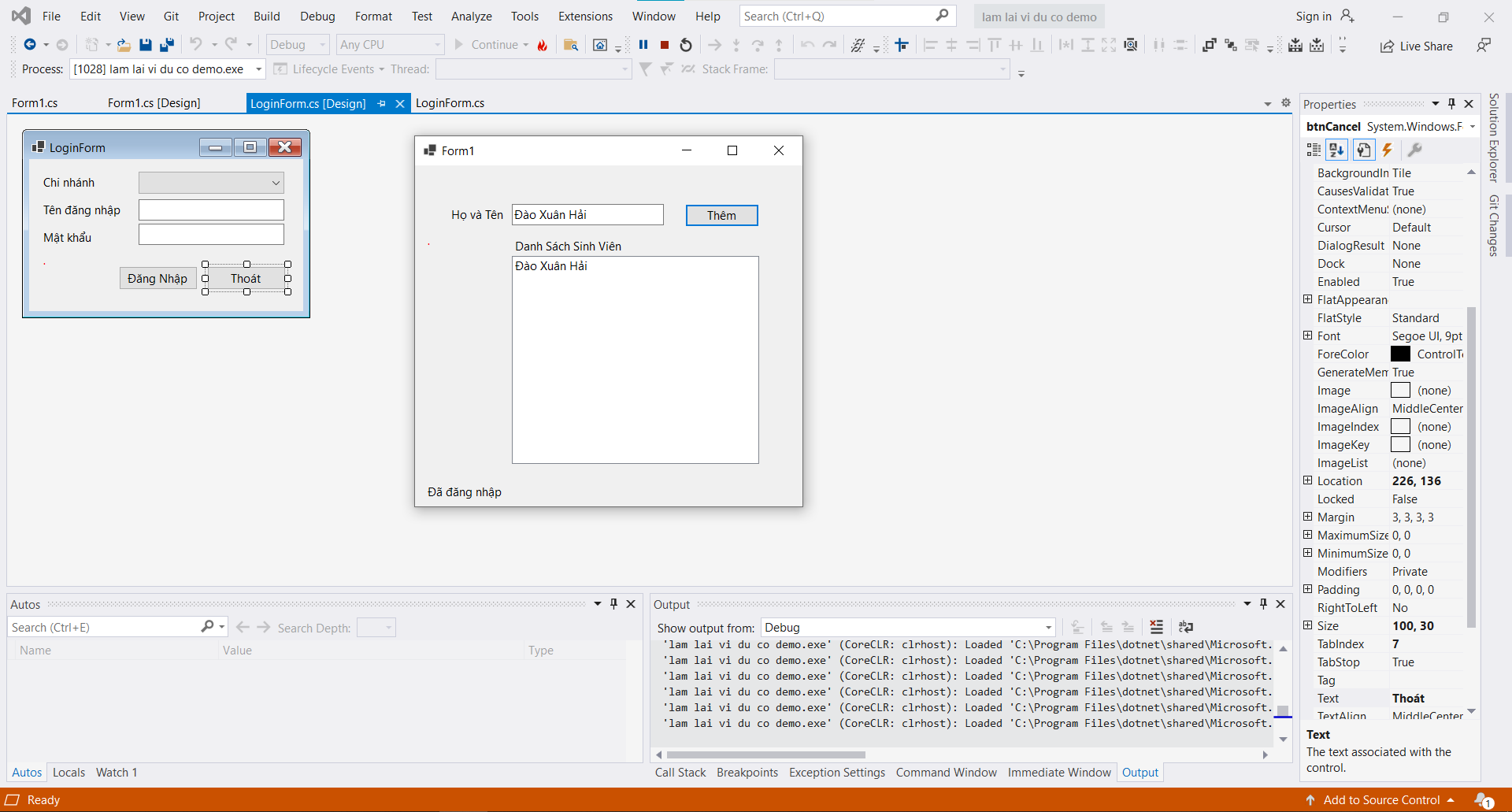
}

}









**2. Bài tập làm thêm từ trang functionX:**

-         [Boolean Algebra](http://functionx.com/vcsharp2010/controls/booleanalgebra.htm)

Code:

namespace BooleanAlgebra1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void chkOperand1\_CheckedChanged(object sender, EventArgs e)

{

if (chkOperand1.Checked == true)

chkOperand1.Text = "True";

else

chkOperand1.Text = "False";

}

private void chkOperand2\_CheckedChanged(object sender, EventArgs e)

{

if (chkOperand2.Checked == true)

chkOperand2.Text = "True";

else

chkOperand2.Text = "False";

}

private void chkResult\_CheckedChanged(object sender, EventArgs e)

{

if (chkResult.Checked == true)

chkResult.Text = "True";

else

chkResult.Text = "False";

}

private void btnNewOperation\_Click(object sender, EventArgs e)

{

string[] strBooleanValues = { "True", "False" };

string[] strOperations = { "A ^ B", "A V B" };

Random rnd = new Random();

chkOperand1.Text = strBooleanValues[rnd.Next(2)];

chkOperand2.Text = strBooleanValues[rnd.Next(2)];

if (chkOperand1.Text == "True")

chkOperand1.Checked = true;

else

chkOperand1.Checked = false;

if (chkOperand2.Text == "True")

chkOperand2.Checked = true;

else

chkOperand2.Checked = false;

lblOperation.Text = strOperations[rnd.Next(2)];

}

private void btnCheckOperation\_Click(object sender, EventArgs e)

{

if (lblOperation.Text == "A ^ B")

{

if (chkOperand1.Checked == true)

{

if (chkOperand2.Checked == true)

{

if (chkResult.Checked == true)

{

MessageBox.Show("Bravo - Good Answer");

}

else if (chkResult.Checked == false)

{

MessageBox.Show("Wrong - Get it right next time!");

}

}

else if (chkOperand2.Checked == false)

{

if (chkResult.Checked == false)

{

MessageBox.Show("Bravo - Good Answer");

}

else if (chkResult.Checked == true)

{

MessageBox.Show("Wrong - Get it right next time!");

}

}

}

else if (chkOperand1.Checked == false)

{

if (chkOperand2.Checked == true)

{

if (chkResult.Checked == false)

{

MessageBox.Show("Bravo - Good Answer");

}

else if (chkResult.Checked == true)

{

MessageBox.Show("Wrong - Get it right next time!");

}

}

else if (chkOperand2.Checked == false)

{

if (chkResult.Checked == false)

{

MessageBox.Show("Bravo - Good Answer");

}

else if (chkResult.Checked == true)

{

MessageBox.Show("Wrong - Get it right next time!");

}

}

}

}

else if (lblOperation.Text == "A V B") // Logical Disjunction:

{

if (chkOperand1.Checked == true)

{

if (chkOperand2.Checked == true)

{

if (chkResult.Checked == true)

{

MessageBox.Show("Bravo - Good Answer");

}

else if (chkResult.Checked == false)

{

MessageBox.Show("Wrong - Get it right next time!");

}

}

else if (chkOperand2.Checked == false)

{

if (chkResult.Checked == true)

{

MessageBox.Show("Bravo - Good Answer");

}

else if (chkResult.Checked == false)

{

MessageBox.Show("Wrong - Get it right next time!");

}

}

}

else if (chkOperand1.Checked == false)

{

if (chkOperand2.Checked == true)

{

if (chkResult.Checked == true)

{

MessageBox.Show("Bravo - Good Answer");

}

else if (chkResult.Checked == false)

{

MessageBox.Show("Wrong - Get it right next time!");

}

}

else if (chkOperand2.Checked == false)

{

if (chkResult.Checked == false)

{

MessageBox.Show("Bravo - Good Answer");

}

else if (chkResult.Checked == true)

{

MessageBox.Show("Wrong - Get it right next time!");

}

}

}

}

}

private void btnClose\_Click(object sender, EventArgs e)

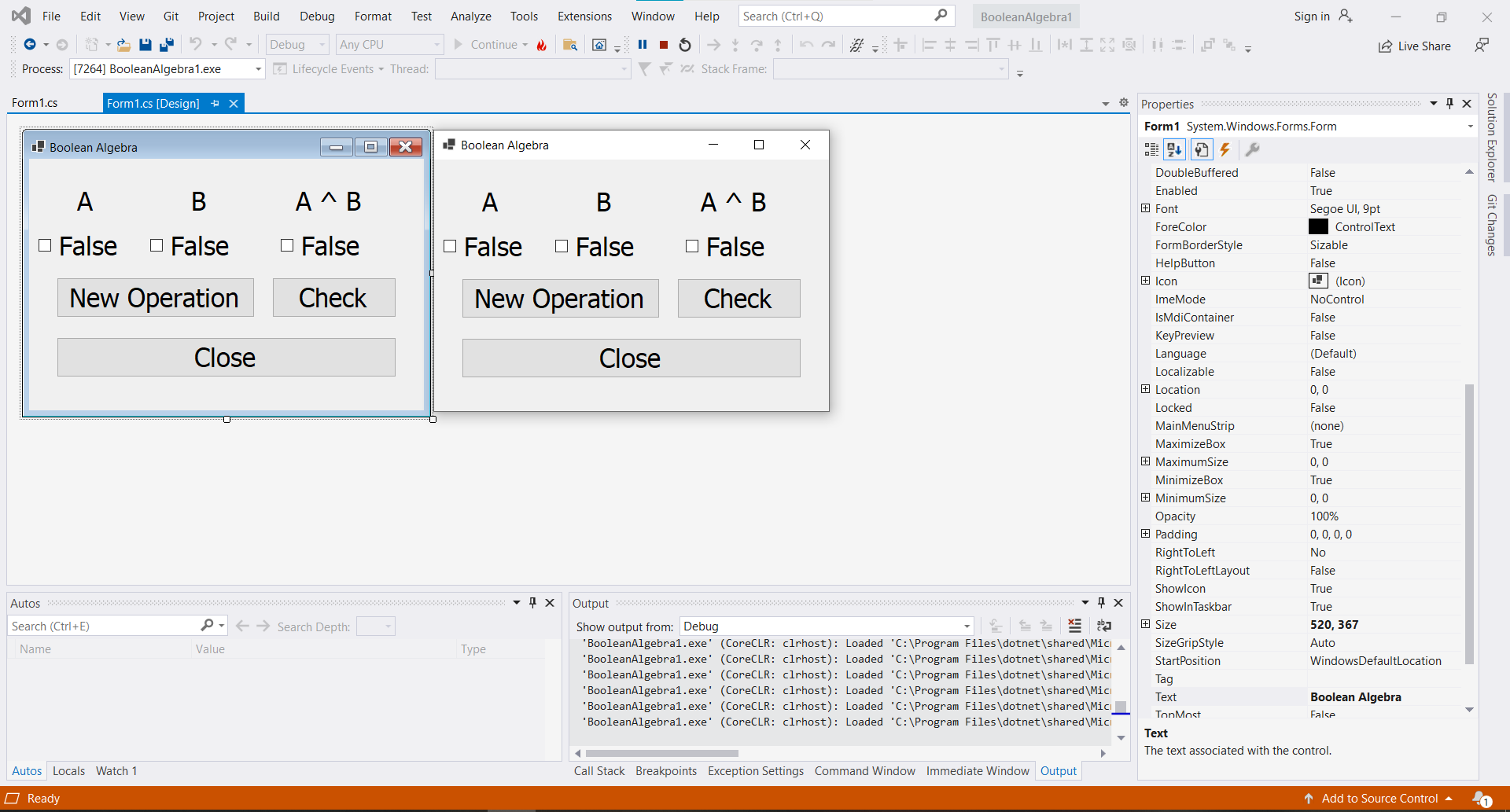
{

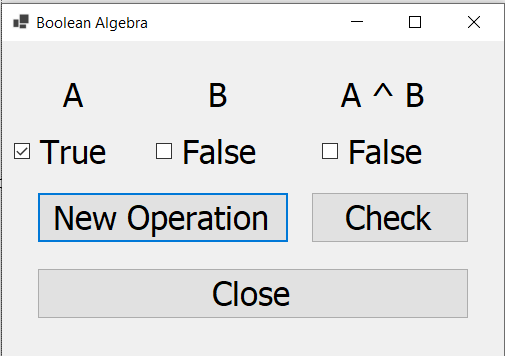
Close();

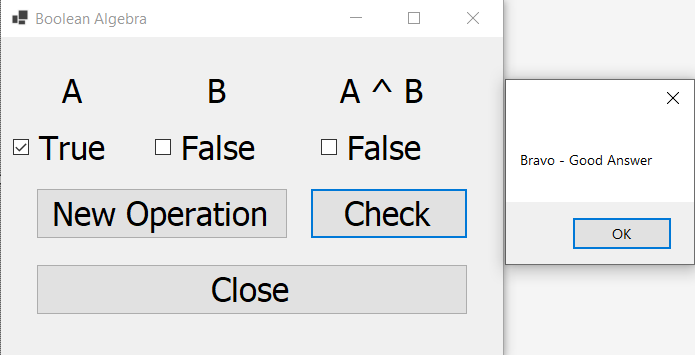
}

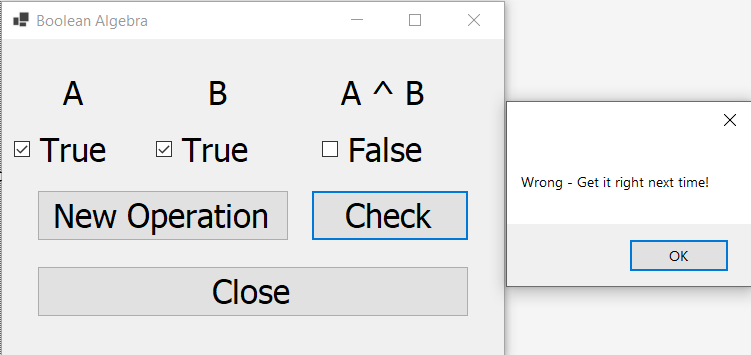
}

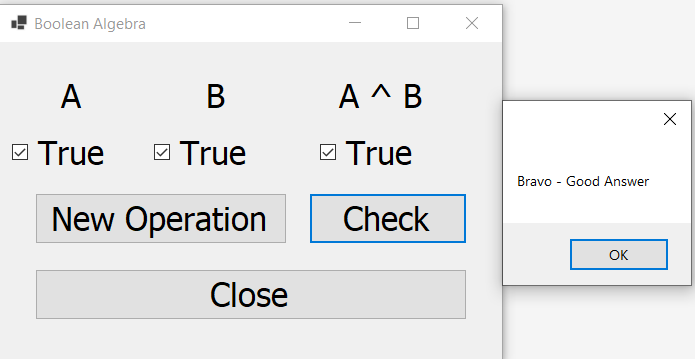
}











-         [Compounded Interest](http://functionx.com/vcsharp2010/controls/compoundinterest.htm)

namespace CompoundedInterest

{

public partial class txtInterestRate : Form

{

public txtInterestRate()

{

InitializeComponent();

}

private void btnClose\_Click(object sender, EventArgs e)

{

Close();

}

private void btnCalculate\_Click(object sender, EventArgs e)

{

double principal= 0.00,

interestRate = 0.00,

interestEarned = 0.00;

double futureValue = 0.00,

ratePerPeriod = 0.00,

periods = 0;

int compoundType = 0;

principal = double.Parse(txtPrincipal.Text);

interestRate = double.Parse(txt.Text) / 100;

if (rdoMonthly.Checked)

compoundType = 12;

else if (rdoQuarterly.Checked)

compoundType = 4;

else if (rdoSemiannually.Checked)

compoundType = 2;

else

compoundType = 1;

periods = double.Parse(txtPeriods.Text);

double i = interestRate / compoundType;

double n = compoundType \* periods;

ratePerPeriod = interestRate / periods;

futureValue = principal \* Math.Pow(1 + i, n);

interestEarned = futureValue - principal;

txtInterestEarned.Text = interestEarned.ToString("C");

txtFutureValue.Text = futureValue.ToString("C");

}

}

}

