Lesson 2 form property & lesson 3 showing messagebox & lesson 4 variables

Toolbox 里面有具体的项目，property 可以改颜色字体

在property 上 有个lightening button, 是可以改变它的event，

从messagebox 打印出东西来：

int a = 10;

MessageBox.Show(a.ToString());

Object 可以fold anything, object myObj = true;

Lesson 5 change property:

button2.Text = "hover"; 用button的 design name

button1.Enabled = false; 会把button disable当点击后

button1.height = 60; 改变button的高度

textBox1.Text = “Adam” 通过button 改变textbox 的内容

textBox1.MaxLength = 2; 改变textBox 的 string 的长度

lesson 6 If statement

lesson 7 more if statement

检查checkbox 是否被check

if (checkBox1.Checked == true) {

MessageBox.Show("THe checkBox is checked");

}

Lesson 9 switch statement

Switch(textBox1.text)

{

Case “Adam”

MessageBox.show(“Hello”);

break;

Case “bob”

MessageBox.show(“yo”);

break;

Default

MessageBox.show(“yoo”)’

break;

}

Lesson 10 mathmatic operation

Int a = 5;

Int b = 3;

MessageBox.show((a-b).ToString());

A++; a--;

Lesson 11 array

String [] Names = {“Adam”,”Bob”,”Joe”};

String [] Names = new string [3];

Lesson 12 List

Do not need to speciliaze how many elements inside the list, 但是array中 需要说明多少个element

List<object> Names = new List<object>();

Names.Add(1);

Lesson 13 for and foreach loop

Names.Add("admas");

Names.Add("yii");

foreach (string s in Names)

{

MessageBox.Show(s);

}

Lesson 14 while and do while loop

Lesson 15 Try, Catch and Finally

try

{

string[] names = new string[2];

string s = names[2];

}

catch {

MessageBox.Show("There was a error");

}

Finally: no matter we got error in try catch or not, we will

try

{

string[] names = new string[2];

string s = names[2];

}

catch (Exception ex)

{

MessageBox.Show("There was a error " + ex.Message);

}

finally {

MessageBox.Show("your code is done");

}

Lesson 16, 17 Methods

Method is any blocks of codes that you want to use more than once

比如function

void Message(string message, string title) {

MessageBox.Show(message,title);

}

Messagebox.Show(Message(“adams”));

void Message(string name) {

return name;

}

Lesson 18 continue and break

Lesson 19 Namespace and classes

Namespace is used for organize class, struct and interface

Can create a namespace inside a namespace

Namespace myNamespace{

Namespace space1{

}

}

using Mynamespace; 加上 别的cs file里面当你想用这个namespace

form 自带的namespace，using System.Windows.Forms;

windows是在system 里面的namespace, Forms 是在system 里面的namespace

System.Windows.Forms.MessageBox.show()

Lesson 20 Constructors

namespace Mynamespace

{

class Class1

{

string Name;

public Class1(string name){

Name = name;

}

}

}

Constructor can has multiple constructors

Lesson 21 Access Modifier and static

private string name() {

return Name;

}

如果不加private 或者public default 是public

Static methods is to access without create the class

public static void showMessage(string message) {

System.Windows.Forms.MessageBox.Show("static method");

}

Static is called modifier and public/private is called as access modifier

Lesson 22 Overloading Methods and Enumerations

public static void showMessage(string message) {

System.Windows.Forms.MessageBox.Show("static method");

}

public static void showMessage(int message) {

System.Windows.Forms.MessageBox.Show(message.ToString());

}

根据parameter 的不同，overload function

enum Names1

{

Adam = 1,

Joe = 4,//similar to array, adam is 0, Joe = 0, 可以改变他们的值，next element 总比前一个大一

Bob,

}

Names1 myName = Names1.Adam; // My names hold adam which is default 0

可以改变enum的数值 成为bytes

enum Names1：byte

{

Adam,

Joe,

Bob,

}

Lesson 23 create your own properties

public string Name

{

get; // let user to read

set; // let user to change property

}

namespace Mynamespace

{

class Class1

{

public Class1(string name){

Name = name; // 把名字给到property

}

public string Name // 不用括号

{

get; // let user to read

set; // let user to change property

}

}

}

Class1 mc = new Class1("adams");

MessageBox.Show(mc.Name); // 不用括号

public string Name

{

get; // let user to read

private set; // let user to change property 只能在class 内部修改，不能外部修改，

}

Read only property

string mystring;

public Class1(string name){

mystring = name;

}

public string Name

{

get { return mystring; } // let user to read

//private set; // let user to change property

}

set {

if (value == "") System.Windows.Forms.MessageBox.Show("You can't do that ");

}

Lesson 24 Throwing Exceptions

if (name == "") throw new Exception("You can't set string equal to an empty string.");

Exception exp = new Exception("You can't set string equal to an empty string.")

public checkstring(string name)

{

if (name == "") throw new Exception(exp) }

之后就可以在main 加上打印exception

Try{

Myclass.checkstring(“”)

}

Exception (exception ex){

MessageBox.show(ex.Message)

}

Lesson 25 Inheritance and overriding

class Class1

{

}

class MysecondClass : Class1 {

} 继承的可以所有parent class function

假如用base class的内容(non private member(pubkic))

用base.age

Class 也可以用protected， protected int Age = 10;

Protected 的话，derived class 可以用parent 的protected的，但是outside class 不能用

Override function 用new

namespace Mynamespace

{

class Class1

{

public string Name = "Bob";

public int age = 10;

public void showMessage(string Message) {

System.Windows.Forms.MessageBox.Show(Message);

}

}

class MysecondClass : Class1 {

public new void ShowMessage(string message) // override parent的class function

{

System.Windows.Forms.MessageBox.Show(message, "title");

}

}

}

Msc = new MysecondClass();

Msc.showMessage(“Message”) 显示有title的

Virtual function 用virtual and override keywords

namespace Mynamespace

{

class Class1

{

public string Name = "Bob";

public int age = 10;

public virtual void showMessage(string Message) {

System.Windows.Forms.MessageBox.Show(Message);

}

// public static void CheckString(string My)

}

class MysecondClass : Class1 {

public override void showMessage(string message) // override parent的class function

{

System.Windows.Forms.MessageBox.Show(message, "title");

}

}

}

Lesson 26 More on Inheritance

We inherit multiple class

Interface 用I 在名字前，为了用户方便识别

所有的member inside in the interface are default public 不能put access modifier inside interface

Interface的用处在于，you have several classes and you want those class do the similar things

class Class1

{

public string Name = "Bob";

public int age = 10;

public virtual void showMessage(string Message) {

System.Windows.Forms.MessageBox.Show(Message);

}

}

class MysecondClass : Class1, IMyinterface

{

public override void showMessage(string message) // override parent的class function

{

System.Windows.Forms.MessageBox.Show(message, "title");

}

public void Myvoid() {

base.showMessage("my void");

}

}

interface IMyinterface

{

void Myvoid();

}

Lesson 27 Indexers 一个class 变成一个string

Class1 mc = new Class1();

MessageBox.Show(mc[0]);

Mc[1] = “Dylan”

class Class1

{

public string this[int index] { //可以传入多个variable [int index,string a]

get { return MyArray[index]; }// must set curly brace;

set { MyArray[index] = value; }//value 是user set equal to

}

string[] MyArray = { "Adam", "Bob", "Joe" };

}

Lesson 28 Structs

Struct 里面可以有method， struct 可以有 constructor

Struct 的constructor必须有parameter

struct Client: IClient {

public Client(string name) {

Name = name;

age = 0;

}

public string Name;

public int age;

public void ClearClientInfo() {

Name = "";

age = 0;

}

}

Class 不能inherit class，但是可以inherit from interface

Interface IClient{

Public string Name;

}

Lesson 29 Partial 部分建立class 或者method

在两个class files 里面分别建立class，名字，名字前加partial意思是，几个partial 的同样名字的class combine together

partial class Myclass {

public string Name = "Mr .smith";

}

Partial void message(void message)

加上partial 名字是在一个file 里declare，一个file里implement，但是有了partial的话，不能加上public/protected 在function 前，所以想要这个method public 必须在class 里面再建一个function call 他

Partial void message(string msg)

{

System.Windows.Forms.MessageBox.show(msg);

}

Public void showmessage(string msg){

message(msg);

}

Lesson 30 abstract

Abstract 不能initializae object

非abstract class 不能有abstract member

abstract class myclass {

public static string Name = "Adams";

public static int Age = 15;

public abstract void showMessage(string message);

}

class MysecondClass : myclass

{

public override void showMessage(string message) {

}

}

Lesson 31 delegates 通过一个delegate call 很多function

Delegates is basically the list of methods

class Myclass {

delegate void Mydelegate(string msg); // in order to add method to delegate 必须跟delegate 的signature 一样， 比如这个要是function input 是int，就不行了

public void showThoseMesasges() {

Mydelegate md = new Mydelegate(ShowMessage); // just add method in to delegate not execute functions ShowMessage 后面不用加括号

md += ShowAnotherMessage;

md("Adam");

}

void ShowAnotherMessage(string message) {

System.Windows.Forms.MessageBox.Show(message, "Test");

}

void ShowMessage(string message)

{

System.Windows.Forms.MessageBox.Show(message);

}

}

Lesson32 Events

Events is a special types of delegate, meaning you can only use += or -= operator

public event EventHandler OnPropertyChanged; 不能OnPropertyChanged = null;

class Myclass {

public event EventHandler OnPropertyChanged; // 用default event handler， 所以

default 的return type 是void, parameter 是object sender EventArgs e

string name = "";

public string Name {

get { return name; }

set { name = value;

OnPropertyChanged(this, new EventArgs()); // Event is arise when property is changed

}

}

}

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

{

Myclass mc = new Myclass();

mc.OnPropertyChanged += new EventHandler(Mc\_OnPropertyChanged); // event的初始化和delegate 初始化一样，都要传入一个function

mc.Name = "steven";

}

void Mc\_OnPropertyChanged(object sender, EventArgs e)

{

MessageBox.Show("The property has changed");

}

Lesson 33 Ternary Operator

string myString = (checkBox1.Checked) ? "it is checked " : "it is unchecked";

MessageBox.Show(myString);

Lesson 34 Open File dialog

OpenFileDialog ofd = new OpenFileDialog();

if (ofd.ShowDialog() == System.Windows.Forms.DialogResult.OK) {

MessageBox.Show("ok");

} // display dialog to user

当user真的打开file， 会打印出来

MessageBox.Show(ofd.FileName); file的path + name

MessageBox.Show(ofd.SafeFileName); file 的name 不会显示path

OpenFileDialog ofd = new OpenFileDialog();

ofd.Filter = "PNG Image|\*.png|BIK|\*.bik";只能打打开.png 和bik结尾的file

ofd.Title = "Open Image";

lesson 35 More Variable Types

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

{

byte mybyte = 255; // one bytes 最大是8 bits， 最大数是255， byte 是 unsigned， sbyte 是 signed bytes

short myshort = 0; // 两个bytes， 16 bits, ushort 是unsigned number

Int16 myint16= myshort; // represent how many bits inside of it, 16 bits, myint16 和 myshort function 基本以一样，

//UInt16 是 unsigned

int MyInt = 0;

Int32 myint32 = MyInt; // Myint32 和 int 一样

long mylong = 0; // 8 bytes 64 bits

Int64 myint64 = mylong; // myint64 和 mylong 一样

//UInt32,uint, ulong, UInt64

float myfloat = 52342141241241; // 只能保证seven digit long， 大于7digit， 用科学计数法表示

float myFloat = 0.5F;// 小于1，告诉compiler this is a float

double d = 0.52352;

char s;

}

Lesson 36， 37， 38 StreamReader pt 1

记得加上 using System.IO; this is providing system reader buffer

OpenFileDialog ofd = new OpenFileDialog();

if (ofd.ShowDialog() == System.Windows.Forms.DialogResult.OK) {

StreamReader sr = new StreamReader(File.OpenRead(ofd.FileName));

textBox1.Text = sr.ReadToEnd();

sr.Dispose(); // close the file

//如果用streamReader 读取同样的file，without sr.Dispose(); 会报错，file is processing

// 关于怎么读的txt，google，下载hxd， 会把每个字符转换成每个数字字母读取

}

sr.BaseStream.ReadByte(); // 读取第一个16 进制的转化成10进制

sr.BaseStream.ReadByte().ToString("x"); // 保留16进制的

sr.BaseStream.Position = 4; // 把初始的位置设置成 第四个byte

sr.BaseStream.Position = 0x0C; //假如读12位，是hexi demical, 0x表示hexi decimla, 0C 代表012

sr.BaseStream.Read(buffer, 0, 3); // 0是读的起点从sr.BaseStream.Position开始的位置算第0个位置，3 是读几个bytes

foreach (byte mybyte in buffer)

{

textBox1.Text += mybyte.ToString("x") + " ";

}

Sr.peek(); // read character at the position 读取但不会改变character 的位置， peek 会返回integer，要convert to character

Char c = (char)sr.peek();

Sr.Read(); // read and advance the position by one

Lesson 39， 40 StreamWriter

private void button3\_Click(object sender, EventArgs e)

{

StreamWriter sw = new StreamWriter(File.OpenWrite(Path));

sw.Write("Hi!");

sw.Dispose();

}

//会接着file 写，原有的位置被覆盖

StreamWriter sw = new StreamWriter(File.Create(Path)); // 会覆盖原来的file，以前的都会删除

sw.Write("Hi!");

sw.Dispose();

sw.WriteLine("Hi!"); 会接着上一行之后，重新下一行接着写

sw.write(“Adam”) 会重启下一行，因为writeline 结束后重启下一行

sw.BaseStream.Position = 0x28;

byte[] buffer = { 0x08, 0x09, 0x0A };

sw.BaseStream.Write(buffer, 0, 3);

lesson 41 Binary Reader

BinaryReader br = new BinaryReader(File.OpenRead(Path));

br.BaseStream.Position = 0x10;

br.ReadChar(); //starting at position 0

string txt = "";

foreach (char a in br.ReadChars(4)) {

txt += a;

}; //read 4 characters

br.ReadInt16().ToString("x"); // Int16() 是2 bytes，会读取两个bytes，会从左向右读取， 假如0001（00 01），会读成100，

byte[] buffer = br.ReadBytes(2);

Array.Reverse(buffer); // reverse 整个array

textBox1.Text = BitConverter.ToInt16(buffer, 0).ToString("x"); 从ToInt16 的位置0，开始convert 整个buffer 成Int

buffer = BitConverter.GetBytes(4312); // convert 4312 到 buffer byte array

Lesson 43 BinaryWriter

int myShort = 0x2345;

byte[] buffer = BitConverter.GetBytes(myShort);

Array.Reverse(buffer); // 因为位置从右向左写，所以Reverse

bw.Write(buffer);

bw.Dispose();

lesson 44 Save File Dialog

SaveFileDialog sfd = new SaveFileDialog();

sfd.Filter = "Text File|\*.txt";

sfd.FileName = "My Text File"; // 默认的class

sfd.Title = "Save Text File";

if (sfd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

string path1 = sfd.FileName;

BinaryWriter bw = new BinaryWriter(File.Create(path1));

bw.Write("Example text file");

bw.Dispose();

}

Lesson 45 Convert Class 在不同的类型的数据中转换用Convert

string a = "123";

Convert.ToInt32(a);

Convert.ToChar(a);

Lessom 47 Is as and converting

object myobj = "Adam";

if (myobj is string) {

string new\_string = myobj as string

MessageBox.Show((string));

}

会显示这个messagebox, 因为它不是string

object myobj = "Adam";

Control myControl = button1;

if (myControl is Button) {

Button myButton = (Button)myControl; //作用一样Button myButton = myControl as Button;

MessageBox.Show(myButton.Text);

}

Lesson 49 substring

String Name = “John Smith”;

String FirstName = Name.Substring(0,4); 第一个参数是起始值，第二个是选择的长度

String FirstName = Name.Substring(5); 从第五个开始，一直到结束

Lesson 50 IndexOf and Trim

Name

String Name = “John Smith”;

String FirstName = Name.Substring(0,Name.IndexOf(“ ”));

Name = “ John Smith ”;

String rawname = Name.Trim(); 去掉空格，像strip 打出来是John Smith

Rawname = Name.TrimStart(); 仅去掉前面的空格不去掉后面的空格“”John Smith ”;

Name.TrimEnd() 去掉结尾的空格，不去开头的空格

Lesson 51 Remove and Replace

string sentence = "Hello, my name is adam";

string after = sentence.Remove(0, 7);// 从0开始，去掉前7个

after = sentence.Remove(17);//从位置17位置开始， 一直remove 到结束

after = sentence.Replace("Hello", "Hi"); // replace hello with hi

lesson 52 Split and ToCharArray

string names = "Adams;Bobs;Joe;Steve;Allen;Matt";

string[] nameArray = names.Split(';');

string letters1 = "abcdefg"; // convert each char in string to char[]

char[] letters = letters1.ToCharArray();

lesson 53 Math class

Math.Abs(-4);

Math.PI; // Pi 3.1415926

Math.Pow(2, 3);

Math.Round(4.723,2); // default is integer, 2 是round to 2位小数

Lesson 54 Generate Random Number

Random r = new Random();

r.Next(); // return random number but not negative

r.Next(0,100);//generate random number between 0 and 100

byte[] buffer = new byte[5];

r.NextBytes(buffer); // pass bytes array which will be filled

r.NextDouble(); // generate double return number between 0 and 1

lesson 55 Generate Random String

char []letters = "qwertyuiopasdfghjklzxcvbnm".ToCharArray();

Random r = new Random();

letters[r.Next(0, 25)];

lesson 56 Folder browser Dialog 显示folder browser

FolderBrowserDialog fbd = new FolderBrowserDialog();

fbd.RootFolder = Environment.SpecialFolder.MyDocuments; // 当展示folder，直接显示document

fbd.Description = "Adam was here.";

if (fbd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

MessageBox.Show(fbd.SelectedPath); //显示选择的地址

}

Lesson 57， 58， 59 Directory class ， 得到directory的名称， 得到directory生成时间，读取时间，处理directory

FolderBrowserDialog fbd = new FolderBrowserDialog();

fbd.RootFolder = Environment.SpecialFolder.MyDocuments; // 当展示folder，直接显示document

fbd.Description = "Adam was here.";

if (fbd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

string [] files = Directory.GetFiles(fbd.SelectedPath);

// 会得到所有的file， return string array which include all files inside the folder 但不会得到文件夹 里面的文件夹

files = Directory.GetDirectories(fbd.SelectedPath);

Directory.GetLogicalDrives(); // 得到电脑上所有的drive

}

FolderBrowserDialog fbd = new FolderBrowserDialog();

fbd.RootFolder = Environment.SpecialFolder.MyDocuments; // 当展示folder，直接显示document

fbd.Description = "Adam was here.";

if (fbd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

Directory.GetCreationTime(fbd.SelectedPath); // 最后一次创建的时间

Directory.GetLastAccessTime(fbd.SelectedPath); //最后一次更改的时间

Directory.GetLastWriteTime(fbd.SelectedPath); // 得到最后一次改的时候

Directory.GetParent(fbd.SelectedPath); // 得到这个文件夹的parent 文件夹的名字

}

FolderBrowserDialog fbd = new FolderBrowserDialog();

fbd.RootFolder = Environment.SpecialFolder.MyDocuments; // 当展示folder，直接显示document

fbd.Description = "Adam was here.";

if (fbd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

Directory.CreateDirectory(fbd.SelectedPath + "\\Becks"); // create a directory 在选择的folder中,新文件夹叫becks

Directory.Move(fbd.SelectedPath, "C:\\Users\\Destop"); // move 选择的folder 到之后的folder

Directory.Delete(fbd.SelectedPath); // 删除选择的folder

}

Lesson 60, 61 File class

OpenFileDialog ofd = new OpenFileDialog();

if (ofd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

File.Exists(ofd.FileName); // check if file exists, 返回值是true or false

File.Delete(ofd.FileName); // 删除这个file

}

Lesson 62 path class

OpenFileDialog ofd = new OpenFileDialog();

if (ofd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

Path.GetDirectoryName(ofd.FileName); // 返回file 的directory name

Path.GetFileName(ofd.FileName);//返回 file 的名字

Path.GetFullPath(ofd.FileName); // 返回file 全名

}

Lesson 63, 64 process class

OpenFileDialog ofd = new OpenFileDialog();

if (ofd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

Process.Start(ofd.FileName); // start ext process;

}

Process.Start("Notepad.exe"); // Don't need to tell path, 因为在system32 folder

string name = Process.GetCurrentProcess().ProcessName; // 得到现在process name

// kill method or end process

Process.GetCurrentProcess().Kill();

OpenFileDialog ofd = new OpenFileDialog();

if (ofd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

Process.Start(ofd.FileName); // start ext process;

}

foreach (Process p in Process.GetProcesses())

{

MessageBox.Show(p.ProcessName); //得到所有的process 不仅仅是program的

bool respond = p.Responding; // if program freeze, return false

if (!respond) p.Kill();

}

foreach (Process p in Process.GetProcessesByName("Skype")) // 如果有多个process，可能返回多个name

{

MessageBox.Show(p.ProcessName);

}

Lesson 65 Null Coalesce Operator

string myString = null;

MessageBox.Show(myString ?? "The string is null"); //如果myString 是null， 显示 "The string is null"， 如果不是null，显示string itself

int? i = null; // int i = null; 会报错，因为intger 不能是null

int x = i ?? 8; // 如果i 是 null，给它8 如果不是null， 给它i的值

当时null的时候，？？后面的内容成立，如果不是null，就是它自己

Lesson 66, 67, 68 Bitwise operator

short myshort = ~3; // 把3 invert， 取非

Convert.ToString(myshort, 2); // 想要它成为binary

short myshort = 3 & 5; // 取binary and 011 & 101 = 001

myshort = 3 | 5; // 011 | 101 = 111;

short myshort = 3 ^ 5; //XOR(exception or) 011 ^ 101 = 110

myshort = 3 >> 1; // shift to the left 011 >> 1 = 1 ;

myshort = 5 << 1; // shift to the right 101 << 1 = 1010 ;

Convert.ToString(myshort, 2); // 想要它成为binary

Lesson 69,70,71 Threading

在头文件加上using System.Threading;

Start 开始,abort结束

Thread t = new Thread(freeze); // freeze is a function

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

{

for (;;) { }// some method freeze your application

// let your code in the background and your gui never freeze

t.Start();// start the thread 当点击exit button 不会退出threading ，因为code 在background running

}

private void Form1\_FormClosing(object sender, FormClosingEventArgs e)

{

t.Abort(); // End the thread

}

不能把form 的property， 比如button。Text， textbox.Text 放进别的thread，因为他们生成在别的thread，所以不能access in another thread

想要pass variable into form， 必须建立别的varaible，然后thread 完成后pass

Thread t;

string mystring = "";

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

{

t = new Thread(write);

t.Start();

while (t.IsAlive) ;

textBox1.Text = mystring;

}

void write() {

for (int i = 0; i < 1000; i++) {

mystring += "Adam" + i.ToString() + "\r\n";

}

}

private void Form1\_FormClosed(object sender, FormClosedEventArgs e)

{

t.Abort();

}

Pass into thread 必须是object variable

Thread t;

string mystring = "";

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

{

t = new Thread(write);

t.Start("Adam");

while (t.IsAlive) ;

textBox1.Text = mystring;

}

void write(object name) { // pass into thread must be object

for (int i = 0; i < 1000; i++) {

mystring += name.ToString() + i.ToString() + "\r\n";

}

}

// pass into thread must be object, 只能pass 一个object，想要pass 多个，必须pass by object array

// 而且pass array，名字只能是object 。。。，想要用这个array，必须 加上 object[] o = name as object[];

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

{

t = new Thread(write);

object[] obj = { "Bob", 500 };

t.Start(obj);

while (t.IsAlive) ;

textBox1.Text = mystring;

}

void write(object name) { // pass into thread must be object, 只能pass 一个object，想要pass 多个，必须pass by object array

// 而且pass array，名字只能是object 。。。，想要用这个array，必须 加上 object[] o = name as object[];

Thread.Sleep(100); // 100 milliseconds

object[] o = name as object[];

for (int i = 0; i < Convert.ToInt32(o[1]); i++) {

mystring += o[0].ToString() + i.ToString() + "\r\n";

}

}

Lesson 72 webconnect status log

using System.Net;

从网页的text 文件放进text box 里面

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

{

WebRequest.DefaultWebProxy = null;

webClient wc = new WebClient();

textBox1.Text = wc.DownloadString(“Http://....”)

}

Lesson 72， 73 webconnect download file

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

{

SaveFileDialog sfd = new SaveFileDialog();

if (sfd.ShowDialog() == System.Windows.Forms.DialogResult.OK) {

WebClient wc = new WebClient();

wc.DownloadFileAsync(new Uri("https://www.youtube.com/watch?v=pzl5VMf4Dlg&index=73&list=PL0EE421AE8BCEBA4A"), sfd.FileName);

wc.DownloadFileCompleted += new AsyncCompletedEventHandler(wc\_DownFileComplieter); // 建立一个event 显示progress finished

wc.DownloadProgressChanged += new DownloadProgressChangedEventHandler(wc\_progressChanged);

}

}

void wc\_progressChanged(object sender, DownloadProgressChangedEventArgs e)

{

label1.Text = "Progress: " + e.ProgressPercentage.ToString(); // 显示出progress 的进程

}

void wc\_DownFileComplieter(object sender, AsyncCompletedEventArgs e) {

MessageBox.Show("File is downloaded");

}

Lesson 74,75,76 Send Email

Tooltip 将会显示，假如在输入的时候, TextBox 有passwordchar 可以将输入的数字变成\*

Form 的property MaximizeBox and MinimizeBox 选择false，可以不让user change form

添加图片在button 的image，然后点resource context 然后import， 然后button的property的iage Align 可以选择图片位置

private void button2\_Click(object sender, EventArgs e)

{

try

{

button2.Enabled = false;

MailMessage message = new MailMessage();

message.From = new MailAddress(textBox1.Text);

message.Subject = textBox4.Text;

message.Body = textBox5.Text;

foreach (string s in textBox3.Text.Split(';'))

{

message.To.Add(s); // 把发送对象加进To 对象

}

SmtpClient client = new SmtpClient();

client.Credentials = new NetworkCredential(textBox1.Text, textBox2.Text); // 1 是用户名，2是密码

client.Host = "smtp.gmail.com"; // gmail的host

client.Port = 587;//gmail的host

client.EnableSsl = true;

client.Send(message);

}

catch

{

MessageBox.Show("There was an error sending the Message. Make sure Crendential correctly");

}

finally {

button2.Enabled = true;

}

}

Lesson 77 Datetime Picker

Datetimepicker property 可以改变 format of datetime

可以 google datetime picker format

Long version of format: dddd, MMMM d, yyyy h:mm:ss tt

Short version of format: M/d/yyyy h:mm:ss tt

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

{

DateTime dt = dateTimePicker1.Value; // 返回时DateTime format

}

Lesson 78 Datetime Struct

DateTime dt = new DateTime(1995, 10, 16); // 会显示1995/10/16 12:00:00AM, 12:00:00AM 是 default 选项

MessageBox.Show(dt.ToString());

DateTime dtt = DateTime.Today; // 得到今天日子，没有时间

dtt = DateTime.Now; // 显示现在日期和时钟

bool isleapyear = DateTime.IsLeapYear(2011);// 检测是不是闰年

int dateinMonth = DateTime.DaysInMonth(2011, 7);//得到2011年7月 日子长

MessageBox.Show(DateTime.Now.ToFileTime().ToString()); // 显示file的时间 是long

dtt = DateTime.FromFileTime(); // 一个long

lesson 79 Picture Box and Image Class

在picture box 的右上角 可以选择 image size mode

可以在picturebox 选择 borderstyle 的形式， 比如fixed3D 或者normal

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

{

OpenFileDialog ofd = new OpenFileDialog();

if(ofd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

pictureBox1.ImageLocation = ofd.FileName;

}

}

OpenFileDialog ofd = new OpenFileDialog();

if(ofd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

Image image = Image.FromFile(ofd.FileName); // can set multiple into one picture box and save image for later use

pictureBox1.Image = image;

pictureBox1.ImageLocation = "http://...";//也可以是url

}

Lesson 80 clipboard

Clipboard is the board when you copy something

当你copy something, the data is stored in clipboard

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

{

textBox2.Text = Clipboard.GetText();

// get test retrieve text in clipboard if any, 点复制了，然后就会出现在textbox2, 甚至可以copy 从电脑任何地方

// 都会显示在clipboard 上

}

pictureBox2.Image = Clipboard.GetImage(); 复制图片

pictureBox2.Image = Clipboard.GetImage();

Clipboard.GetData(DataFormats.Rtf); // 可以retrive 很多形式的format 通过dataformats

往clipboard 设置

try

{

Clipboard.SetText("Adam");//可以往clipboard set 值，然后点粘贴，就可以把adam 粘贴了}

Clipboard.SetImage(pictureBox1.Image);

}

catch { }

清楚clipboard

Clipboard.Clear();

Lesson 81 Color Dialog

ColorDialog cd = new ColorDialog();

if (cd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

button1.BackColor = cd.Color;

// 选择的颜色变成back的color

//

}

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

{

ColorDialog cd = new ColorDialog();

cd.AllowFullOpen = false; // so cannot choose custom color

cd.FullOpen = true; // 选择custom color

cd.ShowHelp = true; // 会显示 help button 在按键的下面

cd.HelpRequest += Cd\_HelpRequest; // 建一个event

if (cd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

button1.BackColor = cd.Color;

// 选择的颜色变成back的color

//

}

}

private void Cd\_HelpRequest(object sender, EventArgs e)

{

MessageBox.Show("choose a color for the background of your button");

}

Lesson 82 Color Struct

ColorDialog cd = new ColorDialog();

if (cd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

Color c = cd.Color; // Color c is color struct

if (c.IsKnownColor) { // known color is the colors that used for windows properties, like scroball

// 如果选择了color， 会显示名字，但是如果是customer color，不会显示color

MessageBox.Show(c.ToKnownColor().ToString());

}

}

ColorDialog cd = new ColorDialog();

if (cd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

Color c = cd.Color; // Color c is color struct

if (c.IsNamedColor) {

// 如果选择了color

MessageBox.Show(c.Name.ToString());

}

}

也可以 从Color 里面选择颜色

Color c = Color.Red;

也可以 从knownColor 里面选择颜色

c = Color.FromKnownColor(KnownColor.Red);

把color 转换成int

c = Color.FromKnownColor(KnownColor.GradientActiveCaption);

c.ToArgb(); // convert color to integer A alpha, r is red, g is green, b is blue

从int 转换成color

int i = c.ToArgb();

Color b = Color.FromArgb(i);

Lesson 83 Font Dialog 可以选择你想要的字体和颜色

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

{

FontDialog fd = new FontDialog();

fd.MinSize = 10; // 最小的字符必须是10

fd.MaxSize = 20; // 最大字符是20

fd.ShowColor = true; // 可以选择颜色

fd.ShowHelp = true;

fd.HelpRequest += Fd\_HelpRequest;

if (fd.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

textBox1.Font = fd.Font;//让text font 等于 fd的dialog

textBox1.ForeColor = fd.Color; // 选择字符的颜色

}

}

private void Fd\_HelpRequest(object sender, EventArgs e)

{

MessageBox.Show("Select the fond you like")

}

Lesson 84 Timer control

Timer 的property 选择interval 是按millisecond 算的, timer不会显示在GUI上

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

{

timer1.Start();// wait for that interval then tick

}

private void timer1\_Tick(object sender, EventArgs e)

{

timer1.Stop(); // stop the timer 如果不stop，每一秒都会蹦出来一个hello

MessageBox.Show("Hello");//get hello after one second

}

Lesson 85 playing sounds

加上这个在头文件

using System.Media;

Wav file

OpenFileDialog ofd = new OpenFileDialog();

if (ofd.ShowDialog() == DialogResult.OK) {

SoundPlayer s = new SoundPlayer(ofd.FileName);

s.Play();

s.PlayLooping(); // 持续播放

s.PlaySync(); // 不能move window 必须等声音结束才能move gui

}

系统的声音

SystemSounds.Asterisk.Play();

SystemSounds.Beep.Play();

SystemSounds.Exclamation.Play();

SystemSounds.Hand.Play();

Lesson 86 MaskedTextBox Control

假如你只想让user text number,然后string 成 int 然后不能输入别的值，可以用它

在property 中mask 可以选择，你想让user type 的种类

还可以自己定义 mask

谷歌 masktestbox mask property

在mask中定义 Literal 是不能更改的，remain where they are

Lesson 87 multiple forms

先点project 🡪 add windows form

在form 1 中button 定义，然后

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

{

Form2 f = new Form2();

f.Show();

f.ShowDialog();

//点击它只能在form2 上操作，就不能点击form1了，但是f.show() 可以在1,2 中选择

}

把form1 的变量传入form2

Form2 f = new Form2(textBox1.text);

f.ShowDialog();

public Form2(string mystring) // 从form1 中传入variable 到 form2

{

InitializeComponent();

label2.Txt = mystring;

}

Lesson 88 Multi Document Interface MDI

We can form inside form

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

{

// 告诉compiler form 1 hold the entire form

this.IsMdiContainer = true; // have form inside

Form2 f = new Form2();

f.MdiParent = this; // 告诉它的parent 是form 1

f.Show(); // 不能选showDialog(); 因为不能click and not go to form1 因为inside form1

Form3 ff = new Form3();

ff.MdiParent = this;

ff.show()

}

private void button2\_Click(object sender, EventArgs e)

{

this.LayoutMdi(MdiLayout.ArrangeIcons); // 可以range them 当他们minisize 几个form，点击它可以让他们在下面arrange

this.LayoutMdi(MdiLayout.Cascade); // Makethem wider and taller

this.LayoutMdi(MdiLayout.TileVertical); // 点击他们让他们水平放置

}

Lesson 89 ComboBox Control

Allow you to select items from huge list of items

点击右上角的edit items，可以编辑item

每个item 只能在一行， 比如

Adam

Bob

Joe

选择combo box 的property，change DropDownStyle from DropDown to DropDown List 只能是读取，不能写

Simple 是会显示所有的内容用list 不用 dropdown 也可以写

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

{

// 当选择不同item 会显示

MessageBox.Show("test");

}

Lesson 90 ProgressBar Control

ProgressBar 的property 的maximum value 是多少这个bar 可以hold

如果设置maxi 是100, value = 50， progress bar 就是到一半的位置

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

{

progressBar1.Value += 10; // 每次progress bar 加100 如果ProgessBar1.value 大于100， 会报错

progressBar1.PerformStep();//每次加的值就是property 里面设置的step的值

}

private void button2\_Click(object sender, EventArgs e)

{

progressBar1.Value = 0;

}

Property 里面MarqueeAnimationSpped

让他自己循环中表示正在进行

progressBar1.MarqueeAnimationSpeed = 100;

设置数越大，越慢

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

{

progressBar1.Style = ProgressBarStyle.Marquee; // 开始让bar 进行

progressBar1.MarqueeAnimationSpeed = 100;

}

private void button2\_Click(object sender, EventArgs e)

{

progressBar1.Style = ProgressBarStyle.Blocks; // 让bar 结束

}

Lesson 91, 92, 93 ListView Control

如果选择listview 的checked box = True, 就在每行前面都有checked box

ListView is used to sort data

点击listView 右上角的edit columns, 然后点击view 选detail

然后listview property grid line 为 true FullRowSelect 为True

之后假如选择一个数据，一行都会被选择上

之后点击multiselect 为False， 就一次不能选多个数据了

ListViewItem lvi = new ListViewItem(textBox1.Text);

lvi.SubItems.Add(textBox3.Text); ;

lvi.SubItems.Add(textBox2.Text);

//传入三个textbox 到subitems

listView1.Items.Add(lvi);

textBox1.Text = "";

textBox2.Text = "";

textBox3.Text = "";

Content Menu Strip 是menu 当你right click something, 然后在ListView 的property里面选择 你建立好的ContextMenuStrip

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

{

if (listView1.SelectedItems.Count != 0) //确保user 确实选择某些东西

{

MessageBox.Show(listView1.SelectedItems[0].SubItems[0].Text); // 得到选择的选择的第一列的值

// listView1.SelectedItems 选返回array, 如果选择一个SelectedItems的 index都是0，SubItem 如果是第一列，就是0， 第二列是1， 第三列是2

}

}

通过listview 然后右键选择多行，打印所有选择的第一列

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

{

if (listView1.SelectedItems.Count != 0) //确保user 确实选择某些东西

{

MessageBox.Show(listView1.SelectedItems[0].SubItems[0].Text); // 得到选择的选择的第一列的值

// listView1.SelectedItems 选返回array, 如果选择一个SelectedItems的 index都是0，SubItem 如果是第一列，就是0， 第二列是1， 第三列是2

foreach (ListViewItem lvi in listView1.SelectedItems) { // loop through 所有的listview的selected的items

MessageBox.Show(lvi.SubItems[0].Text);

}

}

}

Loop 去掉选择listview

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

{

if (listView1.SelectedItems.Count != 0) {

foreach (ListViewItem lvi in listView1.SelectedItems) {

lvi.Remove();

}

}

}

去掉所有的item

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

{

listView1.Items.Clear(); //

}

去掉所有checked 的item

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

{

foreach (ListViewItem lvi in listView1.Items) {

if (lvi.Checked)

lvi.Remove();

}

Lesson 95 ToolStrip and StatusStrip Controls

Toolstrip 是在GUi 上面， StatusStrip是在下面

ToolStrip 是在file 上面的下拉菜单

点击Form 上的ToolStrip 然后点dropdown button

可以点击toolStripDropDownButton1 的property 然后选display style, 可以选择file下拉菜单的显示种类是image 还是text

还可以选择file， separator

Status 显示status, 假如open file显示open file

Lesson 96 NotifyIcon Control

System control allow us to have a balloon tip message show from the system tray

就像windows 右下角的文件夹

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

{

this.Hide();//把这个form hide 起来

notifyIcon1.ShowBalloonTip(3000,"My Title","My Text",ToolTipIcon.Info); //会显示3秒，在你windos 上,ToolTipIcon.Info说没有error 是info, “My title”是text 的title "My Text" 是balloontip的内容

//第一个parameter

}

private void notifyIcon1\_MouseDoubleClick(object sender, MouseEventArgs e)

{

this.Show();

}

双击 icon显示这个exe

Lesson 97 Opening Files With your app

在program.cs 改

Drag files into program, 然后告诉你文件的位置

namespace WindowsFormsApp2

{

static class Program

{

/// <summary>

/// The main entry point for the application.

/// </summary>

[STAThread]

static void Main(string [] files) // 让文件带着file 开始

{

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

foreach (string s in files) {

Application.Run(new Form1(s)); // run more files and then pass into file

}

}

}

}

namespace WindowsFormsApp2

{

public partial class Form1 : Form

{

public Form1(string s) // s is the file program open

{

MessageBox.Show(s);

InitializeComponent();

}

}

Lesson 98 settings

when you open program and close program, then setting still be there

点击vs的project， 然后project 的property， 点击setting

打开后会显示上次保存的信息

public Form1() // s is the file program open

{

InitializeComponent();

textBox1.Text = WindowsFormsApp2.Properties.Settings.Default.Name;

}

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

{

WindowsFormsApp2.Properties.Settings.Default.Name = textBox1.Text;

WindowsFormsApp2.Properties.Settings.Default.Button = button1;

WindowsFormsApp2.Properties.Settings.Default.Save();//save the property

}

Lesson 99, 100, 101 Tree View

就是一个Item 接着一个sub item

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

{

treeView1.Nodes.Add("People"); // 会显示default 最top 的 node

treeView1.Nodes.Add("Animal");

treeView1.Nodes[0].Nodes.Add("adam"); //在people node(reeView1.Nodes[0].Nodes.) 下面加node，

treeView1.Nodes[0].Nodes.Add("Bob");

treeView1.Nodes[0].Nodes.Add("Joe");

}

private void button2\_Click(object sender, EventArgs e)

{

treeView1.SelectedNode.Remove(); // 删除选择的nodes

treeView1.Nodes.Clear(); // 删除所有的nodes

}

选择checkbox 的property为True，

private void button2\_Click(object sender, EventArgs e)

{

RemovedChecked(treeView1.Nodes);

}

List<TreeNode> tnList = new List<TreeNode>();

void RemovedChecked(TreeNodeCollection tnc) {

foreach (TreeNode tn in tnc)

if (tn.Checked) tnList.Add(tn);

else if (tn.Nodes.Count != 0) RemovedChecked(tn.Nodes);

// have children nodes inside those nodes recurssion 用recurssion 删除

foreach (TreeNode tn in tnList)

treeView1.Nodes.Remove(tn);

}

往treeview 里面加image

首先加ImageList, 然后选择images, 选择image 文件，再选择imageSize, 16\*`16 是default

然后记得返回treeview 的property，点击ImageList 选择imageList1

TreeNode tn = new TreeNode();

tn.Text = "Person"; // 文字是person

tn.ImageIndex = 0; //ImageIndex 来自Imagelist

tn.SelectedImageIndex = 0; // 保证选择的ImageIndex 也是3

treeView1.Nodes.Add(tn);

SelectedImageIndex 跟ImageIndex 要一样，因为可能一点击，就换图片了

Lesson 102 Property 显示class 的property

propertyGrid1 不光显示自己定义的class的property，还可以显示，toolbox 的 property

在property grid的property中 点selected object 中选择你想要的tool的property

自己定义的class

，比如

Person p = new Person();

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

{

p.Name = "Adam";

p.Age = 15;

p.Email = "123@gmail.com";

propertyGrid1.SelectedObject = p; // 把person 的property 改property grid

//Reload();

}

class Person

{

public String Name {

get;

set;

}

public int Age {

get;

set;

}

public string Email {

get;

set;

}

}

private void propertyGrid1\_PropertyValueChanged(object s, PropertyValueChangedEventArgs e)

{

textbox.Text = p.name; 当在property grid 中改变这个property， 会call这个function，显示在textbox上

}

Lesson 103, 104 Accessing all controls

因为control 里面有control，所以用recurssion, 循环所有的control

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

{

AccessAll(this.Controls);

}

void AccessAll(Control.ControlCollection cc) {

foreach (Control c in cc)

{

c.Text = "Adams";// recurssive method

if (c.HasChildren) AccessAll(c.Controls); // access 所有的control

}

}

只enable button的control, 让所有checkbox 都checked

void AccessAll(Control.ControlCollection cc) {

foreach (Control c in cc)

{

if(c is Button) c.Enabled = true;// recurssive method

if (c is CheckBox) {

CheckBox ch = c as CheckBox;

ch.Checked = true;

}

if (c.HasChildren) AccessAll(c.Controls); // access 所有的control

}

}

void AccessAll(Control.ControlCollection cc) {

foreach (Control c in cc)

{

if (c is Button) {

Button b = c as Button;

b.Click += new EventHandler(b\_Click);

}

c.Enabled = true;// recurssive method

if (c is CheckBox) {

CheckBox ch = c as CheckBox;

ch.Checked = true;

}

if (c.HasChildren) AccessAll(c.Controls); // access 所有的control

}

}

void b\_Click(object sender, EventArgs e) {

MessageBox.Show("You click the button");

}

105 WebBrower Control

webBrowser1.Navigate(textBox1.Text);//需要Url 显示web的内容， textBox1.Text 是url 比如google.com

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

{

webBrowser1.Navigate(textBox1.Text);//需要Url

//如果想要url display when user click to update

Navigate 到user 输入的url

}

private void webBrowser1\_Navigated(object sender, WebBrowserNavigatedEventArgs e)

{

textBox1.Text = webBrowser1.Url.ToString(); // Convert Url to string

当url改变，textbox 显示新的url

}

让webrower 前进 后退，刷新

private void button3\_Click(object sender, EventArgs e)

{

webBrowser1.Refresh();

}

private void button2\_Click(object sender, EventArgs e)

{

webBrowser1.GoBack();

}

private void button4\_Click(object sender, EventArgs e)

{

webBrowser1.GoForward();

}

private void button5\_Click(object sender, EventArgs e)

{

webBrowser1.GoHome(); // GoHome 是由internet explore 决定的

}

Scrap 网页

WebBrowser wb = new WebBrowser();

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

{

wb.Navigate("https://www.fxstreet.com/economic-calendar");

wb.DocumentCompleted += Wb\_DocumentCompleted;//当被加载出来，显示这个

}

private void Wb\_DocumentCompleted(object sender, WebBrowserDocumentCompletedEventArgs e)

{

//chrome 点击inspect element

MessageBox.Show(wb.Document.GetElementById("fxit - eventInfo - time fxs\_event\_time").InnerText); //get test for this id

}

在webBrower property 点击url 可以写默认的url

WebBrowser wb = new WebBrowser();

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

{

webBrowser1.Document.GetElementById("uh-search-box").InnerText = textBox1.Text; //首先inspect element 找他的id

}

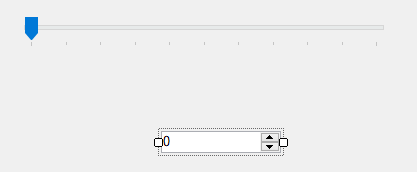
private void button2\_Click(object sender, EventArgs e)

{

webBrowser1.Document.GetElementById("yui\_3\_18\_0\_4\_1502253635248\_1663").InvokeMember("Click"); // Invoke Click Butoon

}

Lesson 108 TrackBar Control and NumericUpDown Control



Property 里面的maximum value 决定了最大可以drag 多少

Minmum 是最小可以drag 多少，tick 是个value 一个tick

比如max 10， min 0 tick 1， 就是每次drag 一次往上走一

Property 可以选择orientation horizontal 水平放置，vertical竖直放， 还可以选择tickstyle，选择trackbar 箭头的位置，

显示track bar的值

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

{

MessageBox.Show(trackBar1.Value.ToString());

}

Numerical Up and down 只能enter number

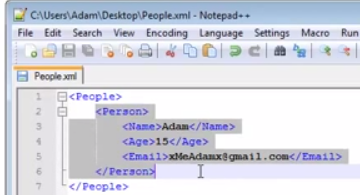
Property incremental 可以更改每次增加的数值，是1，还是10,还是别的数

也可以更改Maximum, minimum 的值

numericalUpDown1.value = Convert.toInt32(32);

Lesson 109， 110, 111, 112, 113， 114 Reading, Edit, Writing XML

Used for store data . Similar to tree view



在头文件加上using System.Xml;

读取XML 的 tag

OpenFileDialog ofd = new OpenFileDialog();

ofd.Filter = "XML|\*.xml";

if (ofd.ShowDialog() == DialogResult.OK)

{

XmlDocument xDoc = new XmlDocument();

xDoc.Load(ofd.FileName);//path document， 也可以url的path

MessageBox.Show(xDoc.SelectSingleNode("People/Person/Name").InnerText); //列举了tag 的Innertest

}

假如xml 里面有多个一样的class， 要通过loop 读取所有的XmlNode node in xDoc.SelectNodes("People/Person")

OpenFileDialog ofd = new OpenFileDialog();

ofd.Filter = "XML|\*.xml";

if (ofd.ShowDialog() == DialogResult.OK)

{

XmlDocument xDoc = new XmlDocument();

xDoc.Load(ofd.FileName);//path document

//MessageBox.Show(xDoc.SelectSingleNode("People/Person/Name").InnerText); //列举了tag 的Innertes

foreach (XmlNode node in xDoc.SelectNodes("People/Person"))

// return xml nodes, 所以return 所有的person nodes inside people

{

MessageBox.Show(node.SelectSingleNode("Name").InnerText);

}

}

Edit XML

xDoc.SelectSingleNode("People/Person/Name").InnerText = textBox1.Text;

xDoc.Save(path); //xml file 的路径

Writing XML

XmlWriter writer = XmlWriter.Create("C:\\Users\\Xu Zhao\\Desktop\\file.xml");// 路径

writer.WriteStartElement("People");

writer.WriteStartElement("Person"); // <Person>

writer.WriteStartElement("Name"); // <Name>

writer.WriteString("Maggi");

writer.WriteEndElement(); // </Name>

writer.WriteStartElement("Age"); // <Age>

writer.WriteString("18");

writer.WriteEndElement(); // </Age>

writer.WriteEndElement(); //</Person>

writer.WriteEndElement(); //</People>

writer.Close();

Write Nodes to Existing XML File

CreateElemetn 之后 AppendChild

XmlDocument doc = new XmlDocument();

doc.Load("C:\\Users\\Xu Zhao\\Desktop\\file.xml");

XmlNode person = doc.CreateElement("Person"); // 再加一个person <Person>

XmlNode name = doc.CreateElement("Name");

name.InnerText = "cc";

person.AppendChild(name); // Person append name

XmlNode age = doc.CreateElement("Age");

age.InnerText = "20";

person.AppendChild(age); // Person append name

doc.DocumentElement.AppendChild(person); //这个会append child into XML root node

// 因为xml的root node 是 people， 所以加在root node 下面

doc.DocumentElement.AppendChild(person);

doc.Save("C:\\Users\\Xu Zhao\\Desktop\\file.xml");

Delete XML name

xNode.ParentNode.RemoveChild(xNode); //得到XNode的parent就是Person然后remove person中属于bob的

XmlDocument doc = new XmlDocument();

doc.Load("C:\\Users\\Xu Zhao\\Desktop\\file.xml");

foreach (XmlNode xNode in xDoc.SelectNodes("People/Person")) {

if (xNode.SelectSingleNode("Name").InnerText == "Bob")

xNode.RemoveAll(); // remove all inside entry， 就是把所有的内容inside person 等于bob的删除了，但是不会删除Person node（属于bob的）

xNode.ParentNode.RemoveChild(xNode);// 这样会删除所有的node

}

doc.Save(路径)

lesson 115 MD5 and SHA1

MD5 is 16 bytes

SHA1 is 20 bytes

using System.Security.Cryptography; 加上这个在头文件

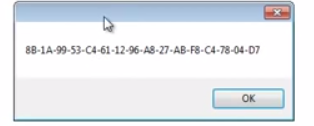
MD5CryptoServiceProvider md5 = new MD5CryptoServiceProvider();

UTF8Encoding utf8 = new UTF8Encoding(); // text of format to store text

string a = BitConverter.ToString(md5.ComputeHash(utf8.GetBytes("Adam")));

// Hash for Adam, 16 bytes 长， 如果改变一个字母都completely different from before

MessageBox.Show(a);



SHA1CryptoServiceProvider sha1 = new SHA1CryptoServiceProvider();

UTF8Encoding utf8 = new UTF8Encoding();

string a = BitConverter.ToString(sha1.ComputeHash(utf8.GetBytes("Adam")));

116 TripleDES Encryption; EnCryption Data

Difference between Encryption and hashing 是 hashing 不能decryption

//16 bytes long

MD5CryptoServiceProvider md5 = new MD5CryptoServiceProvider();

UTF8Encoding utf8 = new UTF8Encoding();

TripleDESCryptoServiceProvider tDes = new TripleDESCryptoServiceProvider();

//set key which user for encrpyted

tDes.Key = md5.ComputeHash(utf8.GetBytes("Becks")); //改变key，就=改变encrypted 所有的内容

tDes.Mode = CipherMode.ECB;

tDes.Padding = PaddingMode.PKCS7; // set those property correctly

ICryptoTransform trans = tDes.CreateEncryptor(); // Encrpt our data

string c = BitConverter.ToString(trans.TransformFinalBlock(utf8.GetBytes("I am Becks"),0, utf8.GetBytes("I am Becks").Length)); // do the actual encrption

// 第二个参数 是 从这个string起始位置，0表示从头开始

// 第三个参数 代表 encrpyted 多少的长度， utf8.GetBytes("I am Becks").Length表示所有的

MessageBox.Show(c);

117 TripleDES Decryption

Decryption 与 Encryption 前面设置MD5，utf8 都一样，不一样的是ICryptoTransform trans = tDes.CreateDecryptor();

byte[] encrptyed;

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

{

MD5CryptoServiceProvider md5 = new MD5CryptoServiceProvider();

UTF8Encoding utf8 = new UTF8Encoding();

TripleDESCryptoServiceProvider tDes = new TripleDESCryptoServiceProvider();

//set key which user for encrpyted

tDes.Key = md5.ComputeHash(utf8.GetBytes("Becks")); //改变key，就=改变encrypted 所有的内容

tDes.Mode = CipherMode.ECB;

tDes.Padding = PaddingMode.PKCS7; // set those property correctly

ICryptoTransform trans = tDes.CreateDecryptor(); // Encrpt our data

string a = utf8.GetString(trans.TransformFinalBlock(encrptyed, 0, encrptyed.Length));

MessageBox.Show(a);

}

private void button2\_Click(object sender, EventArgs e)

{

//16 bytes long

MD5CryptoServiceProvider md5 = new MD5CryptoServiceProvider();

UTF8Encoding utf8 = new UTF8Encoding();

TripleDESCryptoServiceProvider tDes = new TripleDESCryptoServiceProvider();

//set key which user for encrpyted

tDes.Key = md5.ComputeHash(utf8.GetBytes("Becks")); //改变key，就=改变encrypted 所有的内容

tDes.Mode = CipherMode.ECB;

tDes.Padding = PaddingMode.PKCS7; // set those property correctly

ICryptoTransform trans = tDes.CreateEncryptor(); // Encrpt our data

encrptyed =trans.TransformFinalBlock(utf8.GetBytes("I am Becks"),0, utf8.GetBytes("I am Becks").Length); // do the actual encrption

// 第二个参数 是 从这个string起始位置，0表示从头开始

// 第三个参数 代表 encrpyted 多少的长度， utf8.GetBytes("I am Becks").Length表示所有的

MessageBox.Show(BitConverter.ToString(encrptyed));

}

Lesson 118 Drag and Drop

需要panel， 去toolbox drag panel

然后property 设置 BorderStyle 为 FixedSingle, 这样就可以看见Border, 然后AllowDrop 点True,这样user 可以drag 进panel, 然后设置event， Drag Over

Drag Drop 是让用户drag drop file into table

private void panel1\_DragOver(object sender, DragEventArgs e)

{

e.Effect = DragDropEffects.All; // 可以Drag file over there

}

Dragdrop 显示file 的位置

private void panel1\_DragDrop(object sender, DragEventArgs e)

{

string [] files = e.Data.GetData(DataFormats.FileDrop) as string []; // return a object

foreach (string s in files) {

MessageBox.Show(s);

}

}

Lesson 119 Draw Shapes

点击panel，然后选event， 点click button

SolidBrush, Graphics, panel1.CreateGraphics()

private void panel1\_Click(object sender, EventArgs e)

{

SolidBrush sb = new SolidBrush(Color.Red); // 建立一个brush, 选取颜色为红色

Graphics g = panel1.CreateGraphics(); // 可以paint graphic

g.FillRectangle(sb, 20, 20, 50, 50);// 先选取brush，（0,0）是在左上角

//然后选position, 20是横坐标，30 是纵坐标, 50X50 是size

g.FillEllipse(sb, 60, 60, 30, 50); // 30 是横的距离，50是竖的距离

}

Lesson 120 Drawing More Shape

private void panel1\_Click(object sender, EventArgs e)

{

SolidBrush sb = new SolidBrush(Color.Blue); // 建立一个brush, 选取颜色为红色

Graphics g = panel1.CreateGraphics(); // 可以paint graphic

g.FillPie(sb,30,40,60,60,0,120);

//Pie Graph, 0 是start angle, 120 是 sweep angle，就是扫荡的角度, swweep is clockwise

Point[] points = { new Point(0, 20), new Point(0, 0), new Point(20, 0) }; // 选举 画的点，三个点就是三角形

//围着（0,20）， （0,0）, (20,0) 画线，组成三角形

g.FillPolygon(sb, points);

}

Lesson 121, 122 Drawing with Pen Class

Pen 和 Brush 区别是 一个是Draw（Pen）,一个是fill（Brush 全部上色）

Pen pen = new Pen(Color.Red);

Graphics g = panel1.CreateGraphics();

g.DrawRectangle(pen, 20, 20, 50, 50);// 20, 是横坐标位置，第二个20是纵坐标，50, 50 是长宽

Pen pen = new Pen(Color.Red);

Graphics g = panel1.CreateGraphics();

g.DrawRectangle(pen, 20, 20, 50, 50);// 20, 是横坐标位置，第二个20是纵坐标，50, 50 是长宽

g.DrawArc(pen, 20, 20, 100, 100, 0, 90); //如果90 变成360， 将是一个圆

//similar to fillpie, 第一个100是画的弧度的width，第二个100是画的高度， 0 是起始点，90 是画的弧度

g.DrawBezier(pen,new Point(20,20),new Point(30,60), new Point(70,40), new Point(50,80)); // 画一个蛇形的线，\

g.DrawLine(pen, new Point(0, 0), new Point(100, 100));

// point 是上面的点

Lesson 123 Drawing Strings Text

private void panel1\_Click(object sender, EventArgs e)

{

SolidBrush sb = new SolidBrush(Color.Blue);

Graphics g = panel1.CreateGraphics();

FontFamily ff = new FontFamily("Arial"); // 字体

System.Drawing.Font font = new System.Drawing.Font(ff,10, FontStyle.Bold);// 10 号字体,最后一个是想加粗，下划线，还是别的

g.DrawString("Adams", font, sb, new Point(20,20)); // Adams 是名字，第二参数是font,new Point(20,20) 是画的位置

}

Lesson 123 LinearGradientBrush

让颜色渐变的

LinearGradientBrush lgb = new LinearGradientBrush(new Point(20, 20), new Point(20,70),Color.Red,Color.Yellow);

//Points to start at (20,20), (20,70), 是vertical gradient， 横向的颜色不变, 颜色将会从红色逐渐变成黄色

//Points to start at (20,20), (70,20), horizontal gradient

//Points to start at (20,20), (70,70), 斜着的 gradient

Graphics g = panel1.CreateGraphics();

g.FillEllipse(lgb,20,20,50,50)

lesson 125 Multiple Colors in a LinearGradientBrush

LinearGradientBrush lgb = new LinearGradientBrush(new Point(20, 20), new Point(20,70),Color.Black,Color.Red);

//Points to start at (20,20), (20,70), 是vertical gradient， 横向的颜色不变, 颜色将会从红色逐渐变成黄色

//Points to start at (20,20), (70,20), horizontal gradient

//Points to start at (20,20), (70,70), 斜着的 gradient

Graphics g = panel1.CreateGraphics();

ColorBlend cb = new ColorBlend(); // store all colors

cb.Colors = new Color[] { Color.Black, Color.Blue, Color.White }; // set colors in the gradient, 可以set as many as color you want

cb.Positions = new float[] { 0, .5F, 1F }; //set the center of array, .5F convert to Floar, 1F convert to float

// 0 是 black, 0.5 Blue, 1 是 white

lgb.InterpolationColors = cb; // 插入的颜色贴到lgb上

g.FillRectangle(lgb, 20, 20, 50, 50);

lesson 126, 127 PathGradientBrush

private void panel1\_Click(object sender, EventArgs e)

{

GraphicsPath gp = new GraphicsPath(); // tell PathGradientBrush what to do what to draw

gp.AddEllipse(20,20,50,50); // circle 20, 20 起始位置，

PathGradientBrush pgb = new PathGradientBrush(gp);

pgb.CenterColor = Color.Red;

//从中间的红色逐渐变成黄色

pgb.SurroundColors = new Color[] { Color.Yellow };

Graphics g = panel1.CreateGraphics();

g.FillEllipse(pgb, 20, 20, 50, 50);

}

画长方形

Rectangle r = new Rectangle(20,20,50,50); // 起始20,20，长宽50,50

gp.AddRectangle(r);//因为addRectangle 不能直接加坐标，必须用Rectangle initialize

PathGradientBrush pgb = new PathGradientBrush(gp);

pgb.CenterColor = Color.Red;

//从中间的红色逐渐变成黄色

pgb.SurroundColors = new Color[] { Color.Yellow };

Graphics g = panel1.CreateGraphics();

g.FillRectangle(pgb, 20, 20, 50, 50);

画多边形

GraphicsPath gp = new GraphicsPath(); // tell PathGradientBrush what to do what to draw

Point[] points = { new Point(20, 20), new Point(20, 70), new Point(70, 20)};

gp.AddPolygon(points);

PathGradientBrush pgb = new PathGradientBrush(gp);

pgb.CenterColor = Color.Red;

//从中间的红色逐渐变成黄色

pgb.SurroundColors = new Color[] { Color.Yellow };

Graphics g = panel1.CreateGraphics();

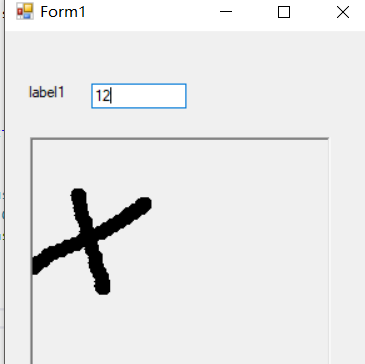
g.FillPolygon(pgb, points);

如果画多边形，Point 定义成

Point[] points = { new Point(20, 20), new Point(20, 70), new Point(70, 20), new Point(70, 70)};会显示成botai



Class 128,129,130,131,132 Paint project



public Form1()

{

InitializeComponent();

g = panel1.CreateGraphics();

}

bool canPaint = false;

Graphics g;

private void panel1\_MouseDown(object sender, MouseEventArgs e) // Move Down 是鼠标点击的时候

{

canPaint = true;

}

private void panel1\_MouseUp(object sender, MouseEventArgs e) // Move Down 是鼠标点击松开的时候

{

canPaint = false;

}

private void panel1\_MouseMove(object sender, MouseEventArgs e) // Move Down 是鼠标点击动的时候

{

if (canPaint) {

SolidBrush s = new SolidBrush(Color.Black);

g.FillEllipse(s, e.X, e.Y, Convert.ToInt32(textBox1.Text), Convert.ToInt32(textBox1.Text)); // e.X, e.Y mouse的x,y轴

//然后用户设置 线的长宽高, basically it draw ellipse 根据我们的鼠标，假如鼠标变得快乐，就是线乐

}

}

用pen

public Form1()

{

InitializeComponent();

g = panel1.CreateGraphics();

}

bool canPaint = false;

Graphics g;

private void panel1\_MouseDown(object sender, MouseEventArgs e)

{

canPaint = true;

}

private void panel1\_MouseUp(object sender, MouseEventArgs e)

{

canPaint = false;

int? preX = null;

int? prey = null;

}

int? preX = null;

int? prey = null;

private void panel1\_MouseMove(object sender, MouseEventArgs e)

{

if (canPaint) {

Pen pen = new Pen(Color.Black, float.Parse(textBox1.Text)); //

g.DrawLine(pen, new Point(preX ?? e.X, prey??e.Y), new Point(e.X, e.Y));

preX = e.X;

prey = e.Y;

}

}

改变canvas 的颜色, 清除画布，

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

g = panel1.CreateGraphics();

}

bool canPaint = false;

Graphics g;

private void panel1\_MouseDown(object sender, MouseEventArgs e)

{

canPaint = true;

}

private void panel1\_MouseUp(object sender, MouseEventArgs e)

{

canPaint = false;

preX = null;

prey = null;

}

int? preX = null;

int? prey = null;

private void panel1\_MouseMove(object sender, MouseEventArgs e)

{

if (canPaint) {

Pen pen = new Pen(toolStripButton2.ForeColor, float.Parse(toolStripTextBox1.Text)); //

g.DrawLine(pen, new Point(preX ?? e.X, prey??e.Y), new Point(e.X, e.Y));

preX = e.X;

prey = e.Y;

}

}

private void toolStripButton2\_Click(object sender, EventArgs e)

{

ColorDialog cd = new ColorDialog();

if (cd.ShowDialog() == DialogResult.OK) {

toolStripButton2.ForeColor = cd.Color;

}

}

private void toolStripButton3\_Click(object sender, EventArgs e)

{

g.Clear(panel1.BackColor);//只clear 画布的canvas的颜色，不clear 别的, 清楚panel1.BackColor 上的

}

private void toolStripButton4\_Click(object sender, EventArgs e)

{

ColorDialog cd = new ColorDialog();

if (cd.ShowDialog() == DialogResult.OK)

{

toolStripButton4.ForeColor = cd.Color;

panel1.BackColor = cd.Color;

}

}

画长方形，矩形，

public Form1()

{

InitializeComponent();

g = panel1.CreateGraphics();

}

bool canPaint = false;

Graphics g;

private void panel1\_MouseDown(object sender, MouseEventArgs e)

{

canPaint = true;

if (drawsquare) {

SolidBrush sb = new SolidBrush(toolStripButton2.ForeColor);

g.FillRectangle(sb, e.X, e.Y,50,50);

canPaint = false;

drawsquare = false;

}

else if (drawRectangle) {

SolidBrush sb = new SolidBrush(toolStripButton2.ForeColor);

g.FillRectangle(sb, e.X, e.Y, 100, 50);

drawRectangle = false;

canPaint = false;

}

}

private void panel1\_MouseUp(object sender, MouseEventArgs e)

{

canPaint = false;

preX = null;

prey = null;

}

int? preX = null;

int? prey = null;

private void panel1\_MouseMove(object sender, MouseEventArgs e)

{

if (canPaint) {

Pen pen = new Pen(toolStripButton2.ForeColor, float.Parse(toolStripTextBox1.Text)); //

g.DrawLine(pen, new Point(preX ?? e.X, prey??e.Y), new Point(e.X, e.Y));

preX = e.X;

prey = e.Y;

}

}

private void toolStripButton2\_Click(object sender, EventArgs e)

{

ColorDialog cd = new ColorDialog();

if (cd.ShowDialog() == DialogResult.OK) {

toolStripButton2.ForeColor = cd.Color;

}

}

private void toolStripButton3\_Click(object sender, EventArgs e)

{

g.Clear(panel1.BackColor);//只clear 画布的canvas的颜色，不clear 别的, 清楚panel1.BackColor 上的

}

private void toolStripButton4\_Click(object sender, EventArgs e)

{

ColorDialog cd = new ColorDialog();

if (cd.ShowDialog() == DialogResult.OK)

{

toolStripButton4.ForeColor = cd.Color;

panel1.BackColor = cd.Color;

}

}

bool drawsquare = false;

bool drawRectangle = false;

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

{

drawsquare = true;

}

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

{

drawRectangle = true;

}

把图片drop 到panel 上，在property 中设置allowdrop 为true

private void panel1\_DragEnter(object sender, DragEventArgs e)

{

e.Effect = DragDropEffects.All; //make sure you can drop 可以drag drop 在上面

}

private void panel1\_DragDrop(object sender, DragEventArgs e)

{

string[] imagepath = e.Data.GetData(DataFormats.FileDrop) as string[];

foreach(string s in imagepath)

{

g.DrawImage(Image.FromFile(s),new Point(0,0));//把图片放在（0，0）

}

}

133, 134, 135, 136,137,138 Making Controls，建造自己的control，然后可以有自己定义的property

在project 上选择user control

public UserControl1()

{

InitializeComponent();

}

protected override void OnPaint(PaintEventArgs e)

{

SolidBrush s = new SolidBrush(Color.FromKnownColor(KnownColor.Control));

Graphics g = this.CreateGraphics();

g.FillRectangle(s, 0, 0, this.Width, this.Height);//Fill 整个control的长和高

label1.Location = new Point(this.Width / 2 - label1.Width / 2, this.Height / 2 - label1.Height / 2); //永远在中间

}

然后在property中显示 buttontext，

public string ButtonText

{

get { return label1.Text; }

set { label1.Text = value; }

}

让button 上下显示不同的颜色

protected override void OnPaint(PaintEventArgs e)

{

SolidBrush s = new SolidBrush(Color.FromKnownColor(KnownColor.Control));

Graphics g = this.CreateGraphics();

g.FillRectangle(s, 0, 0, this.Width, this.Height);//Fill 整个control的长和高

s.Color = Color.FromKnownColor(KnownColor.ControlLight);

g.FillRectangle(s, 0, this.Height / 2, this.Width, this.Height / 2);

label1.Location = new Point(this.Width / 2 - label1.Width / 2, this.Height / 2 - label1.Height / 2); //永远在中间

}

把鼠标f放在usercontrol上变颜色

string text = "";

protected override void OnPaint(PaintEventArgs e)

{

DrawButton(Color.FromKnownColor(KnownColor.Control));

}

public string ButtonText

{

get { return text; }

set { text = value; }

}

private void MyButton\_MouseHover(object sender, EventArgs e)

{

Color mycolor = Color.FromArgb(255, Color.FromKnownColor(KnownColor.Control).R - 30, Color.FromKnownColor(KnownColor.Control).R - 5, 255);

DrawButton(mycolor);

}

void DrawButton(Color c) { // default private

SolidBrush s = new SolidBrush(c);

Graphics g = this.CreateGraphics();

g.FillRectangle(s, 0, 0, this.Width, this.Height);//Fill 整个control的长和高

s.Color = Color.FromArgb(255, c.R - 13, c.G - 13, c.B - 13);

g.FillRectangle(s, 0, this.Height / 2, this.Width, this.Height / 2);

PointF fpoint = new Point(this.Width / 2 - text.Length / 2, this.Height / 2 - text.Length / 2); //永远在中间

//PointF stands for flow

FontFamily ff = new FontFamily("Arial");

Font f = new System.Drawing.Font(ff, 8);

s.Color = Color.Black;

g.DrawString(text, f, s, fpoint);// f 是font, fpoint 位置

}

private void MyButton\_MouseLeave(object sender, EventArgs e)

{

DrawButton(Color.FromKnownColor(KnownColor.Control));

}

可以自己改button 的颜色，拖动，放下，是一种颜色，鼠标离开一种颜色

public MyButton()

{

InitializeComponent();

}

string text = "";

protected override void OnPaint(PaintEventArgs e)

{

DrawButton(myButtonColor);

}

public string ButtonText

{

get { return text; }

set { text = value; }

}

private void MyButton\_MouseHover(object sender, EventArgs e)

{

Color mycolor = Color.FromArgb(255, Color.FromKnownColor(KnownColor.Control).R - 30, Color.FromKnownColor(KnownColor.Control).R - 5, 255);

DrawButton(mycolor);

}

Color myButtonColor;

public Color ButtonColor {

set {

try

{

myButtonColor = value;

}

catch {

myButtonColor = Color.FromKnownColor(KnownColor.Control);

MessageBox.Show("Select a valid color");

}

}

get { return myButtonColor; }

}

void DrawButton(Color c) { // default private

SolidBrush s = new SolidBrush(c);

Graphics g = this.CreateGraphics();

g.FillRectangle(s, 0, 0, this.Width, this.Height);//Fill 整个control的长和高

s.Color = Color.FromArgb(255, c.R - 13, c.G - 13, c.B - 13);

g.FillRectangle(s, 0, this.Height / 2, this.Width, this.Height / 2);

PointF fpoint = new Point(this.Width / 2 - text.Length / 2, this.Height / 2 - text.Length / 2); //永远在中间

//PointF stands for flow

FontFamily ff = new FontFamily("Arial");

Font f = new System.Drawing.Font(ff, 8);

s.Color = Color.Black;

g.DrawString(text, f, s, fpoint);// f 是font, fpoint 位置

}

private void MyButton\_MouseLeave(object sender, EventArgs e)

{

DrawButton(myButtonColor);

}

private void MyButton\_MouseDown(object sender, MouseEventArgs e) //点击鼠标是一个颜色

{

Color mycolor = Color.FromArgb(255, Color.FromKnownColor(KnownColor.Control).R +15, Color.FromKnownColor(KnownColor.Control).R - 15, 150);

DrawButton(mycolor);

}

}

Lesson 139 Inherit from existing Controls

如果只想让user button，可以inherit from button class

把

public partial class MyButton : UserControl

改成

public partial class MyButton : Button

继承 button class， 让usercontrol 不能设置text 为adam

public override string Text {

get { return base.Text}

set {

if (value == "Adam")

{

MessageBox.Show("You are not authorize to set that value");

base.Test = "UserControl";

return;

}

base.OnTextChanged = value;

}

}

还可以override event，都不用点event，选取Click，设置的button自动，每次点，都会跳出 hello

protected override void OnClick(EventArgs e)

{

MessageBox.Show("Hello");

base.OnClick(e);

}

Lesson 140 Splash Screen, program 开始结束的界面

Tell screen is loading

点击project add new windows form，然后选new form

Make no border, Go to form 的property 选FormBorderStyle 为 None

然后拉一个picturebox, 然后点击右上角的edit，choose Image -> local resource -> import

然后点击splashscreen 的view code，让这个form show 2 seconds 之后show form 1

然后点击splashscreen event 选取 shown

Timer t;

private void SplashScreen\_Shown(object sender, EventArgs e)

{

t = new Timer();

t.Interval = 2000; // 2 seconds 2 秒后show form

t.Start();

t.Tick += T\_Tick;

}

private void T\_Tick(object sender, EventArgs e)

{

t.Stop();

Form1 f = new Form1();

f.Show();

this.Hide(); //让splash screen hide

}

然后去program.cs view code，让splash screen 先开始

static class Program

{

/// <summary>

/// The main entry point for the application.

/// </summary>

[STAThread]

static void Main()

{

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

Application.Run(new SplashScreen());

}

}

最后当form1 close，也close splash screen

private void Form1\_FormClosed(object sender, FormClosedEventArgs e)

{

Application.Exit(); //让这个程序都退出

}

Lesson 141 making DLL (Dynamic Link Library)

Put all namespace created inside DLL and then give DLL to someone else, 之后他们就可以use all the namespace inside DLL

DLL 优势，People cannot see ur code and people cannot edit your code

首先选择create a new project 然后选择class library

可以在一个classlibray中，有多个namespace

写好file 之后，点击build solution 然后点击debug, 然后即使debug有error，its okay it is building dll

namespace ClassLibrary1

{

namespace Client

{

public class Client

{

public int Age {

get;

set;

}

public string Name {

get;

set;

}

}

}

}

然后点击别的C#, 然后在别的vs 中点add reference, 然后就可以写using

Client a = new Client（）；

Lesson 142 Internal Access Modifier (Internal) 是只能在这个file access

Internal class 是只能在这个file 中用，及时用了包含这个file的

internal class Client

{

Internal int Age {

get;

set;

}  
}

// 当Age是internal，只能access age在这个file中，即使client 是public ，但是age 是internal，所以在别的file 可以access client class，但不能access age

Lesson 143 Comments and Description （写在你想用的Description 的前面）

可以对class，property，variable 做description，

如果是function （method），也可以对parameter 做description

///三个代表Description 是xml

namespace Client

{

/// <summary>

/// Represent a person Many properties about person

/// </summary> 代表class的description

internal class Client

{

public int Age {

get;

set;

}

public string Name {

get;

set;

}

}

对method 和 parameter description

/// <summary>

/// Fix the Person's name

/// </summary>

/// <param name="s"> The person's name </param>

void name(string s) { }

lesson 144 Goto Keyword And Regions

Goto Statement

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

{

string Adam = "Blah";

switch (Adam) {

case "Adam":

MessageBox.Show("Hello");

break;

default:

MessageBox.Show("The Default");

goto case "Adam";

}

}

还可以自己定label，然后goto到定义到的label

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

{

for (;;) {

goto Mycode; // 会调到Mycode, 然后不再回来原来的位置，for loop 在这会断开

}

Mycode: {

}

}

Region: Biscally hide code

然后有个可以hide code

#region MyRegion

/\*定义了

\* 的method \* /

#endregion

Lesson 145 Capturing Screen

Using bitmap to store screen capture

需要button 和picturebox

Bitmap b = new Bitmap(Screen.PrimaryScreen.WorkingArea.Width,Screen.PrimaryScreen.WorkingArea.Height); // Create a new Bitmap, use Bitmap to store screen capturing

//想要screen的长宽，多个screen的话，算primaryScreen

Graphics g = Graphics.FromImage(b); // 来自BitMap

g.CopyFromScreen(Point.Empty, Point.Empty,Screen.PrimaryScreen.WorkingArea.Size);//Point.Empty起始点

pictureBox1.Image = b;

用threading， 跟录像一样

private void pictureBox1\_Click(object sender, EventArgs e)

{

Thread t = new Thread(Blah);

t.Start();

}

void Blah()

{

while (true) {

Bitmap b = new Bitmap(Screen.PrimaryScreen.WorkingArea.Width, Screen.PrimaryScreen.WorkingArea.Height); // Create a new Bitmap, use Bitmap to store screen capturing

//想要screen的长宽，多个screen的话，算primaryScreen

Graphics g = Graphics.FromImage(b); // 来自BitMap

g.CopyFromScreen(Point.Empty, Point.Empty, Screen.PrimaryScreen.WorkingArea.Size);//Point.Empty起始点

pictureBox1.Image = b;

}

Lesson 146 Making keyboard

Keyboard shorcut

在form的property 中设置keyPreview = true

然后选择form的key down event, key down event 是key press event 之前的

private void Form1\_KeyDown(object sender, KeyEventArgs e)

{

if (e.Control && e.KeyCode.ToString() == "A")

{

// 如果user press control + A

MessageBox.Show("Adam");

}

}

如果是test box, 那么不用设置KeyPreview, 直接选择key down event

private void Form1\_KeyDown(object sender, KeyEventArgs e)

{

if (e.Alt && e.Control)

{

// 如果user press control + Alt

MessageBox.Show("Adam");

}

}

Lesson 147 Checking Controls on Leave

public Form1()

{

InitializeComponent();

comboBox1.SelectedIndex = 0; 设置default 选项，这样进来就是选择第一个

}

private void Form1\_KeyDown(object sender, KeyEventArgs e)

{

if (e.Alt && e.Control)

{

// 如果user press control + Alt

MessageBox.Show("Adam");

}

}

private void textBox1\_Leave(object sender, EventArgs e)

{

if (textBox1.Text == "") {

MessageBox.Show("You must provide a nmae");

textBox1.Select();

}

}

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

{

if (comboBox1.SelectedIndex == 0) {

MessageBox.Show("You must select a item");

comboBox1.Select();

}

}

Lesson 148，149,150,151 Overloading Operators

Operator 符号（符号左边的，符号右边的）

有些符号必须配对overload

1. == 必须overload !=
2. >= 必须overload <=
3. > 必须overload <

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

{

Item item1 = new Item();

item1.Price = 4;

Item item2 = new Item();

item2.Price = 7;

Item item3 = item1 + item2;

MessageBox.Show(item3.Price.ToString());

}

}

class Item

{

public int Price

{

set;

get;

}

public static Item operator +(Item i1, Item i2) // static 是all operator 必须是static

//返回值是Item

//i1 是left side of 加号, i2 是right side of 加号

{

Item i3 = new Item();

i3.Price = i1.Price + i2.Price;

return i3;

}

}

如果overload == 必须overload !=

class Item

{

public int Price

{

set;

get;

}

public static bool operator ==(Item i1, Item i2)

{

return i1.Price == i2.Price ? true: false;

}

public static bool operator !=(Item i1, Item i2)

{

return i1.Price != i2.Price? true: false;

}

}

如果overload < 必须overload >, 如果overload <=, 必须overload >=

public static Item operator ++(Item i1) // 可以Item i1++ 或者 ++i1

{

Item i = new Item();

i.Price = i1.Price + 1;//自动赋值值给新的item

return i;

}

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

{

Item item2 = new Item();

item2.Price = 7;

item2++;

++item2;

MessageBox.Show(item2.Price.ToString());

}

Lesson 152 Making Converison Operators

Explicit(user do it byitself to tell compiler) vs Implicit( do it automatically)

Item i = (item)3; // Explicit

Item item = 3; // Implicit

不能同时有explicit or implict

public static explicit operator Item(int ItemPrice)// 得告诉compiler 是explicit 还是implicit 符号是item

{

Item i = new Item();

i.Price = ItemPrice;

return i;

}

Explicit 和implicit 区别就是关键词

public static implicit operator Item(int ItemPrice)// 得告诉compiler 是explicit 还是implicit 符号是item

{

Item i = new Item();

i.Price = ItemPrice;

return i;

}

Lesson 153 Ref and Out keywords

Ref 表示是pass by reference

Method 参数和call 的参数都要带ref

要带入method 的参数必须要初始化，给他值int age = 3;

同时，也不能说 Modify(ref 5); 必须代入variable

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

{

int age = 3;

string name;

Modify(ref age, out name);

MessageBox.Show(age.ToString());

}

void Modify(ref int age, out string name) {

age += 5; //

name = "becks";

}

Out 代入method，不能赋值，只用带着variable 的名字

如果代入out 进method，必须带着

Lesson 154

Lesson 166 Optional Parameters

void ShowMessage(string message,string title = "") { // default value

MessageBox.Show(message, title);

}

Make Sure that default message is in the end,

ShowMessage("Adam")； 这个就不知道Adam 是属于哪个parameter

void ShowMessage(string message = "",string title = "") { // default value

MessageBox.Show(message, title);

}

Lesson 167 IEnumerable and Yield Returns

IEnumerable: collections of objects

using System.Collections; //添加这个在头文件

function 会把每个值return 出去

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

{

GetNumbers(0, 10); //only use when we need them, 如果不use, 不会run

foreach (int i in GetNumbers(0, 10)) {

MessageBox.Show(i.ToString());//show every number in 0, 10

}

}

IEnumerable GetNumbers(int min, int max) {

for (; min <= max; min++) {

yield return min; // return everything inside in

}

}

Log4net

Appender： log 的形式

在App.Config 中更改一下, 可以在running time 更改它

Create a file called log4net

<configSections>

<section name = "log4net" type = "log4net.Config.Log4NetConfigurationSectionHanlder, log4mnet" />

</configSections>

Appender 是what do we want to log to

可以解决do we have permission to log file / DB ?

<appender name ="ConsoleAppender" type ="log4net.Appender.ConsoleAppender">

Inside append, we need to tell appender how to write message: layout

How do you want for the layout

</conversionPattern value ="%date{ABSOLUTE} [%thread] %level - %message%newline%exception" // the pattern you want to write out, date is local date

{ABSOLUTE} pattern that tell us that specific way to layout time

<root> tell which appender to use and specify level for logging to use

</root>

<level value ="DEBUG" /> low value of debug, log debug, info, warn, Error, Fatal

<level value ="ALL" /> Similar to Debug

<level value ="ERROR" /> only log Error and fator

<level value ="OFF" /> nothing to log

<appender-ref ref ="ConsoleAppender"/> 与 <appender name ="ConsoleAppender" type ="log4net.Appender.ConsoleAppender"> ref 与name 名字

<?xml version="1.0" encoding="utf-8" ?>

<configuration>

<configSections>

<section name = "log4net" type = "log4net.Config.Log4NetConfigurationSectionHanlder, log4net" />

</configSections>

<log4net>

<appender name ="ConsoleAppender" type ="log4net.Appender.ConsoleAppender">

<layout type ="log4net.Layout.PatternLayout">

<conversionPattern value ="%date{ABSOLUTE} [%logger] %level - %message%newline%exception"/>

</layout>

</appender>

<appender name ="FileAppender" type ="log4net.Appender.FileAppender">

<file value ="C:\Users\Xu Zhao\Desktop\Divisa Capital\11.txt" />

<appendToFile value ="true"/>

<lockingModel type ="log4net.Appender.FileAppender+MinimalLock"/>

<layout type ="log4net.Layout.PatternLayout">

<conversionPattern value ="%date{ABSOLUTE} [%logger] %level - %message%newline%exception"/>

</layout>

</appender>

<root>

<level value ="DEBUG" />

<appender-ref ref ="ConsoleAppender"/>

<appender-ref ref ="FileAppender"/>

</root>

</log4net>

<startup>

<supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5" />

</startup>

</configuration>

在cs program 里面

[assembly: log4net.Config.XmlConfigurator(Watch =true)]

读取xmlconfigurator watch = true 等于 在running time如果更改，它会发现

<conversionPattern value ="%date{ABSOLUTE} [%logger] %level - %message%newline%exception"/>

%logger, 说是那个class 产生的file

private static readonly log4net.ILog log = log4net.LogManager.GetLogger(System.Reflection.MethodBase.GetCurrentMethod().DeclaringType);

<file name ="C:\Users\Xu Zhao\Desktop\1.txt"/>

<appendToFile value ="true"/>

<lockingModel type ="log4net.Appender.FileAppender+MinimalLock"/>

LockingModel is to tell to log as easyily as possible from different thread

添加完fileappender 之后，记得在root里面添加 appender -ref

Gmt time with milliseconds

<conversionPattern value ="%date{yyyyMMdd-HH:mm:ss.fff} [%logger] %level - %message%newline%exception"/>

RollingFileAppender is when you get get 10 MB, 将会remane them as FollingFile.1 直到 5

<appender name ="RollingFileAppender" type ="log4net.Appender.RollingFileAppender">

<file value ="C:\Users\Xu Zhao\Desktop\Divisa Capital\FollingFile.txt">

<appendToFile value ="true" />

<rollingStyle value = "Size" />

<maximumFileSize value ="10MB" />

<maxSizeRollBackups value ="5" />

<staticLogFileName value ="true"/> 保证filename 不改变

我们将会有5个backup files,

然后在root 里面

<root>

<level value ="DEBUG" />

<appender-ref ref ="ConsoleAppender"/>

<appender-ref ref ="FileAppender"/>

<appender-ref ref =" RollingFileAppender"/>

</root>

<appender name ="AdoNetAppender" type ="log4net.AdoNetAppender"> 在传入sql 用的

<bufferSize value ="10"/> instead of writing query into mysql, every

如果我们不想改root的信息，只capture warn and error message， (debug 是最低的level), fatal 是最高的

<appender name ="ConsoleAppender" type ="log4net.Appender.ConsoleAppender">

<filter type ="log4net.Filter.LevelRangeFilter">

<levelMin value ="WARN" />

<levelMax value ="ERROR"/>

</filter>

<layout type ="log4net.Layout.PatternLayout">

<conversionPattern value ="%date{ABSOLUTE} [%logger] %level - %message%newline%exception"/>

</layout>

</appender>

Find pattern inside of our log message 如果只想要特定的string 一定写上<filter type ="log4net.Filter.DenyAllFilter"/> 因为当filter 看见不满足 Maintence， 就会去别的filter，但是既然没有别的filter了，就assume log this info，所以必须加上denyallfilter

<filter type ="log4net.Filter.StringMatchFilter">

<stringToMatch value ="Maintence" />

</filter>

<filter type ="log4net.Filter.DenyAllFilter"/>

只想要debug 和 Error level 的message

<filter type ="log4net.Filter.LevelMatch">

<levelToMatch value ="Error"/>

</filter>

<filter type ="log4net.Filter.LevelMatch">

<levelToMatch value ="Debug"/>

</filter>

<filter type="log4net.Filter.DenyAllFilter"/>

<layout type ="log4net.Layout.PatternLayout">

<conversionPattern value ="%date{ABSOLUTE} [%logger] %level - %message%newline%exception"/>

</layout>

<conversionPattern value ="%date{yyyyMMdd-HH:mm:ss.fff} [%logger] %timestamp "/>

Timestamp 是the number of millisecond since application

Work

<conversionPattern value ="%utcdate{ABOSULTE}%newlineTimestamp = %timestamp identity%identity%newline username: %username %location" %method />

%identity 有时为空

%username 是login username

Work great in debug, 但是在running 的时候，有可能是空的，有可能message doesn’t make sense

%location 是 program的位置

%method 是哪个method，比如哪个function called

%.5message .5 代表5个character，如果多余5个只能留5个， 最后5个character

%2.5message 最少两个character， 最多5个character，如果不到2个，会留space，add left side， 比如传入H， 不到两个，会写上space H

在code 里加上

如果定义成classm Counter j = new Counter(); 然后把j assigned 到log4net的property，以后就不用改了，改了，log4net property 的就改了，因为j 设置了Property, 所以每次call property 时候call class j 就会call class 的property

log4net.GlobalContext.Properties["Counter"] = j;//put properties

<conversionPattern value ="%preoperty{Counter} [%logger] %timestamp "/>

自己上property 然后log 到log 上

Step for using log4net:

[assembly: log4net.Config.XmlConfigurator(Watch =true)]

1. Create a config to do the log4net

完整的log4net 的config

<?xml version="1.0" encoding="utf-8" ?>

<configuration>

<configSections>

<section name = "log4net" type = "log4net.Config.Log4NetConfigurationSectionHanlder, log4net" />

</configSections>

<log4net>

<appender name ="ConsoleAppender" type ="log4net.Appender.ConsoleAppender">

<layout type ="log4net.Layout.PatternLayout">

<conversionPattern value ="%date{ABSOLUTE} [%logger] %level - %message%newline%exception"/>

</layout>

</appender>

<appender name ="FileAppender" type ="log4net.Appender.FileAppender">

<file value ="C:\Users\Xu Zhao\Desktop\Divisa Capital\11.txt"/>

<appendToFile value ="true"/>

<lockingModel type ="log4net.Appender.FileAppender+MinimalLock"/>

<layout type ="log4net.Layout.PatternLayout">

<conversionPattern value ="%date{yyyyMMdd-HH:mm:ss.fff} [%logger] %level - %message%newline%exception"/>

</layout>

</appender>

<appender name ="RollingFileAppender" type ="log4net.Appender.RollingFileAppender">

<file value ="C:\Users\Xu Zhao\Desktop\Divisa Capital\RollFile.txt" />

<appendToFile value ="true" />

<rollingStyle value = "Size" />

<maximumFileSize value ="1MB" />

<maxSizeRollBackups value ="5" />

<staticLogFileName value ="true"/>

<layout type ="log4net.Layout.PatternLayout">

<conversionPattern value ="%date{yyyyMMdd-HH:mm:ss.fff} [%logger] %level - %message%newline%exception"/>

</layout>

</appender>

<root>

<level value ="DEBUG" />

<appender-ref ref ="ConsoleAppender"/>

<appender-ref ref ="FileAppender"/>

<appender-ref ref ="RollingFileAppender"/>

</root>

</log4net>

<startup>

<supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5" />

</startup>

</configuration>

SQL

namespace SAXO\_SOUP

{

class MysqlWrapper

{

private const string server = "ukdb.divisacapital.com";

private const int port = 3306;

private const string UID = "becks";

private const string password = "qSH@hnNZT%D3mdRr9$";

private const string DATABASE = "becks";

private static MySqlConnection dbConn;

public static void InitializeDB() {

MySqlConnectionStringBuilder build = new MySqlConnectionStringBuilder();

build.Server = server;

build.UserID = UID;

build.Password = password;

build.Database = DATABASE;

build.Port = port;

string connString = build.ToString();

build = null;

Console.WriteLine(connString);

dbConn = new MySqlConnection(connString);

}

public static void data() {

string query = "SELECT \* FROM TEST";

MySqlCommand cmd = new MySqlCommand(query, dbConn);

dbConn.Open();

MySqlDataReader reader = cmd.ExecuteReader();

while (reader.Read())

{

Console.WriteLine(reader[0]);

Console.WriteLine(reader[1]);

}

dbConn.Close();

}

}

}