1. OpenFileDialog打开文件对话框

ofd.Filter = "DAT文件|\*.DAT|所有文件|\*.\*";//过滤打开对话框可选的文件格式

ofd.Multiselect//判断是否可以多选，默认false

ofd.CheckFileExists//判断打开的文件是否存在，不存在发警告

ofd.AddExtension//是否自动添加拓展名

ofd.AutoUpgradeEnabled//是否自动升级外观和行为

1. ListView列表控件

属性：

View//显示视图样式，其中list为横行列表

事件：

DragEnter//鼠标拖放文件进入时发生

DragDrop//鼠标拖放的文件操作完成时发生

方法：

Contains//判断元素是否存在，传入列表类型的参数，返回bool值

Find//指示是否包含某元素，传入string类型参数，返回元素的数组

PS：

列表元素属性text可以取到项中的内容，而使用item.tostring获得的不只是内容，而是列表名：{"内容"}

PS：判断列表中某元素是否存在

当需要判断列表中是否存在某个字符串时，可以通过Find返回的数组的长度判断是否存在

1. 拖放文件相关操作

PS：控件事件方法中MouseEventArgs e 为事件鼠标参数,因此,e可以看成是鼠标

DataFormats指示拖入的文件格式

e.Data.GetDataPresent判断拖入的文件格式是否符合，返回bool类型

DragDropEffects对拖入的文件操作效果

e.Effect获得操作效果的属性

e.Data.GetData获取拖入的符合指定格式的文件，返回一个object对象，里面包含所有已选文件

1. object赋值给数组

PS：数组虽然也是一个对象，但是object对象无法赋值给明确定义的数组，如string[]、int[]。object可以赋值给数组的基类Array，再通过该基类遍历成明确定义的数组

1. 一些动态库.DLL

kernel32.dll：

是Windows 9x/Me中非常重要的32位[动态链接库文件](http://baike.so.com/doc/6322493.html)，属于内核级文件。它控制着系统的[内存管理](http://baike.so.com/doc/5020618.html)、数据的输入输出操作和[中断处理](http://baike.so.com/doc/4973549.html)，当Windows启动时，kernel32.dll就驻留在内存中特定的写保护区域，使别的程序无法占用这个内存区域。

1. 反射

微软官方的MSDN定义：

通过 [System.Reflection](http://msdn.microsoft.com/zh-cn/library/system.reflection.aspx) 命名空间中的类以及 [System.Type](http://msdn.microsoft.com/zh-cn/library/system.type.aspx)，您可以获取有关已加载的[程序集](http://msdn.microsoft.com/zh-cn/library/k3677y81.aspx)和在其中定义的类型（如[类](http://msdn.microsoft.com/zh-cn/library/2s9w552e.aspx)、[接口](http://msdn.microsoft.com/zh-cn/library/f9846y7x.aspx)和[值类型](http://msdn.microsoft.com/zh-cn/library/34yytbws.aspx)）的信息。 您也可以使用反射在运行时创建类型实例，以及调用和访问这些实例。

我们把它理解成计算机里的反射就是我用某中对象去反射我无法直接获取到的东西

反射实例范本：

在A项目中，

调用Assembly的方法LoadFile（项目B路径字符串）获得B项目，此方法返回值为Assembly类型

再调用上一返回对象的方法GetType（类名字符串，bool）获得类，返回Type类型

再调用上一返回对象的方法GetMethod（方法名字符串）获得该类的方法，返回MethodInfo类型

最后调用上一返回对象，完成A项目中调用B项目中方法的反射

1. ZK读取DAT文档另存为txt文档的解密算法：

传入要转换的文件的内容的字节数组，遍历字节数组，第i个元素-‘x’，并赋值给h。第i+1个元素强转成short类型后-‘z’。接着将h左移4位再+1后强转成byte。然后将此字节数组第一位赋给一个char数组的第j位，执行j++、i+=2

1. 设置XML文档的内容

8.1 获得XML文档

即创建XmlDocument类的实例化对象doc

By the way：实例化DirectoryInfo类对象并通过构造方法设置参数可以获得目录，同理，实例化FileInfo类对象可以获得文件路径

若存在XML文件，则调用doc.Load（文件全路径）加载XML文档；若不存在XML文档，则将XML开头标志字符串通过流写入文件全路径中，再调用doc.LoadXml（字符串）从字符串中加载XML文档

8.2 获得根节点

即创建XmlNode类的实例化对象root，调用doc.DocumentElement属性

8.3 获取配置节点，即子节点

若根节点存在，则调用root.ChildNodes返回一个XmlNodeList对象、节点集合，若不存在，则调用doc.CreateNode（XmlNodeType.Element，指定的节点名，空字符串）创建返回一个XmlNode类节点，然后通过root.AppendChild将刚创建的节点追加为根节点的子节点

8.4 ZKAccess中的ConfigManager项目中的initLang类中子节点的操作

1）从根节点中获取子节点”ComboBoxs”，（得到combobox类型的节点）

2）若1）中获取的节点存在，那么从这个子节点中获取再下一级的节点”combobox”，且这个combobox包含属性（即子节点）--node.Attributes.Count>0，第一个属性的值.Attributes[0].Value为指定的名称。否则类似8.3，调用doc.CreateNode（XmlNodeType.Element，指定的节点名，空字符串）创建返回一个XmlNode类节点，然后给这个节点添加已经有名称的属性，即调用doc.CreateAttribute(属性名)返回XmlAttribute对象并调用newnode.Attributes.Append(XmlAttribute对象)添加属性，再通过node.AppendChild（newnode）将刚创建的节点追加为根节点的子节点。

3）若2）中获取的节点存在，那么遍历将要保存入XML文档的二维键值对集合，从这个子节点中获取再下一级的节点”Type”（反潜或联动的类型，一般值为数字，即用数字代表类型，二维键值对集合的键），且这个Type包含属性（即子节点）--node.Attributes.Count>0，第一个属性的值.Attributes[0].Value为指定的名称。与2）中否则一样。

4）若3）中获取的节点存在，那么遍历二维键值对中的值，即一维键值对，从这个子节点中获取再下一级的节点（反潜或联动中代表门关系的数字，即一维键值对的键），最后将一维键值对的值赋给对应的节点的具体数据。与2）中否则一样。

1. 类KeyPressEventArgs：指定在用户按键时撰写的字符

当用户按下任意键时，发生 KeyPress 事件。与 KeyPress 事件紧密相关的两个事件为 [KeyUp](https://msdn.microsoft.com/zh-cn/library/system.windows.forms.control.keyup(v=vs.80).aspx) 和 [KeyDown](https://msdn.microsoft.com/zh-cn/library/system.windows.forms.control.keydown(v=vs.80).aspx)。当用户按下某个键时，KeyDown 事件先于每个 KeyPress 事件发生；当用户释放某个键时发生 KeyUp 事件。当用户按住某个键时，每次字符重复时，KeyDown 和 KeyPress 事件也都重复发生。一个 KeyUp 事件在释放按键时生成。

属性：

KeyChar：获取或设置用户按下键对应的字符，如e.KeyChar>=65297&&…<65307表示按下键对应字符是全角字符

Handled：将 [Handled](https://msdn.microsoft.com/zh-cn/library/system.windows.forms.keypresseventargs.handled(v=vs.80).aspx) 设置为 true，以取消 KeyPress 事件。这可防止控件处理按键，即为true时说明已处理过事件。

1. 正则表达式

浅解：

**正则表达式**，又称**正规表示法**、**常规表示法**(英语:Regular Expression，在代码中常简写为regex、regexp或RE)，计算机科学的一个概念。在[计算机](http://baike.so.com/doc/3435270.html)科学中，是指一个用来描述或者匹配一系列符合某个句法规则的[字符串](http://baike.so.com/doc/630752.html)的单个字符串。在很多文本[编辑器](http://baike.so.com/doc/6781941.html)或其他工具里，正则表达式通常被用来检索和/或替换那些符合某个模式的文本内容。

正则表达式由一些普通[字符](http://baike.so.com/doc/2529629.html)和一些[元字符](http://baike.so.com/doc/6706942.html)（metacharacters）组成。普通字符包括大小写的字母和数字，而元字符则具有特殊的含义（可查询元字符表）

基本术语：

正则表达式应用——替换指定内容到行尾

正则表达式应用——数字替换

正则表达式应用——删除每一行行尾的指定[字符](http://baike.so.com/doc/2529629.html)

正则表达式应用——替换带有半角括号的多行

正则表达式应用——删除空行

正则表达式应用——实例应用：通过普通字符和元字符组成并完成一系列命令

类Regex：表示不可变的正则表达式

.IsMatch：指定的正则表达式在输入字符串中是否找到匹配

1. 一些陌生的类
2. ManagementClass：管理类，构造函数可指定路径：Win32-LogicalDisk和Win32-Process，前者表示磁盘驱动器，后者表示进程
3. ManagementObjectCollection：管理对象和管理类的集合类
4. ManagementObjectSearcher：可执行WMI查询
5. Default

1）熟悉的使用

在switch语句中，如果没有任何 case 表达式与开关值匹配，则控制传递给跟在可选 default 标签后的语句。如果没有 default 标签，则控制传递到 switch 以外

2）泛型应用

需要在不知道类型参数为值类型还是引用类型的情况下，为对象实例赋初值

如果我们用int型来绑定泛型参数，那么T就是int型，那么注释的那一行就变成了 int t = null；显然这是无意义的。

class TestDefault<T>

{  
     public T foo()  
     {  
         T t = null; //???  
         return t;  
     }  
}

class TestDefault<T>  
{  
    public T foo()  
    {  
        return default(T);  
    }  
}

总结

给定参数化类型 T 的一个变量 t，只有当 T 为引用类型时，语句 t = null 才有效；只有当 T 为数值类型而不是结构时，语句 t = 0 才能正常使用。解决方案是使用 default 关键字，此关键字对于引用类型会返回空，对于数值类型会返回零。对于结构，此关键字将返回初始化为零或空的每个结构成员，具体取决于这些结构是值类型还是引用类型。

1. where（泛型类型约束）

可以实现接口约束（public class MyGenericClass<T> where T:IComparable { }

）、基类（结构）约束（class MyClass<T, U>

where T : class

where U : struct

{ }）

还可以包括构造函数约束。 可以使用 new 运算符创建类型参数的实例；但类型参数为此必须受构造函数约束 new() 的约束。 [new() 约束](https://msdn.microsoft.com/zh-cn/library/sd2w2ew5.aspx)可以让编译器知道：提供的任何类型参数都必须具有可访问的无参数（或默认）构造函数。new() 约束出现在 where 子句的最后

1. 位域

位域是指信息在存储时，并不需要占用一个完整的字节， 而只需占几个或一个二进制位。为了节省存储空间，并使处理简便。

在枚举定义前+[Flags]即可以把此枚举作为位域处理

1. sql语句
2. VS中对sql规定DateTime的格式

Access数据库：

string.Format("[time]>=#{0}# and [time]<#{1}#", dateTime.Date.ToString("yyyy-MM-dd HH:mm:ss"), dateTime.Date.AddDays(1).AddSeconds(-1).ToString("yyyy-MM-dd HH:mm:ss"))

SQL Server数据库：

string.Format("aml.[time]>={{ts '{0}'}} and aml.[time]<{{ts '{1}'}}", dateTime.Date.ToString("yyyy-MM-dd HH:mm:ss"), dateTime.Date.AddDays(1).AddSeconds(-1).ToString("yyyy-MM-dd HH:mm:ss"))

存放类名：AccMonitorLog

1. 排序

Order by time desc根据时间降序排序

1. 给数据表添加字段

alter table UserLevel add column 字段名 数据类型 如:varchar(6)

1. Access数据库中的自动编号

在Access中，自动编号是一个数据类型，用AUTOINCREMENT代替int等直接作为数据类型即为自动编号

1. mid函数

如MID（列1，起始位置int（1开始），几位int） from 表1，表示在表1的列1中提取起始位置开始到后几位。

1. weekday函数

16.DevExpress.XtraReports第三方报表控件

xrTable：可右键添加行列，默认无边框

属性Borders可以设置边框

属性TextAlignment可以设置文本位置，可居中

在代码中赋值：

通过自定义内存表存储从数据库中得到的表格（为了避免为空）

在这个控件中，一般是通过第一行的绑定而自动填充下面的数据的，使用方法.DataBindings.Add(“表中代表一列的字段”,数据源（即要录入的自定义表）,”自定义表中的对应字段”；当然若要使其自动填充，那么在此之前需要将自定义表先绑定到数据源中，即是将表赋值给this.DataSource。

17.combobox控件

其值可以通过ValueMember与DisplayMember属性来获取，类似于键值对，例如定义一个键值对，将其存入一个表中，而此表作为combobox的数据源，然后此前将表的列名与ValueMember、DisplayMember两属性的名称一致即可绑定数据

此控件的项不能出现null，因此.items.add（）不能添加null，否则将抛异常

18.将一个窗体的控件拖入另一个窗体

1）在新窗体中声明该控件，并实例化该控件对象

2）给该控件的基本属性赋初值

3）将该控件添加到父窗体中

19. DevExpress.GridControl与DevExpress.Gridview

两个都是列表控件，control中列visibled为true便是view

但是，在列表控件中配有过滤器，即可以根据条件显示要显示的行，而这样做不会使得control的列的visibled变为false，即过滤器不影响view。

By the way：1）可以通过列的.FilterInfo.FilterString属性得到此列的过滤条件

2）通过DevExpress.Gridview的ActiveFilterString属性可以得到全部的过滤条件

**PS**：在窗体中，有关闭与取消选中两种方式确定是否使用过滤条件，若选择关闭即是清除过滤条件，而若选择取消选中即是依然存在过滤条件但是不使用

3）DevExpress.Gridview的ActiveFilterEnabled可以判断过滤条件是否使用，即过滤条件是否被选中

20.师傅传授：破解版制作

1）使用次数存储在注册表中，可以写一个dll文件，每打开一次软件就删一次注册表

2）写一个配置文件，使每次使用时的时间与次数不符合实际

3）写一个配置文件，使验证号每次更新

21.值传递与引用传递的传参问题

值传递通过形实参传递数值时，不论方法中的值怎么改变，外面的那个变量也不会改变；而引用传递则不同，内外变量的数值是同步的（string除外）

22. dynamic与var

dynamic被编译后，实际是一个object类型，只不过编译器会对dynamic类型进行特殊处理，让它在编译期间不进行任何的类型检查，而是将类型检查放到了运行期。

var实际上是编译期抛给我们的“语法糖”，一旦被编译，编译期会自动匹配var 变量的实际类型，并用实际类型来替换该变量的申明，这看上去就好像我们在编码的时候是用实际类型进行申明的。

以var声明的变量，支持“智能感知”，因为visual studion能推断出var类型的实际类型，而以dynamic声明的变量却不支持“智能感知”，因为编译器对其运行期的类型一无所知。对dynamic变量使用“智能感知”，会提示“此操作将在运行时解析”。

23. MessageBox

使用MessageBox时，如果使用了YESNO 或者OKCANCEL时，右上角的关闭按钮会变灰。即Eabled被设置为false

24. #if-#endif

#if一般以#endif结束，指告诉编译器，当满足条件时才编译。

而if是条件判断，是直接编译与执行的。

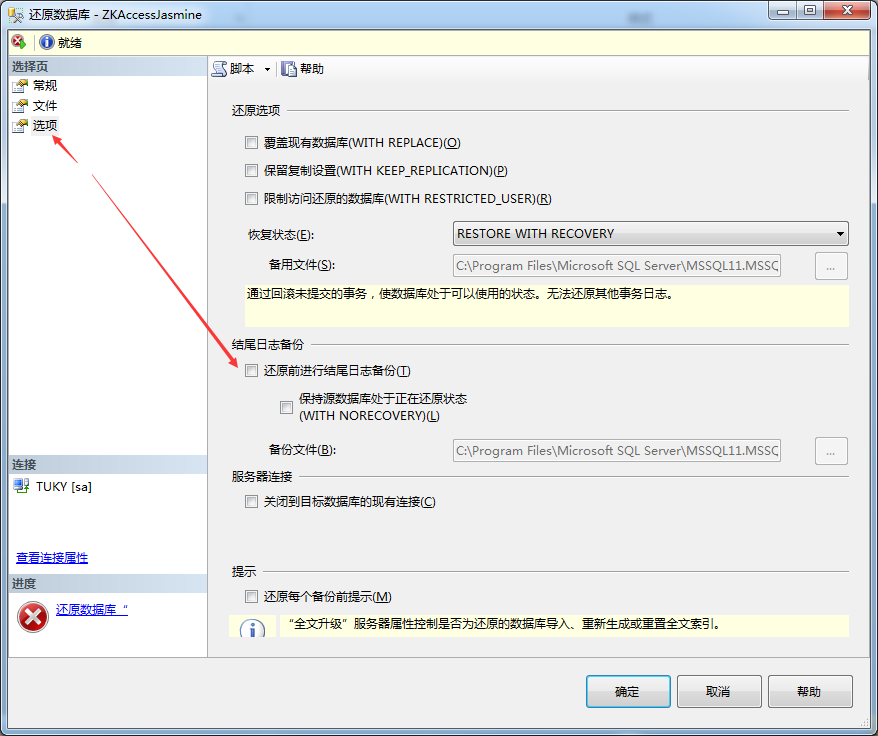
25．类库的引用

在每一个C#的引用类库中，当右键选择属性时，均有嵌入选项可选，当选择true时，则说明此类库将从指定路径引用，此时若将生成的dll给客户使用，则软件也将从指定路径获取此类库，而当选择false时，则说明此类库从指定路径引用后当生成dll时会一并复制，客户使用时软件从当前路径获取生成时复制的类库。

26.sqlserver数据库还原

问题1：System.Data.SqlClient.SqlError: 无法执行 BACKUP LOG，因为当前没有数据库备份。

解决方案1：结尾日志问题，如图去掉结尾日志即可

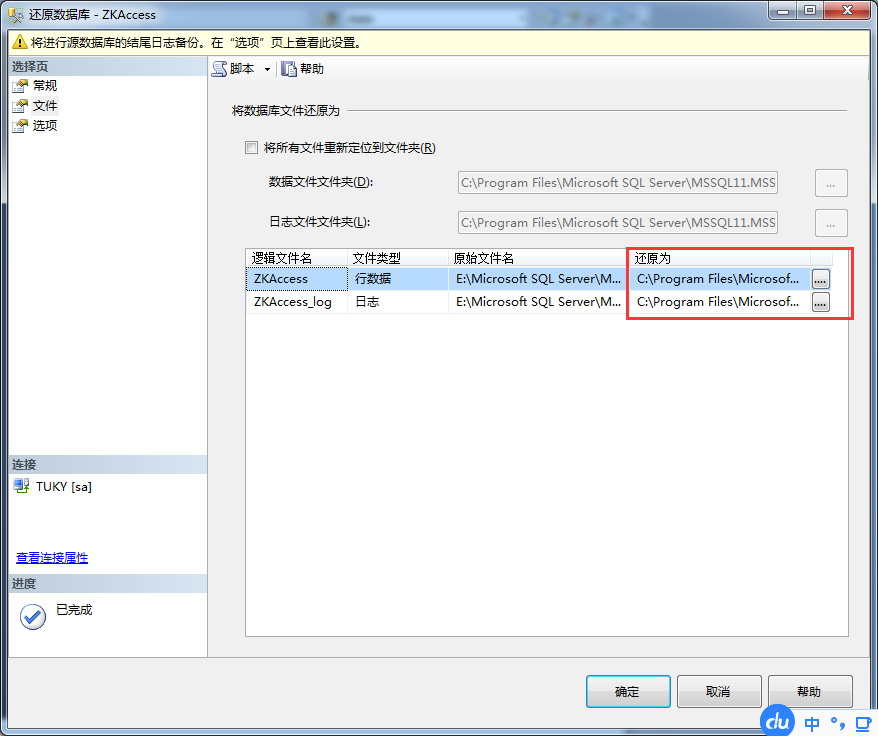


问题2：System.Data.SqlClient.SqlError: 备份集中的数据库备份与现有的 'xxx' 数据库不同。

解决方案2: 1）删除已存在的同名数据库 2）将问题1配图中覆盖现有数据库勾选

问题3：[sql System.Data.SqlClient.SqlError: 无法覆盖文件 'C:\Program Files\Microsoft SQL Server\MSSQL\data\xxx.MDF'。数据库 'xxx' 正在使用该文件。](http://www.cnblogs.com/dekevin/archive/2012/07/24/2605968.html)

解决方案3：与2有些类似，路径下的xxx.MDF已经存在，只需要将图示路径中的xxx.MDF改名即可



问题4：提示“不能将值NULL插入列'id'，表'web.dbo.player'；列不允许有空值。Insert失败”

解决方案：首先，此列被设置了不能为空值，其次，若此列还作为主键列，那么它必须是被标识的，即将属性中的标识改为是，这样才会自增长，不予与赋值也不会为null。

27. DevExpress控件GridView

DevExpress.XtraGrid.Views.Grid.GridView也是一个列表控件

属性：ColumnEdit可以控制列所显示的类型，如check或者text等

事件：CustomDrawCell、CustomDrawColumnHeader—这两个事件会为列表中除了列名一行外的所有行自动绑定列名行的所设置的图片，如构造函数中为列名行设置了复选框，经过这两个事件中的其中一个，列表中其他所有行都会被设置好复选框。

28.List集合与自定义对象实例

当需要循环给List添加对象实例，即使用List方法Add时，每进一次循环均需要重新new一次对象实例，重新分配空间，否则直接修改对象的值的话会令已经添加到List的对象的值也修改。

注意：

其sort排序规则比较难懂，测试：若list只有两个元素，在某种排序规则下会比对5次。值1与值2，值2与值1，值2与值2，值2与值1，值1与值2

29.SMTP协议

SMTP协议用来发送和接收邮件

1. 使用时一定要先设置 EnableSsl和UseDefaultCredentials，再实例化Credentials，否则会抛出异常Error: need EHLO and AUTH first
2. 异常：mail from address must be same as authorization user

解决方案：发件人的PC端上所用的邮箱客户端必须开启POP3和SMTP，且对应SMTP的账号是活跃状态。

30. c# 软件安装或者打开权限设置

1.打开

在源码的Properties目录中找到 app.manifest,将其中level="asInvoker" 改成 level="requireAdministrator"，即可实现自动适应管理员权限运行。



2.安装

需要修改Setup脚本

31.C#设置数据库密码

Access：sql语句——ALTER DATABASE PASSWORD 新密码 旧密码，其中null表示无密码

若只用普通的连接字符串，将会出现“不能改变一个打开的共享数据库的密码。”的异常，此时需要在连接字符串中添加”mode=12;”表示以独占的方式打开，这样才可以修改密码。

32.特殊字符

1.$字符

可以相当于string.Format，只是将其中的占位符里面的数字改为对应的变量名而已。