

WebFrame 简介

一、WebFrame 命名空间

系统框架所使用的枚举、接口和基础类。

1、枚举

枚举名称	说明
public enum SearchOperator	Equal, BiggerAndEqual, Bigger, SmallerAndEqual, Smaller, NotEqual, Contains, LeftContains, RightContains, Collection, NullValue, NotNullValue, UserDefine, None 作用：用于定义查询字段表达式，例如： <code>SearchField search1 = new SearchField("sname", "jin shouji", SearchOperator.Contains);</code>
public enum SearchFieldType	StringType, NumericValue
public enum JoinType	InnerJoin, LeftJoin, RightJoin, FullJoin
public enum WebDbType	SqlServer, Oracle, Oledb, Odbc
public enum JDataType	String, Integer, Numeric, DateTime, Email
public enum JCommandType	SqlID, Sql, Table
public enum JParaType	String, RequestForm, RequestQueryString, Controllist, Control, Cookie, Cookies, Session, Application, ContextItem 作用：定义参数源的类型
public enum JButtonType	BusinessRule = 1, SimpleAction, RedirectButton, SearchButton, OrderButton, NoJButton = 0
public enum SimpleButtonType	NoAction, DeleteData, UpdateData

2、接口

接口定义	说明
✧ public interface ISearchControl	实现查询控件的接口
String SearchExpress { get; set; } SearchField GetSearchCondition();	SearchExpress="sname='{0}'"
✧ public interface IJControl	实现 JControl 控件的接口
String Caption { get; set; } Object ObjValue { get; set; }	

Boolean AllowNullValue { get; set; } String ErrorInfo { get; set; } Control Me { get; } JDataType DataType { get; set; } String DataField { get; set; } String NextFocusControl { get; set; } String EqualControl { get; set; } bool CheckData(); bool AutoFillValue { get; set; }	
✧ public interface IJTextControl : IJControl	文本框的扩展接口
int MinLength { get; set; } int MaxLength { get; set; } String RegExpress { get; set; } String RegExpressSample { get; set; }	
✧ public interface IAppListControl	用于数据字典的接口
String KeyItem { get; set; } String BlankValue { get; set; } String DefaultValue { get; set; }	
✧ public interface IJButton	定义 JButton 的接口
JButtonType JButtonType { get; set; } Control Me { get; } String DataSourceID { get; set; } String UniqueValidatorControlID { get; set; } JDataSource DataSource { get; } bool IsValidatorData { get; set; } bool ValidatorDataSucc { get; set; } String ControlList { get; set; } String NoControlList { get; set; } List<Control> GetControlList(); List<ParameterItem> ParaItems { get; } AssemblePara AssemblePara { get; } ButtonSearchPara SearchPara { get; } ButtonSortPara SortPara { get; } ButtonExecutePara ExecutePara { get; } ButtonSimpleActionPara SimpleActionPara { get; } String UrlReferrer { get; set; }	

3、类

类定义	说明
✧ public sealed class FrameLib	框架基础类
public static String Root {get;}	得到当前程序运行的根目录
public static String UserID {get;set;}	当前登录的用户 ID
public static String GetSpecialVariableStr(String str1);	处理字符串中的特殊变量
public static String HostAndPort {get;}	得到当前程序的域名和端口字符串
public static void CreateCheckCodeImage()	创建一个特殊的随机码图片，可通过 CheckCode 的 Cookie 得到该图片的随机值
public static Object ExecuteMethod (String assemFile, String className, object[] conParams, String methodName, object[] methParams);	执行一个反射 程序集文件名 类的全名称 构造函数的参数值，可为 null 实例方法名 方法值，也可为 null
public static bool CheckData(IJButton button);	检查一个 IJbutton 按钮所控制的 UI 控件的值是否有效
public static void ExecuteButtonInfo(IJButton button);	按定义的参数显示的转向，适合自定义的有效性检查后的显示

二、WebFrame.Util 命名空间

常用的实用类。

1、类 JAjax

类属性和方法	说明
✧ public sealed class JAjax	Ajax 脚本类
public static void Alert(string message)	显示一条信息
public static void AlertAndFocusControl(string message, String focusControlID)	显示一条信息，并聚焦到某控件
public static void AlertAndClose(string message)	显示一条信息，并关闭当前窗口
public static void AlertAndGoBack(string message)	显示一条信息，并返回到上一页
public static void AlertAndGoUrl(string message, string newurl)	显示一条信息，并转向到 newurl，支持 #
public static void AlertAndGoUrl(string message, string newurl, bool show)	Show 为 true，同上，为 false 不显示信息，直接转向
public static void ExecuteJS(string JS)	执行普通的 js 脚本

2、类 JControl

类属性和方法	说明
✧ public sealed class JControl	Control 实用类
public static string GetListControlValue(ListControl listcontrol, String sep)	得到 ListControl 控件的多个值，多个值之间使用 sep 分隔
public static string GetListControlValue(ListControl listcontrol)	同上，使用逗号分隔
public static string GetListControlText(ListControl listcontrol, String sep)	得到 ListControl 控件的多个文本，文本之间使用 sep 分隔
public static string GetListControlText(ListControl listcontrol)	同上，使用逗号分隔
public static void SetListControlByValue(ListControl listcontrol, object mulValue, String sep)	设置 ListControl 控件的多个值，多个值之间使用 sep 分隔
public static void SetListControlByValue(ListControl listcontrol, object mulValue)	同上，使用逗号分隔
public static void SetListControlByText(ListControl listcontrol, object mulValue, char sep)	
public static void SetListControlByText(ListControl listcontrol, object mulValue)	

public static object GetControlValue(Control con1)	得到控件的值
public static void SetControlValue(Control con1, object value1)	设置控件的值
public static void SetControlValues (Control[] cons,DataRow data)	使用 DataRow，设置多个控件的值
public static void SetControlValues (Control[] cons,Dictionary<String,object> data)	使用 Dictionary 设置多个控件的值
public static DataRow GetControlValues(Control[] cons)	得到多个控件的值

3、类 JCookie

类属性和方法	说明
✧ public sealed class JCookie	Cookie 实用类
public static void SetCookieValue(String name, String value)	设置一个 Cookie
public static void SetCookieValue(String name,String value, int hours)	同上，指定 Cookie 的时间，可指定为负值，表示 Cookie 仅在当前会话中起作用（关闭浏览器失效）
public static String GetCookieValue(String name)	得到 Cookie 的值
public static void SetCookieValues(String name,DataRow values, int hours)	使用 DataRow，设置多值 Cookie
public static void SetCookieValues(String name,DataRow values)	同上
public static void SetCookieValues (String name,Dictionary<String,object> values, int hours)	使用 Dictionary 设置一个多值 Cookie
public static void SetCookieValues (String name,Dictionary<String, object> values)	同上
public static DataRow GetCookieValues(String name)	得到一个多值 Cookie
public static void RemoveCookie(String name)	移去一个 Cookie
public static void ClearAllCookie()	清除所有的 cookie

4、类 JString

类属性和方法	说明
✧ public sealed class JString	String 实用类
public static string GetRandomString(byte Length,	得到一个随机字符串

int Up, int Down)	
public static string GetRandomString(byte Length)	同上 (0—Z)
public static string EnString(String strCryptThis,string strKey)	字符串的加密和解密
public static string EnString(String strCryptThis);	使用默认的密钥加密和解密
public static String MD5(String str1)	MD5 加密的字符串
public static string GetSimpleShowFormat(string str);	将\n 转换成 , 将空格转换成 nbsp;
public static string GetMaxLenth(string str1,int maxlength,String extends)	按最大长度截取字符串, 使用 extends 扩展
public static string GetMaxLenth(string str1,int maxlength)	同上, 使用...扩展
public static String GetDaXieMoney(decimal num)	得到数字的大写金额
public static string SqlFilter(string value)	Sql 语句非法字符串过滤
public static String GetStringToHex(String str1)	将字符串转换成 16 进制表示的字符串
public static String GetHexToString(String str1)	恢复 16 进制字符串为一个正常的字符串

5、类 JMyStringExpand

类属性和方法	说明
✧ public static class JMyStringExpand	字符串的扩展类
public static String JSimpleShowFormat(this String str1)	
public static String JMaxLenth (this string str1,int maxlength,String extends)	
public static String JMaxLenth(this string str1,int maxlengths)	
public static String JStringToHex(this String str1)	
public static String JHexToString(this String str1)	
public static String JDaXieMoney(this String num1)	

6、类 JValidator

类属性和方法	说明
✧ public sealed class JValidator	验证类
public static bool IsEmail(string source)	验证是否是正确的 Email
public static bool HasEmail(string source)	验证是否包含正确的 Email
public static bool IsUrl(string source)	
public static bool HasUrl(string source)	

public static bool IsDateTime(string source)	
public static bool IsMobile(string source)	
public static bool HasMobile(string source)	
public static bool IsIP(string source)	
public static bool HasIP(string source)	
public static bool IsIDCard(string Id)	
public static bool IsIDCard18(string Id)	
public static bool IsIDCard15(string Id)	
public static bool IsInt(string source)	
public static bool IsDouble(string source)	
public static bool IsLengthStr(string source, int begin, int end)	
public static bool IsPostCode(string source)	
public static bool IsChinese(string source)	
public static bool hasChinese(string source)	
public static bool IsNormalChar(string source)	

7、类 JDebug

类属性和方法	说明
✧ public sealed class JDebug	调试类
public static void Print(object obj1)	输出一个对象的值，目前对 Dictionary 和 Hashtable 进行了处理

三、WebFrame.Data 命名空间

数据库编程类。

1、类 SearchField

类属性和方法	说明
✧ [Serializable] public sealed class SearchField	查询条件类
public SearchField(string FieldName, string FieldValue, SearchOperator Operator, SearchFieldType Valuetype, bool SqlFilter)	说明：构造函数 字段名； 查询的值； 查询条件操作符（枚举）； 查询值类型（枚举）； 是否过滤非法字符
public SearchField(string FieldName, string FieldValue, SearchOperator Operator, SearchFieldType Valuetype)	同上，其中 SqlFilter 为 true
public SearchField(string FieldName, string FieldValue, SearchOperator Operator);	同上，其中值类型默认为 String
public SearchField(string FieldName, string FieldValue, SearchFieldType Valuetype)	同上，其中 SearchOperator 默认为等于
public SearchField(string FieldName, string FieldValue)	同上
public static SearchField operator +(SearchField su1, SearchField su2)	运算符重载，查询条件的 or
public static SearchField operator &(amp;SearchField su1, SearchField su2)	运算符重载，查询条件的 and
public static SearchField operator (SearchField su1, SearchField su2)	运算符重载，查询条件的 or
public static string GetSearchCondition(List<SearchField> SearchFieldList);	根据查询条件 List，得到查询条件的字符串表达式。
public static List<SearchField> GetSearchFieldList(Control[] control)	根据控件数值，得到查询条件表达式

2、类 **RelaTable**

类属性和方法	说明
✧ public sealed class RelaTable	相关表定义
public RelaTable(string tablename)	构造函数
public RelaTable(string tablename, string aliasname)	表名和别名
public string TableName;{get;}	表名称（包含关联表的名称）
public void Join(string jointablename, String JoinExpress ,JoinType jointype);	功能：连接 连接表名 连接表达式 连接类型
public void CommonJoin(String jointablename, String parentJoinField, String childJoinField)	常用连接 连接的表名； 父表字段名 子表字段名
public void InnerJoin(string jointablename, String JoinExpress)	内连接
public void LeftJoin(string jointablename, String JoinExpress)	左连接
public void RightJoin(string jointablename, String JoinExpress)	右连接
public void FullJoin(string jointablename, String JoinExpress)	完全连接

3、类 **JConnect**

类属性和方法	说明
✧ public sealed class JConnect : IDisposable	通用连接类
public WebDbType DbType {get;}	数据库连接类型
public void BeginTrans();	开始一个事务
public void CommitTrans();	提交一个事务
public void RollBackTrans();	回滚一个事务
public void Dispose();	释放连接资源
public int GetIdentity(String tableName);	得到表 tableName 的最近自增列值
public int GetIdentity();	得到 SQL SERVER 服务器的最近自增列值
public String GetSqlbyID(String sqlID);	根据 SQLID，得到 SQL 语句表达式
public List<DbParameter> GetDbParametersByDictionary	根据 Dictionary 得到操作参数列表

(Dictionary<string, object> htParameters)	
public static JConnect GetConnect(String ConnectName);	根据 WebConfig 中配置的连接字符串名，得到连接
public static JConnect GetConnect();	得到第一个连接字符串的连接
public static void CloseConnect();	关闭连接

4、类 JCommand

类属性和方法	说明
✧ public class JCommand : IDisposable	通用 Command 类，主要用于操作 SQL 语句和存储过程
public JCommand();	使用默认的连接构造
public JCommand(JConnect conn);	
public JConnect MyConnect; {get;}	得到 JCommand 中的 JConnect
public DbCommand MyDbCommand; {get;}	得到 JCommand 中的 DbCommand
public String CommandText; {get;set;}	Command 文本
public CommandType CommandType; {get;set;}	Command 类型
public List<DbParameter> Parameters; {get;set;}	Command 相关的参数
public int PageSize; {get;set;}	分页大小
public int ExecuteNonQuery()	执行非查询语句
public object ExecuteScalar()	得到第一行第一列的值
public virtual void Update(DataTable dt)	成批更新 DataTable，DataTable 必须是通过下面的 SearchData 得到的 DataTable
public virtual DataSet SearchData(int curpage)	返回 curpage 的 DataSet 值
public virtual DataRow GetFirstDataRow()	得到第一行数据
public int GetTotalRow()	得到总行数
public void Restore()	重置 CommandType 和 CommandText
public void Dispose()	释放资源
public void Close();	释放资源

5、类 JQuery

类属性和方法	说明
✧ public class JQuery:JCommand	多表查询类
public JQuery()	
public JQuery(JConnect conn)	
public JQuery(RelaTable tab)	使用关联表构造
public JQuery(JConnect conn, RelaTable tab)	
public RelaTable RelaTable; {get;set;}	设置多表查询中的关联表

public List<SearchField> Condition; {get;}	设置查询条件
public string OrderBy; {get;set;}	设置排序表达式
public string GroupBy; {get;set;}	设置分组表达式
public string Having; {get;set;}	设置分组后过滤表达式
public void SetCommandText(String commandtext);	直接设置 CommandText
public DataSet SearchData(string fields,int curpage);	返回查询结果

6、类 JTable

类属性和方法	说明
✧ public class JTable:JCommand	一般用于单表业务操作
public int Identity; {get;}	最近的自增列
public String TableName; {get;set;}	设置和得到表名
public String OrderBy; {get;set;}	设置和得到排序条件
public JTable(JConnect conn,string tableName);	
public JTable(string tableName);	
public JTable();	
public virtual void InsertData(Dictionary<String, object> data);	插入一条数据
public virtual void DeleteData(List<SearchField> condition);	根据条件，删除数据
public virtual void EditData (Dictionary<String, object> NewData, List<SearchField> condition, bool Append);	根据条件，编辑数据，如果没有满足条件的数据， 当 Append 为 true 时，则表示新增数据
public object SearchScalar(List<SearchField> condition, String FieldName);	查询满足条件的数据
public DataSet SearchData(List<SearchField> condition, int curpage,params String[] FieldList)	
public int GetDataTotalRow(List<SearchField> condition);	得到满足条件的总数据行
public DataRow GetFirstDataRow(List<SearchField> condition, params String[] FieldList)	得到满足条件的第一条数据
public bool HasData(List<SearchField> condition)	判断是否有数据

四、WebFrame.UI 命名空间

1、参数类

类定义	说明
✧ public class ParameterItem	数据源和按钮等参数类
public String ParaName { get; set; }	参数名称
public JParaType ParaType { get; set; }	参数类型
public String DataName { get; set; }	相关的字段名称，不填为 ParaName
public SearchFieldType DataType { get; set; }	参数类型
public bool IsMulValue { get; set; }	是否为多值
public bool IsNullValue { get; set; }	当没值时，是否转成 Null
✧ public class AssemblePara	程序集参数类
public String AssembleFile { get; set; }	程序集文件名
public String ClassLibName { get; set; }	类的全名
public String MethodName { get; set; }	方法名
✧ public class ButtonExecutePara	按钮执行结果参数
public bool Success { get; set; }	执行成功标志位
public object ExecuteResult { get; set; }	执行结果
public String SuccInfo { get; set; }	成功提示信息
public String SuccUrl { get; set; }	成功后的转向，支持#、0、和-1 等特殊地址
public String FailUrl { get; set; }	失败后的专向
public String FailInfo { get; set; }	失败后的提示信息
✧ public class ButtonSearchPara	查询按钮参数
public String SearchControlList { get; set; }	查询空间的 ID 列表
public bool IsCancelSearch { get; set; }	是否为取消查询
✧ public class ButtonSortPara	排序按钮参数
public String SortExpress { get; set; }	排序表达式
public bool FixSort { get; set; }	是否为固定排序
✧ public class ButtonSimpleActionPara	简单按钮操作参数
public SimpleButtonType ButtonType { get; set; }	简单按钮操作类型，例如为删除、新增、更新等
public String ConnectName { get; set; }	连接名称，不填为默认连接
public String TableName { get; set; }	表名称
public bool AppendFlag { get; set; }	是否追加数据，用于更新操作

2、UI 扩展类

类定义	说明
public class JCheckBox:CheckBox,ISearchControl,IJControl;	
public class JCheckBoxList:CheckBoxList,ISearchControl,IJControl;	
public class JDropDownList:DropDownList,ISearchControl,IJControl;	
public class JFileUpload:FileUpload,IJControl,ISearchControl;	
public class JLabel:Label,IJControl;	
public class JListBox:ListBox,ISearchControl,IJControl;	
public class JRadioButton:RadioButton,ISearchControl,IJControl;	
public class JTextBox : TextBox, ISearchControl, IJTextControl;	
public class AppCheckBoxList : JCheckBoxList, IAppListControl;	
public class AppDropDownList : JDropDownList, IAppListControl;	
public class AppRadioButtonList : JRadioButtonList, IAppListControl;	
public class AppListBox : JListBox, IAppListControl;	

3、Button 扩展类

类定义	说明
public class JButton : Button, IJButton	
public class JImageButton : ImageButton, IJButton	
public class JLinkButton : LinkButton, IJButton	

4、数据类

类定义	说明
◇ public class JDataSource	数据源类
public String ConnectName {get;set;}	连接名称
public bool IsAddWhere {get;set;}	是否追究 where，一般不用
public String OrderBy {get;set;}	排序表达式
public int CurPage {get;set;}	显示的数据页
public int RandomPage {get;set;}	随机显示的页
public JCommandType JType {get;set;}	数据源类型，不指定，则为表
public int PageSize {get;set;}	页大小
public String SqlID {get;set;}	数据源 ID，可为表名称、SQL 语句和 SQLID
public int TotalRow {get;}	结果总行数

public int TotalPage {get;}	结果总页数
public List<ParameterItem> ParaItems {get;}	数据源中的参数列表
public DataTable GetListData();	得到数据源中的列表数据
✧ public class JDetailView	数据明细类
public String DataSourceID {get;set;}	数据源控件 ID
public JDataSource DataSource {get;}	数据源对象
public String ControlList {get;set;}	相关控件 ID 列表
public String NoControlList {get;set;}	不相关控件 ID 列表
✧ public class JUniqueValidator	唯一验证控件
public String ConnectName {get;set;}	连接名称
public String TableName {get;set;}	表名称
public String UniqueControlList {get;set;}	相关控件的 ID 列表