System.Collections.ArrayList类是一个特殊的数组。通过添加和删除元素，就可以动态改变数组的长度。

一、优点

1. 支持自动改变大小的功能

2. 可以灵活的插入元素

3. 可以灵活的删除元素

4. 可以灵活访问元素

二、局限性

跟一般的数组比起来，速度上差些

三、添加元素

1．public virtual int Add(object value);

将对象添加到ArrayList的结尾处

ArrayList aList=new ArrayList();

aList.Add("a");

aList.Add("b");

aList.Add("c");

aList.Add("d");

aList.Add("e");

内容为abcde

2．public virtual void Insert(int index,object value);

将元素插入ArrayList的指定索引处

ArrayList aList=new ArrayList();

aList.Add("a");

aList.Add("b");

aList.Add("c");

aList.Add("d");

aList.Add("e");

aList.Insert(0,"aa");

结果为aaabcde

3．public virtual void InsertRange(int index,ICollectionc);

将集合中的某个元素插入ArrayList的指定索引处

ArrayList aList=new ArrayList();

aList.Add("a");

aList.Add("b");

aList.Add("c");

aList.Add("d");

aList.Add("e");

ArrayList list2=new ArrayList();

list2.Add("tt");

list2.Add("ttt");

aList.InsertRange(2,list2);

结果为abtttttcde

四、删除

a)public virtual void Remove(object obj);

从ArrayList中移除特定对象的第一个匹配项,注意是第一个

ArrayList aList=new ArrayList();

aList.Add("a");

aList.Add("b");

aList.Add("c");

aList.Add("d");

aList.Add("e");

aList.Remove("a");

结果为bcde

2.public virtual void RemoveAt(intindex);

移除ArrayList的指定索引处的元素

aList.Add("a");

aList.Add("b");

aList.Add("c");

aList.Add("d");

aList.Add("e");

aList.RemoveAt(0);

结果为bcde

3．public virtual void RemoveRange(int index,int count);

从ArrayList中移除一定范围的元素。Index表示索引，count表示从索引处开始的数目

aList.Add("a");

aList.Add("b");

aList.Add("c");

aList.Add("d");

aList.Add("e");

aList.RemoveRange(1,3);

结果为ae

4．public virtual void Clear();

从ArrayList中移除所有元素。

五、排序

a)public virtual void Sort();

对ArrayList或它的一部分中的元素进行排序。

ArrayListaList=newArrayList();

aList.Add("e");

aList.Add("a");

aList.Add("b");

aList.Add("c");

aList.Add("d");

DropDownList1.DataSource=aList;//DropDown ListDropDownList1;

DropDownList1.DataBind();

结果为eabcd

ArrayList aList=new ArrayList();

aList.Add("a");

aList.Add("b");

aList.Add("c");

aList.Add("d");

aList.Add("e");

aList.Sort();//排序

DropDownList1.DataSource=aList;//DropDownListDropDownList1;

DropDownList1.DataBind();

结果为abcde

b)public virtual void Reverse();

将ArrayList或它的一部分中元素的顺序反转。

ArrayList aList=new ArrayList();

aList.Add("a");

aList.Add("b");

aList.Add("c");

aList.Add("d");

aList.Add("e");

aList.Reverse();//反转

DropDownList1.DataSource=aList;//DropDownListDropDownList1;

DropDownList1.DataBind();

结果为edcba

六、查找

a)public virtual int IndexOf(object);

b)public virtual int IndexOf(object,int);

c)public virtual int IndexOf(object,int,int);

返回ArrayList或它的一部分中某个值的第一个匹配项的从零开始的索引。没找到返回-1。

ArrayList aList=new ArrayList();

aList.Add("a");

aList.Add("b");

aList.Add("c");

aList.Add("d");

aList.Add("e");

intnIndex=aList.IndexOf(“a”);//1

nIndex=aList.IndexOf(“p”);//没找到，-1

d)public virtual int LastIndexOf(object);

e)public virtual int LastIndexOf(object,int);

f)public virtual int LastIndexOf(object,int,int);

返回ArrayList或它的一部分中某个值的最后一个匹配项的从零开始的索引。

ArrayList aList=new ArrayList();

aList.Add("a");

aList.Add("b");

aList.Add("a");//同0

aList.Add("d");

aList.Add("e");

intnIndex=aList.LastIndexOf("a");//值为2而不是0

g)public virtual bool Contains(objectitem);

确定某个元素是否在ArrayList中。包含返回true,否则返回false

七、获取数组中的元素

下面以整数为例，给出获取某个元素的值的方法

ArrayList aList=new ArrayList();

for(int i=0;i<10;i++)

     aList.Add(i);

for(i=0;i<10;i++)

    Textbox1.text+=(int)aList[i]+" ";//获取的方式基本与一般的数组相同，使用下标的方式进行

结果为：0 1 2 3 4 5 6 7 8 9

八、其他

1．public virtual intCapacity{get;set;}

获取或设置ArrayList可包含的元素数。

2．public virtual intCount{get;}

获取ArrayList中实际包含的元素数。

Capacity是ArrayList可以存储的元素数。Count是ArrayList中实际包含的元素数。Capacity总是大于或等于Count。如果在添加元素时，Count超过Capacity，则该列表的容量会通过自动重新分配内部数组加倍。

如果Capacity的值显式设置，则内部数组也需要重新分配以容纳指定的容量。如果Capacity被显式设置为0，则公共语言运行库将其设置为默认容量。默认容量为16。

在调用Clear后，Count为0，而此时Capacity切是默认容量16，而不是0

3．public virtual void TrimToSize();

将容量设置为ArrayList中元素的实际数量。

如果不向列表中添加新元素，则此方法可用于最小化列表的内存系统开销。

若要完全清除列表中的所有元素，请在调用TrimToSize之前调用Clear方法。截去空ArrayList会将ArrayList的容量设置为默认容量，而不是零。

ArrayList aList=new ArrayList();

aList.Add("a");

aList.Add("b");

aList.Add("c");

aList.Add("d");

aList.Add("e");//Count=5,Capacity=16,

aList.TrimToSize();//Count=Capacity=5;

Aaaa