

ArrayList představuje třídu .NET Frameworku ([System.Collections.ArrayList](#), knihovna mscorlib). Dokáže to, co musíme jinak horko těžko při práci s prostým polem ve VBA programovat – třídění, přidávání a odebrání položek, test existence položky, spojování polí aj. Vtip je v tom, že s jistými omezeními je třída ArrayList dostupná i pod VBA. Musím přiznat, že na tuto skutečnost jsem narazil poprvé cca před rokem a půl. Žádná další teorie nás nečeká, pojďme si zkusit pár příkladů. Pozn. Doporučuji kód nekrokovat, ale spouštět až do místa vložené záložky. I tak budete muset leckdy pro zobrazení skutečného obsahu proměnné v okně Locals ji sbalit a znovu rozbalit.

## ArrayList v příkladech

```

1 Sub ProceduraArrayList()
2
3 Dim objArray
4 Dim objArrayList1
5 Dim objArrayList2
6 Dim objArrayList3
7 Dim objArrayList4
8 Dim objArrayList5
9
10 Set objArrayList1 = CreateObject("System.Collections.ArrayList")
11 Set objArrayList2 = CreateObject("System.Collections.ArrayList")
12 Set objArrayList3 = CreateObject("System.Collections.ArrayList")
13 Set objArrayList4 = CreateObject("System.Collections.ArrayList")
14 Set objArrayList5 = CreateObject("System.Collections.ArrayList")
15
16 Dim intIndex As Integer
17 Dim intPocetPolozek As Integer
18
19 Dim binPolozkaExistuje As Boolean
20
21 'plnění pole 1
22 With objArrayList1
23 .Add ("žirafa")
24 .Add ("čížek")
25 .Add ("motýl")
26 End With
27
28 'plnění pole 2
29 With objArrayList2
30 .Add ("pěnkava")
31 .Add ("chobotnice")
32 End With
33
34 'plnění pole 3
35 With objArrayList3
36 .Add ("výr")
37 .Add ("mravenec")
38 End With
39
40 'přidání jednoho pole na konec druhého
41 'závorky nutné
42 objArrayList1.AddRange (objArrayList2)
43
44 'vlození jednoho pole do druhého na pozici
45 'indexy 0, 1, ...
46 'závorky nutné
47 objArrayList1.InsertRange 1, (objArrayList3)
48
49 'přepsání položek jednoho pole druhým
50 'indexy 0, 1, ...
51 'závorky nutné
52 objArrayList1.SetRange 1, (objArrayList2)
53
54 'odstranění části pole (2 položky od indexu 1)
55 objArrayList1.RemoveRange 1, 2
56
57 'přidání položky za n-tou položku
58 '0 ... počátek
59 objArrayList1.Insert 1, "tučňák"
60
61 'test existence položky
62 binPolozkaExistuje = objArrayList1.Contains("tučňák")
63
64 'index hledané položky s definováním
65 'počátku vyhledávání
66 'indexy 0, 1, ...
67 intIndex = objArrayList1.IndexOf("tučňák", 0)
68
69 'odstranění položky dle jména
70 objArrayList1.Remove ("tučňák")
71
72 'odstranění položky dle indexu
73 'indexy položek 0, 1, ...
74 'objArrayList1.RemoveAt 1
75
76 'vzestupné seřazení položek
77 objArrayList1.Sort
78
79 'výpis položek do okna Immediate
80 Debug.Print Join(objArrayList1.ToArray(), vbCrLf)
81
82 'počet položek
83 intPocetPolozek = objArrayList1.Count
84
85 'přizpůsobení velikosti naplněným položkám
86 'objArrayList1.TrimToSize
87
88 'revers pole
89 objArrayList3.Reverse
90
91 'kopie ArrayList do obyčejného pole
92 objArray = objArrayList3.ToArray()
93
94 'převzetí části pole
95 'indexy 0, 1, ...
96 Set objArrayList4 = objArrayList1.GetRange(1, 2)
97
98 'klonování pole
99 Set objArrayList5 = objArrayList3.Clone
100
101 'vyčištění pole
102 objArrayList1.Clear
103
104 'odstranění proměnných z paměti
105 Set objArrayList1 = Nothing
106 Set objArrayList2 = Nothing
107 Set objArrayList3 = Nothing
108 Set objArrayList4 = Nothing
109 Set objArrayList5 = Nothing
110
111 End Sub

```

Největší problémy mi dělaly metody SetRange a InsertRange. Podotýkám, že práce s objektovými proměnnými pod VBA vyžaduje příkaz Set, navíc VBA prapodivně pracuje se závorkami u volání.

Pokud se pamatuji, prvně jsem ArrayList využil při třídění položek. Metoda Sort vrací korektnější výsledky, než třeba vlastní procedura (technika řazení) QuickSort. Na ukázkou:

čížek  
chobotnice  
motýl  
pěnkava  
žirafa

## Reference

Na knihovnu mscorlib.dll je možné se napojit i přes reference a pak není potřeba metody CreateObject. Doporučuji do knihovny nahlédnout přes Object Browser (F2).

1	<b>Sub</b> KnihovnaMSCORLIB()
2	
3	'Tools / References / mscorlib.dll
4	
5	<b>Dim</b> objArrayList <b>As New</b> ArrayList
6	
7	'plnění pole 1
8	<b>With</b> objArrayList
9	.Add ("žirafa")
10	.Add ("čížek")
11	.Add ("motýl")
12	<b>End With</b>
13	
14	<b>End Sub</b>

Co se mi bohužel nepodařilo zrealizovat, je vytvoření pole s opakováním položek, např.

```
Set objArrayList = ArrayList.Repeat(„abc“, 7)
```

Je velká škoda, že ArrayList se neumí přímo zbavit duplicit. Museli bychom si zkombinovat metodu Contains třeba s GetRange nebo použít jiné objekty (HashSet). Ve VBA už kdysi pro tyto účely navrhl John Walkenbach datový typ vypůjčený z VBScriptu, a to Dictionary (neumožňuje přidání již existující

položky v metodě Add). Je-li potřeba v polích pracovat s klíčem a hodnotou, nezapomínejte ani na přímo ve VBA dostupný objekt Collection.

## Zásobník a fronta

V knihovně se lze odvolávat na některé další třídy (System.Collections.Hashtable, System.Collections.SortedList). My se podíváme na zásobník (zbraně), neboli [stack](#) a frontu (na poště), čili [queue](#), známé to typy z algoritmů.

```
1      Sub ProceduraStack()  
2  
3      Dim objStack  
4  
5      'zásobník, také jinak LIFO  
6      'tj. Last In First Out (poslední dovnitř, první ven)  
7  
8      Set objStack = CreateObject("System.Collections.Stack")  
9  
10     'plnění zásobníku  
11     With objStack  
12     .Push ("1. náboj")  
13     .Push ("2. náboj")  
14     .Push ("3. náboj")  
15     End With  
16  
17     'vystřelení 3. náboje  
18     'tj. naposled přidané položky  
19     MsgBox objStack.Pop  
20  
21     'vystřelení 2. náboje  
22     MsgBox objStack.Pop  
23  
24     'je v zásobníku 1. náboj?  
25     MsgBox objStack.Contains("1. náboj")  
26  
27     'odstranění proměnné z paměti  
28     Set objStack = Nothing  
29  
30     End Sub
```

```

1  Sub ProceduraQueue()
2
3  Dim objQueue
4
5  'fronta, také jinak FIFO
6  'tj. First In First Out (první dovnitř, první ven)
7
8  Set objQueue = CreateObject("System.Collections.Queue")
9
10 'plnění fronty
11 With objQueue
12 .enqueue ("1. člověk")
13 .enqueue ("2. člověk")
14 .enqueue ("3. člověk")
15 End With
16
17 'odchod 1. člověka
18 'tj. odebrání první (nejstarší) přidané položky
19 MsgBox objQueue.Dequeue
20
21 'odchod 2. člověka
22 MsgBox objQueue.Dequeue
23
24 'kdo je na řadě?
25 MsgBox objQueue.Peek
26
27 'odstranění proměnné z paměti
28 Set objQueue = Nothing
29
30 End Sub

```

Příloha

[pole\\_arraylist.zip](#)