

5.2 Arrayens initieringslista

Precis som det finns skillnader i definitionen av arrayvariabler jämfört med vanliga variabler, finns även skillnader vid initieringen dvs första tilldelningen. T.ex. är initieringen av arrayen **no** i programexemplet **ArrayDef** – en sats för varje element – inte särskilt lämplig för arrays, speciellt om man skulle tillämpa samma teknik på större arrays. Men just hanteringen av stora datamängder var ju motiveringen för att syssla med array. Kan man inte effektivisera initieringen? Jo, till en viss gräns. Det finns i huvudsak två möjligheter: antingen att använda **for**-satsar eller att slå ihop definitionen med tilldelningen till en kortform som använder sig av en s.k. *initieringslista*. Båda har vi använt i följande program:

```
1 <!-- InitLista.html
2     Kortform för definition och initiering av en array med
3     en initieringslista. Direkt tilldelning till en kopia -->
4 <title>Initieringslista</title>
5 <script>
6     no = [64, 86, 34]           // Kortform på definition och
7                               // initiering med initieringslistan
8     copy = no                 // Tilldelning med arraynamnet
9                               // Elementvis utskrift av kopian:
10    document.writeln('<h3>Kopians första element copy[0]' +
11                    ' har värdet ' + copy[0] + '<br><br>' +
12                    'Kopians andra element no[1]' +
13                    ' har värdet ' + copy[1] + '<br><br>' +
14                    'Kopians tredje element copy[2]' +
15                    ' har värdet ' + copy[2] + '<br><br>' +
16                    'Kopians fjärde element copy[3]' +
17                    ' har värdet ' + copy[3] + '</h3>' )
18                               // Utskrift av kopian med arraynamnet:
19    document.writeln('<h3>Utskrift med arraynamnet: ' +
20                    'copy = ' + copy + '</h3>' )
21 </script>
```

En körning av program-exemplet **InitLista** visar att värdena från arrayen **no** verkligen kopierats över till arrayen **copy**:

Både definitionssatsen och initieringssatserna i **ArrayDef** – det är de 4 första satserna – kan slås ihop till en enda sats.

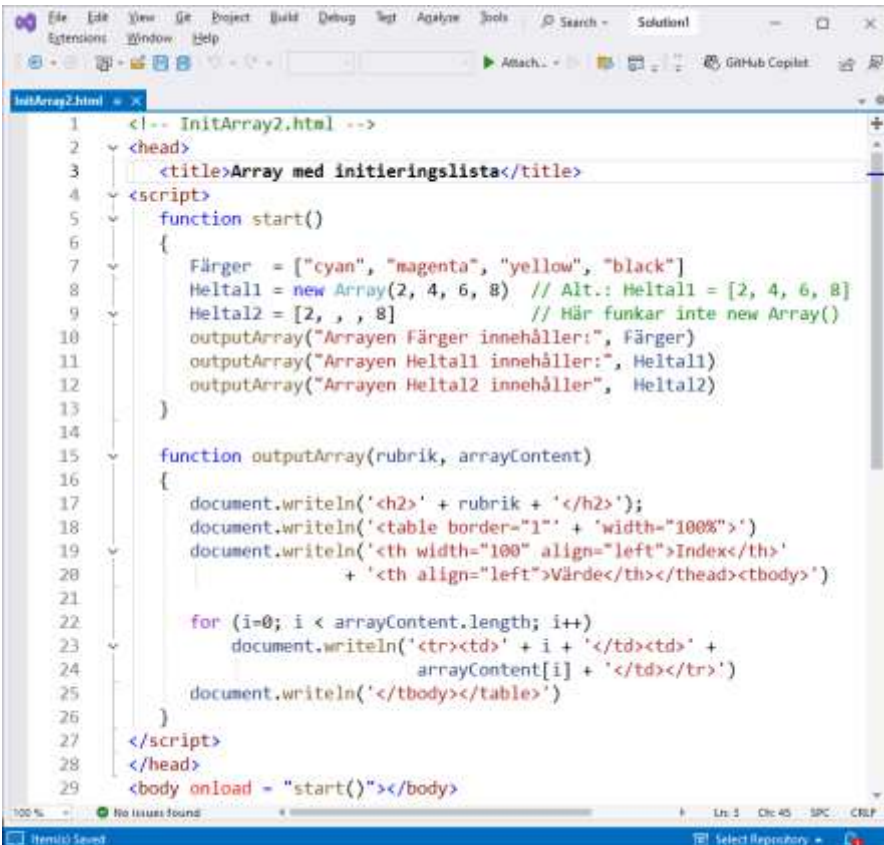
```
Kopians första element copy[0] har värdet 64.
Kopians andra element copy[1] har värdet 86.
Kopians tredje element copy[2] har värdet 34.
Kopians fjärde element copy[3] har värdet undefined.
Utskrift med arraynamnet: copy = 64,86,34
```

Initieringslista

```
no = [64, 86, 34] // Kortform för definition och
                 // initiering med initieringslistan
```

Satsen gör två saker: Först, fram till tilldelningstecknet definieras arrayen `no` utan någon uppgift om arrayens storlek. Sedan, från och med tilldelningstecknet tilldelas arrayen `no`:s element fyra värden som står i en kommaseparerad lista grupperad inom hakparenteserna `[]` som kallas arrayens *initieringslista*. Satsen ovan är endast en kortform för de fyra första satserna i `<script>`-taggen till `ArrayDef` och gör precis samma sak som de. JavaScript-interpretatorn får informationen om arrayens storlek i initieringslistan, dvs räknar antalet kommaseparerade element inom hakparenteserna `[]`. Observera att man får använda kortformerna ovan endast i samma sats som definitionen.

Exempel med funktion



```
1 <!-- InitArray2.html -->
2 <head>
3   <title>Array med initieringslista</title>
4 <script>
5   function start()
6   {
7     Färger = ["cyan", "magenta", "yellow", "black"]
8     Heltal1 = new Array(2, 4, 6, 8) // Alt.: Heltal1 = [2, 4, 6, 8]
9     Heltal2 = [2, , , 8] // Här funkar inte new Array()
10    outputArray("Arrayen Färger innehåller:", Färger)
11    outputArray("Arrayen Heltal1 innehåller:", Heltal1)
12    outputArray("Arrayen Heltal2 innehåller:", Heltal2)
13  }
14
15  function outputArray(rubrik, arrayContent)
16  {
17    document.writeln('<h2>' + rubrik + '</h2>');
18    document.writeln('<table border="1" + 'width="100%">')
19    document.writeln('<th width="100" align="left">Index</th>'
20                      + '<th align="left">Värde</th></thead><tbody>')
21
22    for (i=0; i < arrayContent.length; i++)
23      document.writeln('<tr><td>' + i + '</td><td>' +
24                        arrayContent[i] + '</td></tr>')
25    document.writeln('</tbody></table>')
26  }
27 </script>
28 </head>
29 <body onload = "start()"></body>
```