

5.6 Sortering av arrays

```
Sort.html
1  <!-- Sort.html -->
2  <head>
3  <script>
4    function start()
5    {
6      NoSort = [10, 1, 9, 2, 8, 3, 7, 4, 6, 5, 100]
7      document.writeln("<h1>Sortering med inbyggd metod sort()</h1>")
8      outputArray("Array osorterad: ", NoSort)
9      NoSort.sort(asc)           // Sortering i stigande ordning
10     outputArray("Array sorterad i stigande ordning: ", NoSort)
11     NoSort.sort(desc)         // Sortering i fallande ordning
12     outputArray("Array sorterad i fallande ordning: ", NoSort)
13   }
14
15   function outputArray(label, a)
16   {
17     document.writeln("<p><h3>" + label + a.join(" ") + "</h3></p>")
18   }
19
20   function asc(value1, value2) // Comparator function för
21   {                             // stigande ordning
22     return value1 - value2     // Returnerar + om value1 > value2
23   }
24
25   function desc(value1, value2) // Comparator function för
26   {                             // fallande ordning
27     return value2 - value1     // Returnerar - om value1 > value2
28   }
29 </script>
30 </head><body onload = "start()"></body>
```

Scriptet **Sort** skapar och initierar en array på rad 6 samt skriver ut den på rad 8 i samma ordning som den initierades. På rad 9 skickas arrayen till funktionen **sort()** för sortering. Syntaxen för detta anrop visar att **sort()** inte är en vanlig funktion:

NoSort.sort(asc)

För det första har vi ingenstans i scriptet definierat en funktion vid namnet **sort()**. För det andra anropas den med punkt efter arrayen **NoSort** och för det tredje anropas



den med en parameter (**asc**) som vi inte har definierat som en variabel. Faktiskt är **sort()** en s.k. *metod* som är fördefinierad i JavaScript, närmare bestämt i *klassen Array*. Därför kallar vi den för en *Array-metod*. Den anropas med punktnotation, vilket i sin tur innebär att **NoSort** är ett *objekt* av JavaScript-klassen **Array**. Vi kommer att förklara dessa begrepp senare och nöjer oss med att närmare gå in på *metoder*.

Vad är en metod?

En *metod* är en funktion som är definierad i en *klass*. Dvs den enda skillnaden mellan metod och funktion är *placeringen* av definitionen. Så man skiljer mellan *funktioner* som är *fristående* och *metoder* som är placerade i klasser. Därför kan metoder endast anropas i ett *objekt* (ett exemplar) av klassen. Närmare bestämt kan metoder anropas endast genom att först kalla på objektet och sedan med hjälp av punktnotation anropa metoden, t.ex. **NoSort.sort()**. För att anropa **sort()** måste vi skriva så, därför att metoden **sort()** är definierad i klassen **Array** och **NoSort** är ett objekt av denna klass. Det går inte att anropa **sort()** utan objekt, som en fristående funktion.

Array-metoden sort()

En annan egenskap av den inbyggda metoden (*built-in method*) **sort()** är att den kan ta som parameter namnet på en s.k. *comparator function* som jämför sina två parametrar och returnerar:

- ett positivt värde om den första parametern är större än den andra
- noll om båda parametrarna är lika stora eller
- ett negativt värde om den första parametern är mindre än den andra.

I scriptet **Sort** har vi definierat två comparator functions: **asc()** på raderna **20-23** och **desc()** på raderna **25-28**. Båda anropas i parameterlistan av metoden **sort()** för att bestämma om sorteringen ska göras i stigande eller i fallande ordning.

Det är anmärkningsvärt att anropen av comparator functions görs utan de sedvanliga parenteserna som är obligatoriska vid funktionsanrop. Anledningen är att funktioner anses i JavaScript som *data* som kan tilldelas variabler och skickas som parametrar till andra funktioner, precis som vilken annan data som helst. Eftersom anropet sker i parameterlistan av en *annan* funktion vore det fel att skriva parenteserna. Därför skriver vi i scriptet **Sort** som anrop av metoden **sort()**

på rad **9**: **NoSort.sort(asc)**

och på rad **11**: **NoSort.sort(desc)**

Dvs våra comparator functions **asc()** och **desc()** anropas i parameterlistan av metoden **sort()** utan parenteser och överför informationen om sorteringsordningen (stigande eller fallande) som en flagga till metoden. **asc** och **desc** anses här vara *variabler*. Skriver man parenteserna blir det fel syntax.