

a. Taulukot (Arrays)

Aikaisemmin olemme tutustuneet muuttujiin ja havainneet ne hyväksi tiedon väliaikaiseen tallentamiseen ohjelman ajon aikana. Kuhunkin muuttujaan voi kuitenkin tallentaa vain yhden arvon. On kuitenkin usein tilanteita, joissa tarvitaan tallentaa samalla kertaa useita tietoja. Tällaisia voisivat olla esimerkiksi osallistujien tietojen tallentaminen tai tuotteiden hinta tiedot tms. Käyttökohteita olisi paljon. Tähän muuttuja ei sovellu, vaan tarvitaan oma tietorakenne – taulukko. Taulukkoon voi tallentaa (sijoittaa, assign) esimerkiksi vaikkapa osallistujien tiedot. Myös taulukko luodaan (määritellään, define) var tai uudemmalla let avainsanalla JavaScriptissä:

```
var osallistujat = [];
```

Saman voisi tehdä myös:

```
let ehdokkaat = [];
```

Loimme tässä kaksi tyhjää taulukkoa, osallistuja taulukon ja ehdokkaat taulukon. Voisimme lisätä osallistujat taulukkoon ensimmäiset osallistujat, lauseilla:

```
osallistujat[0] = "Virta";  
osallistujat[1] = "Niemi";  
osallistujat[2] = "Ranta";
```

Taulukon sisältö näyttäisi tässä vaiheessa seuraavanlaiselta:

osallistujat	
0	Virta
1	Niemi
2	Ranta

Huomaa miten taulukon ensimmäinen tieto on indeksi arvossa 0, eikä 1:ssä. Taulukossa on nyt kolme arvoa, jonka saamme selville taulukon length arvon avulla:

```
var taulukonPituus = osallistujat.length;
```

Taulukossa olevia arvoja saamme selville lauseella:

```
document.write(osallistujat[0]);
```

joka tulostaa web –sivulle osallistujat taulukossa ensimmäisenä olevan tiedon, eli tässä tapauksessa "Virta".

Taulukkoon voi myös heti sijoittaa sen määrittelyn yhteydessä tarvittavat arvot:

```
var hinnat = [12, 8.5, 5, 7.25];
```

Hinnat taulukko näyttää heti sen luonnin jälkeen seuraavalta:

hinnat	
0	12
1	8.5
2	5
3	7.25

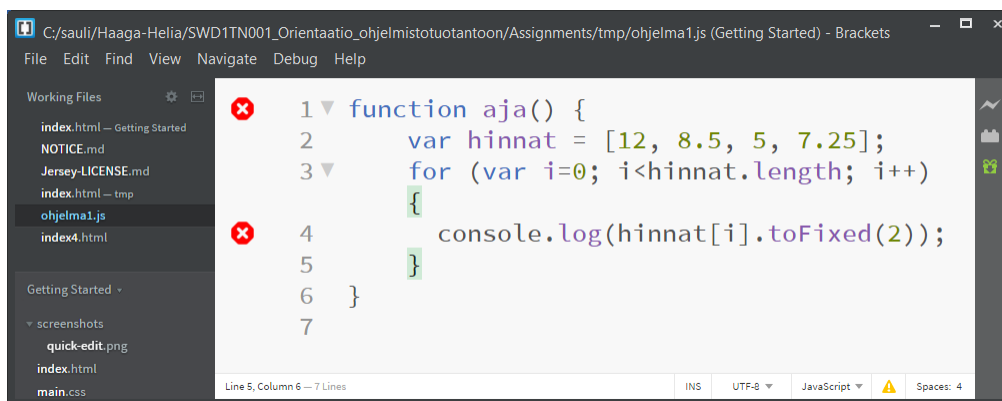
Taulukossa olevat tiedot saa kätevästi tulostettua toistorakenteen ja taulukon .length ominaisuuden avulla:

```
for (var i=0; i<hinnat.length; i++) {  
    console.log(hinnat[i]);  
}
```

joka tulostaa hinnat taulukossa olevat arvot web –selaimen konsoli-ikkunaan. Myös arvojen pyöristäminen kahden desimaalin tarkkuuteen onnistuu:

```
for (var i=0; i<hinnat.length; i++) {  
    console.log(hinnat[i].toFixed(2));  
}
```

JavaScript ohjelma näyttäisi Adobe Brackets:issa:



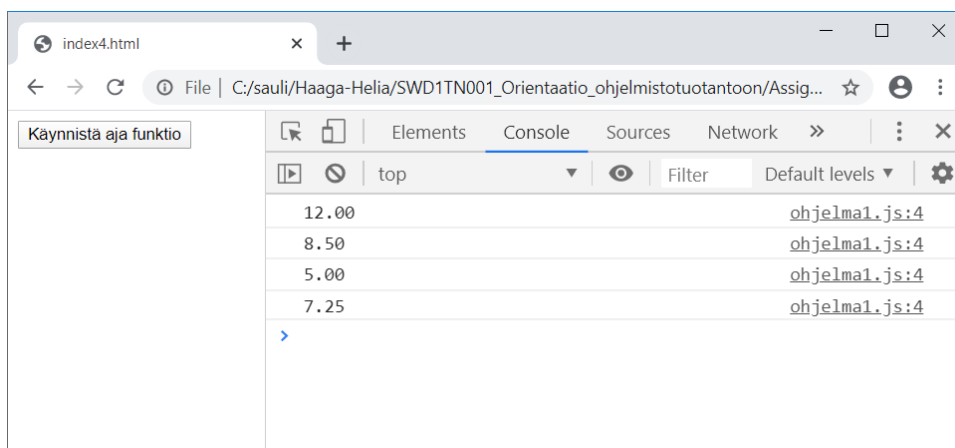
Lisäksi tarvitaan html tiedosto, joka on samassa kansiossa yllä olevan ohjelma1.js tiedoston kanssa:

A screenshot of the Brackets code editor. The main window displays the file 'index4.html' with the following HTML code:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title></title>
5   <meta charset="utf-8">
6 </head>
7 <body>
8   <button onclick="aja()">Käynnistä aja
   funktio</button>
9   <script src="ohjelma1.js"></script>
10 </body>
11 </html>
```

The left sidebar shows the 'Working Files' list with 'index4.html' selected. The bottom status bar indicates 'Line 11, Column 8 - 11 Lines', 'INS', 'UTF-8', 'HTML', and 'Spaces: 4'.

Tulos näyttää Chromessa seuraavalta kun Käynnistä aja funktio nappulaa on klikattu:



Taulukko on JavaScriptissä täysin dynaaminen, joten sen kokoa ja sinne tallennettavaa tietoa voi vaihtaa tietotyyppistä toiseen. Taulukkoon voi myös tallentaa eri tietotyyppisiä tietoja, kuten tekstiä, numeroita jne.

Seuraava on siis mahdollista:

```
var JavaScriptKussinTiedot = ["JavaScript", 5];
```

Tosin tietorakenteesta saisi helpommin ymmärrettävän olioiden avulla. Nythän ei ole varmaa, mitä numero 5 tarkoittaa yllä olevassa taulukossa.

Taulukon kokoa voi myös dynaamisesti kasvattaa halutessaan. Esimerkiksi hinnat taulukkoon voi lisätä hinta tietoja seuraavalla tavalla:

```
hinnat[4] = 10;
```

Nyt taulukko näyttää seuraavalta:

hinnat	
0	12
1	8.5
2	5
3	7.25
4	10

b. Taulukon data numerojärjestykseen tai aakkosjärjestykseen (.sort())

Taulukon sisältämän datan voi järjestää numerojärjestykseen jos ne ovat numeroita .sort() metodilla. Vastaavasti jos datat ovat taulukossa tekstiä, saa ne järjestettyä aakkosjärjestykseen. Esimerkiksi jos on taulukko:

```
var painot = [60.5, 70.2, 59.3, 87];
```

Taulukon saa järjestettyä numerojärjestykseen lauseella:

```
painot.sort();
```

Numerot ovat nyt seuraavassa järjestyksessä taulukossa keskusmuistissa:

```
[59.3, 60.5, 70.2, 87]
```

Kyseisestä JavaScript metodista on lisää tietoa kiinnostuneille osoitteessa:

https://www.w3schools.com/jsref/jsref_sort.asp

.reverse() metodilla puolestaan pystyy kääntämään aakkosjärjestyksen käänteiseen eli ”ööökkösjärjestykseen”. Numeroiden kyseessä ollessa, kääntää .reverse() numerot suuremmasta pienempään:

```
painot.reverse();
```

Numerot ovat nyt seuraavassa järjestyksessä taulukossa:

```
[87, 70.2, 60.5, 59.3]
```

c. Lisä materiaalia kiinnostuneille

- Edellä käytimme for toistorakennetta taulukon läpikäyntiin. Kuten monissa muissakin ohjelmointikielissä, on myös JavaScriptissä nykyään tähän nimenomaan tarkoitettu `for in` -toistorakenne:

```
for (i in hinnat)
{
    document.write(i + "<br>");
}
```

- Myös päivämääriä voi tallentaa JavaScriptissä taulukoihin:

```
var flightDate = new Date("September 22, 2014");
var myValues = ["Finnair", 221, flightDate];
```

- JavaScriptistä löytyy myös niin sanotut assosiatiiviset taulukot (Assosiative Arrays)

Esimerkiksi:

```
var scoreList = [];
scoreList["Mattila"] = 9;
scoreList["Mikkonen"] = 5;
scoreList["Aalto"] = 7;
alert("Score: " + scoreList["Mikkonen"] + " points");
```