



Taulukot

(English from page 4)

Ennen kuin alat tehdä näitä harjoituksia, sinun tulee katsoa kaikki videot ja lukea web-linkit otsikon 'Before you start the exercises' alta.

Lataa Moodlesta *Arrays*-zip-tiedosto ja pura omalle työasemallesi. Avaa projekti NetBeansissa. Projekti sisältää yhden *www*-sivun, pari tyylitiedostoa ja javascript-tiedoston. Mukana on myös kirjoitetun ohjelmakoodin testaamiseen tarvittava osa. Tehtävänäsi on kirjoittaa vaadittu toiminta JavaScript-tiedoston funktioihin.

Avaa projekti NetBeansissa, ota esille tiedostot *index.html* ja *arrays.js*. Kunkin tehtävän ratkaisun kirjoitat omaan funktioonsa. Avaa *index.html*-tiedosto myös selaimessa. Kun olet kirjoittanut ratkaisusi tehtävään, virkistä sivu selaimessa ja kokeile ratkaisiasi kirjoittamalla datan selaimen syöttökenttään ja klikkaamalla painiketta.

Älä muuta html- ja css-koodia. Esimerkkitulokset ovat tehtävien jälkeen.

Näissä tehtävissä sinun tulee tyhjentää tulostuselementit jokaisen painike-funktion alussa, jotta funktio toimii oikein.

Asioita, joita tarvitset eri tehtävissä

Kopioi ja liitä tämä taulukko määrittely globaaliksi taulukoksi.

```
let numbers = [15, 4, 68, 23, 22, 7, 15, 10, 32, 15];
```

Kopioi myös tähän js-tiedostoon satunnaisluvun arvontafunktio, jota olet käyttänyt edellisissä tehtävissäkin. Etsi kommentti, joka ohjaa sinua kopioimaan funktion oikeaan paikkaan.





T1: Taulukon peruskäsittelyä - functions t01a() ... t01b()

Print 1 - function t01a()

Sivulla on kaksi elementtiä tulostusta varten. Tulosta elementin `result_a1` ominaisuuteen innerHTML edellä oleva taulukko numbers. Elementtiin `result_a2` tulostat taulukon arvot yksi kerrallaan alkaen ensimmäisestä taulukkoalkiosta käyttäen for-toistoa. Laita välilyönti (" ") tulostettujen arvojen väliin.

Print 2 - function t01b()

Elementtiin `result_b` tulostat taulukon arvot yksi kerrallaan alkaen viimeisestä taulukkoalkiosta käyttäen for-toistoa. Laita välilyönti (" ") tulostettujen arvojen väliin.

Sum and avg - function t01c()

Laske taulukkoalkioiden summa ja keskiarvo. Tulosta molemmat.

Smallest and biggest - function t01d()

Etsi suurin ja pienin taulukkoalkion arvo. Logiikka::

- määrittele muuttujat smallest ja biggest
- laita ensimmäinen taulukkoalkio molempien arvoksi
- käy taulukko läpi for-toistolla ja tutki onko käsiteltävä taulukkoalkio pienempi kuin arvo muuttujassa smallest; jos on, aseta käsiteltävän taulukkoalkion arvo muuttujan smallest arvoksi.
- vastaava logiikka suurimman etsimiseksi

Lopuksi tulosta smallest ja biggest.

Vastaavan tyyppinen tehtävä oli Choices, V7: Smallest and biggest.

Sort - function t01e()

JavaScriptissä on funktio `sort()` lajittelua varten. Kokeile sitä; kirjoita `numbers.sort()`. Tulosta taulukkoalkiot yhdelle riville ensimmäisestä viimeiseen ja välilyönti välissä.

Huomaat – lajittelu ei ole oikein. Perehdy lajitteluun w3schools-sivustolla https://www.w3schools.com/jsref/jsref_sort.asp ja kirjoita oikea lajittelukomento.

New numbers - function t01f()

Tyhjennä taulukko luomalla uusi: `numbers = new Array();`

Käytä satunnaislukugeneraattorifunktiota ja luo 20 uutta satunnaislukua taulukkoon väliltä 1 ... 20. Lisää taulukkoon arvoja `push()`-funktiolla, https://www.w3schools.com/jsref/jsref_obj_array.asp

Tämän jälkeen kutsu jokaista painikefunktiota oheisen mallin mukaisesti:

```
t01a();  
t01b();  
...
```

T1: Basic array handling

Print 1

Array:
15,4,68,23,22,7,15,10,32,15
15 4 68 23 22 7 15 10 32 15

Print 2

Array:
15 32 10 15 7 22 23 68 4 15

Sum and avg

Sum: 211
Average: 21.1

Smallest and biggest

Smallest: 4
Biggest: 68

Sort

Array: 4 7 10 15 15 15 22 23 32 68





```
t01e();
```

Mitä tässä tapahtui?

T2: Joker - functions t02()

Luo paikallinen taulukko jokerinumeroille käyttäen lausetta `let jokers = new Array();`

Käytä satunnaislukugeneraattorifunktiota ja luo 7 numeroa taulukkoon väliltä 0 ... 9 (for-toisto). Tämän jälkeen kirjoita toinen for-toisto, jossa käyt läpi jokers-aulukon ensimmäisestä viimeiseen, tulostat jokaisen taulukkoalkion ja välilyönnin numeroiden väliin.





T3: Lotto - functions t03()

Lottoarvontaharjoitus. Arvotaan satunnaislukugeneraattorifunktiolla 7 lottonumeroa väliltä 1 ... 40, yksi lisännumero ja yksi plusnumero. Internetistä löytyy monia ohjeita tämän tekemiseen. Käytät nyt kuitenkin alla olevaa ohjeistusta.

Ohje:

- Luo paikallinen taulukko. Täytä taulukko nolilla siten, että viimeisen taulukkoalkion indeksi on 40 (for loop). Taulukkoalkiota indeksillä nolla ei käytetä tässä tehtävässä.
- Arvo satunnaislukugeneraattoria käyttäen 7 eri lukua väliltä 1 ... 40. Tarvitset tässä while-toistoa. Arvo luku. Käytä sitä indeksinä lottotaulukkoon ja tutki osoittaako se taulukkoalkioon, jossa on nolla. Mikäli näin on, niin aseta ko. taulukkoalkioon 1 (= lottonumero arvottu). Mikäli taulukkoalkiossa on jo luku 1, niin tiedät, että ko. paikka on jo käytössä ja sitä ei voi käyttää uudestaan. While-toistossa sinun täytyy laskea arvottuja lukuja siten, että 7 erilaista on arvottu.
- Kun 7 lottonumeroa on arvottu, niin sen jälkeen arvotaan lisännumero. Tämä ei voi olla sama kuin jo arvottu lottonumero. Tarvitset toisen while-toiston. Käytä samaa logikkaa kuin edellä ja laita taulukkoalkioon luku 2, kun indeksi osoittaa taulukkoalkioon, jossa on nolla.
- Nyt voit tulostaa lottonumerot, käy läpi taulukko for-toistolla. Mikäli taulukkoalkiossa on luku 1, niin indeksi kertoo lottonumeron ja *indeksi tulostetaan*. Tulosta lottonumerot yhdelle riville, välilyönti väliin.
- Lisännumero tulostetaan vastaavalla logiikalla, mutta etsitään taulukkoalkiota, jossa on luku 2. Taas tulostetaan indeksi ei lukua 2.
- Lopuksi arvot plusnumeron. Se on mikä tahansa luku väliltä 1 ... 40. Arvo ja tulosta.

Lotto numbers:
1 4 5 21 22 24 40
Extra number: 29
Plus number: 27

▼ numbers: Array(41) ^e

0: 0
1: 1
2: 0
3: 0
4: 1
5: 1
6: 0
7: 0
8: 0
9: 0
10: 0
11: 0
12: 0
13: 0
14: 0
15: 0
16: 0
17: 0
18: 0
19: 0
20: 0
21: 1
22: 1
23: 0
24: 1
25: 0
26: 0
27: 0
28: 0
29: 2
30: 0
31: 0
32: 0
33: 0
34: 0
35: 0
36: 0
37: 0
38: 0
39: 0
40: 1

T4: Random order - functions t04()

Määrittele paikallinen taulukko, jossa on luvut 1 ... 10 nousevassa järjestyksessä.

let numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];

- Tulosta taulukkoalkiot satunnaisessa järjestyksessä. Tämä tehdään while-toistolla, jota suoritetaan niin kauan kuin taulukossa on taulukkoalkioita, taulukon length > 0.
- Toistossa luot satunnaisluvun väliltä 0 ... array length-1, käytä tätä indeksinä





- Tulosta taulukkoalkio, johon edellä luotu indeksi osoittaa, välilyönti tulostuksen perään.
- Poista tulostettu taulukkoalkio taulukosta splice()-funktiolla
https://www.w3schools.com/jsref/jsref_splice.asp; lue ensin definition ja usage osat, sitten parametrien arvoista ja lopuksi esimerkki
- Jotta voit useamman kerran kokeilla tulostamista, niin tyhjennä tulostuselementti funktion aluksi.

Array exercises

T1: Basic array handling

Print 1

Array:
15,4,68,23,22,7,15,10,32,15
15 4 68 23 22 7 15 10 32 15

Print 2

Array:
15 32 10 15 7 22 23 68 4 15

Sum and avg

Sum: 211
Average: 21.1

Smallest and biggest

Smallest: 4
Biggest: 68

Sort

Array: 4 7 10 15 15 15 22 23 32
68

New numbers

T2: Joker

Joker

2 6 1 3 7 9 9

T3: Lotto

Lotto numbers

Lotto numbers:
1 4 5 21 22 24 40
Extra number: 29
Plus number: 27

T4: Random order

Random order

Array:
3 9 8 4 5 2 10 6 7 1





Arrays

Before you start these exercises you should watch all the videos and read web links under a title 'Before you start the exercises'.

Download from Moodle the zip file *Arrays*. Unzip it in your own computer. The file includes a NetBeans project. The project includes an html, a css and a JavaScript files. There is also a testing part for the project. Your task is to write program code in the JavaScript file.

Open the project in NetBeans, open the files *index.html* and *arrays.js*. Each exercise solution is to be written in a different function. Open the index.html file also in a browser. After you have written your solution to an exercise you refresh the page in a browser and test your solution by inserting the data and clicking the button.

Please, do not change any html and css code.

In these exercises you need to clear the contents of the output elements of a function at beginning of each button function.

Things you need in several tasks

Copy and paste this array definition so that it is a global array.

```
let numbers = [15, 4, 68, 23, 22, 7, 15, 10, 32, 15];
```

Copy-and-paste here in this exercise the random number generator function you have used in the previous exercises. See the note where to copy-and-paste.

T1: Basic array handling - functions t01a() ... t01b()

Print 1 - function t01a()

There are two elements for printing. In the element **result_a1** you just set the array to innerHTML property. In the element **result_a2** you print the array values one by one starting from the first value using for loop. Set a space (" ") between printed values.

Print 2 - function t01b()

In the element **result_b** you print the array values one by one starting from the *last* value using for loop. Set a space (" ") between printed values.

Sum and avg - function t01c()

Calculate the sum and average of the array values. Print the results.

Smallest and biggest - function t01d()

Find the smallest and biggest array value. Logic:

- define variables smallest and biggest
- set the first array element value to both of the variables

T1: Basic array handling

Print 1

Array:
15,4,68,23,22,7,15,10,32,15
15 4 68 23 22 7 15 10 32 15

Print 2

Array:
15 32 10 15 7 22 23 68 4 15

Sum and avg

Sum: 211
Average: 21.1

Smallest and biggest

Smallest: 4
Biggest: 68

Sort

Array: 4 7 10 15 15 15 22 23 32 68





- go through the array values with for loop and find out if the current array element value is smaller than the value in the variable smallest; if so, set the current array element value to smallest
- similar logic to find out the biggest

Finally print smallest and biggest.

Similar kind of exercise is Choices, V7: Smallest and biggest.

Sort - function t01e()

In JavaScript there is a function `sort()` for sorting. Try it; write `numbers.sort()`. Print then the array values in one line from first to last with space between.

You see the sorting is not right. Study the sorting in w3schools site

https://www.w3schools.com/jsref/jsref_sort.asp and write the right sort statement.

New numbers - function t01f()

Empty the array by creating a new array: `numbers = new Array();`

Use the random number generator function and generate 20 new random numbers in the array, range 1 ... 20. You insert values into an array element using `push()`,

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

After call each of the button functions one after another:

`t01a();`

`t01b();`

...

`t01e();`

T2: Joker - functions t02()

Create a local array for joker numbers using statement `let jokers = new Array();`

Use the random number generator function to create 7 numbers in the array, range 0 ... 9. Finally print the numbers from the array in one line with a space between.





T3: Lotto - functions t03()

This is a Finnish lotto exercise. Use the random number generator function to create 7 lotto numbers, range 1 ... 40, one extra number and one plus number. There are many advices to be found in the internet about how to do this task. Now you must use the logic described in this exercise.

Advice:

- Create a local array. Fill the array with zeros so that the last element has an index 40 (for loop). The array element with an index zero is not used in this exercise.
- Use the random number generator function to create 7 different numbers, range 1 ... 40. You need to use while loop here. Generate a random number. Use it as an index to a lotto array and see if the index points to an element having a value zero. If so set the element value to one. If the generated index points to an array element having value 1 you know that this number has been already used and should not be used again. You need to count the generated numbers so that 7 different have been generated.
- After the 7 lotto numbers have been generated you need to generate the extra number. This must be different from lotto numbers. Use the same logic as before (while loop), set value to 2 when the index points to an array element having value zero.
- Now you are ready to print the lotto numbers. Go through the lotto array (for loop) and find the array elements that has a value 1 (= a lotto number). Lotto number is not 1 but it is the index of that array element. *So print the index value.* Printed index values have a space between.
- The next task is to print the extra number. Similar way go through lotto array (for loop) and find an array element that has a value 2. That is the extra number. Again print the index value, not the array value.
- Finally you generate the plus number. It is any number between 1 and 40. Generate and print.

Lotto numbers:
1 4 5 21 22 24 40
Extra number: 29
Plus number: 27

```
▼ numbers: Array(41)
0: 0
1: 1
2: 0
3: 0
4: 1
5: 1
6: 0
7: 0
8: 0
9: 0
10: 0
11: 0
12: 0
13: 0
14: 0
15: 0
16: 0
17: 0
18: 0
19: 0
20: 0
21: 1
22: 1
23: 0
24: 1
25: 0
26: 0
27: 0
28: 0
29: 2
30: 0
31: 0
32: 0
33: 0
34: 0
35: 0
36: 0
37: 0
38: 0
39: 0
40: 1
```

T4: Random order - functions t04()

Define a local array to have numbers from 1 to 10 in ascending order.

```
let numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];
```

Your task is to print them in the random order. Advice here:





- You do it with a while loop as long there are elements in the array, array length > 0.
- Generate an index using random number generator, range 0 ... array length-1
- Print the value from the array using the random index, a space between the printed values
- Remove the printed value from the array, use splice() function, https://www.w3schools.com/jsref/jsref_splice.asp; read at first the definition and usage section, then parameter values and finally the very last example.
- To be able to run this button several times, clear the print element at the beginning

Array exercises

T1: Basic array handling

Print 1

Array:
15,4,68,23,22,7,15,10,32,15
15 4 68 23 22 7 15 10 32 15

Print 2

Array:
15 32 10 15 7 22 23 68 4 15

Sum and avg

Sum: 211
Average: 21.1

Smallest and biggest

Smallest: 4
Biggest: 68

Sort

Array: 4 7 10 15 15 15 22 23 32
68

New numbers

T2: Joker

Joker

2 6 1 3 7 9 9

T3: Lotto

Lotto numbers

Lotto numbers:
1 4 5 21 22 24 40
Extra number: 29
Plus number: 27

T4: Random order

Random order

Array:
3 9 8 4 5 2 10 6 7 1

