

Toistot

(English from page 4)

Perehdy oppimateriaaliin ennen kuin alat tehdä näitä harjoituksia.

Lataa Moodlesta *Loops*-zip-tiedosto ja pura omalle työasemallesi. Harjoitus sisältää html-, css- ja javascript-tiedostoja. Mukana on myös kirjoitetun ohjelmakoodin testaamiseen tarvittava osa. Tehtävänäsi on kirjoittaa vaadittu toiminta JavaScript-tiedoston funktioihin.

Avaa harjoituksen kansio Visual Studiossa, ota esille tiedostot *index.html* ja *loops.js*. Kunkin tehtävän ratkaisun kirjoitat omaan funktioonsa. Avaa index.html-tiedosto myös selaimessa. Kun olet kirjoittanut ratkaisusi tehtävään, virkistä sivu selaimessa ja kokeile ratkaisuasi 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.

L1: Your name - function l01()

Lue etu- ja sukunimesi. Tulosta etunimesi for-toistolla 5 kertaa elementtiin **first_name**. Tulosta sukunimesi while-toistolla 5 kertaa elementtiin **last_name**. Tulosta kukin nimesi esiintyminen eri riville, käytä `
`.

L2: Numbers - function l02()

Tulosta luvut viiden välein luvusta 5 lukuun 50 käyttäen for-toistoa. Katso esimerkkitulostus.

Tulosta luvut kuuden välein luvusta 6 lukuun 60 käyttäen while-toistoa.

L3: Stars - function l03()

Tulosta tähtirakennelma käyttäjän antaman luvun mukaan. Ensimmäisellä rivillä on yksi tähti, toisella kaksi, kolmannella kolme, jne. kunnes viimeisellä rivillä tähtien määrä on sama kuin käyttäjän antama luku.

Tee ensin for-toistolla, sitten while-toistolla.

```
*
**
***
****
*
**
***
****
```

L4: Numbers - function l04()

Tulosta numerot 10, 9, 8, ..., -8, -9, -10 elementtiin **numbers1**.

Tulosta numerot -20, -18, -16, ..., 16, 18, 20 elementtiin **numbers2**.

Tulosta numerot 29, 27, ..., 1 elementtiin **numbers3**.

L5: Divisibility - function l05()

Tulosta 3:lla jaolliset luvut käyttäjän antamalta lukuväliltä (min, max) elementtiin **divisibility3**.

Tulostaa 5:llä jaolliset luvut käyttäjän antamalta lukuväliltä elementtiin **divisibility5**.



Liisa Auer

Luku on jaollinen toisella numerolla, kun jakojäännös on nolla.

L6: Heads or tails - function l06()

Kirjoita ohjelma, joka arpoo 1000 kertaa luvun 0 tai 1. Laske kuinka monta kertaa tuli 1 ja kuinka monta kertaa 0. Tulosta jakauma. Tarvitset tässä muuttujat, joihin lasket yksi kerrallaan lukujen 0 ja 1 esiintymiset. Määrittele muuttujat ennen for-toistoa ja tulosta sisältö for-toiston jälkeen.

Kopioi myös tähän js-tiedostoon satunnaisluvun arvontafunktio, jota olet käyttänyt edellisissä tehtävissäkin.

L7: Joker number - function l07()

Tämä tehtävä simuloi jokerinumeron arvontaa, joka on osa viikkoittaista lottoarvontaa.

Kirjoita koodi, joka arpoo viikkonumeron (1 ... 52) ja tulostaa sen. Sen jälkeen kirjoita koodi, joka arpoo 7 satunnaista numeroa väliltä 0 ... 9 ja tulostaa ne sivulle. Tulostettujen lukujen välissä on välilyönti (" ").

L8: Multiply - function l08()

Lue kertoja input-elementistä. Tulosta kertotaulu numerosta 0 numeroon 10.

L9: Multiply all - function l09()

Tee ohjelma (kahta sisäkkäistä for -rakennetta käyttäen), joka tulostaa kaikki kertotaulut 0-10. Jokaisen kertotaulun jälkeen on puolipiste ja kaksi välilyöntiä (= välilyönti, joka aina tulostuu, nbsp = non-breakable space). Käytä määriteltyä vakiota. Sisemmän toiston loppumisen jälkeen on rivinvaihto (
).

Loop exercises

L1: Your name

First name:

Last name:

Liisa
Liisa
Liisa
Liisa
Liisa

Auer
Auer
Auer
Auer
Auer

L2: Numbers

5 10 15 20 25 30 35 40 45 50

6 12 18 24 30 36 42 48 54 60

L3: Stars

Amount:

*
**

*
**

L4: Numbers

10 9 8 7 6 5 4 3 2 1 0 -1 -2 -3 -4
-5 -6 -7 -8 -9 -10-20 -18 -16 -14 -12 -10 -8 -6 -4
-2 0 2 4 6 8 10 12 14 16 18 2029 27 25 23 21 19 17 15 13 11 9
7 5 3 1

L5: Divisibility

Min: Max:

3 6 9 12 15 18 21 24 27

5 10 15 20 25





Liisa Auer

Random numbers

L6: Heads or tails

Zeros: 499
Ones: 501

L7: Joker number

Week: 29
Number: 1 6 0 6 1 5 8

Multiplication

L8: Multiply

Multiplier:

4 * 0 = 0
4 * 1 = 4
4 * 2 = 8
4 * 3 = 12
4 * 4 = 16
4 * 5 = 20
4 * 6 = 24
4 * 7 = 28
4 * 8 = 32
4 * 9 = 36
4 * 10 = 40

L9: Multiply all

0 * 0 = 0; 0 * 1 = 0; 0 * 2 = 0; 0 * 3 = 0; 0 * 4 = 0; 0 * 5 = 0; 0 * 6 = 0; 0 * 7 = 0; 0 * 8 = 0; 0 * 9 = 0; 0 * 10 = 0;
1 * 0 = 0; 1 * 1 = 1; 1 * 2 = 2; 1 * 3 = 3; 1 * 4 = 4; 1 * 5 = 5; 1 * 6 = 6; 1 * 7 = 7; 1 * 8 = 8; 1 * 9 = 9; 1 * 10 = 10;
2 * 0 = 0; 2 * 1 = 2; 2 * 2 = 4; 2 * 3 = 6; 2 * 4 = 8; 2 * 5 = 10; 2 * 6 = 12; 2 * 7 = 14; 2 * 8 = 16; 2 * 9 = 18; 2 * 10 = 20;
3 * 0 = 0; 3 * 1 = 3; 3 * 2 = 6; 3 * 3 = 9; 3 * 4 = 12; 3 * 5 = 15; 3 * 6 = 18; 3 * 7 = 21; 3 * 8 = 24; 3 * 9 = 27; 3 * 10 = 30;
4 * 0 = 0; 4 * 1 = 4; 4 * 2 = 8; 4 * 3 = 12; 4 * 4 = 16; 4 * 5 = 20; 4 * 6 = 24; 4 * 7 = 28; 4 * 8 = 32; 4 * 9 = 36; 4 * 10 = 40;
5 * 0 = 0; 5 * 1 = 5; 5 * 2 = 10; 5 * 3 = 15; 5 * 4 = 20; 5 * 5 = 25; 5 * 6 = 30; 5 * 7 = 35; 5 * 8 = 40; 5 * 9 = 45; 5 * 10 = 50;
6 * 0 = 0; 6 * 1 = 6; 6 * 2 = 12; 6 * 3 = 18; 6 * 4 = 24; 6 * 5 = 30; 6 * 6 = 36; 6 * 7 = 42; 6 * 8 = 48; 6 * 9 = 54; 6 * 10 = 60;
7 * 0 = 0; 7 * 1 = 7; 7 * 2 = 14; 7 * 3 = 21; 7 * 4 = 28; 7 * 5 = 35; 7 * 6 = 42; 7 * 7 = 49; 7 * 8 = 56; 7 * 9 = 63; 7 * 10 = 70;
8 * 0 = 0; 8 * 1 = 8; 8 * 2 = 16; 8 * 3 = 24; 8 * 4 = 32; 8 * 5 = 40; 8 * 6 = 48; 8 * 7 = 56; 8 * 8 = 64; 8 * 9 = 72; 8 * 10 = 80;
9 * 0 = 0; 9 * 1 = 9; 9 * 2 = 18; 9 * 3 = 27; 9 * 4 = 36; 9 * 5 = 45; 9 * 6 = 54; 9 * 7 = 63; 9 * 8 = 72; 9 * 9 = 81; 9 * 10 = 90;
10 * 0 = 0; 10 * 1 = 10; 10 * 2 = 20; 10 * 3 = 30; 10 * 4 = 40; 10 * 5 = 50; 10 * 6 = 60; 10 * 7 = 70; 10 * 8 = 80; 10 * 9 = 90; 10 * 10 = 100;

Liisa Auer, Oulu University of Applied Sciences, [CC-BY 4.0](#)

Opetus- ja
kulttuuri-
ministeriö

TURKU AMK



OAMK
OULUN AMMATTIKORKEAKOULU

Centria
Jyväskylän ammattikorkeakoulu

jamk.fi
Jyväskylän ammattikorkeakoulu

VAMK
VAASAN AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES

KAMK • University
of Applied Sciences





Loops

Study the studying material before you start these exercises .

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

Open the exercise folder in Visual Studio, open the files *index.html* and *loops.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.

L1: Your name - function l01()

Read your first name and last name. Print using for loop your first name 5 times in the element **first_name**. Print using while loop your last name 5 times in the element **last_name**. Print each of your names in a different row, use
.

L2: Numbers - function l02()

Print numbers from 5 to 50. The next number in sequence should be 5 greater than the previous. Use for loop. See the example image.

Print numbers from 6 to 60. The next number in sequence should be 6 greater than the previous. Use while loop.

L3: Stars - function l03()

Print number of stars based on the input number. In the first row there is one star, in the second row two stars, etc. until in the last row is the number of stars of the input number.

Write at first a for loop, the other using while loop.

```
*
**
***
****
*
**
***
****
```

L4: Numbers - function l04()

Print numbers 10, 9, 8, ..., -8, -9, -10 to an element **numbers1**.

Print numbers -20, -18, -16, ... , 16, 18, 20 to an element **numbers2**.

Print numbers 29, 27, ..., 1 to an element **numbers3**.

L5: Divisibility - function l05()

Print numbers divisible by 3 from the min to max range to an element **divisibility3**.

Print numbers divisible by 5 from the min to max range to an element **divisibility5**.





Number is divisible by the other number when the remainder is zero.

L6: Heads or tails - function l06()

Write a code that draws randomly 1000 times a number 0 or 1. Count how many times there was a number 1 and how many times number 0. Print the result. Here you need variables to count the occurrences of the numbers 0 and 1 one at a time. Define variables before for loop and print their content after for loop.

Copy-and-paste here in this exercise the random number generator function you have used in the previous exercises.

L7: Joker number - function l07()

This task simulates a weekly a joker number draw that is a part of weekly lotto draw in Finland.

Write a code that draws a random week number (1 ... 52) and prints it. After write a code that draws 7 random numbers, range 0 ... 9 and prints them in an element. Between printed numbers there is a space (" ").

L8: Multiply - function l08()

Read the multiplier from the input element. Print the multiplication table from the number 0 to number 10.

L9: Multiply all - function l09()

Write a program using two for loops inside each other that prints all multiplication tables 0 ... 10. Between each multiplication is a semicolon (;) and two non-breakable spaces (" "). Use the defined constant for spaces. After the inner for loop there should be a line break (
).

Loop exercises

L1: Your name

First name:

Liisa

Last name:

Auer

Names

Liisa
Liisa
Liisa
Liisa
Liisa

Auer
Auer
Auer
Auer
Auer

L2: Numbers

Numbers

5 10 15 20 25 30 35 40 45 50

6 12 18 24 30 36 42 48 54 60

L3: Stars

Amount: 6

Stars

*
**

*
**

L4: Numbers

Numbers

10 9 8 7 6 5 4 3 2 1 0 -1 -2 -3 -4
-5 -6 -7 -8 -9 -10

-20 -18 -16 -14 -12 -10 -8 -6 -4
-2 0 2 4 6 8 10 12 14 16 18 20

29 27 25 23 21 19 17 15 13 11 9
7 5 3 1

L5: Divisibility

Min: 2

Max: 27

Divisibility

3 6 9 12 15 18 21 24 27

5 10 15 20 25





Liisa Auer

Random numbers

L6: Heads or tails

Zeros: 499
Ones: 501

L7: Joker number

Week: 29
Number: 1 6 0 6 1 5 8

Multiplication

L8: Multiply

Multiplier:

4 * 0 = 0
4 * 1 = 4
4 * 2 = 8
4 * 3 = 12
4 * 4 = 16
4 * 5 = 20
4 * 6 = 24
4 * 7 = 28
4 * 8 = 32
4 * 9 = 36
4 * 10 = 40

L9: Multiply all

0 * 0 = 0; 0 * 1 = 0; 0 * 2 = 0; 0 * 3 = 0; 0 * 4 = 0; 0 * 5 = 0; 0 * 6 = 0; 0 * 7 = 0; 0 * 8 = 0; 0 * 9 = 0; 0 * 10 = 0;
1 * 0 = 0; 1 * 1 = 1; 1 * 2 = 2; 1 * 3 = 3; 1 * 4 = 4; 1 * 5 = 5; 1 * 6 = 6; 1 * 7 = 7; 1 * 8 = 8; 1 * 9 = 9; 1 * 10 = 10;
2 * 0 = 0; 2 * 1 = 2; 2 * 2 = 4; 2 * 3 = 6; 2 * 4 = 8; 2 * 5 = 10; 2 * 6 = 12; 2 * 7 = 14; 2 * 8 = 16; 2 * 9 = 18; 2 * 10 = 20;
3 * 0 = 0; 3 * 1 = 3; 3 * 2 = 6; 3 * 3 = 9; 3 * 4 = 12; 3 * 5 = 15; 3 * 6 = 18; 3 * 7 = 21; 3 * 8 = 24; 3 * 9 = 27; 3 * 10 = 30;
4 * 0 = 0; 4 * 1 = 4; 4 * 2 = 8; 4 * 3 = 12; 4 * 4 = 16; 4 * 5 = 20; 4 * 6 = 24; 4 * 7 = 28; 4 * 8 = 32; 4 * 9 = 36; 4 * 10 = 40;
5 * 0 = 0; 5 * 1 = 5; 5 * 2 = 10; 5 * 3 = 15; 5 * 4 = 20; 5 * 5 = 25; 5 * 6 = 30; 5 * 7 = 35; 5 * 8 = 40; 5 * 9 = 45; 5 * 10 = 50;
6 * 0 = 0; 6 * 1 = 6; 6 * 2 = 12; 6 * 3 = 18; 6 * 4 = 24; 6 * 5 = 30; 6 * 6 = 36; 6 * 7 = 42; 6 * 8 = 48; 6 * 9 = 54; 6 * 10 = 60;
7 * 0 = 0; 7 * 1 = 7; 7 * 2 = 14; 7 * 3 = 21; 7 * 4 = 28; 7 * 5 = 35; 7 * 6 = 42; 7 * 7 = 49; 7 * 8 = 56; 7 * 9 = 63; 7 * 10 = 70;
8 * 0 = 0; 8 * 1 = 8; 8 * 2 = 16; 8 * 3 = 24; 8 * 4 = 32; 8 * 5 = 40; 8 * 6 = 48; 8 * 7 = 56; 8 * 8 = 64; 8 * 9 = 72; 8 * 10 = 80;
9 * 0 = 0; 9 * 1 = 9; 9 * 2 = 18; 9 * 3 = 27; 9 * 4 = 36; 9 * 5 = 45; 9 * 6 = 54; 9 * 7 = 63; 9 * 8 = 72; 9 * 9 = 81; 9 * 10 = 90;
10 * 0 = 0; 10 * 1 = 10; 10 * 2 = 20; 10 * 3 = 30; 10 * 4 = 40; 10 * 5 = 50; 10 * 6 = 60; 10 * 7 = 70; 10 * 8 = 80; 10 * 9 = 90; 10 * 10 = 100;

Liisa Auer, Oulu University of Applied Sciences, [CC-BY 4.0](#)

Opetus- ja
kulttuuri-
ministeriö

TURKU AMK



OAMK
OULUN AMMATTIKORKEAKOULU



Centria
Jyväskylän ammattikorkeakoulu

jamk.fi

Jyväskylän ammattikorkeakoulu

VAMK
VAASAN AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES

KAMK • University
of Applied Sciences

