



VILNIUSCODINGSCHOOL

# Javascript

**Populiariausia programavimo kalba**



- Algoritmų, loginių sekų užrašymas pasirinkta programavimo kalba, kurią supranta kompiuteris
- Programuotojas - tarpininkas tarp kompiuterio ir vartotojo
- Programuotojas - logiškai mąstantis žmogus, gebantis sugalvoti sudėtingus algoritmus
- Išmokus vieną programavimo kalbą, tie patys principai panaudojami ir kitoje programavimo kalboje



- **1995 m. Netscape**
- **LiveScript -> JavaScript**
- **JavaScript != Java**
- **Kompiluojamas naršyklėje**

# Prijungiame Javascript prie HTML



1. **Susikuriame naują projektą, pvz.:**  
**webmokymai/js01/**
2. **Susikuriame dvi direktorijas: **css** ir **scripts****
3. ****css/style.css** ir **scripts/custom.js****
4. ****custom.js** pridedame vieną eilutę:**

```
console.log("Pavyko!");
```

# Prijungiame Javascript prie HTML



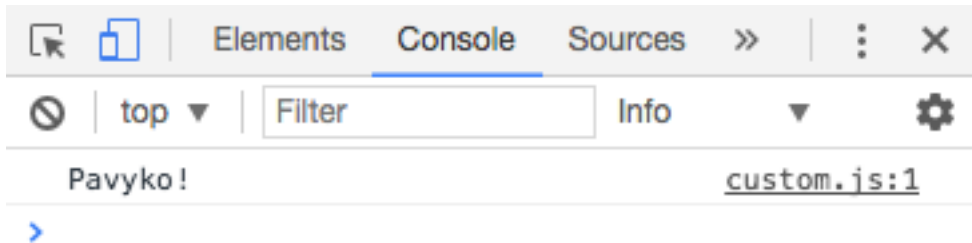
## webmokymai/js01/ sukuriame index.html

```
<!DOCTYPE html>
<html lang="">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title></title>
  <link rel="stylesheet" href="css/style.css">
</head>
<body>
  <h1>Nematau jokio Javascript! Kur jis???</h1>
  <script src="scripts/custom.js"></script>
</body>
</html>
```

# Prijungiame Javascript prie HTML



**Nematau jokio Javascript. Kur jis???**



# Kintamieji ir jų išvedimas konsolėje - let/var/const



```
let num = 6 + 4;  
let vcs = "Labas";  
let relationship = (vcs == num);  
console.log(num + ", " + vcs + ", " + typeof(vcs) + ", " + relationship);  
console.log(`${num}, ${vcs}, ${typeof(vcs)}, ${relationship}`);
```

10, Labas, string, false

custom.js:4:9

10, Labas, string, false

custom.js:5:9

**Pratimas:** sukurti du kintamuosius vardui ir pavardei, priskirti jiems reikšmes, tada sukurti trečią kintamąjį, kuris apjungtų vardo ir pavardės kintamuosius, ir išspausdinti trečio kintamojo reikšmę konsolėje.

# Svarbu!



**1. Javascript yra “case sensitive”:**  
**let one** ir **let One** - ne tas pats!

**2. Nepamirštame rašyti komentarų**

```
15
16 // Vienos eilutės komentaras
17
18 ▾ /*
19   Dviejų eilučių
20   komentaras
21   (gali būti ir daugiau)
22   */|
```



# Svarbu!



**3. Kintamųjų pavadinimai negali prasidėti skaičiumi  
ir negali naudoti Javascript rezervuotų žodžių**

**[http://www.w3schools.com/js/js\\_reserved.asp](http://www.w3schools.com/js/js_reserved.asp)**



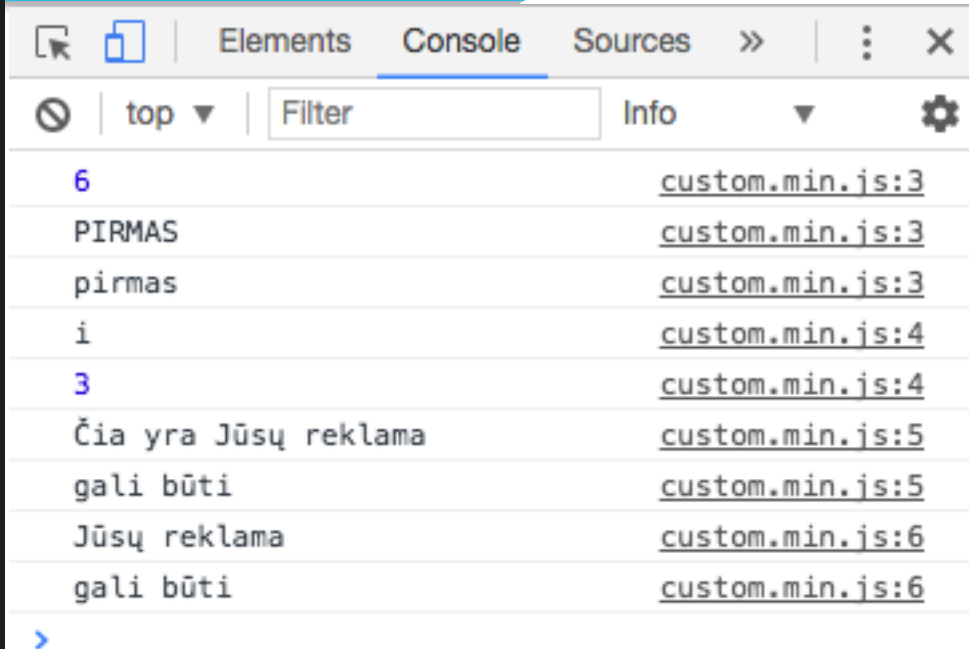
**Savybės pateikia tam tikrą informaciją apie kintamojo duomenis**

**Metodai - funkcijos, kurios manipuliuoja duomenimis esančiais kintamuosiuose**

# Savybės ir metodai tekstinėms eilutėms



```
let tekstas = "Pirmas";  
  
// tekstinės eilutės ilgis  
console.log(tekstas.length);  
  
// didžiosiomis raidėmis  
console.log(tekstas.toUpperCase());  
  
// mažosiomis raidėmis  
console.log(tekstas.toLowerCase());  
  
// išveda nurodytą tekstinės eilutės simbolį pagal eilę  
console.log(tekstas.charAt(1));  
  
// išveda ieškomos raidės ar žodžio pradžios poziciją eilutėje  
console.log(tekstas.indexOf("ma"));  
  
tekstas = "Čia gali būti Jūsų reklama";  
  
// pakeičia nurodytą eilutės dalį į naują  
console.log(tekstas.replace("gali būti", "yra"));  
  
// išveda atitinkamą eilutės dalį pagal mūsų nurodytas pozicijas  
console.log(tekstas.substring(4,14));  
console.log(tekstas.substring(14));  
  
/* skirtingai nuo substring, substr antroje funkcijos  
reikšmėje nurodome ne pabaigos poziciją, o ilgį */  
console.log(tekstas.substr(4, 9));
```



# Math funkcijos



```
let x = 6.34;  
console.log(Math.round(x));  
console.log(Math.ceil(x));  
console.log(Math.floor(x));
```

```
let y = 5.8;  
let z = 9;  
console.log(Math.max(x,y,z));  
console.log(Math.min(x,y,z));
```

```
let i = Math.random() * 30; // nuo 0 iki 30  
console.log(i);
```

```
let a = Math.floor(Math.random() * (50 - 10 + 1) + 10) // nuo 10 iki 50  
console.log(a);
```



# Metodai su skaičiais



```
let skaicius = 12.14;

// išveda sveikąjį skaičių
console.log(parseInt(skaicius, 10));

// išveda mišrųjį skaičių
console.log(parseFloat(skaicius));

// skaičių paverčia tekstine eilute
console.log(skaicius.toString());

// nurodo kiek skaičių po kablelio išvesti
console.log(skaicius.toFixed(3));

// nurodo kiek skaičių išvesti bendrai
console.log(skaicius.toPrecision(3));
```

Elements Console Sources >> ⋮ ✕	
⊘ top ▼	Filter Info ▼ ⚙
12	<u>custom.min.js:3</u>
12.14	<u>custom.min.js:3</u>
12.14	<u>custom.min.js:3</u>
12.140	<u>custom.min.js:4</u>
12.1	<u>custom.min.js:4</u>

# Lyginimo operatoriai



<b>==</b>	<b>// lygu, lyginami gali būti skirtingų rūšių kintamieji</b>
<b>===</b>	<b>// lygu, lyginami gali būti tik tos pačios rūšies kintamieji, kitu atveju - false</b>
<b>&gt;</b>	<b>// daugiau</b>
<b>&gt;=</b>	<b>// daugiau arba lygu</b>
<b>&lt;</b>	<b>// mažiau</b>
<b>&lt;=</b>	<b>// mažiau arba lygu</b>
<b>!=</b>	<b>// nelygu, lyginami gali būti skirtingų rūšių kintamieji</b>
<b>!==</b>	<b>// nelygu, lyginami gali būti tik tos pačios rūšies kintamieji, kitu atveju - false</b>

# Switch



```
let projektoBalas = 10;

switch(projektoBalas) {
  case 10:
    console.log('Tobulas darbas.');
```

break;

```
  case 9 :
    console.log('Labai gerai.');
```

break;

```
  case 8:
    console.log('Geras darbas.');
```

break;

```
  default:
    console.log('Prastokai.');
```

}

# While ciklas



```
/* jei norėtume išvėti skaičius nuo 1 iki 10  
galėtume rašyti taip */
```

```
let number = 1;  
console.log(number);  
console.log(number + 1);  
console.log(number + 2);  
console.log(number + 3);  
console.log(number + 4);  
console.log(number + 5);  
console.log(number + 6);  
console.log(number + 7);  
console.log(number + 8);  
console.log(number + 9);
```

```
// ...bet daug geriau taip!
```

```
let number = 1;  
while (number <= 10) {  
    console.log(number);  
    number++;  
}
```

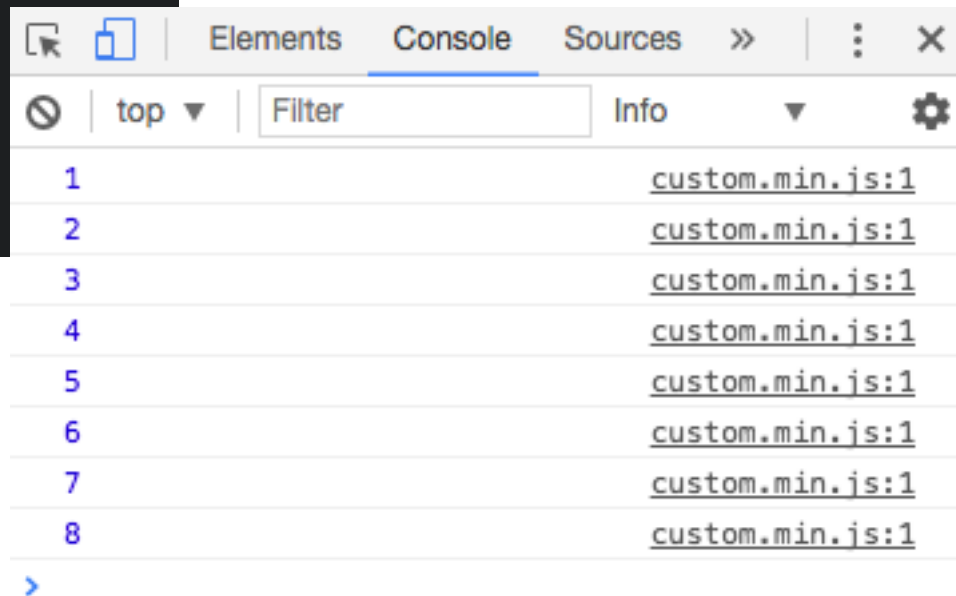
Elements Console Sources >> ⋮ ✕	
⊘ top ▾ Filter Info ▾ ⚙	
1	<u>custom.min.js:1</u>
2	<u>custom.min.js:1</u>
3	<u>custom.min.js:1</u>
4	<u>custom.min.js:1</u>
5	<u>custom.min.js:1</u>
6	<u>custom.min.js:1</u>
7	<u>custom.min.js:1</u>
8	<u>custom.min.js:1</u>
9	<u>custom.min.js:1</u>
10	<u>custom.min.js:1</u>



# While ciklas



```
// pakeitus bigNumber, mūsų loop vis tiek veiks  
  
let number = 1;  
let bigNumber = 8;  
  
while (number <= bigNumber) {  
  console.log(number);  
  number++;  
}
```



# For ciklas



```
let suma = 0;

for (let i = 0; i < 5; i++) {
    suma += i;
}

console.log(suma);
```

```
let suma = 0;
let skaiciai = [10, 20, 30, 40, 50, 60, 70, 80];

for (let i = 0; i < skaiciai.length; i++) {
    suma += skaiciai[i];
}

console.log(suma);
```

# For ir while kartu (if irgi)



```
let number = 1;
let niceNumber = 10;
let bigNumber = 15;

while(number < niceNumber) {
  console.log(`Skaičius ${number} yra mažesnis už 10`);
  number++;
}

for(number; number <= bigNumber; number++) {
  if(number == niceNumber) {
    console.log(`Skaičius ${number} yra lygus 10`);
  } else {
    console.log(`Skaičius ${number} yra didesnis už 10`);
  }
}
```

Skaičius 1 yra mažesnis už 10
Skaičius 2 yra mažesnis už 10
Skaičius 3 yra mažesnis už 10
Skaičius 4 yra mažesnis už 10
Skaičius 5 yra mažesnis už 10
Skaičius 6 yra mažesnis už 10
Skaičius 7 yra mažesnis už 10
Skaičius 8 yra mažesnis už 10
Skaičius 9 yra mažesnis už 10
Skaičius 10 yra lygus 10
Skaičius 11 yra didesnis už 10
Skaičius 12 yra didesnis už 10
Skaičius 13 yra didesnis už 10
Skaičius 14 yra didesnis už 10
Skaičius 15 yra didesnis už 10

# if, else if, else



```
let number = 1;
let niceNumber = 10;
let bigNumber = 15;

while(number < niceNumber) {
  console.log(`Skaičius ${number} yra mažesnis už 10`);
  number++;
}

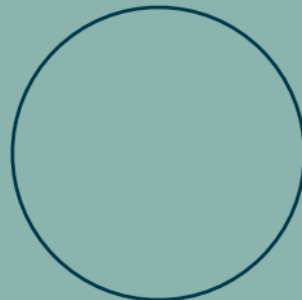
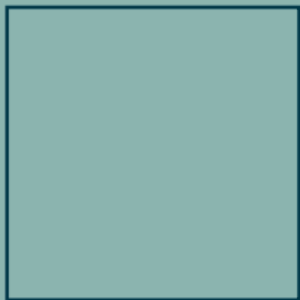
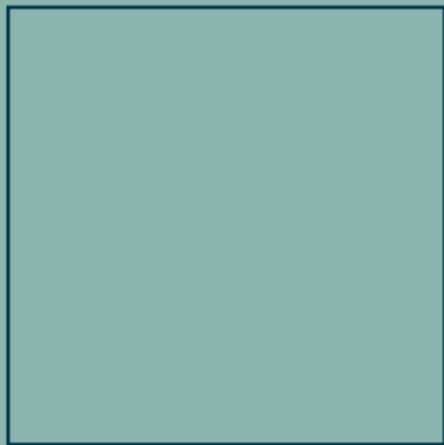
for(number; number <= bigNumber; number++) {
  if (number < niceNumber) {
    console.log(`Skaičius ${number} yra mažesnis už 10`);
  } else if (number == niceNumber) {
    console.log(`Skaičius ${number} yra lygus 10`);
  } else {
    console.log(`Skaičius ${number} yra didesnis už 10`);
  }
}
```

```
Skaičius 1 yra mažesnis už 10
Skaičius 2 yra mažesnis už 10
Skaičius 3 yra mažesnis už 10
Skaičius 4 yra mažesnis už 10
Skaičius 5 yra mažesnis už 10
Skaičius 6 yra mažesnis už 10
Skaičius 7 yra mažesnis už 10
Skaičius 8 yra mažesnis už 10
Skaičius 9 yra mažesnis už 10
Skaičius 10 yra lygus 10
Skaičius 11 yra didesnis už 10
Skaičius 12 yra didesnis už 10
Skaičius 13 yra didesnis už 10
Skaičius 14 yra didesnis už 10
Skaičius 15 yra didesnis už 10
```

# Nested sąlygos



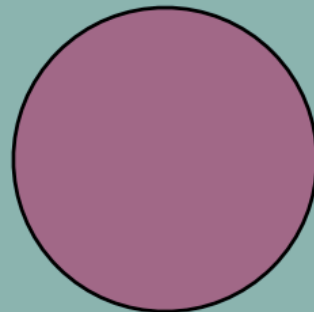
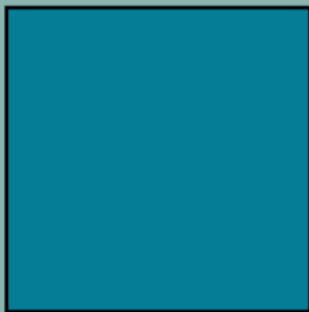
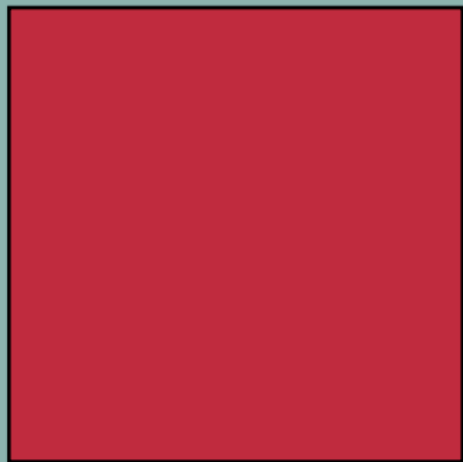
**Turime du skirtingų dydžių kvadratus ir vieną apskritimą**



# Nested sąlygos



**Didelį kvadratą norime nudažyti raudonai, mažą kvadratą - mėlynai, o apskritimą - violetine spalva**



# Nested sąlygos



```
if (*kvadratas*) {  
    if (didelisKvadratas) {  
        *nuspelvink raudonai*  
    } else {  
        *kadangi tai ne didelis, tuomet tai mažas kvadratas, nuspelvink mėlynai.*  
    }  
} else {  
    *kadangi tai ne kvadratas, tuomet tai apskritimas, nuspelvink violetine.*  
}
```

# Sudėtingesnės sąlygos



**Galime duoti daugiau nei vieną sąlygą:**

**&& - ir**  
**|| - arba**

**true && false**  
**false**

**true && true**  
**true**

**false && false**  
**false**

**false || true**  
**true**

**false || false**  
**false**

**true || true**  
**true**



# Sudėtingesnės sąlygos



```
( 11 >= 11 ) && ( -7 < 6 )
```

```
true      &&      true
```

```
→ true
```

```
( 5 < 7 ) || ( 8 > 10 )
```

```
true      ||      false
```

```
→ true
```

```
( 2 >= 0 ) && ( 9 < 4 )
```

```
true      &&      false
```

```
→ false
```

```
( 3 > 8 ) || ( 7 < 3 )
```

```
false     ||      false
```

```
→ false
```

# Esamos Javascript funkcijas



```
alert("Dēmesio! Jūsu kompiuteryje virusas!");
```

This page says:

Dēmesio! Jūsų kompiuteryje virusas!

OK

# Esamos Javascript funkcijos



```
confirm("Ar tikrai norite uždaryti šį puslapį?");
```

This page says:

Ar tikrai norite uždaryti šį puslapį?

Cancel

OK

# Esamos Javascript funkcijos



```
prompt("Koks Tavo vardas?");
```

This page says:

Koks Tavo vardas?

Cancel

OK

# Esamos Javascript funkcijos



```
let turimVarda = false;

while (turimVarda == false) {
    let userVardas = prompt("Koks tavo vardas?");

    if(userVardas) {
        if(confirm(`Ar tikrai tavo vardas ${userVardas}?`)) {
            alert(`Labas, ${userVardas}!`);
            turimVarda = true;
        } else {
            alert("Bandyk dar kart.");
        }
    } else {
        alert("Bandyk dar kart.");
    }
}
```

# Funkcijos



```
function sumOfCubes(a, b) {  
    return a * a * a + b * b * b;  
}  
  
let mySum = sumOfCubes(4, 9);  
console.log(mySum);
```

Funkcija priima duomenis, atlieka veiksmą ir pateikia rezultatą. Jos pavadinimas atspindi funkcijos paskirtį. Skliausteliuose jos parametrai, kurie naudojami funkcijos atliekamam veiksmui. **return** pateikia rezultatą.

# Funkcijos



```
// Funkcijos aprašymas
function hello () {
    console.log("Hello World!");
}

// Funkcijos iškvietimas
hello();
```

```
// Funkcijos aprašymas
function hello(a) {
    console.log(a);
}

// Funkcijos iškvietimas
hello('Hello World!');
```

```
let hi = "Hello World!";

// Funkcijos iškvietimas
hello(hi);

// Funkcijos aprašymas
function hello(a) {
    console.log(a);
}
```

**Funkcijas aprašome ir tuomet galime jas išsikviesti toje vietoje, kur norime. Kad funkcijos būtų kuo universalesnės, geriau rašyti jas taip, kad kiekviena atliktų tik vieną veiksmą.**

# Funkcijos



```
function kiekE() {  
    let fraze = prompt("Kokią frazę norėtum analizuoti?");  
    if(typeof(fraze) != "string") {  
        alert("Tai ne frazė.");  
        return false;  
    } else {  
        let eSkaicius = 0;  
        for (let i = 0; i < fraze.length; i++) {  
            if (fraze.charAt(i) == 'e' || fraze.charAt(i) == 'E') {  
                eSkaicius++;  
            }  
        }  
        alert(`${fraze} \nŠioje frazėje yra ${eSkaicius} E raidės`);  
        return true;  
    }  
}  
  
console.log(kiekE());
```



# Masyvai (arrays)



**Ką daryti, jei norime sukurti studentų sąrašą?**

```
function studentList(){  
  let studentas1 = "Lukas Kuprys";  
  let studentas2 = "Robertas Buila";  
  let studentas3 = "Ilzė Cibulskaitė";  
}
```

**ir taip toliau... o jei tas sąrašas labai ilgas, ir gali keistis? Turi būti geresnis sprendimas!**

# Masyvai (arrays)



**Masyvas** - duomenų struktūra su automatiškai indeksuotomis pozicijomis, kurios prasideda nuo 0. Masyvuose galime talpinti bet kokius duomenis - tekstą, skaičius, nuotraukas ir t.t.

```
let studentai = ["Lukas Kuprys", "Pavel Novak", "Robertas Buila", "Ilzė Cibulskaitė", "Kristina Kurpičevič", "Artiom Fomenko"];
```

```
studentai[5]; //5 laukelyje esantis turinys - "Artiom Fomenko"
```

# Masyvai (arrays)



**Jei norėtume pakeisti informaciją masyve, kuriame nors laukelyje:**

**studentai[2] = “Arūnas Matusevičius”;**

**Kaip ir strings, galime rasti masyvo ilgį:**

**studentai.length;**

# pop() ir push() funkcijos



**pop( )** funkcija sunaikina paskutinį masyvo laukelį ir išveda jo turinį:

**studentai.pop( );** // rezultatas: “Artiomas Fomenko” ir masyvas sutrumpėja

**push( )** funkcija prideda laukelį ir jo turinį į masyvo galą

**studentai.push(“Lukas Čepauskas”);**

# forEach ciklas masyve



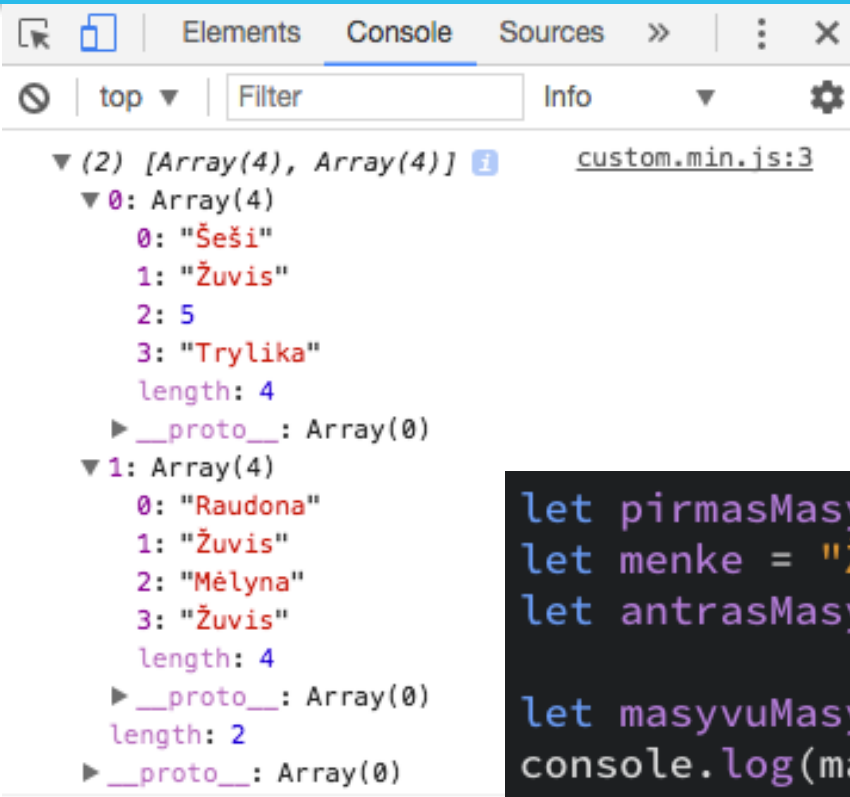
```
let skaiciai = [5, 7, 6];

skaiciai.push(8, 9, 10);
skaiciai.pop();

skaiciai[0] = 1;

skaiciai.forEach((element) => {
    console.log(element);
});
```

# Masyvai masyvuose



```
let masyvuMasyvas =  
[pirmasMasyvas, antrasMasyvas];
```

```
let pirmasMasyvas = ["Šeši", "Žuvis", 5, "Trylika"];  
let menke = "Žuvis";  
let antrasMasyvas = ["Raudona", menke, "Mėlyna", menke];  
  
let masyvuMasyvas = [pirmasMasyvas, antrasMasyvas];  
console.log(masyvuMasyvas);
```

# Masyvai masyvuose



["Šeši", "Žuvis", 5, "Trylika"]

0

1

2

3

0

["Raudona", "Žuvis", "Mėlyna", "Žuvis"]

0

1

2

3

1

```
console.log(masyvuMasyvas[1][3]); // "Žuvis"
```

```
console.log(masyvuMasyvas[0][2]); // "5"
```

# For ciklas masyve



```
let numList = [2,3,5,8,4,7,6,9,1,12];  
  
for(let i = 0; i < numList.length; i++) {  
    console.log(`Laukelyje ${i} yra skaičius ${numList[i]}`);  
}
```

Laukelyje 0 yra skaičius 2

Laukelyje 1 yra skaičius 3

Laukelyje 2 yra skaičius 5

Laukelyje 3 yra skaičius 8

Laukelyje 4 yra skaičius 4

Laukelyje 5 yra skaičius 7

Laukelyje 6 yra skaičius 6

Laukelyje 7 yra skaičius 9

Laukelyje 8 yra skaičius 1

Laukelyje 9 yra skaičius 12



# Tušti laukeliai masyve



Tuščią laukelį masyve sukuriame naudodami **undefined**

```
newNumbers[5] = undefined;
```

2	5	8	4	7		6	9	3	11
0	1	2	3	4	5	6	7	8	9

# Suraskime lyginių skaičių kiekį ir pašalinkime nelyginius



```
let numberList = [2,3,5,8,4,7,6,9,1,12];

let lyginiuKiekis = 0;
for (let i = 0; i < numberList.length; i++) {
  if (numberList[i] % 2 == 0) {
    lyginiuKiekis++;
  } else {
    numberList[i] = undefined;
  }
}
console.log(lyginiuKiekis);
```