

BALTIC TALENTS ACADEMY

JAVASCRIPT

KARTOJIMAS

- ▶ Ciklas **while**
- ▶ Ciklas **for**
- ▶ Elemento radimas pagal selektorių

CIKLAS - WHILE

```
while ( <loginė išraiška> ) {  
    <komanda 1>;  
    <komanda 2>;  
    ...  
}
```

1. Pradžioje tikriname ar <loginė išraiška> teisinga
2. Jei neteisinga tai išeiname iš ciklo
3. Jei teisinga tai vykdome paeiliui komandas:
<komanda 1>, <komanda 2> ir t.t.
4. Einame atgal į 1.

CIKLAS - FOR

```
for ( <pradinė> ; <sąlyga> ; <žingsnis> ) {  
    <komanda 1>;  
    <komanda 2>; ...  
}
```

1. Vykdomė išraišką <pradinė>
2. Tikriname ar <sąlyga> teisinga
3. Jei neteisinga tai išeiname iš ciklo
4. Jei teisinga tai vykdomė paeiliui komandas:
 <komanda 1>, <komanda 2> ir t.t.
5. Vykdomė išraišką <žingsnis>
6. Einame atgal į 2

HTML ELEMENTO PAIEŠKA PAGAL SELEKTORIŲ

- ▶ `document.querySelector(<selector>);`
- ▶ kur *selector* - tekstinė eilutė kur vienas ar daugiau CSS selektorių atskirtų kableliais
- ▶ Gražinamas rezultatas - pirmas rastas elementas arba `null` jei elementas nerastas
- ▶ `var e1 = document.querySelector(".myclass");`
- ▶ `var e2 = document.querySelector("h1+div");`

CIKLO VALDYMAS

Kartais reikia iš ciklo išeiti ankščiau laiko.

Tarkime mums reikia susumuoti visus masyvo **a** elementus pradedant nuo pirmo, bet tik iki elemento kurio reikšmė yra 0.

```
var i, s = 0;
for (i = 0; i < a.length; i++) {
  if (a[i] == 0) {
    break;
  }
  s += a[i];
}
```

CIKLO VALDYMAS

Kartais reikia cikle nevykdyti likusių ciklo sakinių ir grįžti į ciklo pradžią.

Tarkime mums reikia sudauginti visus masyvo **a** elementus, išskyrus tuos kurių reikšmė yra 0

```
var i, s = 1;
for (i = 0; i < a.length; i++) {
  if (a[i] == 0) {
    continue;
  }
  s *= a[i];
}
```

SWITCH KOMANDA

Labai panaši į **if** komandą tik ją patogiau užrašyti jei yra daug skirtingų sąlygų:

```
switch(<išraiška>) {  
  case <išraiška 1>:  
    <komandos 1>  
    break;  
  case <išraiška 2>:  
    <komandos 2>  
    break;  
  ....  
  default:  
    <komandos default>  
}
```


SWITCH KOMANDA

Switch komandos pvz.:

```
switch (kodas) {  
  case 'LT':  
    pavadinimas = 'Lietuva';  
    break;  
  case 'UK':  
    pavadinimas = 'Jungtinė Karalystė';  
    break;  
  case 'LV':  
    pavadinimas = 'Latvija';  
    break;  
  case 'EE':  
    pavadinimas = 'Estija';  
    break;  
}
```

FUNKCIJOS

Funkcija - tai komandų rinkinys sugrupuotas viename bloke tam, kad būtų paprasčiau naudoti vienas ir tas pačias komandas ten kur jų reikia.

Standartinės matematinės funkcijos:

Math.sqrt(x) - suskaičiuoja kvadratinę šaknį iš x

Math.sin(x) - suskaičiuoja $\sin(x)$

Math.ceil(x) - suapvalina į viršų, t.y. $2.1 \rightarrow 3$

Math.floor(x) - suapvalina žemyn, t.y. $2.9 \rightarrow 2$

Math.round(x) - suapvalina 'normaliai', t.y. $2.5 \rightarrow 3$

ir t.t.

FUNKCIJOS

Kaip parašyti savo funkciją ir ją panaudoti?

Vienas iš būdų aprašyti funkciją tai panaudoti specialų raktinį žodį **function** ir po jo nurodyti funkcijos vardą ir jos parametrus (jei tokių yra)

Jei norime, kad funkcija grąžintų kažkokią reikšmę, tą galima padaryti su **return** sakiniu

Funkcija panaudojama, t.y. sakoma 'kviečiama' nurodant jos vardą su argumentais

FUNKCIJOS

```
function greitis(atstumas, laikas) {  
  if (laikas > 0) {  
    var km = atstumas / 1000;  
    var val = laikas / 3600;  
    return km / val;  
  } else {  
    return 0;  
  }  
}
```

```
var g1 = greitis(95300, 3500);  
var g2 = greitis(60000, 3599);  
var g3 = greitis(1000);
```

FUNKCIJOS

Funkcija savo viduje gali kviesti kitas funkcijas arba pačią save, pvz.:

```
function faktorial(n) {  
    if (n == 1) return 1;  
    return n * faktorial(n - 1);  
}
```

```
var f1 = faktorial(1); // = ?
```

```
var f2 = faktorial(6); // = ?
```

ĮVYKIAI - ONCLICK

Kiekvienas HTML elementas gali sureguoti į “pelės” (kompiuteris) ar “piršto” (telefonas) paspaudimą. Tada tame elemente įvyksta taip vadinamas ‘**onclick**’ įvykis ir jei mes buvome parašę/užregistravę tam įvykiui javascript funkciją tai ji ir bus iškviesta.

ĮVYKIAI - ONCHANGE

HTML elemente, kuris gali turėti kažkokią reikšmę (<input>, <select>), po reikšmės pasikeitimo įvyksta taip vadinamas 'onchange' įvykis

Jei norime nuskaityti ir/arba įrašyti elemento reikšmę, tai reikia naudoti elemento atributą 'value'

JVYKIAI - ONKEYPRESS, ONKEYUP, ONKEYDOWN

Šie įvykiai leidžia reaguoti į klaviatūros paspaudimus. Geriausia juos naudoti jei norime iškart reaguoti į lauko pasikeitimus.

UŽDAVINYS

Padarykite įvedimo laukus automobilių įrašams įvesti, t.y. datai-laikui, numeriui, atstumui ir laikui ir mygtukas “Saugoti”. O apačioje tegul būna pradžioje tuščia lentelė.

Padarykite taip, kad įvedus duomenis ir paspaudus mygtuką, naujai suvesti duomenys atsirastų lentelėje, o įvedimo laukai išsivalytų ir t.t.