

BALTIC TALENTS ACADEMY

JAVASCRIPT

KARTOJIMAS

- ▶ Datos / laiko objektas
- ▶ Dinaminiai objektai
- ▶ Objekto prototipas
- ▶ Formos

DATOS / LAIKO OBJEKTAS

- ▶ `new Date()`
- ▶ `new Date(value)`
value - milisekundės nuo 1970 Sausio 1d. 00:00:00 UTC
- ▶ `new Date(dateString)`
dateString - "2016-01-01 12:45:00" - nenaudoti
- ▶ `new Date(year, month[, date[, hours[, minutes[, seconds[, milliseconds]]]])`

DATOS / LAIKO OBJEKTO METODAI (FUNKCIJOS)

- ▶ `.getFullYear()` - metai
- ▶ `.getMonth()` - mėnuo: 0..11 !!!
- ▶ `.getDate()` - mėnesio diena: 1..31
- ▶ `.getDay()` - savaitės diena: 0..6 (0 - sekmadienis)
- ▶ `.getHours()` - valandos: 0..23
- ▶ `.getMinutes()` - minutės: 0..59
- ▶ `.getSeconds()` - sekundės: 0..59
- ▶ `.getTime()` - milisekundės nuo 1970 Sausio 1d. 00:00:00 UTC
- ▶ analogiškai yra ir 'set' funkcijos - pvz. `.setMonth(4)`

OBJEKTO KONSTRUKTORIUS

```
function Automobilis(nr, s, t) {  
  this.numeris = nr;  
  this.atstumas = s;  
  this.laikas = t;  
  this.greitis = function () {  
    return this.atstumas / this.laikas * 3.6;  
  };  
}
```

```
var a1 = new Automobilis('LRS001', 5000, 150);  
var a2 = new Automobilis('ABC222', 5000, 125);
```

```
console.log( a1.greitis() );  
console.log( a2.greitis() );
```

OBJEKTO PROTOTIPAS

- ▶ Kiekvienas objektas turi taip vadinamą to objekto prototipą, t.y. to objekto tėvyninį “objektą” iš kurio jis paveldi metodus ir atributus.
- ▶ Norint sukurti naują arba perrašyti seną objekto metodą (funkciją) arba objekto atributo pradinę reikšmę reikia juos aprašyti objekto prototipe.

FORMOS

HTML:

```
<form name="forma" onsubmit="return onFormSubmit()">  
  <input name="atstumas" type="number">  
  <input name="laikas" type="number">  
  <button>Saugoti</button>  
</form>
```

JS:

```
function onFormSubmit() {  
  // ... document.forms['forma']['atstumas'].value ...  
  return false;  
}
```

Funkcija **onFormSubmit** grąžina **true** jei formą siųsti į serverį arba **false** - jei ne.

p.s. Tas pats galioje visiems įvykiams - jei grąžinama true - įvykis toliau apdorojamas, false - ne.

JSON

- ▶ JSON - kartais verčiamas kaip JavaScript Object Notation - duomenų pateikimo formatas
- ▶ JSON formatas pirmą kartą aprašytas 2000 pradžioje. 2013 standartizuotas kaip ECMA-404 ir RFC 7159
- ▶ Nors kilęs iš JavaScript bet yra nuo kalbos nepriklausomas
- ▶ pvz.:
 - ▶ { "numeris": "ABC001", "atstumas": 100, "laikas": 0.9 }
 - ▶ [
 { "vardas": "Jonas", "pavarde": "Onas" },
 { "vardas": "Ona", "pavarde": "Jona" }
]
- ▶ JSON funkcijos: `JSON.stringify(object)`, `JSON.parse(string)`
- ▶ <https://postit.lt/data/?address=K.+Baršausko+g.+59,+Kaunas>

jQuery

- ▶ jQuery - <https://jquery.com/>
- ▶ JavaScript biblioteka skirta palengvinti darbą su HTML DOM elementais, CSS, animacijomis, DOM įvykiais (onchange, oninput...), AJAX
- ▶ AJAX - Asinchroninis JavaScript ir XML (dabar daugiau JSON) programavimas

jQuery - PRADŽIA

- ▶ Kaip taisyklingai naudoti jQuery?

- ▶ HTML:

```
<head>
```

```
...
```

```
<script src="...jquery.min.js"></script>
```

- ▶ JS:

```
...
```

```
$(function() {
```

```
... ← čia bus naudojami jQuery metodai
```

```
});
```

jQuery

\$(selektorius).veiksmas1().veiksmas2()...

selektorius - DOM elemento ar kelių elementų paieškos sąlyga (kaip CSS selektorius)

veiksmas - funkcija kurią reikia atlikti su atrinktu pagal selektorių HTML elementu ar elementais (!!!)

pvz:

\$('#div.alert').show()

\$('.tooltip').hide()

jQuery - VEIKSMAI / ANIMACIJA

`.hide(speed, callback)`

`.show(speed, callback)`

`.fadeIn(speed, callback)`

`.fadeOut(speed, callback)`

`.fadeToggle(speed, callback)`

`.slideDown(speed, callback)`

`.slideUp(speed, callback)`

`.slideToggle(speed, callback)`

kur:

speed - 'slow', 'fast' arba skaičius milisekundėmis,

callback - funkcija kuri bus įvykdytą po atlikto veiksmo

jQuery - VEIKSMAI / ANIMACIJA

`.animate(params, speed, callback)` - animuojamas esamų elemento savybių pasikeitimas į parametruose nurodytas.

p.s. animuoti galima tik skaitinę reikšmę turinčius parametrus, pvz.: `width`, `height`, `margin`, `padding`, `top`, `left` bet ne `color`.

`.stop()` sustabdyti vykstančią animaciją

jQuery - ELEMENTŲ REIKŠMĖ

.text() - funkcija gražinanti arba pakeičianti tekstinę HTML elemento reikšmę, t.y. ignoruojant HTML tagus

.html() - funkcija gražinanti arba pakeičianti HTML reikšmę kartu su HTML tagais - analogas **.innerHTML**

.val() - funkcija gražinanti arba pakeičianti **<input>**, **<select>** elementų reikšmę - analogas **.value**

jQuery - ĮVYKIAI

`$(selektorius).ivykis(function() {...})`

ivykis: click, dblclick, mouseenter, mouseleave, keypress, keydown, keyup, submit, focus, blur, resize, ...

pvz.:

```
$('#button').click(function() {  
    $('#alert').show();  
});
```

KLAUSIMAS

HTML:

```
<p>Pirmas</p>
```

```
<p>Antras</p>
```

```
<p>Trečias</p>
```

JS:

```
$(‘p’).hide()
```

Kas atsitiks?

PRATIMAS

1. Sukurkite HTML puslapį su paragrafu su kažkokiu tekstu ir apačioje pridėkite `<a>` elementą su tekstu 'Slėpti'. Padarykite taip, kad užvedus pelę ant to `<a>` elemento paragrafas išnyktų, o nuvedus vėl atsirastų
2. Sukurkite HTML su trimis spalvotais 100px spindulio apskritimais. Padarykite taip, kad paspaudus ant kažkurio apskritimo kiti du išnyktų per 2 sekundes. Du kart paspaudus vėl atsirastų.

Nenaudokite HTML onclick, CSS animacijų, o naudokite tik jQuery

BOOTSTRAP MODALINIS LANGAS

- ▶ Tai toks atsidarantis langas, kai kituose vietose esantys komponentai (menu, įvedimo laukai it pan.) blokuojami, o aktyvus tampa tik modaliniame lange esantys
- ▶ Dokumentacija:
 - ▶ <http://getbootstrap.com/javascript/#modals>
 - ▶ https://www.w3schools.com/bootstrap/bootstrap_modal.asp

BOOTSTRAP MODALINIS LANGAS

```
<button data-toggle="modal" data-target="#m1">  
Atidaryti langą</button>
```

```
<div class="modal fade" id="m1">  
  <div class="modal-dialog">  
    <div class="modal-content">  
      ...  
    </div>  
  </div>  
</div>
```

BOOTSTRAP MODALINIS LANGAS

```
<button onclick="atidaryti()">Atidaryti langą</button>
```

```
<div class="modal fade" id="m1">  
  <div class="modal-dialog">  
    <div class="modal-content">  
      ...  
    </div>  
  </div>  
</div>
```

JS:

```
function atidaryti() {  
  $('#m1').modal('show');
```

UŽDAVINYS

Praplėsti uždavinį su automobiliais:

Pradžioje puslapio turi matytis tik mygtukas “Naujas” ir tik tuščia lentelė. Paspaudus “Naujas” turi atsirasti modalinis langas su trimis įvedimo laukais ir mygtukais “Saugoti” ir “Atšaukti”. Paspaudus “Saugoti” reikia sukurti naują įrašą ir parodyti jį lentelėje. Paspaudus “Atšaukti” - paslėpti įvedimo formą ir nieko nedaryti.

Pridėkite mygtukus prie kiekvieno lentelės stulpelio “Taisyti” ir “Naikinti”

Paspaudus naikinti turi atsirasti langas su klausimu “Ar tikrai norite ištrinti?” ir mygtukais “Taip” ir “Ne”. Paspaudus “Taip” reikia ištrinti ir perpiešti lentelę

Paspaudus “Taisyti” - turi vėl atsirasti langas su įvedimo laukais. Paspaudus “Saugoti” pataisytos reikšmės turi atsirasti lentelėje

Naudokite Bootstrap biblioteką. Atsirandantiems langams naudokite bootstrap modal componentą. Saugokite automobilio duomenis ne masyve, o objekte.