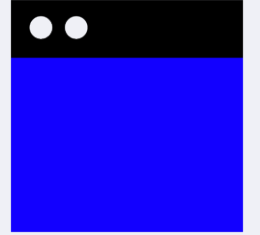




**Code
Academy**

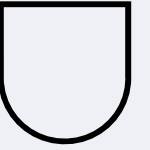


Gytis Juozenas

JavaScript Formos

2022

JavaScript programavimo kalba



Šiandien išmoksite

01

Kaip veikia Javascript forma



Javascript formos

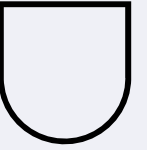
Prisiminkime kokius žinome formos įvesties būdus:

- `<input type="....">` - text, number, email, password, url, radio, checkbox, taip pat ir `<select>`, `<textarea>` ir t.t.

SVARBU!

-Kiekvienoje formoje reikalingas mygtukas, kurio tipas yra "submit", tai gali būti ir `<input type="submit">` ir `<button type="submit">`. Nes kitaip be jo, forma negali būti užpildoma galutinai ir pateikiama.

[Daugiau informacijos, kodėl naudojame mygtuką su tipu submit](#)



HTML formos elementas

- HTML forma yra naudojama vartotojo duomenims rinkti, tai pagrindinis tarpininkas tarp vartotojo ir web svetainės / aplikacijos.
- Formos atributai:
 - method – **GET** (numatytasis, siunčiama kaip URL kintamieji key/value poromis) arba / **POST** (HTTP request - siunčiama apsaugant duomenis);
 - action – kur peradresuoti vartotoją po formos užpildymo;

CONTACT

Name:

E-mail:

Message:

SEND YOUR MESSAGE

[Daugiau informacijos](#) apie GET ir POST



JavaScript formos reikšmių pasiėmimas

- Pasiimti formos elementą įprastu būdu naudojant jums patogų selektorių (querySelector, getElementById ir pan.)

Pvz.: const myForm = document.querySelector('form');

- Prie formos prisegti addEventListener, kur paleisime funkciją 'submit' evento metu.

Pvz.: myForm.addEventListener('submit', function() {});

Svarbu: įvykio registravimas priskiriamas ne mygtukui su type="submit", o pačiai FORMAI.

- Laukdami submit event'o turime nurodyti funkcionalumą, t.y. ką norime padaryti vartotojui paspaudus **submit** mygtuką

*Pvz.: myForm.addEventListener('submit', function() {
console.log('form submitted')*
})*

* forma pagal numatytuosius nustatymus persikraus, todėl console.log nespės atsispausdinti;

[Daugiau informacijos apie submit event;](#)



preventDefault();

-Kad galėtume pasiimti vartotojo įvestus duomenis iš formos, taip pat, kad galėtume juos patikrinti ir validuoti prieš išsiunčiant į serverį apdorojimui, mes galime/turime sustabdyti numatytą puslapio persikrovimą.

- Tai mes darome, kuomet formai priskirtam `addEventListener()`, prie pirmojo argumento (event'o), iškviečiame funkciją, kuriai bus perduotas **event** parametras, pvz:

```
myForm.addEventListener("submit", function (event) {  
    event.preventDefault();  
    console.log(event);  
});
```

- event parametras leidžia:

- sustabdyti puslapio persikrovimą;
- pasirinkti target, t.y. formą;
- susirinkti vartotojo įvestas reikšmes.

[Daugiau informacijos apie preventDefault\(\);](#)



Funkcijas paleidimo būdai

1. Tiesioginis funkcijas paleidimas:

```
myForm.addEventListener("submit", function (event) {  
  event.preventDefault();  
});
```

2. Funkcijas apsirašymas ir vėliau iškvietimas:

```
function formValues(event) {  
  event.preventDefault();  
}  
  
myForm.addEventListener("submit", formValues);
```



Event Target Elements

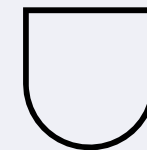
- Norėdami pasiekti formoje pateiktus duomenis, krepiames į **event.target**, ir nurodome savybę **elements**.

Pvz.: `const formInputs = event.target.elements;`

- Tokiu būdu gausime objektą, kuriame užfiksuoti visi panaudoti elementai**.

***Skirtingose naršyklėse rodomas vaizdas gali skirtis, pavyzdyje naudojama Mozilla Firefox Developer Edition*

```
▼ HTMLFormControlsCollection { 0: input, 1: input, 2: input, 3: input, length: 4, ... }  
  ▶ 0: <input type="text" name="firstName" placeholder="Your Name">  
  ▶ 1: <input type="number" name="age" placeholder="Your Age">  
  ▶ 2: <input type="email" name="email" placeholder="Your Email">  
  ▶ 3: <input type="submit" name="submit" value="Submit">  
  ▶ age: <input type="number" name="age" placeholder="Your Age">  
  ▶ email: <input type="email" name="email" placeholder="Your Email">  
  ▶ firstName: <input type="text" name="firstName" placeholder="Your Name">  
    length: 4  
  ▶ submit: <input type="submit" name="submit" value="Submit">  
  ▶ <prototype>: HTMLFormControlsCollectionPrototype { namedItem: namedItem(), ... }
```

`even.target.elements.inputName.value`

- Kiekvieną iš input'ų lengvai galime pasiekti per jam suteikto atributo name reikšmę.

Pvz. Jei formoje nurodyta: `<input type="text" name="firstName" placeholder="Your Name" />`

- Javascript'e išsisaugome įvestą reikšmę naudodami event, target, elements, atributo pavadinimas, ir reikšmė:

Pvz. `const firstName = event.target.elements.firstName.value;`

- Taip pat galite išsisaugoti naują kintamąjį, kuris laikys visas formos vertes, pvz.:

```
// stop the browser from refreshing the page
event.preventDefault();

// save new variable for shorter syntax
const formInputs = event.target.elements;

// 1. retrieve data from name input
const firstName = formInputs.firstName.value;
```



Įvestos reikšmės - įprasti input types, textarea

1. **input type text, number, email, password** ir visi kiti įprasti input'ai, kuriuose įrašoma tekstinė vertė, taip pat ir textarea, pasiimami nurodytu būdu:

```
const firstName = event.target.elements.firstName.value;
```

SVARBU! Nepamiršti, kad visos reikšmės iš formos yra gaunamos **string** duomenų tipu, todėl reikia būti atsargiems su `<input type="number">` ir kitomis reikšmėmis, jeigu jas norite gauti skaičiumi, **ypač jeigu vėliau atlikinėsite skaičiavimus**.

Konvertuojame į number duomenų tipą:

```
const age = Number(event.target.elements.age.value);
```

arba

```
const age = +event.target.elements.age.value;
```



Įvestos reikšmės - atspausdinimas

-Kuomet susirenkame įvestas reikšmes, galime atlikti tolimesnius veiksmus, pvz.: *atspausdinti, tikrinti ar atitinka tam tikras sąlygas (validuoti), išsiųsti į serverį.*

-HTML'e galime susikurti tuščią elementą, į kurį norėsime atspausdinti informaciją.

Pvz.: `<p class="output"></p>`

-Pasiimsime jį savo JS kode, išsisaugosime kintamajame.

Pvz.: `const output = document.querySelector('.output')`

-Naudosime element.**textContent** savybę teksto atspausdinimui.

Pvz.: `output.textContent = firstName;`

Pvz.: `output.textContent = `Sveiki, mano vardas yra ${firstName}.`;`



Užduotis nr. 1

Susikurkite forma su 2 input ir label, vienas vardui, kitas pavardei. Priskirkite on submit eventa ir jo metu pasiimkite irasytas reiksmes ir jas sudekite i paragrafa(sukurtas html'e), kiekviena karta kazka ivedus ir paspaudus submit buvusi reiksme turi persirasyti.