ლაბორატორიული 6

1. **innerHTML, outerHTML, ატრიბუტების**

**innerHTML**

<body>

<p>რაღაც ტექსტი პარაგრაფში</p>

<div>რაღაც ტექსტი div-ში</div>

<script>

alert( document.body.innerHTML ); //გამოვიტანით მიმდინარე კონტენტი

document.body.innerHTML = '<h1>ახალი კონტენტი</h1>!'; // შევცვალოთ კონტენტი

</script>

</body>

**outerHTML**

<div>js <b>rocks</b></div>

<script>

var div = document.body.children[0];

alert( div.outerHTML ); // <div>js <b>rocks</b></div>

</script>

**ატრიბუტები**

* elem.hasAttribute(name) – ამოწმებს ატრიბუტის არსებობას
* elem.getAttribute(name) – იღებს ატრიბუტის მნიშვნელობას
* elem.setAttribute(name, value) – ანიჭებს ატრიბუტს მნიშვნელობას
* elem.removeAttribute(name) – შლის ატრიბუტს

მოყვანილ სკრიპტში ნაჩვენებია ატრიბუტებთან მუშაობის მაგალითი:

<body>

<div id="elem" about="Elephant"></div>

<script>

alert( elem.getAttribute('About') ); // (1) 'Elephant', ატრიბუტუს აღება

elem.setAttribute('Test', 123); // (2) ატრიბუტუ Test მინიჭებულია

alert( document.body.innerHTML ); // (3) HTML-ში ჩანს ყველა ატრიბუტი

var attrs = elem.attributes; // (4) შეიძლება ატრიბუტების კოლექციის აღება

for (var i = 0; i < attrs.length; i++) {

alert( attrs[i].name + " = " + attrs[i].value );

}

</script>

</body>

**დავალებები:**

1. **L601** დაწერეთ სკრიპტი რომელიც სთხოვს მომხმარებელს შეიყვანოს სტუდენტის id და შემდგომ იპოვის ყველა ელემენტს ატრიბუტით data-student-id რომლის მნიშვნელობა მომხმარებლის მიერ შეყვანილიs ტოლია და შეუცვლის ფონს მწვანე ფერით.

1. **ელემენტების დამატება, წაშლა**

განვიხილოთ მაგალითი, როდესაც გვჭირდება ვებ გვერდზე შეტყობინების გამოჩენა/გამოტანა. მაგალითისთვის ჩავთვალოთ რომ გვდინა ესეთი შეტყობინების ჩვენება:

<style>

.alert {

padding: 15px;

border: 1px solid #d6e9c6;

border-radius: 4px;

color: #3c763d;

background-color: #dff0d8;

}

</style>

<div class="alert">

<strong>ყურადღება!</strong> მალე გამოცდები იწყება.

</div>



**ელემენტის შექმნა:**

var div = document.createElement('div');

**ტექსტური node-ს შექმნა განსაზღვრული ტექსტით:**

var textElem = document.createTextNode('ტექსტი');

ჩვენი შეტყობინების შექმნა:

var div = document.createElement('div');

div.className = "alert alert-success";

div.innerHTML = "<strong>ყურადღება!</strong> მალე გამოცდები იწყება.";

შეტყობინება შექმნილია, მაგრმა ჯერ არ ჩანს გვერდზე რადგან ჯერ არ არის ჩასმული. ელემენტის ჩასმისთვის გამოიყენება შემდეგი მეთოდები:

**parentElem.appendChild(elem)**

მაგალითად მოცემული სკრიპტი ჩასვავს ელემენტს სიის ბოლოში:

<ol id="list">

<li>0</li>

<li>1</li>

<li>2</li>

</ol>

<script>

var newLi = document.createElement('li');

newLi.innerHTML = '3!';

list.appendChild(newLi);

</script>

ელემენტის ჩასმა კონკრეტული ელემენტის წინ:

**parentElem.insertBefore(elem, nextSibling)**

მოცემული სკრიპტი ჩასვავს ელემენტს სიის 2 პუნქტის წინ:

<ol id="list">

<li>0</li>

<li>1</li>

<li>2</li>

</ol>

<script>

var newLi = document.createElement('li');

newLi.innerHTML = 'test';

list.insertBefore(newLi, list.children[1]);

</script>

თუ გვინდა სიის დასაწყისში ჩასმა შეიძლება გამოვიყენოთ შემდეგი კოდი:

list.insertBefore(newLi, list.firstChild);

თუ სია ცარიელია, და შესაბამისად firtsChild არის null, ან გადავცემთ მეორე პარამეტრად null, მაშინ იმუშავებს როგორც appendChild

**შეტყობინების ჩვენებისთვის საჭირო სრული კოდი:**

<style>

.alert {

padding: 15px;

border: 1px solid #d6e9c6;

border-radius: 4px;

color: #3c763d;

background-color: #dff0d8;

}

</style>

<body>

<h3>სტუდენტის პირადი გვერდი</h3>

</body>

<script>

var div = document.createElement('div');

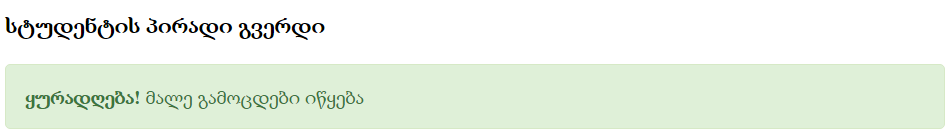
div.className = "alert alert-success";

div.innerHTML = "<strong>ყურადღება!</strong> მალე გამოცდები იწყება.";

document.body.appendChild(div);

</script>

**შედეგი:**



იგივე იმ გასხვავებით რომ შეტყობინება გვერდის თავში არის მოქცეული:

<style>

.alert {

padding: 15px;

border: 1px solid #d6e9c6;

border-radius: 4px;

color: #3c763d;

background-color: #dff0d8;

}

</style>

<body>

<h3>სტუდენტის პირადი გვერდი</h3>

</body>

<script>

var div = document.createElement('div');

div.className = "alert alert-success";

div.innerHTML = "<strong>ყურადღება!</strong> მალე გამოცდები იწყება.";

document.body.insertBefore(div, document.body.firstChild);

</script>

**ელემენტების კლონირება, cloneNode**

<style>

.alert {

padding: 15px;

border: 1px solid #d6e9c6;

border-radius: 4px;

color: #3c763d;

background-color: #dff0d8;

}

</style>

<body>

<h3>სტუდენტის პირადი გვერდი</h3>

</body>

<script>

var div = document.createElement('div');

div.className = "alert alert-success";

div.innerHTML = "<strong>ყურადღება!</strong> მალე გამოცდები იწყება.";

document.body.insertBefore(div, document.body.firstChild);

// ასლის შექმნა

var div2 = div.cloneNode(true);

// ასლში ცვლილებების შეტანა

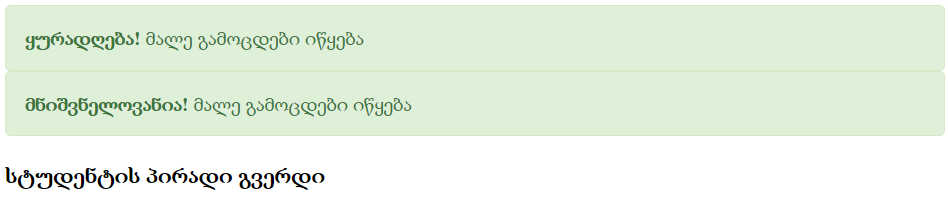
div2.querySelector('strong').innerHTML = 'მნიშვნელოვანია!';

// მიმდინარე შეტყობინების შემდეგ ჩასმა

div.parentNode.insertBefore(div2, div.nextSibling);

</script>

შედეგი:



**ელემენტის წაშლა, removeChild, replaceChild**

**parentElem.removeChild(elem)**

წაშლის ელემენტს elem;

**parentElem.replaceChild(newElem, elem)**

წაშლის ელემენტს elem და მის მაგივრად ჩასვავს ელემენტს newElem

ელემენტების ადგილის შესაცვლელად არ არის საჭირო მათი წაშლა, ეს ავტომატურად მოხდევა ახალ ადგილზე ჩასმისას.

<div>1</div>

<div>2</div>

<script>

var first = document.body.children[0];

var last = document.body.children[1];

// არ არის საჭირო removeChild(last)

document.body.insertBefore(last, first); // ადგილების შეცვლა

</script>

გავაკეთოთ ისე რომ შეტყობინება გაქრეს 5 წამის შემდეგ:

<style>

.alert {

padding: 15px;

border: 1px solid #d6e9c6;

border-radius: 4px;

color: #3c763d;

background-color: #dff0d8;

}

</style>

<body>

<h3>შეტყობინება გაქრება 5 წამში</h3>

</body>

<script>

var div = document.createElement('div');

div.className = "alert alert-success";

div.innerHTML = "<strong>ყურადღება!</strong> მალე გამოცდები იწყება.";

document.body.appendChild(div);

setTimeout(function() {

div.parentNode.removeChild(div);

}, 5000);

</script>

**დავალებები:**

1. **L602** დაწერეთ ფუნქცია insertAfter(elem, refElem) რომელიც ჩასვავს ერთ ელემენტს მეორეს შემდეგ
2. **L603** დაწერეთ ფუნქცია removeChildren(elem) რომელიც წაშლის ელემენტის ყველა შვილს
3. **L604** დაწერეთ სკრიპტი რომელიც სთხოვს მომხმარებელს სახელების შეყვანას და დინამიურად ქმნის მათ სიას გვერდზე, ანუ ყოველი შეყვანილი სახელი ემატება სიაში. პროცესი გრძელდება სანამ მომხმარებელი არ დააჭერს esc ღილაკს ან დატოვებს სახელის შესავსებ ველს ცარიელს
4. **L605** დაწერეთ ფუნქცია რომელიც პარამეტრად მიიღებს სტუდენტების ობიექტები სიას ფორმატით { FirstName: ‘test’, LastName: ‘test’, PersonalNumber: ‘1234'} და გამოიტანს მათ გვერდზ ცხრილის სახით (შექმნის html table-ს).