1. **მინიმალური პროგრამა js-ზე (hello world)**

<!DOCTYPE HTML>

<html>

<head>

<!—meta ტეგი, კოდირების მისათითებლად -->

<meta charset="utf-8">

</head>

<body>

<p>გვერდის დასაწყისი...</p>

<script>

alert( 'js ძალიან მარტივია!' );

</script>

<p>...გვერდის დასასრული</p>

</body>

</html>

**დავალება L101:** დაწერეთ js-ზე სკრიპტი რომელიც გამოიტანს შეტყობინებას “უფრო რთულ რაღაცეებზე გადასვლის დროა!”

1. **გარე სკრიპტები**

გარე სკრიპტის ჩართვის მაგალითი: <script src="/path/to/script.js"></script>

/path/to/script.js - ფაილის მისამართი საიტის root -დან

შეიძლება სრული მისამართის მითითებაც, მაგალითად:

<script src="https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.3.0/lodash.js">  
</script>

რამდენიმე სკრიპტის ჩართვისთვის რამდენიმე script ტეგი უნდა გამოვიყენოთ. მაგალითად:

<script src="/js/script1.js"></script>

<script src="/js/script2.js"></script>

თუ მითითებულია src ატრიბუტი მაშინ ტეგის შიგნით მოთავსებულ კოდს ბრაუზერი იგნორირებას უკეთებს. მაგალითად:

<script src="file.js">

alert(1); //ეს კოდი არ შესრულდება

</script>

**დავალება L102:** გამოიტანეთ შეტყობინება გარე სკრიპტის გამოყენებით. გამოიტანეთ ორი შეტყობინება სხვა და სხვა გარე სკრიპტის ფაილის გამოყენებით. დააკვირდით რაზე არის დამოკიდებული შეტყობინების გამოსვლის თანმიმდევრობა.

1. **სკრიტების შესრულების თანმიმდევრობა, ასინქრონული სკრიპტები: defer/async**

**მაგალითი 1:**

<!DOCTYPE HTML>

<html>

<head>

<meta charset="utf-8">

</head>

<body>

<p>დათვლის დაწყება:</p>

<script>

alert( 'პირველი სტუდენტი!' );

alert( 'მეორე სტუდენტი!' );

alert( 'მესამე სტუდენტი!' );

</script>

<p>ყველა სტუდენტი დათვლილია!</p><!-- ეს ტექსტი არ გამოჩნდება სანამ ყველა alert არ შესრულდება -->

</body>

</html>

**მაგალითი 2:**

<html>

<head>

<script src="big.js"></script>

</head>

<body>

ეს ტექსტი არ გამოჩნდება სანამ ბრაუზერი არ ჩატვირთავს და არ შეასრულებს big.js.

</body>

</html>

ამ შეზღუდვის გადასაწყვეტად არის ატრიბუტები defer, async

<script src="async.js" async></script>

<script src="defer.js" defer></script>

განსხვავება: defer გარანტიას იძლევა რომ სკრიპტები შესრულდება იმ თანმიმდევრობით რა თანმიმდევრობითაც არიან ჩართული.

**მაგალითი:**

თანმიმდევრობა შეიძლება ნებისმიერი იყოს:

<script src="1.js" async></script>

<script src="2.js" async></script>

სრულდება ჯერ 1 მერე 2:

<script src="1.js" defer></script>

<script src="2.js" defer></script>

**რეალური გამოყენების მაგალითი:**

<p>მნიშვნელოვანი ინფომრაცია არ გამოჩნდება სანამ რეკლამის სკრიპტი არ შესრულდება </p>

<script src="https://js.cx/hello/ads.js?speed=0"></script>

<p>...მნიშვნელოვანი ინფომრაცია!</p>

<p>მნიშვნელოვანი ინფომრაცია ჩნდება თავიდანვე და არ ელოდება რეკლამის სკრიპტის შესრულებას </p>

<script async src="https://js.cx/hello/ads.js?speed=0"></script>

<p>...მნიშვნელოვანი ინფომრაცია!</p>

**დავალება:**

ქვემოთ მოყვანილ მაგალითში ჩართულია 2 სკრიპტი small.js და big.js.

თუ ჩავთვლით რომ small.js ბევრად სწრაფად იტვირთება ვიდრე big.js – რომელი სკრიპტი შესრულდება უფრო ადრე:

<script src="big.js"></script>

<script src="small.js"></script>

big.js შესრულდება უფრო ადრე

ასეთ შემთხვევაში?

<script async src="big.js"></script>

<script async src="small.js"></script>

small.js

და ასე?

<script defer src="big.js"></script>

<script defer src="small.js"></script>

big.js

1. **ცვლადები**

ცვლადის გამოცხადებისთვის გამოიყენება საკვანძო სიტყვა **var:**

var message;

გამოცხადების შემდეგ ცვლადში შეიძლება მონაცემების ჩაწერა:

var message;

message = 'გამარჯობა'; //სტრიქონის შენახვა ცვლადში

შემდგომში მონაცმებზე წვდომა ცვლადის სახელით გამოყენებით არის შესაძლებელი:

var message;

message = 'გამარჯობა!';

alert( message ); //გამოიტანს ცვლადის მნიშვნელობას

შემოკლების მიზნით შეიძლება გამოცხადების და მონაცმების ჩაწერის გაერთიანება:

var message = 'გამარჯობა!';

შეიძლება რამდენიმე ცვლადის ერთად გამოცხადება:

var user = 'John', age = 25, message = 'Hello';

შეიძლება ცვლადში არსებული მნიშვნელობის რამდენჯერმე შეცვლა:

var message;

message = 'გამარჯობა!';

message = 'ნახვამდის!'; //მნიშვნელობის ცვლილება

alert( message );

შეიძლება მნიშვნლობსი ერთი ცვლადიდან მეორეში კოპირება:

var hello = 'გამარჯობა';

var message;

//მნიშვნელობსი კოპირება

message = hello;

alert( hello ); //გამარჯობა

alert( message ); //გამარჯობა

**დავალება L103:** გამოაცხადეთ ორი ცვლადი admin და name. name ცვლადში ჩაწერეთ „დავითი“. Name ცვლადის მნიშელობა ჩაწერეთ admin ცვლადში. გამოიტანეთ (alert) admin ცვლადის მნიშვნელობა.