

# FRONT-END

## 6 თვიანი კურსი



კურსის გავლა შეუძლია ნებისმიერ მსურველს ვისაც სურს აითვისოს ახალი პროფესია, საწყისებიდან პროფესიონალის დონემდე , კურსის მიმდინარეობისას შეისწავლით ისეთ ენებს როგორც არის : HTML , CSS , JAVASCRIPT , ჯავასკრიფტის DOM ბიბლიოთეკას JQUERY CSS პრე პროცესორს SASS , FRAMEWORK BOOTSTRAP

(კურსი არ მოიცავს **framework angular\_s** და 3 თვიან სტაჟირების პროგრამას)

- 1) რისგან შედგება და როგორ მუშაობს ვებ გვერდი
- 2) როგორ გამოიყურება სერვერზე განლაგებული ვებ საიტის ფაილური სისტემა
- 3) ფაილების გაფართოება და მათი დანიშნულება ვებ გვერდის ფუნქციონირებაში
- 4) რა არის HTTP მეთოდები , მისი მაგალითები
- 5) რა არის HTML , CSS, JAVASCRIPT მათი დანიშნულება ვებ დეველოპმენტში
- 6) რისგან შედგება HTML , ედიტორის შერჩევა
- 7) ტერმინების განმარტება
- 8) მეტა ტეგების დანიშნულება
- 9) ტექსტური ტეგები
- 10) HTML ტექსტური ფორმატი .
- 11) ციტირება

12) ჰიპერლინკები და მისი აგრიბუტები

13) სიები

- \_\_ცხრილები
- \_\_მულტიმედია
- \_\_IFRAME
- \_\_სემენტიკური & ბლოკური ტიპის ტეგები
- \_\_მათი მნიშვნელობა თანამედროვე ვებსა და SEO ოპტიმიზაციაში
- \_\_HTML - ფორმები

- \_\_CSS მიმოხილვა , სინტაქსი , თვისებები და მნიშვნელობები , HTML დოკუმენტში შემოგანის გზები , გამოყენების რეკომენდირებული მეთოდები
- \_\_სელექტორები , მარტივი სელექტორი ტეგის აიდი და კლასის გამოყენებით
- \_\_სელექტორების უპირატესობა
- \_\_CSS კომენტარი , ფერები
- \_\_ფონები
- \_\_ფონის თვისებები
- \_\_საზღვრები
- \_\_საზღვრის თვისებები
- \_\_□არე საზღვრები
- \_\_TEXT - ტექსტური თვისებები
- \_\_ფონტების შემოგანა

- \_\_ CSS - დაშორებები
- \_\_ Box-sizing , content-box , border-box
- \_\_ ელემენტების WIDTH, HEIGHT
- \_\_ განხილვა და რეკომენდაციები, რა შემთხვევებში არ შეიძლება ფიქსირებული სიმაღლის & სიგანის გამოყენება
- \_\_ იუნიტები PX, %
- \_\_ BOX-MODEL განხილვა
- \_\_ □უქჩრდილები (BOX-SHADOW)

- \_\_ აიქონების შემოგანა ვებ გვერდზე
- \_\_ ლინკები - A:VISITED , A:HOVER, A:ACTIVE
- \_\_ HOVER PSEUDO-CLASS
- \_\_ სიის თვისებები
- \_\_ DISPLAY
- \_\_ DISPLAY FLEX თვისებები
- \_\_ FLEX მიმართულებები
- \_\_ FLEX განოყენებით ვებ საიტის მარტივი მაკეტის აწყობა , სხვადასხვა განლაგებით ,

- \_\_ MIN-WIDTH, MIN-HEIGHT , MAX-HEIGHT, MAX-WIDTH , თვისებების განხილვა მათი როლი რესფონსივ დიზაინში
- \_\_ პოზიციები , STATIC , RELATIVE, ABSOLUTE , FIXED, STICKY
- \_\_ Z-INDEX
- \_\_ პოზიციონირება , LEFT , RIGHT , TOP , BOTTOM , თვისებების გამოყენებით
- \_\_ ABSOLUTE პოზიციის ურთიერთქმედება RELATIVE პოზიციასთან
- \_\_ პრაქტიკული საშუალო პოზიციების გამოყენება რეალური პროექტებიდან

- \_\_OVERFLOW - თვისებები , HIDDEN , SCROLL , AUTO
- \_\_OVERFLOW თვისების პრაქტიკული მაგალითები
- \_\_სქროლბარის გასვლა
- \_\_ელემენტების გაცენტრება , ვერტიკალური/ჰორიზონტალური გაცენტრება
- \_\_პრაქტიკული მაგალითები , ბლოკური ტიპის ელემენტის გაცენტრება ჰორიზონტალურად ,  
RELATIVE/ABSOLUTE ელემენტის გაცენტრება ვერტიკალურად/ჰორიზონტალურად,  
TRANSFORM:TRANSLATE თვისების გამოყენებით , გაცენტრებები FLEX თვისების  
შეგნით
- \_\_კომბინაციები

- \_\_PSEUDO – ELEMENT
- \_\_::BEFORE, ::AFTER , CONTENT,
- \_\_სელექტორები აგრიბუგის გამოყენებით,
- \_\_PSEUDO – CLASSES
- \_\_OPACITY
- \_\_OBJECT-FIT , თვისებები COVER , CONTAIN, OBJECT - POSITION

- \_\_ერთეულების გამოყენება პრაქტიკული მაგალითები
- \_\_ერთეულები
- \_\_იუნთების სწორი გამოყენება RESPONSIVE დიზაინში
- \_\_CALC ფუნქცია
- \_\_ATTR ფუნქცია
- \_\_VAR ფუნქცია
- \_\_ფუნქციების პრაქტიკული მაგალითები

- \_\_ 2/3D ტრანსფორმაციები
- \_\_ გადასვლები (TRANSITION)
- \_\_ ანიმაციები KEYFRAME გამოყენებით
- \_\_ მელია ქუერები
- \_\_ FILTER
- \_\_ BACKDROP-FILTER
- \_\_ მელია ქუერები გამოყენებით , ადაპტირებული ვებ საიტის მაკეტის შექმნა

- \_\_ რა არის SCSS ზოგადი მიმოხილვა , გამოყენება თანამედროვე ვებ დეველოპმენტში
- \_\_ NODE.JS ინსტალაცია
- \_\_ NPM განხილვა
- \_\_ NPM გამოყენებით SASS ინსტალაცია
- \_\_ SCSS - სინტაქსი
- \_\_ PARENT-SELECTOR, @EXTEND , VIRIABLES , ფსეულო კლასების/ელემენტის გამოყენება , @MIXIN, @MIXIN პარამეტრები , @INCLUDE, @USE, @FUNCTION, @FOR , @RETURN , @EACH
- \_\_ OPERATOR , @IF , @ELSE

## მცირე ფინალური პროექტი (მარჯაფი)

- \_\_ BOOTSTRAP - მიმოხილვა , გამოყენება თანამედროვე ვებ პროგრამირებაში უემოგანის გზების განხილვა
- \_\_ ვერსიების განხილვა

- `__`კომპონენტების ზოგადი მიმოხილვა
- `__`რამდენიმე ძირითადი კომპონენტების შემოგანა თვალსაჩინობისთვის
- `__`სელექტორების კონფლიქტი და უპირატესობა
- `__`BOOTSTRAP კლასებთან მუშაობის სტანდარტი
- `__`დაშორებების კლასები
- `__`ტექსტებთან სამუშაო კლასების განხილვა

- `__`UTILITIES - ძირითადი კლასები განხილვა ,
- `__`BORDERS, COLORS, DISPLAY
- `__`FLEX - კლასის და მისი თვისებების შემადგენელი კლასების განხილვა BOOTSTRAP-ში flex-direction, justify-content , align-items
- `__`POSITION კლასების განხილვა
- `__`SHADOW კლასები
- `__`HELPERS - ძირითადი კლასების განხილვა
- `__`POSITION - კლასები
- `__`FORM - კლასების განხილვა

- `__`LAYOUT - ყველა ძირითადი კლასის განხილვა , CONTAINER , ROW , CONTAINER-FLUID
- `__`BREAKPOINTS – SM , MD, LG, XL , XXL ,
- `__`BREAKPOINTS გამოყენება პრაქტიკაში , მათი დანიშნულება RESPONSIVE ვებ დიზაინში
- `__`BREAKPOINTS - გამოყენება , GRID სისტემის შიგნით
- `__`ანლაგები GRID სისტემის შიგნით , justify-content, align-items
- `__`BREAKPOINTS გამოყენება utilities კლასების შიგნით
- `__`რაქტიკული სამუშაო BOOTSTRAP გამოყენებით RESPONSIVE საიგის მარჯაფის შექმნა

- ვებ გვერდის შექმნა .figma , .xd ფაილიდან
- figma/xd - ფაილთან მუშაობა
- დაშორებების განსაზღვრა , ფონის ფერების, ფონტის ზომების ფერის font-family განსაზღვრა, opacity თვისების განსაზღვრა , box-shadow თვისების განსაზღვრა
- ფაილების ჩამოგვირთვა svg/png ფორმატში
- დიზაინში გათვალისწინებული დინამიური კონტენტის დაშორებების განსაზღვრის პრინციპი

## ინდივიდუალური მარქაფის ფინალური პროექტი

- მარქაფის ფინალური  პროექტების წარდგენა
- პროექტის  თვის გარჩევა/გამარტივება
- აგების არსებობის შემთხვევაში გამომწვევი მიზების განხილვა
- Responsive ვერსიის განხილვა
- რიგიკული აცდენების შემთხვევაში სამუშაო დიზაინთან (psd, xd) შესაბამისი მითითებების მიცემა
- Javascript - ზოგადი მიმოხილვა
- Javascript - გამოყენება ვებ საიტებში , მაგალითები რეალური პროექტებიდან
- რაქტიკული მაგალითების ჩვენება , (ივენტების გამოყენებით)

- ინტაქსის განხილვა ,  ველალების შექმნა , ცვლადების როლი javascript ენაში, ქეისენსიტივის მნიშვნელობა
- ველადის შექმნა
- ა სხვაობაა ქივორდს შორის

- ა შემთხვევაში რომელი ქვივორდის გამოყენებაა რეკომენდებული
- ომენტარი
- ვლადების გამოყენებით მარტივი არითმეტიკული მოქმედებების შესრულება
- ამოგანის ფუნქციები
- ვლადის გამოგანა html დოკუმენტში
- ონაცემთა ტიპები
- მონაცემის ტიპის გადამოწმება
- ოპერატორები

- მოქმედებები სტრინგებზე
- ტრინგის სტრიქონების გადაბმა “+” ოპერატორის გამოყენებით
- ტრინგის აწყობა html ტეგების გამოყენებით და მისი გამოგანა html დოკუმენტში
- ემპლეიტ სინტაქსის გამოყენება ` \${ // } `
- ეთოდები სტრინგებზე , length, charAt(), startwith(), endwith(), includes(), indexOf(), match(), search(), replace(), slice(), split(), toString(), touppercase() , tolowercase()
- Regexp – search(), replace() , მოლიფიკატორები ( l , g , m ) , ნიმუშები , metacharacter

- მოქმედებები რიცხვებზე
- აშენებული ფუნქციები
- ა შემთხვევაში ბრუნება NaN ტიპი
- უნქციები , ფუნქციის მნიშვნელობა ჯავასკრიფტში,
- უნქციის შექმნა , ფუნქციის პარამეტრები , რესტ პარამეტრი , RETURN ქვივორდი ,
- ოკალური და გლობალური ცვლადების განსაზღვრა ,
- ARROW ფუნქციები , სხვაობა გრადიციულ და ARROW ფუნქციას შორის (THIS ქვივორდის გამოყენება)



- ასივები , მასივის დანიშნულება , ინდექსის განმარტება
- ასივის შექმნა
- ასივის ელემენტის გამოძახება ინდექსის მეშვეობით
- For ციკლის გამოყენებით მასივი ყველა წევრის დაბეჭვლა
- ოქმელებები მასივებზე
- ამეორების მეთოდები
- ხვაობა მეთოდებს შორის
- ასივების სორტირება
- იცხვების სორტირება მასივში

- ბიექტები , ობიექტის გამოყენების მაგალითები რეალურ ვებ გვერდებზე
- ინტაქსი , თვისებები და მნიშვნელობები
- ბიექტიდან თვისების / მეთოდი გამოძახება
- ვისების შეცვლა , ახალი თვისების ჩამატება
- This ქივორდის გამოყენება ობიექტის მეთოდებში
- For in loop გამოყენება
- Nested ობიექტის შექმნა
- Getter & setter
- ბიექტების მასივის შექმნა მაგ
- ოქმელებები ობიექტების მასივზე

- არილები
- არილის ობიექტის შექმნა
- Get & set მეთოდები მათი მნიშვნელობები
- PARSE() მეთოდი
- რაქტიკული სამუშაო

- Match - ობიექტი
- Match  ეთოდები
- ენდომული ოპერაციები
- რაქტიკული მაგალითები
- boolean ლოგიკა
- If/else , else if გამოყენება , პრაქტიკული მაგალითები
- Switch / case - გამოყენება პრაქტიკული მაგალითები
- ანხილვა და შედარება , რა უპირატესობები აქვს if/else switch შედარებით და პირიქით

- იკლები
- For
- For in
- For of
- While / do while
- Events - ივენტების მიმოხილვა , გამოყენება front-end დეველოპმენტში
- ამოყენების გზები
- ენტის გამოყენება html ელემენტში
- ენტის გამოყენება addEventListener ფუნქციით
- Event ობიექტი (ზოგადი განხილვა)

- DOM - დოკუმენტის ობიექტის მოდელი
- DOM - განხილვა , მისი გამოყენება FRONT-EDN დეველოპმენტში
- ელემენტები
- ომელი სელექტორის გამოყენება არის რეკომენდებული
- ენტის შექმნა LOOP შიგნით
- ამოგანის მეთოდები
- რაქტიკული მაგალითები

HTML CLASS მეთოდები

CLASSLIST.ADD()

CLASSLIST.REMOVE()

CLASSLIST.REPLACE()

CLASSLIST.CONTAINS()

CLASSLIST.TOGGLE()

ლასების და ივენთების გამოყენებით მარტივი ფუნქციური კომპონენტის შექმნა (MODAL , SIDEBAR ა.შ)

HTML ATTRIBUTE მეთოდები

CREATEATTRIBUTE()

SETATTRIBUTE()

REMOVEATTRIBUTE()

GETATTRIBUTE()

HASATTRIBUTES()

TOGGLEATTRIBUTE()

ტრიბუტების გამოყენების შორთჰენდი (ELEMENT.ATTRIBUTE , ELEMENT.ATTRIBUTE = ‘)

ტრიბუტებთან სამუშაო პრაქტიკული მაგალითები

გილები & HTML ელემენტის შექმნა

CSS სტილების მინიჭება ჯავასკრიფტიდან `ELEMENT.STYLE.PROPERTY = VALUE`

თვითნებური გამოყენება CSS თვისებები რომელიც ორ თვისებას აერთიანებს მაგ: `css (border-color:red ) js (ELEMENT.STYLE.borderColor = 'red')`

HTML ელემენტის შექმნა

`DOCUMENT.CREATEELEMENT(ELEMENT)`

`DOCUMENT.CREATETEXTNODE('TEXT')`

`APPENDCHILD()`

`PREPEND()`

`REMOVE()`

რაქტიკული სამუშაო TODO ლისტის შექმნა

HTML ელემენტების ურთიერთობები

ParentElement

children

NextElementSibling

PreviousElementSibling

ChildElementCount

LastElementChild

FirstElementChild

რაქტიკული გამოყენება ,

ოცემული თვისებების გამოყენებით მარტივი კარუსელის აწყობა

ლემენტის ზომები , სქროლი,

ClientHeight , clientTop , clientLeft, clientWidth

OffsetHeight , offsetWidth , offsetLeft, offsetTop

ScrollHeight, scrollWidth

ScrollLeft , scrollTop

ლემენტის პოზიციების დადგენა , `element.getBoundingClientRect()`

Aria თვისებები

EVENT - ობიექტი

EVENT.TARGET

EVENT.TYPE

EVENT.KET

preventDefault()

აუსის ივენთების განხილვა

აჩის ივენთები , თაჩ დივაისისთვის

CLICK , CHANGE, INPUT , LOAD

□რაქტიკული გამოყენება

□რაუზერის ობიექტი & JQUERY შესავალი

window.history.forward(), window.history.back() , window.history.go() , window.open()

Confirm(),

Screen ობიექტი

JQUERY მიმოხილვა მისი შექმნის მიზანი , გამოყენება თანამედროვე პროგრამირებაში

JQUERY - □ინტაქსი, სელექტორი , ივენთის შექმნა

Jquery – ეფექტები

Hide(), show(), toggle()

Fadein(), fadeout(), fadetoggle()

SlideDown(), slideUp(), slideToggle()

□ანიმაციის მეთოდების პარამეტრები (“slow”, “hide”, milisecend), ანიმაციის callback ფუნქცია

Animate(),

CSS რა თვისებებზე არ აქვს მხარდაჭერა animate მეთოდს

Stop() მეთოდი

□რავალი მეთოდი გაშვება ერთ სელექტორზე მაგ: \$(selector).hide(3000).show(5000)

Jquery – მინიჭების და წაკითხვის მეთოდები

Get/Set მეთოდები , Html(), text(), val() , attr(), data()

Html  ლასებთან მუშაობა, addClass(), removeClass(), toggleClass()

Html ატრიბუტებთან მუშაობა , attr(), removeAttr(),

Prop მეთოდის გამოყენება

CSS - მეთოდის გამოყენება და მისი პარამეტრები

მოცემული მეთოდების გამოყენებით პრაქტიკული სამუშაო

Jquery - ემემენტების დამატების მეთოდები, traversing

Append(),prepend(), before(), after()

აშლის მეთოდები , remove(), empty()

Traversing - მეთოდები

Children(),

parent(), parents(),parentsUntil(),

siblings(),

prev(), prevALL(), prevUntil()

next(), nextALL(),nextUntil()

პრაქტიკული სამუშაო მოცემული მეთოდების გამოყენებით

## Filtering & misc

ეთოდები : eq(), filter(), find() , first() , last(), not(), has(), is()

\$.each()

Ajax - მიმოხილვა რა არის აჯაქსი, მისი გამოყენება პროგრამირებაში

Ajax - პარამეტრების თვისებები

url

Type

Data

DataType

აბრუნების მეთოდები

Success()

Error()

Done()

Fail()

Load() მეთოდი

Jquery - შეჯამება & plugin განხილვა

ექცია მოიცავს ისეთი ფლაგინების განხილვას რომელიც პრაქტიკულია და ხშირად გამოიყენება ვებ დეველოპმენტში

Swiper.js



Fslightbox.js

bootstrap-datepicker

NuiSlider.js

Magnifier.js

Jquery plugin ბაზის განხილვა

Jquery ui განხილვა

Json - ფორმატი , მისი დანიშნულება , სინტაქსი

Json - მონაცემთა ტიპები

Json - გამოყენებით , მასივების, ობიექტების , ობიექტების მასივის აწყობა

Json - სამუშაო მეთოდები

JSON.PARSE(), JSON.STRINGIFY()

JSON გამოყენებით ლოკალური ფაილის შექმნა

რაქტიკული სამუშაო , AJAX გამოყენებით JSON ფაილიდან მონაცემის წამოღება გაპარსვა და HTML გამოგანა

სინქრონული ჯავასკრიფტი

Async ქივორდი

Await ქივორდი

.then() მეთოდი

Promise ობიექტი

Fetch() api

ანხვაება ajax და fetch შორის ,

Fetch გამოყენება პრაქტიკაში , json fake api გამოყენებით (<https://jsonplaceholder.typicode.com/>)

LocalStorage, sessionStorage api

Javascript CLASS & Module

კლასების ზოგადი დანიშნულება პროგრამირებაში , კლასების გამოყენება ჯავასკრიფტში

კლასის შექმნა , წევრების განხილვა

Constructor()

Static მეთოდები & თვისებები

This ქივორდი გამოყენება

ემკვიდრებითობა extends

Super ქივორდი

ობიექტები , მისი გამოყენების საჭიროება

EXPORT , IMPORT

ახელის გადარქმევა as ქივორდი გამოყენებით

ობიექტების ობიექტის შექმნა \* (ყველა export ობიექტის შემოტანა ფაილიდან)

ობიექტების გამოყენება კლასების შიგნით

Typescript - მიმოხილვა , გამოყენების საჭიროება

იპების განხილვა , ტიპიზაცია

უნიონ ტიპები , მასივების ტიპიზაცია , Tuples/Object destructured , interfaces

რასავალდებულო პარამეტრები & თვისებები

ჯენერიკები

ეკორატორები

Tsconfig.json განხილვა

ლასები , კლასის private , public , protected წევრები

იმპლემენტაცია

