

ვებ ტექნოლოგიების საფუძვლები

- სასწავლო კურსის მიზანია სტუდენტებს მისცეს ვებ-საიტის დაპროექტებისა და დამზადებისათვის საჭირო თეორიული ცოდნა და პრაქტიკული გამოცდილება.
- საფუძვლიანად შეასწავლოს საიტის შესაქმნელად აუცილებელი ვებ ტექნოლოგიები: (X)HTML და CSS.
- ასევე გააცნოს დინამიური საიტის დანიშნულება და მისი შექმნის ტექნოლოგიები.

ვებსაიტის შექმნის ენები

HTML -Hyper Text Markup Language - ჰიპერტექსტის მარკირების ენა, რომლის საშუალებითაც ხდება ვებ გვერდის სტრუქტურის შექმნა.

CSS-Cascading Style Sheets - კასკადური სტილების ცხრილების საშუალებით ხდება ვებ გვერდის გაფორმება ანუ ბრაუზერში წარმოსადგენი სახის მიცემა.

JavaScript -კლიენტის მხარეს პროგრამირების ენა.

PHP -სერულყოფილი პროგრამირების ენაა, რომელსაც სერვერის მხარეს პროგრამირების ენას უწოდენა.

რა თქმა უნდა, არსებობს სხვა პროგრამირების ენები. მაგ. XML, ASP, ASP.NET და ა.შ.

HTML (HyperText Markup Language) ჰიპერტექსტური აღწერის/მარკირების ენა შეიქმნა ტიმ ბერნერსლის მიერ, რომელიც განკუთვნილია ვებ-გვერდების და საიტების შესაქმნელად და ინფორმაციის გასავრცელებლად ინტერნეტის საშუალებით.

ვერსია	წელი
HTML	1991
HTML+	1993
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML 1.0	2000
HTML5	2012
XHTML5	2013

XHTML (eXtensible Hyper Text Markup Language) - ჰიპერტექსტების მარკირების გაფართოებად ენას გააჩნია XML (eXtensible Markup Language) ენის მხარდაჭერა.

რეკომენდაციები

- როდესაც წერთ HTML კოდს, დარწმუნებული ვერასდროს იქნებით, თუ როგორ გამოისახება ტექსტი სხვადასხვა ბრაუზერებში. ამიტომ აუცილებელია ვებ გვერდის დათვალიერება ყველა ძირითად ბრაუზერებში.
- მომხმარებელთა ნაწილს აქვს დიდი მონიტორი, ზოგიერთს პატარა. ტექსტი ყოველთვის გარდაიქმნება ეკრანის ზომის შესაბამისად. ამიტომ კოდი ისე უნდა იყოს დაწერილი მონიტორის ზომის ცვლილებამ არ შეცვალოს (გააფუჭოს) ვებ გვერდის დიზაინი.
- ასევე აუცილებელია HTML დოკუმენტის ვალიდაცია, რაც ნიშნავს ვებ-გვერდის HTML სპეციფიკაციასთან შედარებას და სინტაქსური შეცდომების გამოვლინებას. ასეთი შესწორების სისტემას ეწოდება ვალიდატორი (validator). <http://validator.w3.org> შესაძლებელია ვებ-გვერდის ვალიდაცია.

პროგრამული უზრუნველყოფა

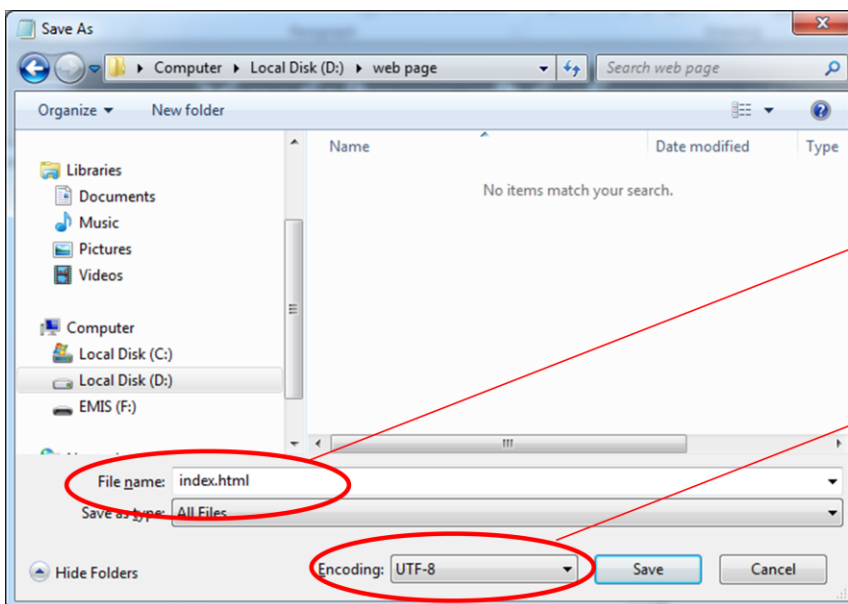
ვებგვერდის შესაქმნელად აუცილებელია:

- ✓ **ტექსტური რედაქტორი** მაგ. Notepad++ სადაც კოდის წერაა შესაძლებელი.
- ✓ **ინტერნეტ ბრაუზერები** შედეგის ანუ ვებგვერდის დასათვალიერებლად.
- ტექსტურ რედაქტორში შექმნილი დოკუმენტი ტექსტური ფაილის სახით რომ არ ჩაიწეროს დისკზე, ამისათვის ვანიჭებთ ფაილს html ან htm გაფართოებას.
- იმისათვის რომ მოხდეს დოკუმენტის უნიკოდის ფორმატით (ქართული ტექსტისათვის აუცილებელია) შენახვა ფაილის დამახსოვრებისას Encoding ველში უნდა ავირჩიოთ კოდირების ტიპი - UTF-8 .

როგორ დავიწყოთ?

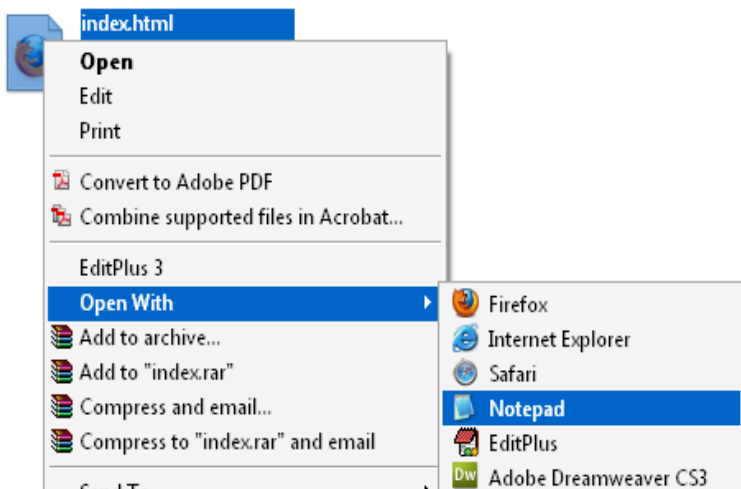
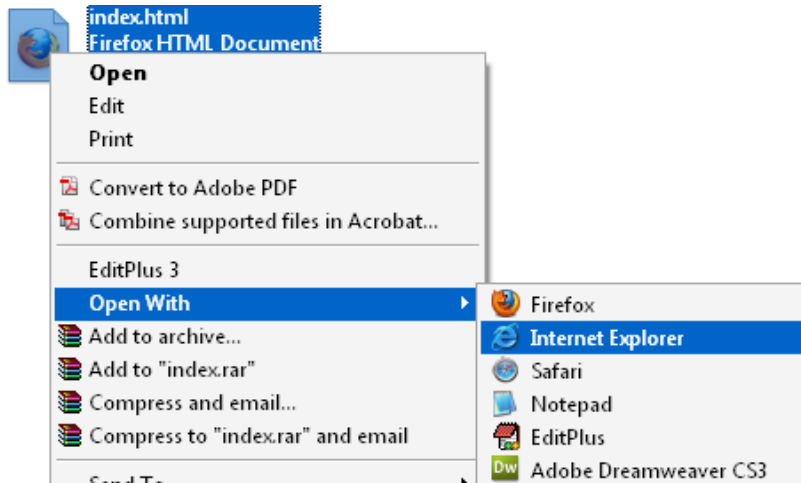
ვქმნით საქაღალდეს, სადაც განვათავსებთ საიტისთვის საჭირო ყველა ფაილს (სურათები, მიბმული ფაილები და html ფაილები).

როგორც წესი, საწყის ფაილს უნდა დავარქვათ index, ვინაიდან შესაბამის საქაღალდეზე მიმართვის შემთხვევაში index.html (index.htm) ფაილი ავტომატურად იხსნება.



*.html ფაილის გახსნა

*.html ფაილი ავტომატურად (მაუსის ორჯერ დაწკაპუნებით) იხსნება ბრაუზერში ან კონტექსტური მენიუდან Open With მენიუდან ვირჩევთ სასურველ ბრაუზერს. თუ გვინდა html კოდის ნახვა და რედაქტირება Open With მენიუდან ვირჩევთ შესაბამის ტექსტურ რედაქტორს მაგ. Notepad



თუ მოვახდენთ კოდის რედაქტირებას, ცვლილების ბრაუზერში ასახვისათვის, ჯერ ვინახავთ ფაილს ტექსტურ რედაქტორში, რის შემდეგაც ბრაუზერში დააჭირებ განახლების ღილაკს Refresh (ან F5 კლავიშს).



HTML დოკუმენტის სტრუქტურა

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">  
<html>  
<head>  
<title> დოკუმენტის სახელი </title>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
</head>
<body>
... დოკუმენტის შემცველობა...
</body>
</html>
```

ძირითადი ტეგები

- <!DOCTYPE> - განაცხადი დოკუმენტის ტიპის შესახებ (შემდგომში დაწვრილებით გავეცნობით);
- <html> ... </html> - არის ვებ დოკუმენტის პირველი და ბოლო ტეგი, რომელიც ბრაუზერს აწვდის ინფორმაციას HTML დოკუმენტის დაწყებისა და დასრულების შესახებ
- <head> ... </head> - სათაურის ბლოკი, სადაც იწერება ზოგადი ინფორმაცია ვებ გვერდის შესახებ, რომელიც არ გამოისახება ბრაუზერის ფანჯარაში.
- <title> ... </title> - ტეგებს შორის იწერება დოკუმენტის სახელი, რომელიც შემდგომ გამოისახება ბრაუზერის სათაურის პანელზე;
- <meta /> - ზოგადი ინფორმაციის ტეგები, რომლებიც განსაზღვრავენ საძიებო სისტემისთვის საკვანძო სიტყვებს, ვებ გვერდის ავტორს, გვერდის კოდირებას და.ა.შ.
- <body> ... </body> - ტეგი ხსნის დოკუმენტის ტანს, სადაც აღიწერება მთლიანად დოკუმენტი (web გვერდის შემცველობა) და გამოისახება ბრაუზერის სათაურების პანელზე

რა არის ტეგი?

ნებისმიერი დოკუმენტი შედგება ბრაუზერში გამოსატანი ტექსტისაგან და მმართველი სიმბოლოებისაგან, რომელსაც ტეგები ეწოდება. ტეგი ეს არის ბრძანება რომელიც მოთავსებულია < .. > სიმბოლოებს შორის.

ტეგები ორნაირია, რომელთა საშუალებითაც ხდება ამა თუ იმ ბრძანების მოქმედების არის განსაზღვრა

1. გამხსნელი <ტეგის სახელი>
2. დამხურავი </ტეგის სახელი>

(მაგ: <h1>.. </h1>, <p>...</p> და ა.შ.)

არსებობს გამონაკლისი ტეგები , რომელთათვისაც არ გამოიყენება დამხურავი ტეგი და იხურებიან თავისივე თავში <ტეგის სახელი/> (მაგ:
 და ა.შ.)

ტეგის თვისებები

ტეგი შეიძლება შეიცავდეს თვისებებს, რომელსაც ენიჭება კონკრეტული მნიშვნელობები. დახურვის დროს მიეთითება მხოლოდ ტეგის სახელი და ავტომატურად მასში შემავალ თვისებებს ენიჭებათ სტანდარტული მნიშვნელობები.

ტეგს საბოლოოდ ექნება შემდეგი სახე:

<ტეგის_სახელი თვისება1="მნიშვნელობა" თვისება2="მნიშვნელობა">

...

მოქედების ველი

....

</ტეგის_სახელი>

Xhtml კოდირების სტანდარები

- ტეგები, ატრიბუტები და მათი მნიშვნელობები აუცილებლად იწერება პატარა სიმბოლოებით!
- ყველა ტეგი აუცილებლად იხურება!
- ატრიბუტის მნიშვნელობები აუცილებლად უნდა ჩავსვათ ბრჭყალებში!
- დაცული უნდა იყოს ტეგების დახურვის თანმიმდევრობა ანუ სტრუქტურა.

მიაქციეთ ყურადღება:

```
<ტეგი1><ტეგი2><ტეგი3> ... </ტეგი3></ტეგი2></ტეგი1>
```

ასეთი მიმდევრობა სწორად ითვლება: ტეგი, რომელიც პირველი იხსნება, ბოლო იხურება.

შემდეგი მიმდევრობა კი შეცდომაა

```
<ტეგი1><ტეგი2><ტეგი3> ... </ტეგი3></ტეგი1></ტეგი2>
```

პირველი ლექციის დასასრული.

გმადლობთ ყურადღებისათვის.