

WEB
design

ზურაბ წვერაიძე
ფილხაზ პაატაშვილი
რამაზ შამუგია

WEB საიტების
ვიზუალური დაკრძეშტეზა

„ტექნიკური უნივერსიტეტი“

საქართველოს ტექნიკური უნივერსიტეტი

ზურაბ წვერაიძე, ფილხაზ პაატაშვილი,
რამაზ შამუგია

WEB საიტების ვიზუალური დაპროექტება



დამტკიცებულია სახელმძღვანელოდ
საქართველოს ტექნიკური უნივერსიტეტის
სარედაქციო-საგამომცემლო საბჭოს
მიერ. 27.12.2016, ოქმი №3

თბილისი

2017

1

სახელმძღვანელოში განხილულია საიტის დაპროექტების ძირითადი ეტაპები და ლოგიკური სტრუქტურის აგების პრინციპები. შემოთავაზებულია ვებსაიტების შექმნის მეთოდოლოგია, კერძოდ განხილულია ვიზუალური რედაქტორი Macromedia Dreamweaver. მოცემულია დინამიკური ვებგვერდების აგების თავისებურებები, სერვერული ტექნოლოგიების ძირითადი ცნებები და საშუალებები.

განკუთვნილია ინფორმატიკისა და მართვის სისტემების ფაკულტეტის პროფესიული და ბაკალავრიატის სტუდენტებისათვის, აგრეთვე ვებსაიტების შექმნით დაინტერესებულ პირთათვის.

სახელმძღვანელო ასევე სტუდენტებსა და დაინტერესებულ მკითხველს დაეხმარება საკუთარი საიტის ინტერნეტში განთავსებაში.

რეცენზენტები: საქართველოს ტექნიკური უნივერსიტეტის ინფორმატიკისა და მართვის სისტემების ფაკულტეტის ასოცირებული პროფესორი ირაკლი ბოჭორიშვილი,

სოხუმის სახელმწიფო უნივერსიტეტის მათემატიკისა და კომპიუტერულ მეცნიერებათა ფაკულტეტის ასოცირებული პროფესორი, დოქტორი მზია ნაჭყებია

© საგამომცემლო სახლი „ტექნიკური უნივერსიტეტი“, 2017

ISBN 978-9941-20-761-7

<http://www.gtu.ge>

ყველა უფლება დაცულია. ამ წიგნის არც ერთი ნაწილის (იქნება ეს ტექსტი, ფოტო, ილუსტრაცია თუ სხვა) გამოყენება არანაირი ფორმით და საშუალებით (იქნება ეს ელექტრონული თუ მექანიკური) არ შეიძლება გამომცემლის წერილობითი ნებართვის გარეშე.

საავტორო უფლებების დარღვევა ისჯება კანონით.



შინაარსი

I თავი

Web საიტების აგების საწყისები

შესავალი	5
1.1 Web საიტების დაგეგმვა	8
1.2 Web საიტის ლოგიკური სტრუქტურა	12
1.3 საიტის რუკა	18
1.4 ვაპროექტებთ ჩვენს პირველ ვებ საიტს.	18
1.5 რა იქნება ამის მერე?	21

II თავი

MACROMEDIA DREAMWEAVER

2. მუშაობა Macromedia Dreamweaver-თან.	22
2.1 Web გვერდის ფრეიმებად დაყოფა	37
2.2 ტექსტის ჩასმა და რედაქტირება.	40
2.3 Image გრაფიკული ობიექტის მოთავსება.	43
2.4 Hyperlink - ჰიპერმიმართვის მოთავსება.	46
2.5 ცხრილის მოთავსება.	49
2.6 Flash ტექსტის და Flash ლილაკების ჩასმა.	51
2.7 ჩამოშლადი მენიუს შექმნა.	54
2.8 Rollover Image - ის მოთავსება.	59
2.9 სანავიგაციო ლილაკების ჩასმა.	61
2.10 Email link ელექტრონული ფოსტის მისამართის მოთავსება.	65
2.11 Named Anchor ნიშნულის მოთავსება.	67
2.12 Comment კომენტარის ჩასმა.	69
2.13 Insert date თარიღის ჩასმა.	70
2.14 ახალი css სტილის შექმნა (css კასკადური სტილების ცხრილი cascading style sheets)	72
2.15 ფორმების შექმნა და გაუქმება.	73
2.16 Hidden Field ფარული ველის ჩასმა.	76
2.17 აღმის Checkbox ჩასმა.	77
2.18 გადამრთველების ჯგუფის Radio group ჩასმა.	79
2.19 გაშლადი მენიუს ტიპის List Menu სიის ჩასმა.	80
2.20 Layer ფენის ჩასმა.	82

2.21. საიტის ატვირთვა სერვეზე	83
---	----

III ტაჰი

გამოყენებითი ვებ პროგრამები

3.1. ზოგადი ინფორმაცია	87
3.2. ვებ პროგრამებთან დაკავშირებული ტერმინოლოგია	108
3.3. კომპიუტერის გამართვა გამოყენებითი პროგრამების შესამუშაოებლად	113

შესავალი

როგორც ცნობილია ვებ საიტი არის საერთო თემატიკით გაერთიანებული და ჰიპერმიმართვებით დაკავშირებული მრავალი ვებ გვერდის ერთობლიობა.

ვებ გვერდის შექმნა Macromedia Dreamweaver-ში ხორციელდება ვიზუალური დაპროექტების რეჟიმში.

ვებ გვერდის შექმნა ჰგავს Word-ში დოკუმენტის შექმნას.

Macromedia Dreamweaver მრავალფუნქციური რედაქტორია, რომლის საშუალებით შეიძლება შეიქმნას ნებისმიერი სირთულის ვებ გვერდი. რედაქტორში მუშაობის ასათვისებლად საჭიროა მინიმუმ ტექსტური რედაქტორის Word-ის ცოდნა.

ვებ გვერდის ფორმირებისას რედაქტორის მიერ ავტომატურად გენერირდება ამ გვერდის შესაბამისი Html კოდი. ბუნებრივია, დაისმის კითხვა, ამ შემთხვევაში აუცილებელია თუ არა Html ენის ცოდნა?

Html ენის ცოდნა, მინიმუმ, ორი მიზეზითაა საჭირო:

პირველი – Macromedia Dreamweaver-ით ვებ გვერდის ფორმირებისას სისტემის მიერ გენერირებული Html კოდი შეიცავს დიდი რაოდენობის ტეგებს, რომლებიც გენერირებულია იმ შემთხვევაშიც, როდესაც მოცემული პარამეტრის მნიშვნელობა სისტემის მიერ გაჩუმებით აღიქმება. მათი ამოშლა კოდიდან ამცირებს გვერდის მოცულობას და, შესაბამისად, ზრდის ბრაუზერში გვერდის ჩატვირთვის სიჩქარეს. Html ენის ცოდნა საშუალებას გვაძლევს, კოდის მოცულობის შემცირების მიზნით, ჩატარდეს სისტემის მიერ გენერირებული კოდის რედაქტირება.

მეორე – Web გვერდზე შეიძლება დავამატოთ ისეთი ეფექტები, რომელთა Html კოდის გენერირება სისტემის მიერ არ ხორციელდება.

Web გვერდის ასაწყობად ასევე ვიყენებთ სხვადასხვა ვიზუალური დაპროგრამების ენებს. გარდა პროგრამირების ცოდნისა, Web გვერდის შესაქმნელად პროგრამისტს აუცილებლად სჭირდება მონაცემთა ბაზების მართვის სისტემების პროგრამის გამოყენება. ჩვენი მიზანია სწორედ, სტუდენტებს და

არა მარტო მათ, ვასწავლოთ შეძლებისდაგვარად ძირითადი მართვის ობიექტები, რომლებსაც გამოიყენებენ პრაქტიკაში Web გვერდების ასაწყობად. მოყვანილი თემები, სტუდენტებს დიდად დაეხმარება პროგრამის შესწავლისას.

I თავი

საიტების აგების საწყისები

როგორც უკვე ვიცით, ვებ საიტი - არის საერთო თემატიკით გაერთიანებული და ჰიპერმინიმუმებით დაკავშირებული მრავალი ვებ გვერდის ერთობლივობა. ნამდვილი საიტის ასაგებად საჭიროა **HTML** და **CSS** ენების ცოდნა, და, რა თქმა უნდა საიტის თემატიკის მოფიქრება და მისი სტრუქტურის დაგეგმვა.

1.1. ვებ საიტების დაგეგმვა

დაგეგმვის ეტაპი უპირველესი და უმთავრესია ნებისმიერი საიტის შექმნისას. ეს მტკიცება სამართლიანია, როგორც მარტივი ერთგვერდიანი, ისე ტრანსნაციონალური სუპერკომპანიების, როგორებიცაა **Intel** და **Coca-Cola**, გიგანტური საიტების შემთხვევაშიც. მართლაც, სანამ რაიმეს კეთებას დავიწყებდეთ, საჭიროა კარგად გავერკვეთ იმაში, თუ რისი მიღება გვსურს. ყოველივე ამის გარკვევა აუცილებელია საწყის ეტაპზე, მანამ, სანამ უშუალოდ საქმეს შევუდგებოდეთ. როდესაც სამუშაო უკვე შესრულებულია, მასში რაიმეს

შეცვლა გაცილებით უფრო ძნელია, ვიდრე თავიდანვე ყველაფრის ისე კეთება, როგორც საჭიროა.

ამიტომ სანამ შევუდგებოდეთ ჩვენი მომავალი საიტის დაგეგმვას, გამოვრთოთ კომპიუტერი, რათა არ წაგვძლიოს ვებ გვერდების კომპიუტერზე აკინძვის სურვილი. ამ ეტაპზე ჩვენ მხოლოდ ფანქარი და ქაღალდი დაგვჭირდება, ან თუ მაინც და მაინც უპირატესობას კომპიუტერს მივანიჭებთ, მაშინ მხოლოდ ტექსტური რედაქტორის პროგრამა და საიტების პროექტირების **Microsoft Visio**-ს მსგავსი, სპეციალური პაკეტები შეიძლება დაგვჭირდეს.

მაშ ასე, რა უნდა გავარკვიოთ, ჩვენს საიტზე მუშაობის დაწყების წინ?

1. უპირველეს ყოვლისა, მკაცრად განვსაზღვროთ საიტის ამოცანები. რა უნდა აკეთოს მან: მოგვითხროს ვინმეს ან რაიმეს შესახებ, მოიზიდოს კლიენტები, დაგვეხმაროს რაიმე პრობლემების გადაწყვეტაში, გაგვანათლოს თუ გაგვართოს. ამოცანებიდან გამომდინარე, საიტის სტრუქტურა შეიძლება ძალიან განსხვავდებოდეს ერთმანეთისაგან.

2. განვსაზღვროთ, კონკრეტულად რა ინფორმაცია უნდა განთავსდეს საიტზე და რა, არა. აქ მთავარი პრინციპია, არაფერი ზედმეტი, მხოლოდ ის, რაც პოტენციურ მომხმარებლებს სჭირდება.

3. დასაწყისშივე შევაგროვოთ ყველა საჭირო ინფორმაცია, რათა არ მოგვიწიოს მისი მოძიება საიტზე მუშაობის დროს (ჩვენ მაშინ ამისთვის არ გვეცლება). ყველა ტექსტი, გამოსახულება, ფაილი, რომელთა განთავსებასაც ვაპირებთ საიტზე, ჩვენი კომპიუტერშივე უნდა იყოს მოთავსებული.

4. გადავწყვიტოთ, საიტის დიზაინი როგორ სტილში უნდა იყოს შესრულებული. იქნება ის კონსერვატიული, მკაცრი თუ სხვ. შესაბამისად, საშინაო გვერდი უნდა ასახავდეს ავტორის ესთეტიკურ მიდრეკილებებს, გასართობი საიტი უფრო მხიარულ ტონებში უნდა იყოს, ხოლო ახალი ამბებისათვის განკუთვნილი საინფორმაციო საიტი-უფრო მოკრძალებული, რათა დიზაინის სიმარტივემ არ დაჩრდილოს მთავარი-

ინფორმაცია. ამ ეტაპზე სასარგებლო იქნება ვებ გვერდების ესკიზების შექმნა ქალაქში.

5. მოვიფიქროთ საიტის ლოგიკური სტრუქტურა. სასურველია, დავხატოთ იგი ფურცელზე.

აქ აჯობებს, არ დავიწყოთ ახალი ველოსიპედის აღმოჩენა, არამედ შევიდეთ მსგავსი თემატიკის რომელიმე უკვე არსებულ და პოპულარულ ვებ საიტზე და ვნახოთ, როგორ არის ის ორგანიზებული.

მაგალითად, საოჯახო საიტისათვის იდეალური იქნება ასეთი სტრუქტურა: თავფურცელი (საშინაო გვერდი) მოკლე ცნობებით საიტის მეპატრონის შესახებ. ამავე გვერდზე შესაძლოა განთავსდეს მიპატიჟება სხვა გვერდების დასათვალიერებლად, შესაბამისი მინიშნებებით, ხოლო დანარჩენი გვერდები დაეთმოს ინფორმაციას ავტორის გატაცებების, პროექტების, ნამუშევრების შესახებ, იმის მიხედვით, თუ ვინ არის ავტორი-პროგრამისტი, მუსიკოსი, მხატვარი (უნდა მიეთითოს მინიშნებების სია პროგრამებზე, აუდიო კლიპებზე ან სურათების ფაილებზე). სასურველია აგრეთვე, რომ საიტის შესაბამის გვერდებზე

განთავსდეს ფოტოგალერეა და მინიშნებები „მეგობარ“ საიტებზე, ასევე ავტორის შესახებ უფრო დაწვრილებითი ინფორმაცია, მისი საფოსტო მისამართითა და ფოტოსურათით.

6. **შევამოწმეთ, ხომ არაფერი გამოგვრჩა.** ეს საიტის დაგეგმვის ბოლო ეტაპია, მაგრამ არანაკლებ მნიშვნელოვანი, ვიდრე სხვები.

ახლა კი ზუსტად ის დროა, რომ დავიწყოთ ლაპარაკი საიტის ლოგიკურ სტრუქტურაზე, ანუ იმაზე, რა განყოფილებებისაგან უნდა შედგებოდეს საიტი და როგორ უნდა იყოს ეს განყოფილებები ერთმანეთთან დაკავშირებული. ეს თემა მარტივი არ არის და, შესაბამისად, მსჯელობა ხანგრძლივი გვექნება.

1.2 ვებ საიტის ლოგიკური სტრუქტურა

მაშ ასე, საიტის ლოგიკური სტრუქტურა აღწერს საიტის სხვადასხვა ვებ გვერდების დანიშნულებასა და მათ შორის არსებულ კავშირებს. ის განისაზღვრება უპირველეს ყოვლისა იმით, თუ როგორ არის ორგანიზებული საიტზე ინფორმაცია. სხვანაირად რომ

ვთქვით, ყველა შესამუშავებელი საიტისათვის უნდა იქნას მოფიქრებული მისი საკუთარი სტრუქტურა.

რათ თქმა უნდა, არსებობენ საიტის სტრუქტურის ფორმის ფორმები, რომელთა დაცვა ყოველთვის აუცილებელია. ჩვენ ახლა მათ განვიხილავთ. ქვემოთ მოცემულია კარგად გააზრებული საიტის მაგალითი:

მთავარი გვერდი

საიტის სიახლეები

სიახლეების არქივი

განყოფილება1

გვერდი1

გვერდი2

.....

განყოფილება2

გვერდი1

გვერდი2

.....

.....

ინფორმაცია დამპროექტებლების შესახებ

საკონტაქტო მონაცემები

საიტის რუკა.

ახლა ვისაუბროთ საიტის ყოველი გვერდის შესახებ ცალცალკე.

მთავარი ან საწყისი გვერდი, როგორც წესი, წარმოადგენს წინასწარ განსაზღვრულ (default) გვერდს. დამთვალიერებელი (ბრაუზერი), საიტის ინტერნეტ-მისამართის აკრეფის შემდეგ სწორედ ამ გვერდზე ხვდება. ის საიტის მოკლე აღწერას, საიტის შესახებ მოკლე შესავალ ინფორმაციას, სიახლეებს (სავალდებულო არ არის) და საიტის სხვა გვერდებზე გადამყვან ჰიპერმინიშნებების სიას (ე.წ. სანავიგაციო ზოლს) შეიცავს. ხანდახან მთავარ გვერდზე ათავსებენ ცნობებს საიტის შემქმნელებისა და მათი უფლებების შესახებ, აგრეთვე ცნობებს საიტის მესაკუთრეებთან და საიტზე მოხსენიებულ სხვა პირებთან და ორგანიზაციებთან კონტაქტების შესახებ.

შეიძლება ითქვას, რომ მთავარი გვერდი-არის მთელი საიტის „სახე“.

ამიტომ ცდილობენ, რომ ის რაც შეიძლება პატარა იყოს, რათა დამთვალეიერებელი საიტიდან დროზე ადრე არ წავიდეს, ისე რომ, არ დაელოდოს მის სრულად ჩატვირთვას. მაგრამ მეორე უკიდურესობაშიც არ უნდა ჩავვარდეთ. მთავარი გვერდი, შინაარსის მიხედვით, იმდენად „სპარტანული“ გამოვიდეს, რომ დამთვალეიერებელი ვერ მიხვდეს, სად მოხვდა. მთავარი გვერდი დამთვალეიერებელს უნდა აწვდიდეს საკმარის ინფორმაციას საიტის შესახებ, მაგრამ ამასთან ერთად არ უნდა გადაიტვირთოს ზედმეტი ცნობებით და თავი არ მოაბეზროს დამთვალეიერებელს, თავისი ხანგრძლივი ჩატვირთვით. ეს ძალიან მნიშვნელოვანია.

საიტის სიახლეებს ხშირად მთავარ გვერდზეც ათავსებენ. ისინი წარმოადგენენ საიტზე მოთავსებული ყველა დამატებებისა და განახლებების ქრონოლოგიურ სიას. როგორც წესი, აქ აისახება მხოლოდ გარკვეული პერიოდის შესაბამისი (თვე, კვარტალი, წელი, იმის და მიხედვით, თუ რამდენად ხშირად ხდება საიტის განახლება) სიახლეები. უფრო ძველ სიახლეებზე

წვდომისათვის გათვალისწინებულია **სიახლეების არქივის ვებ გვერდი**.

საიტის სასარგებლო შიგთავსი-ის ინფორმაციაა, რომლისთვისაც საიტი იყო შექმნილი. მისი სტრუქტურირება ხდება ისევე, როგორც წიგნში: რაიმე თემასთან დაკავშირებული ცალკეული აბზაცები ერთიანდება თავებად, თავები -განყოფილებებად.

ამგვარად, საიტის დამთვალიერებელს მაშინვე შეუძლია იპოვოს მისთვის საინტერესო ინფორმაცია, განყოფილებებიდან თავებისკენ, თავებიდან აბზაცებისაკენ მოძრაობით, მანამ, სანამ ვერ მიაგნებს იმას, რისთვისაც შემოვიდა საიტზე.

ცნობები საიტის დამპროექტებლების შესახებ შეიძლება განთავსებულ იქნას როგორც ცალკე, ისე მთავარ გვერდზეც. თუ დამპროექტებლების რაოდენობა დიდი არ არის (ან სულაც ერთია), უფრო მიზანშეწონილია მეორე ვარიანტი. ამ შემთხვევაში მათ შესახებ ცნობები თავსდება ვებ გვერდის ბოლოს, საავტორო უფლებების შესახებ ცნობების გვერდით. ამასთან, აუცილებლად მიეთითება ელექტრონული

ფოსტის მისამართი, რომელზეც საიტის მომხმარებელს შეეძლება იმ პრობლემების შესახებ შეტყობინებების გაგზავნა, რომელთაც იგი საიტის დათვალიერებისას წააწყდა (არაჩატვირთვადი საიტი, „ცარიელი“ გამოსახულებები, „არსაითკენ“ გადამყვანი ჰიპერმინიშნებები, შეცდომები ტექსტში და სხვ.).

საიტის მფლობელთან კონტაქტის შესახებ ცნობები საჭიროა იმ შემთხვევაში, თუ მოცემული საიტი სარეკლამო მიზნებს ემსახურება. მაგალითად, თუ ეს სავაჭრო ორგანიზაციის საიტია, სავალდებულოა საკონტაქტო მონაცემების მითითება. სხვანაირად პოტენციური მომხმარებლებიდან ვერცერთი ვერ დაუკავშირდება გამყიდველს. ამ შემთხვევაში საჭიროა, როგორც ჩვეულებრივი „ქაღალდის“, ისე ელექტრონული ფოსტის მისამართების მითითება, აგრეთვე ტელეფონი, ფაქსი და პეიჯერი-მოკლედ, ყველა მონაცემი, რომელზეზეც პოტენციურ კლიენტებს შეეძლებათ მომართვა.

1.3 საიტის რუკა

საიტის რუკა არის ვებ გვერდი, რომელზეც განთავსებულია საიტის ლოგიკური სტრუქტურის შესაბამისად ორგანიზებული, საიტის ყველა გვერდზე გადამყვანი ჰიპერმინიშნებების ერთობლიობა. საიტის რუკა გამოიყენება იმისათვის, რომ მომხმარებელს, რომელმაც ზუსტად იცის თუ რა უნა, შეეძლოს მისთვის საჭირო ინფორმაციის სწრაფად მიგნება. რუკა როგორც წესი, აქვს ყველა მეტ-ნაკლებად დიდ და ჩახლართულ საიტებს; მარტივ საიტებზე ისინი ზედმეტია.

1.4 ვაპროექტებთ ჩვენს პირველ ვებ საიტს

ახლა, სანამ ფანქარი და ქაღალდი ან ტექსტური რედაქტორის პროგრამა, ჯერ კიდევ ჩვენს ხელთაა, მოვხაზოთ ჩვენი პირველი ვებ საიტის სტრუქტურა. მის შექმნას ჩვენ შევუდგებით მხოლოდ მესამე თავში, მაგრამ დაგეგმვა უმჯობესია განვახორციელოთ წინასწარ.

ვთქვათ, რომ ჩვენი საიტი წარმოდგენილია კომპიუტერული თემატიკის ფაილებისა და არქივების

სახით. საიტის შემადგენლობაში გვექნება ვებ-გვერდები, ჰიპერმინიშნებებით ფაილებზე (პროგრამები, დოკუმენტები, არქივები და სხვ.) და სტატიებზე, მაგრამ თვით ფაილები და სტატიები არა (ასე ჩვენ თავს დავაღწევთ საავტორო უფლებებთან დაკავშირებულ პრობლემებს).

მთავარ გვერდზე, სანავიგაციო ზოლისა და შესავალი ტექსტის გარდა, გამოტანილი იქნება აგრეთვე საიტის სიახლეები და ცნობები დამპროექტებლების შესახებ. საიტის რუკას ჩვენ არ ვადგენთ-ის ზედმეტი იქნება ჩვენი უმარტივესი საიტისათვის.

დაწვეროთ ქალაქზე ან ტექსტურ რედაქტორში ჩვენი საიტის ლოგიკური სტრუქტურა:

მთავარი გვერდი (საიტის სიახლეები, ცნობები დამპროექტებლების შესახებ)

ფაილები:

ინტერნეტი,

ოფისი,

მულტიმედია,

სისტემა,

პროგრამირება,
თამაშები,
სხვადასხვა,

სტატიები:

ინტერნეტი,
ოფისი,
მულტიმედია,
სისტემა,
პროგრამირება,
გართობა,
სხვადასხვა.

ამ ჩამონათვალის ყოველი სტრიქონი უნდა წარმოადგენდეს ცალკე ვებ-გვერდს.

გვერდები, *ფაილები* და *სტატიები* უნდა შეიცავდეს კატეგორიების სიებს (ინტერნეტი, ოფისი და ა.შ.); ამ სიების ყოველი პოზიცია უნდა წარმოადგენდეს ჰიპერმინიშნებას იმ გვერდზე, რომელიც შეიცავს მინიშნებებს თვით ამ კატეგორიას მიკუთვნებულ ფაილებსა და სტატიებზე. გავაკეთოთ ისე, რომ ყოველი სტატია იხსნებოდეს ვებ დამთვალიერებლის ცალკე

ფანჯარაში-მომხმარებლისათვის ასე უფრო მოხერხებული იქნება.

1.5 რა იქნება შემდეგ?

მაშ ასე, როგორც ჩანს საიტის სტრუქტურასთან დაკავშირებით ყველაფერი ნათელია. ახლა შეიძლება შევუდგეთ საქმეს. მაგრამ ჩვენ რა, ხელით ვაპირებთ ჩვენი ვებ საიტის ყველა ვებ გვერდის შექმნას? რატომ? ხომ არსებობენ მშვენიერი პროგრამები, რომლებიც დაგვეხმარება ამ საქმეში. მათ ვებ რედაქტორები ეწოდებათ. ჩვენ სწორად მათგან ერთ-ერთს-**Macromedia Dreamweaver**-ს გამოვიყენებთ ჩვენი საქმისათვის. მაშ ასე, შევუდგეთ საქმეს!

II ოპ30

MACROMEDIA DREAMWEAVER

2. მუშაობა Macromedia Dreamweaver-თან.

კაცობრიობა მუდმივად მიისწრფვის ცხოვრების გამარტივებისაკენ, ამისათვის მძიმე, მონოტონური და არასაინტერესო სამუშაოების შესასრულებლად სულ უფრო და უფრო სრულყოფილ ინსტრუმენტებს იგონებს. ეს გასაგებიც არის. ხომ შეიძლება, რომ გამოთავისუფლებული დრო, უფრო მეტი ეფექტურობით იქნას გამოყენებული ადამიანის მიერ, ვთქვათ, თვითგანათლებისათვის, სპორტისათვის, ხელოვნების ან სულაც კარგად გამოძინებისათვის. ხოლო, მთელი რუტინა მანქანებმა შეასრულონ.

ასეა ვებ დიზაინშიც. გულმოდგინებით **HTML** ტეგების წერას, მათი ერთმანეთში ჩალაგებულობის წესის დაცვაზე თვალთვალს, შეცდომების გასწორებას, მიღებული შედეგების ყოველწუთიერად შემოწმებას ვებ-დამთვალიერებელში, კაცობრიობას ურჩევნია, საჭირო ტექსტის მარტივად დაწერა და მზა ვებ გვერდის

მიღება. სწორად ამისათვის წერენ პროგრამისტები განსაკუთრებულ პროგრამებს, რომელთა დანიშნულება მდგომარეობს ვებ გვერდების შექმნაში-**ვებ-რედაქტორებს**.

ერთ-ერთი ასეთი პროგრამა დაწერილია ფირმა **Macromedia**-ს პროგრამისტების მიერ და მას **Macromedia Dreamweaver** -ი ეწოდება.

მისი პირველი ვერსია 1998 წელს გამოვიდა; **Dreamweaver**-ი საოცრად მძლავრი და ამავე დროს გამოსაყენებლად მარტივი პროგრამაა; მასთან მუშაობა შეუძლია დამწყებ მომხმარებელსაც კი, რომელსაც წარმოდგენაც კი არა აქვს **HTML**-ენაზე, მაგრამ პროფესიონალებიც კმაყოფილები რჩებიან მისი გამოყენებით. მრავალი ვებ პროგრამისტი მუშაობს **Dreamweaver**-თან, პირველი ვერსიიდან დაწყებული და ჯერჯერობით არც უფიქრიათ მისი შეცვლა რაიმე სხვა პროგრამით.

Dreamweaver -ი, **WYSIWYG** (**What You See Is What You Get**, „რასაც ხედავ, იმას მიიღებ“) პრინციპით მომუშავე ვიზუალური ვებ რედაქტორების ტიპობრივი

წარმომადგენელია. ამ შემთხვევაში, მომხმარებელს მის მიერ ფორმირებული ტექსტის დანახვა მაშინვე შეუძლია რედაქტორის ფანჯარაში. ამავე პრინციპით მუშაობს ცნობილი ტექსტური რედაქტორიც **Microsoft Word**.

არავიზუალური ვებ რედაქტორები (იგივე-**HTML-რედაქტორები**) აგებულია სხვა პრინციპზე. ისინი უშუალოდ **HTML-კოდთან** მუშაობს და ამავე დროს მომხმარებელს სხვადასხვა დამატებით შესაძლებლობას აძლევს: ტეგების სწრაფად ჩასმა, ატრიბუტებისა და მათი მნიშვნელობების განსაზღვრა მოხერხებულად, წინასწარ განსაზღვრული შაბლონების ანაკრებებს ვებ გვერდების სტანდარტული ელემენტების შესაქმნელად, სტილების ცხრილებისათვის სპეციალურ რედაქტორებს, შედეგების სწრაფად დათვალიერების შესაძლებლობას და სხვ.. ამ შემთხვევაში ისინი წააგავს **NotePad-ს**, მაგრამ მნიშვნელოვნად გაფართოებულს.

შეიძლება ვინმემ დაასკვნას, რომ ვიზუალური რედაქტორები უფრო დამწყები ვებ დიზაინერებისათვისაა განკუთვნილნი, ხოლო

არავიზუალური რედაქტორები უფრო გამოცდილი კოლეგებისათვის. მაგრამ ეს მთლად სწორი დასკვნა არ იქნება.

პროფესიონალი, როგორც წესი, იყენებს ერთსაც და მეორესაც; მას შეუძლია შექმნას ვებ გვერდის ჩონჩხი ვიზუალურ რედაქტორში, ხოლო მისი დახვეწა განახორციელოს-არავიზუალურში. ასე გაცილებით მარტივია, განსაკუთრებით თუ ვებ გვერდი საკმარისად რთულია.

უნდა აღინიშნოს, რომ პრაქტიკულად ყველა მეტ-ნაკლებად სერიოზულ ვებ რედაქტორს აქვს თვით **HTML**-კოდის ჩასწორების რეჟიმი (ანუ ფაქტობრივად ჰიბრიდულ ვებ-რედაქტორებს წარმოადგენენ). ამიტომ უკვე პრაქტიკულად ყოველთვის, როცა „ვიზუალურ ვებ-რედაქტორებზე“ ლაპარაკობენ, სწორედ ჰიბრიდულ პროგრამებს გულისხმობენ. ცხადია, რომ მათ რიცხვს პირველ რიგში მიეკუთვნება **Dreamweaver**-ი, რომელსაც უკვე დროა, უფრო ახლოს გავეცნოთ.

Dreamweaver -ის წინასწარი გაწყობა

Dreamweaver - ბრწყინვალე პროგრამაა, რომელიც მომხმარებელს „პირველი სიტყვისთანავე“ უგებს. მაგრამ მიუხედავად ამისა, ვებ საიტზე მუშაობის დაწყებამდე ჩვენ მოგვიწევს მისი გაწყობა (გამართვა, პარამეტრების განსაზღვრა, დაყენება). საბედნიეროდ სავალდებულოდ გასაწყობი პარამეტრების რაოდენობა არც თუ ისე ბევრია, მაგრამ მათი განსაზღვრა აუცილებელია, სხვა შემთხვევაში ჩვენ სასურველ შედეგს ვერ მივიღებთ.

Dreamweaver-ის ყველა პარამეტრის განსაზღვრა ხორციელდება მრავალფუნქციურ დიალოგურ ფანჯარაში **Preferences**, რომელიც მართვის სხვადასხვა ელემენტის შემცველი ჩანართებისაგან შედგება. მისი გამოძახებისათვის მენიუ **Edit**-ში პუნქტი **Preferences** უნდა ავირჩიოთ ან დავაჭიროთ <Ctrl>+<U> კლავიშების კომბინაციას.

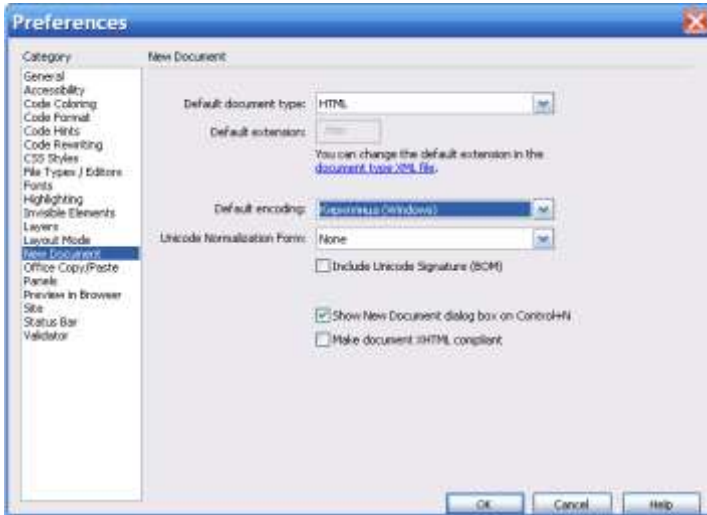
პირველ რიგში, ჩვენ უნდა განვსაზღვროთ კოდირების წესი ახალი ვებ გვერდებისათვის. ამისათვის სიაში **Category**, რომელიც ფანჯრის მარცხენა

ნაწილში მდებარეობს, ავირჩიოთ პუნქტი **New Document**. ამის შემდეგ ფანჯრის მარჯვენა მხარეს გაჩნდება მართვის ელემენტები, რომელთა საშუალებითაც განისაზღვრება ახლად შესაქმნელი ვებ გვერდების პარამეტრები. (ნახ.2.1).

კოდირების წესი აირჩევა განშლად სიაში-**Default encoding**. ინგლისურენოვანი ვებ გვერდებისათვის უნდა ავირჩიოთ პუნქტი *Западноевропейская* (დასავლეთევროპული, კოდირება iso-8859-1, რომელიც ინგლისური ენისთვის გამოიყენება) ან *Центральноевропейская* ცენტრალურევროპული, კოდირება 1250).

შენიშვნა: *Dreamweaver-ი იძლევა ყოველი ვებ-გვერდის კოდირების ცალ-ცალკე განსაზღვრის შესაძლებლობას.*

ამის მერე, ჩვენ აუცილებლად უნდა ჩავრთოთ ალამი **Use when opening existing files that don't specify an encoding**, თუ ის წინასწარ ჩართული არ არის.

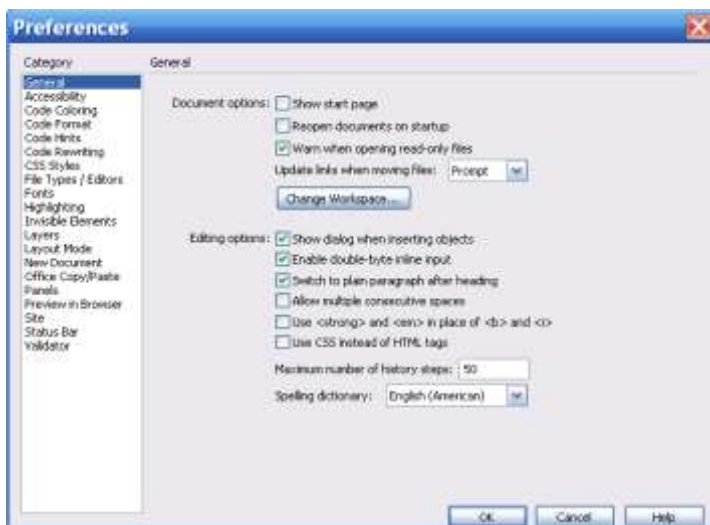


ნახ.2.1. დიალოგური ფანჯარა Preferences-ის კატეგორია New Document

ამის შემდეგ Dreamweaver-ი ავტომატურად გამოიყენებს **Default encoding** სიაში არჩეულ კოდირებას, ყველა გასახსნელი ვებ-გვერდების მიმართ, თუ მათ **HTML**-კოდში არ არის კოდირების განმსაზღვრელი `<META>` ტეგი.

კოდირების განსაზღვრის შემდეგ, **Category** სიაში ავირჩიოთ პუნქტი **General**. ფანჯარა **Preferences** მიიღებს სახეს, რომელიც ნაჩვენებია ნახ.2.2-ზე.

ალამი Use `` and `` in place of `` and `<i>` ჩართავს ან გამორთავს ტექსტის ფიზიკური ფორმატირების `` და `<I>` ტეგების ნაცვლად, ლოგიკური ფორმატირების `` და `` ტეგების გამოყენების შესაძლებლობას. როგორც წესი, ის ჩართულია წინასწარ; თუ არადა ის უნდა ჩაირთოს. ჩვენ კი გვსურს ვებ გვერდების შექმნა უკანასკნელი სტანდარტების შესაბამისად.



ნახ.2.2. დიალოგური ფანჯარა Preferences-ის კატეგორია
General

ალამი **Use CSS instead of HTML tags**, ტექსტის ფორმატირებისათვის **HTML** ტეგების ნაცვლად ჩართავს ან გამორთავს, **CSS** სტილების გამოყენების შესაძლებლობას. წესით, ისიც წინასწარ ჩართული უნდა იყოს, და თუ არა, უნდა ჩაირთოს.

Dreamweaver-ის გაწყობის დასრულებისთანავე დავაჭიროთ ღილაკს **OK**, რათა დავიმახსოვროთ განხორციელებული ცვლილებები. ამის შემდეგ შეიძლება შევუდგეთ მუშაობას.

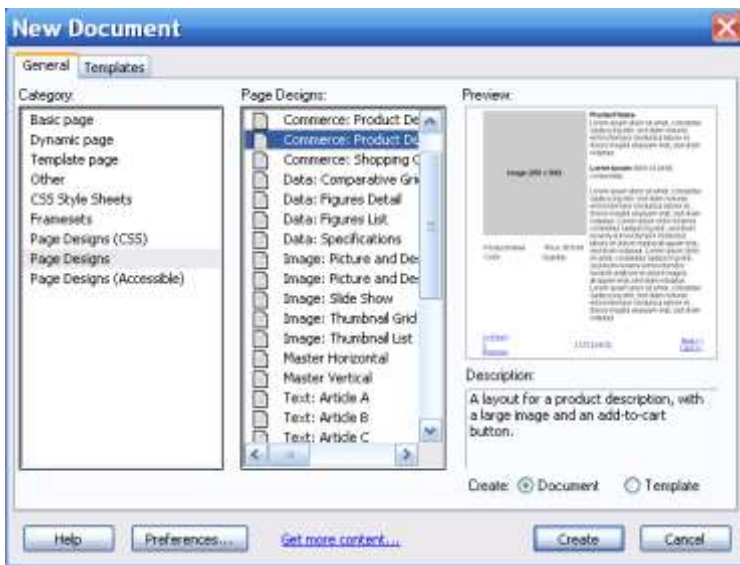
Dreamweaver-ში მუშაობის საფუძვლები

ახლა ჩვენ დავიწყებთ მუშაობას ჩვენს პირველ ვებ საიტზე. მის შემადგენლობაში შესატანი ყველა ფაილისათვის (ვებ-გვერდები, სტილების ცხრილი, გრაფიკული გამოსახულებები) შევქმნათ ცალკე საქაღალდე, დავარქვათ მას ვთქვათ **Site1** და დავიწყოთ. **Windows** სისტემაში პროგრამების გაშვება ყველამ იცის. დავაჭიროთ ჩვენთვის კარგად ნაცნობ ღილაკს **Start-გაშვება**, გახსნილ მენიუში ავირჩიოთ პუნქტი **Programs-პროგრამები**, შემდეგ-პუნქტი **Macromedia** და გახსნილ

ქვემნიშობი-პუნქტი **Macromedia Dreamweave**. გარკვეული დროის შემდეგ (**Dreamweaver**-ის გახსნას დრო სჭირდება) ეკრანზე მისი მთავარი ფანჯარა გაჩნდება, რომელშიც ჩვენ უნდა ვიმუშაოთ.

ახალი ვებ გვერდის შექმნა

ახალ ვებ გვერდზე მუშაობა იწყება მისი შექმნიდან. იმისათვის, რომ ახალი „ცარიელი“ გვერდი შევქმნათ, მენიუ **File**-ში ავირჩიოთ პუნქტი **New**. ეკრანზე გაჩნდება დიალოგური ფანჯარა **New Document**, რომელიც ნაჩვენებია ნახაზზე.



ნახ.2.3.დიალოგური ფანჯარა New Document

იმისათვის, რომ ჩვენი ვებ-შემოქმედებისათვის, მივიღოთ „სუფთა ფურცელი“ სიაში **Category** ავირჩიოთ **Basic page**, ხოლო, სიაში, რომელიც მარჯვნივ მდებარეობს, პუნქტი **HTML** და დავაჭიროთ ღილაკს **Create**. ეკრანზე გაჩნდება დოკუმენტის ფანჯარა, რომელშიც გაიხსნება **Dreamweaver**-ის მიერ ჩვენთვის შექმნილი ვებ გვერდი.

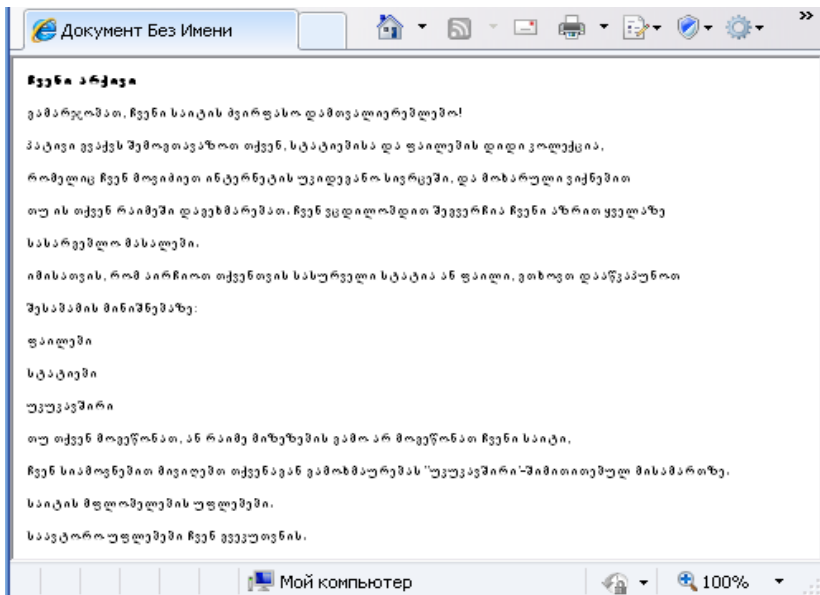
აქ ცოტა რამ მაინც უნდა ითქვას დოკუმენტის ფანჯრების შესახებ. **Dreamweaver**-ი რამდენიმე ვებ გვერდის ერთდროულად გახსნის საშუალებასაც იძლევა; ამავე დროს ყოველი გვერდი იხსნება თავის საკუთარ, ცალკე ფანჯარაში. ეს ძალიან მოხერხებულია საიტის შექმნის პროცესში: საიტის ერთ გვერდზე მუშაობისას, სხვა გვერდებიც თვალწინ გვაქვს.

შენიშვნა: *Dreamweaver*-ი იძლევა ვებ გვერდების შაბლონების (გვერდების თავისებური ნიმუშები, რომლებზეც წინასწარ არის ზოგიერთი ელემენტები განთავსებული) საფუძველზე შექმნის შესაძლებლობას. *New Document* ფანჯრის სიაში *Category* ხდება შაბლონების კატეგორიის არჩევა, ხოლო სიაში,

რომელიც იქვე მარჯვნივ მდებარეობს-არჩეულ კატეგორიაში შემავალი თვით შაბლონის არჩევა. წინასწარი დათვალიერების არეში ჩვენ შეგვიძლია დავინახოთ, თუ რას წარმოადგენს არჩეული შაბლონი.

ტექსტის აკრეფა

აი, დადგა აღსანიშნავი მომენტიც! ახლა ჩვენ ავკრეფთ პირველ ტექსტს Dreamweaver-ის დოკუმენტების ფანჯარაში და ამგვარად შევქმნით ჩვენი საიტის მთავარ გვერდს.



ახლა არ შევუდგებით მის დაფორმატებას-ჯერ მხოლოდ ავკრეფთ. ეს ხორციელდება ზუსტად ისე, როგორც ნებისმიერ ტექსურ რედაქტორში-„ვაკაკუნებთ კლავიატურაზე“, ვუყურებთ შედეგს და ჩასწორებებისათვის ვიყენებთ ისრიან კლავიშებს <Backspace>და. ბოლოს დავიმახსოვრებთ მიღებულ ვებ გვერდს.

ვებ გვერდის დასამახსოვრებლად საჭიროა მენიუ **Файл-File-ფაილში** პუნქტ **Сохранить-Save-შენახვა**-ს გამოყენება ან კლავიშების კომბინაციის <Ctrl>+<C> დაჭერა. თუ აქამდე ჩვენ არ შეგვინახავს ვებ გვერდი (ჩვენ კი ის ჯერ არ შეგვინახავს), ეკრანზე გაჩნდება **Windows**-ის ფაილების შენახვის სტანდარტული დიალოგური ფანჯარა. ავირჩიოთ მასში ადრე შექმნილი საქალაქი **Site1** და ვიფიქროთ იმაზე, თუ რა დავარქვათ ჩვენი პირველი ვებ-გვერდის ფაილს.

ჩვენ უკვე ვიცით, რომ ვებ გვერდებიდან ერთ-ერთი ვებ სერვერის მიერ აღიქმება, როგორც წინასწარ განსაზღვრული საწყისი გვერდი. აგრეთვე ვიცით, რომ ასეთ ვებ-გვერდს ენიჭება სახელი **default** ან **index**

(შესაბამისი გაფართოებით **htm** ან **html**). მოდით, ჩვენ საწყის ვებ-გვერდს დავარქვათ **default.htm**. შევიტანოთ ეს სახელი ფაილის შენახვის დიალოგური ფანჯრის, ფაილის სახელის შეტანის ველში და დავაჭიროთ შენახვის ღილაკს.

მაშ ასე, უკვე შევქმენით ჩვენი პირველი ვებ გვერდი.

მთელი ამ დროის განმავლობაში, ჩვენ ვმუშაობდით **Dreamweaver**-ში web გვერდების წარმოდგენის **WYSIWIG** რეჟიმში. ადრე ნათქვამი იყო, რომ **Dreamweaver**-ი მომხმარებელს აძლევს აგრეთვე **HTML**-კოდის რედაქტირების საშუალებას. იმისათვის, რომ მივიღოთ დაშვება **HTML**-კოდთან, ჩვენ უნდა გადავერთოთ კოდის წარმოჩენის რეჟიმში. ეს ხორციელდება სპეციალური ღილაკების საშუალებით, რომლებიც დოკუმენტების ინსტრუმენტული პანელის მარცხენა ნაწილში მდებარეობს, როგორც ეს ნახაზზეა ნაჩვენები:



მოდით, ჩამოვთვალოთ ეს დილაკები მარცხნიდან მარჯვნივ:


Code - HTML კოდის წარმოჩენის რეჟიმი;

Split - HTML კოდისა და ვებ გვერდის ერთდროულად წარმოჩენის რეჟიმი. ამ დროს, დოკუმენტის ფანჯარა ორ ნაწილად იყოფა; ზედა ნაწილში წარმოდგენილია HTML კოდი, ქვედაში-თვითონ ვებ გვერდი;

Design - ვებ გვერდის წარმოჩენის რეჟიმი.

HTML-კოდის წარმოდგენის რეჟიმში კარგად ჩანს, რომ **Dreamweave**-მა ჩვენ მიერ შეტანილი ყოველი აბზაცი მოათავსა <P> ტეგის შიგნით. ასევე მან ავტომატურად მოახდინა სათაურის სექციის ფორმირება და მოათავსა მასში გვერდის დასახელება <META> ტეგი, რომელიც მიუთითებს გამოყენებულ კოდირებაზე. მაშ ასე, ჩვენ მივიღეთ სწორად გაფორმებული ვებ გვერდი ისე, რომ HTML-კოდის არც ერთი სტრიქონი არ დაგვიწერია.

2.1. WEB გვერდის ფრეიმებად დაყოფა

ფრეიმის მეშვეობით შესაძლებელია Web ფურცლის დაყოფა ნაწილებად. ყველა დაყოფილი ნაწილი ცალკე Web გვერდია, მაგრამ ჯამში იგი წარმოადგენს ერთ გვერდს. ფრეიმის საშუალებით Web გვერდი შეიძლება დავყოთ იმდენ ნაწილად, რამდენიც გვინდა, მაგ., ვდგებით ცარიელ Web გვერდზე და ძირითადი მენიუს Modify ჩანართში ვირჩევთ Frameset ბრძანებას ან ობიექტების ჩასმის ხელსაწყოების ფანჯარაში ვაჭერთ Layout ჯგუფის Frames ღილაკს, გამოსულ ფანჯარაში: ვაჭერთ  ღილაკს.



სურ. 2.1

Frame ვირჩევთ ფრეიმის განლაგების სახეს, თუ სად გვინდა მისი ჩასმა: მარჯვნივ, მარცხნივ, ფურცლის თავში თუ ფურცლის ბოლოში.

Title ვწერთ სათაურს, რომელიც აისახება ბრაუზერის ფანჯარაში. არჩევის შემდეგ ვაჭერთ ok ღილაკს და ჩვენი **Web** გვერდის მარცხენა ნაწილი გაიყოფა შუაზე. ფრეიმის ჩასმის შემდეგ ვახდენთ ფრეიმის რედაქტირებას Properties მენიუში:

Row ბრძანების მეშვეობით ფრეიმის სტრიქონი გაიზრდება პროცენტებით და პიქსელებით.

Colmn ბრძანების მეშვეობით, ფრეიმის სვეტი გაიზრდება პროცენტებით და პიქსელებით.

Border ბრძანების მეშვეობით, ფრეიმის ხაზი ირთვება ან გაუქმდება.

Border width ბრძანების მეშვეობით ფრეიმის ხაზის სისქე გადიდება ან შემცირდება. ხორდერ ცოლორ ბრძანებით კი ფრეიმის ხაზის ფერი შეირჩევა ან გაუქმდება.

დავალება

1. Web ფურცელი ფრეიმის საშუალებით დაყავით სამ ნაწილად.
2. ფრეიმის ხაზს მიეცით წითელი ფერი.
3. სამუშაოს დამთავრების შემდეგ, ფაილი შეინახეთ HTML -ად.

2.2. ტექსტის ჩასმა და რედაქტირება

თუ გვინდა ტექსტის ჩასმა Web გვერდზე და შემდეგ რედაქტირება, ამისათვის კურსორით ვდგებით Web ფურცელზე სასურველ ადგილას და კლავიატურის მეშვეობით ავკრეფთ ტექსტს. შრიფტის ასარჩევად ვასრულებთ ბრძანებას Properties ან ძირითადი მენიუს Text ჩანართში ვირჩევთ:


Font –სასურველ შრიფტს;


Size – ტექსტის ზომას;


Style – ტექსტის სტილს;


Font color – ვაძლევთ ტექსტს ფერს;



 Bold – ვამუქებთ ტექსტს;

 Italic ბრძანების საშუალებით ტექსტს განვათავსებთ მარცხნივ.

 Align left ბრძანების საშუალებით ტექსტს განვათავსებთ მარცხნივ.

 Align center-ბრძანების საშუალებით ტექსტს განვათავსებთ ცენტრში.

 Align right ბრძანების საშუალებით ტექსტს ვასწორებთ მარჯვნივ.

ასევე, ამ პროგრამის მეშვეობით შესაძლებელია ტექსტის  დანომვრა ნუმერაციით და ასევე უნომროდ (წერტილებით) .

სტანდარტული მენიუს Page Properties ლილაკზე დაჭერით გამოვა ფანჯარა, სადაც Appearance ჩანართში ვირჩევთ:



სურ. 2.2.

Background color ხდება web ფურცელზე ფერის ჩასმა ფონად, ხოლო Background image ბრძანების საშუალებით კი web ფურცელზე სურათის ჩასმა ფონად.

Link-ჩანართში ვაძლევთ ლინკს სტილს, სადაც: L
Link color -ვირჩევთ ტექსტის საწყის ფერს.

Rollover link - მასზე მაუსის კურსორის მოთავსებისას, ტექსტი შეიცვლის ფერს.

Active link - მასზე მაუსის კურსორის დაჭერისას, ტექსტი შეიცვლის ფერს.

Underline style ბრძანებით კი ტექსტი, რომელზეც თავსდება ლინკი, შესაძლებელია ქვემოთ გავუსვათ ქვემ ხაზი ან მოვუხსნათ. ამას იმ შემთხვევაში ვიყენებთ, თუ გვინდა ჩვეულებრივი ტექსტი განვასხვავოთ დალინკული ტექსტისაგან. სასურველი ვარიანტის არჩევის შემდეგ ვადასტურებთ ბრძანებას ok..



სურ. 2.3.

Title/Encoding ჩანართში ვირჩევთ ქართულ შრიფტს Unicode (UTF-8). და ვადასტურებთ ბრძანებას ok.

შენიშვნა: Unicode (UTF-8) შრიფტს ნებისმიერი კომპიუტერი აღიქვამს იმ შემთხვევაშიც კი, თუ მასში ქართული შრიფტები არ არის დაყენებული (დაინსტალირებული).

დავალეზა

1. Web გვერდზე აკრიფეთ ტექსტი;
2. აკრეფილ ტექსტს მიეცით ფერი;

3. ტექსტს მიეცით ზომა 16 და შრიფტი AcadNusx;
4. აკრეფილ ტექსტზე ლინკი მოათავსეთ სასურველ ობიექტზე.

2.3. Image გრაფიკული ობიექტის მოთავსება

ძირითადი მენიუს **Insert** პუნქტში შევასრულოთ ბრძანება Image ან ობიექტების ჩასმის ხელსაწყოების ფანჯარაში დავაჭიროთ ღილაკს . გამოსულ ფანჯარაში:



სურ. 2.4.

ვირჩევთ ობიექტს. ჩასმული Image ობიექტის თვისებათა ფანჯარაში: Image შეგვყავს სახელი, Alt გრაფში ტექსტი, რომელიც ბრაუზერში მაუსის კურსორის ობიექტზე მოთავსების დროს გამოდის.

Src გრაფიკული ობიექტის შესაბამისი ფაილის არჩევა (ამ შემთხვევაში, გრაფში მოთავსდება ობიექტის ფაილის სახელი).

Link გრაფიკული ობიექტისთვის ჰიპერმიმართვის არჩევა.

W,H გრაფიკული ობიექტის სიგანისა და სიმაღლის განსაზღვრა, თუ საჭიროა ობიექტის თავდაპირველი ზომების დაბრუნება; ამ ველების მარჯვნივ მდგომ ღილაკზე დაწკაპუნება.

Align ჩასმულ გრაფიკულ ობიექტსა და ტექსტს შორის განლაგების სხვადასხვა ფორმით არჩევა.

Vspace დაშორება ობიექტსა და გვერდის ზედა კიდეს შორის.

Wspace დაშორება ობიექტსა და გვერდის მარცხენა კიდეს შორის.

Border ჩარჩოს სისქე თუ არ არის მითითებული ჩარჩო.

Low Src შეგვძევს გრაფიკული ან ტექსტური გამოსახულების ფაილი, რომელიც წინასწარ მოთავსდება მთავარი გრაფიკული ობიექტის ჩატვირთვამდე.

Edit რედაქტირება *fw Macromedia Fireworks*

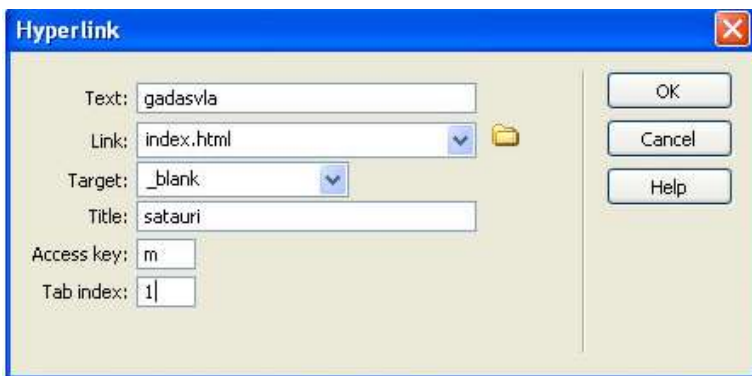
დავალება

1. ჩასვით სურათი;
2. შეუცვალეთ ზომები;
3. დალინკეთ სურათი ნებისმიერ web გვერდზე.

2.4. Hyperlink - ჰიპერმიმართვის მოთავსება

ძირითადი მენიუს Insert პუნქტში ვასრულებთ Hyperlink ბრძანებას ან ობიექტების ჩასმის ხელსაწყოების ფანჯარაში ვაჭერთ Common ჯგუფის Hyperlink ღილაკს.

დიალოგური ფანჯრის Text ველში შევიტანოთ ჰიპერმიმართვის სახელი. Link ველში შევიტანოთ ფაილის სახელი, რომელზეც უნდა მოხდეს მიმართვა ან ფაილის მოსაძებნად დავაჭერთ საქალაღის ნიშნაკს და დავაწკაპუნებთ შესაბამისი ფაილის სახელს.



სურ. 2.5.

Target ველში ავირჩევთ ფანჯარას, რომელშიც უნდა გაიხსნას ჰიპერმიმართვით არჩეული ფაილი. ველის მარჯვნივ მდგომ ისრიან ღილაკზე დაჭერით გამოსული ჩამონათვლიდან შეგვიძლია ავირჩიოთ შემდეგი დარეზერვებული სიტყვები:

Blank მიმართვის ფაილი გამოვა ახალ ფანჯარაში.

Parent მიმართვის ფაილი გამოვა ფრამესეტ-ში ან ბრაუზერის მთლიან ფანჯარაში.

Self მიმართვის ფაილი გამოვა იმავე ფანჯარაში ან ფრეიმში, რომელშიც ჰიპერმიმართვაა მოთავსებული. ეს პარამეტრი სტანდარტულად აირჩევა.


Top მიმართვის ფაილი გამოვა ბრაუზერის მთლიან ფანჯარაში.

Title ველში შევიტანოთ ჰიპერმიმართვის სათაური, რომელიც გამოჩნდება ჰიპერმიმართვაზე მაუსის კურსორის მოთავსებისას.

Access key მივუთითებთ ერთ ასოს, მაგალითად, M. შემდეგში ამ ჰიპერმიმართვის ასარჩევად კლავიატურაზე დავაჭიროთ Alt + M.

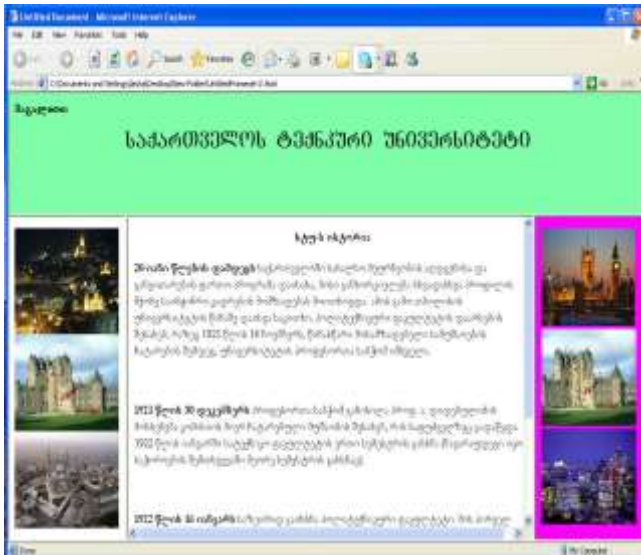
Tab index შევიტანოთ რიცხვი, რომელიც განსაზღვრავს ჰიპერმიმართვის არჩევის რიგს Tab ღილაკზე დაჭერისას.

ობიექტზე ცალკეული მონიშნული უბნისათვის ჰიპერმიმართვის შექმნა.. ობიექტის მოთავსების შემდეგ, მოვნიშნავთ მის ცალკეულ უბანს. ამისათვის

მოვნიშნავთ ობიექტს და თვისებათა ფანჯარაში ავირჩევთ მონიშვნის სხვადასხვა ფორმას, მაგალითად:  Draw Polygon Hotspot მრავალკუთხედი, წრე, წირი. მონიშვნის შემდეგ, Link-ში შევიტანოთ ფაილის სახელი, რომელზეც უნდა მოხდეს მიმართვა.

დავალება

შექმენით მარტივი WEB გვერდი.

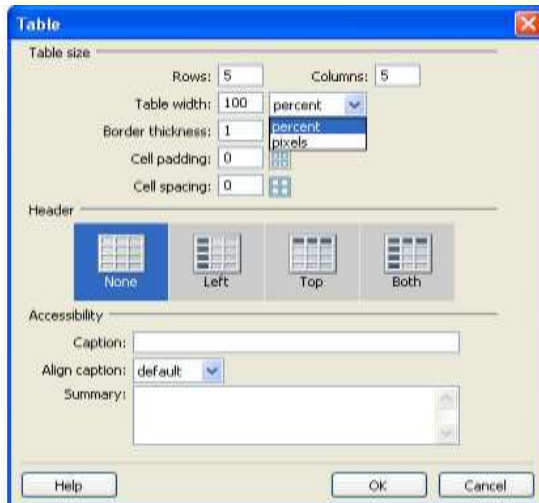


სურ. 2.6.

გამოიყენეთ ყველა ის შესწავლილი ლაბორატორიული სამუშაო, რომელიც აქამდე განვიხილეთ.

2.5. ცხრილის მოთავსება

ძირითადი მენიუს Insert პუნქტში შევასრულოთ ბრძანება Table ან ობიექტების ჩასმის ხელსაწყოების ფანჯარაში დავაჭიროთ Common ჯგუფის Table ღილაკს. გამოსულ ფანჯარაში შევიტანოთ შემდეგი ველების მნიშვნელობები:



სურ. 2.7.

Rows სტრიქონების რაოდენობა;

Columns სვეტების რაოდენობა;

Table width ცხრილის სიგანე პიქსელებით ან პროცენტულად გვერდის სიგანის მიმართ;

Table thickness ცხრილის ჩარჩოს სისქე. თუ სისქე 0-ია, მაშინ ჩარჩო web გვერდზე არ გამოჩნდება;

Cell padding მონაცემების უჯრის გვერდების ირგვლივ დაშორება პიქსელებით;

Cell spacing უჯრებს შორის დაშორება პიქსელებით;

Header სტრიქონის ან სვეტის სათაურის, ან ორივეს ერთად არჩევა.

Caption ცხრილის სათაური;

Align caption ცხრილის სათაურის სწორება top, bottom;

Summary ცხრილის აღწერის ტექსტის მოთავსება, ეს ტექსტი ბრაუზერში არ ჩანს.

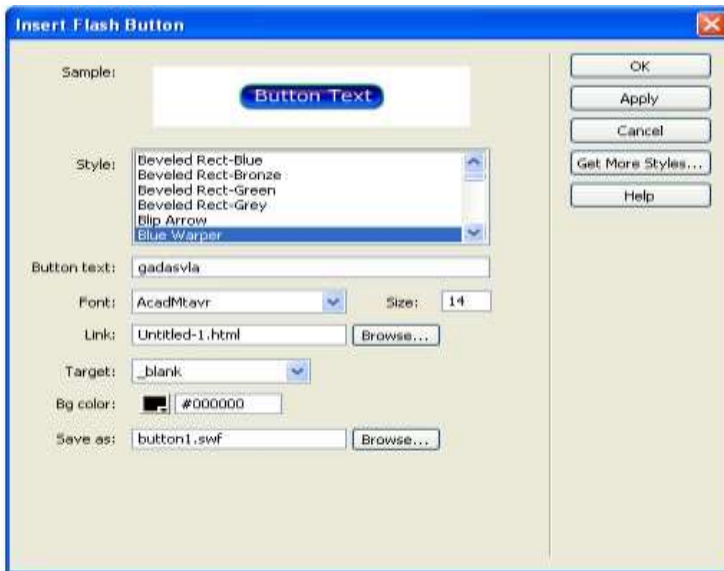
დავალება

1. web ფურცელზე ჩასვით ცხრილი 3 სვეტით და 3 სტრიქონით, სადაც ცხრილის სიგანე იქნება 100%;

2. ცხრილის უჯრაში მოათავსეთ სურათი;
3. ცხრილის უჯრებში ჩასვით ტექსტი.

2.6. Flash ტექსტის და Flash ღილაკების ჩასმა

ძირითადი მენიუს **Insert** პუნქტში შევასრულოთ ბრძანება **media > flash button** ან ობიექტების ჩასმის ხელსაწყოების ფანჯარაში დავაჭიროთ ღილაკს **flash button** გამოსულ ფანჯარაში:



სურ. 2.8.

Style ვირჩევთ სასურველ Flash დილაკს.

Button text შეგვყავს ის ტექსტი, რომელიც აისახება დილაკზე.

Font ვირჩევთ შრიფტის სტილს.

Link ველში შევიტანოთ ფაილის სახელი, რომელზეც უნდა მოხდეს მიმართვა.

Target ველში ვირჩევთ ფანჯარას, რომელშიც უნდა გაიხსნას ჰიპერმიმართვით არჩეული ფაილი.

ასევე შესაძლებელია **Flash Text** ჩასმა, ხელსაწყოების ფანჯარაში დავაჭიროთ დილაკს Flash Text გამოსულ ფანჯარაში:



სურ. 2.9.

ავკრიფოთ ტექსტი და შევუცვალოთ მას შრიფტი, შრიფტის ზომა, მივცეთ ფერი, link ჩანართში მივცეთ მას ლინკი (კავშირი). არჩევის შემდეგ, ვადასტურებთ ბრძანებას ok ღილაკით.

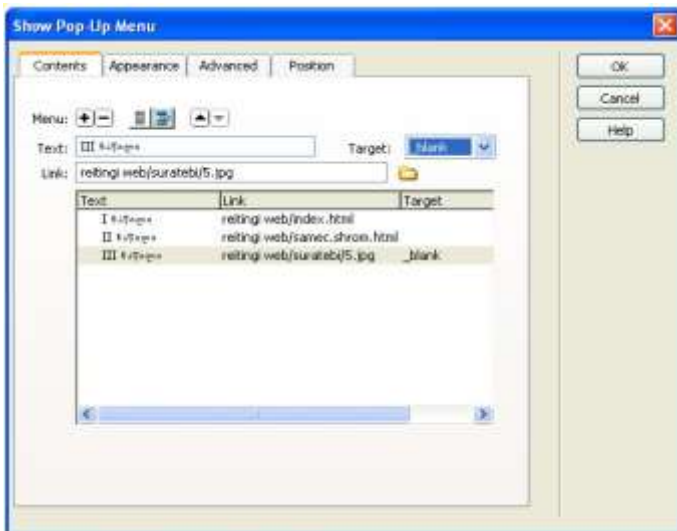
დავალება

1. შექმენით ორი web გვერდი;
2. ერთ გვერდზე ჩასვით flash button, შეარჩიეთ სასურველი ღილაკი, მიეცით მას ფერი, შრიფტი, და ლინკით მეორე გვერდზე გადადით;
3. მეორე გვერდზე ჩასვით Flash Text, მიეცით მას ფერი, შრიფტი, შრიფტის ზომა და ა.შ. ლინკით გადადით პირველ გვერდზე.

2.7. ჩამოშლადი მენიუს შექმნა

ჩამოშლადი მენიუს შესაქმნელად ვირჩევთ ბრძანებას Tag Inspector > behaviors, შემდეგ “+” ღილაკზე დაჭერით გამოდის კონტექსტური მენიუ,

სადაც ვირჩევთ Show Pop-up menu ბრძანებას, გამოსულ ფანჯარაში:



სურ. 2.10.

content ჩანართში Menu “+”, “-” ვუმატებთ ან ვაკლებთ ჩამონათვალს. Link – ეს არის ჰიპერმიმართვა (კავშირი).

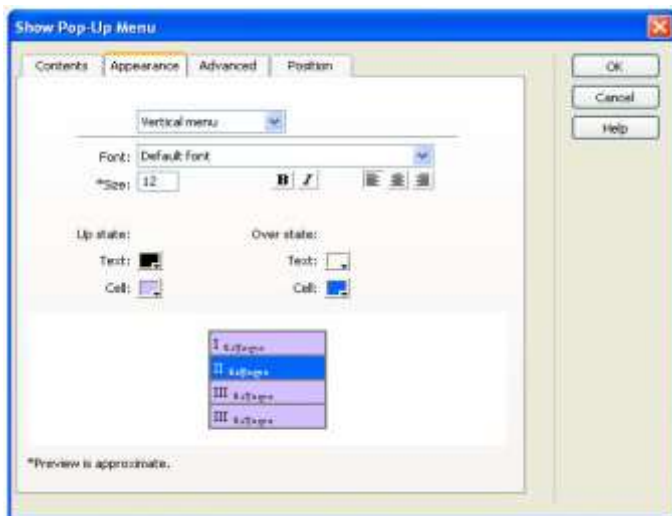
Target- ველში ვირჩევთ ფანჯარას, რომელშიც უნდა გაიხსნას ჰიპერმიმართვით არჩეული ფაილი. ველის მარჯვნივ მდგომ ისრიან ღილაკზე დაჭერით გამოსული ჩამონათვლიდან შეგვიძლია ავირჩიოთ

შემდეგი დარეზერვებული სახელები: Blank – მიმართვის ფაილი გამოვა ახალ ფანჯარაში;

Parent – მიმართვის ფაილი გამოვა frameset- ში ან ბრაუზერის მთლიან ფანჯარაში;

Self – მიმართვის ფაილი გამოვა იმავე ფანჯარაში ან ფრეიმში, რომელშიც ჰიპერმიმართვაა მოთავსებული. ეს პარამეტრი სტანდარტულად აირჩევა;

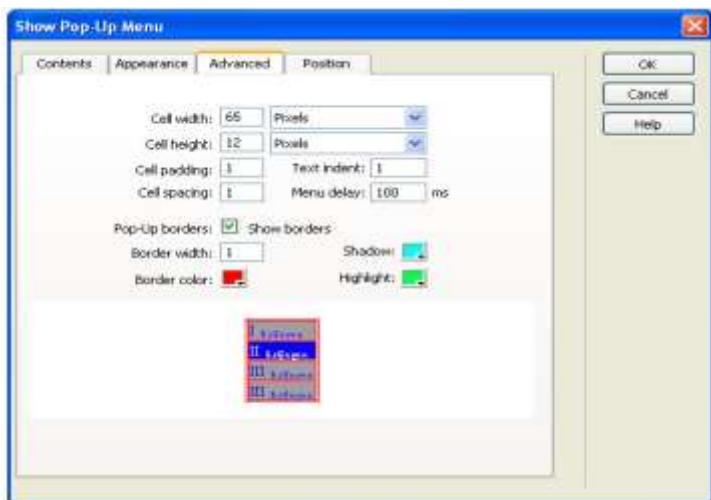
Top – მიმართვის ფაილი გამოვა ბრაუზერის მთლიან ფანჯარაში.



სურ. 2.11.

Appearance ჩანართში ვირჩევთ ვერტიკალურ ან ჰორიზონტალურ მენიუს, აგრეთვე შრიფტს, შრიფტის ზომას, ტექსტს ვასწორებთ მარჯვნივ მარცხნივ ან შუაში და ტექსტს ვაძლევთ ფერს.

Advanced ჩანართში ვირჩევთ:



სურ. 2.12.

Cell width - ჩამოშლადი მენიუს სიგანე.

Cell height - ჩამოშლადი მენიუს სიმაღლე.

Cell padding - უჯრაში მოთავსებული მონაცემებისა და უჯრის გვერდებს შორის დაშორება.

Cell spacing - ცხრილის უჯრებს შორის დაშორება.

Border – ჩარჩოს სისქე პიქსელებით

Border color – ჩარჩოს ფერი.

Shadow – ჩრდილი.

Menu delay – ჩამოშლადი მენიუს დაყოვნების დრო.

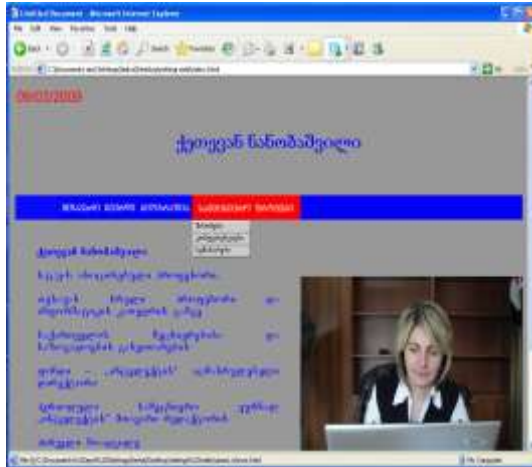
Position ბრძანებით შესაძლებელია ჩამოშლადი მენიუს დაყენება ობიექტის დაბლა, დაბლა და მარჯვნივ, ობიექტის მაღლა, მაღლა და მარჯვნივ.

დავალება

1. web გვერდზე ჩასვით თქვენს მიერ შექმნილი ლილაკი.


2. ჩასმულ web გვერდზე მოათავსეთ ჩამოშლადი მენიუ.

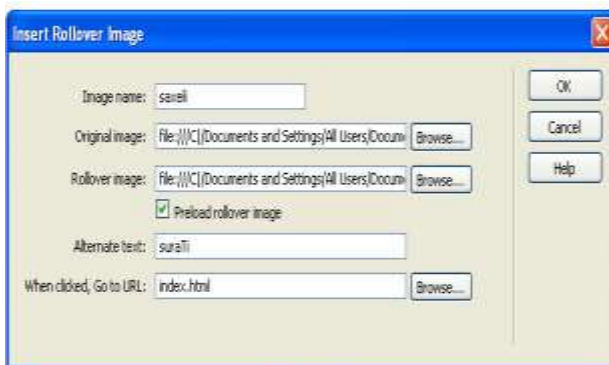
3. შექმენით მსგავსი web გვერდი ჩამოშლადი მენიუთი:



სურ. 2.13.

2.8. Rollover Image -ის მოთავსება

Rollover Image  მასზე მაუსის დაწკაპუნებით გამოვა ფანჯარა:



სურ. 2.14.

აქ შესაძლებელია ცვალებადი გამოსახულების ჩასმა (ცვალებდია გამოსახულება, რომელიც სახეს იცვლის, როდესაც მასზე მაუსის ისარს მივიტანთ).

Image Name - შეგვაქვს ღილაკის სურათის სახელი.

Original Image - ვირჩევთ სურათს, ამ სურათს ექნება პირველადი სახე.

Rollover Image ვირჩევთ გამოსახულებას, მასზე მაუსის კურსორის მოთავსებისას ობიექტი შეიცვლება.

Preload rollover image - ჩართული უნდა იყოს „ალამი“.

Alternate Text - შეგვაქვს ტექსტი, რომელიც ბრაუზერში ღილაკზე მაუსის კურსორის მოთავსებისას გამოდის.

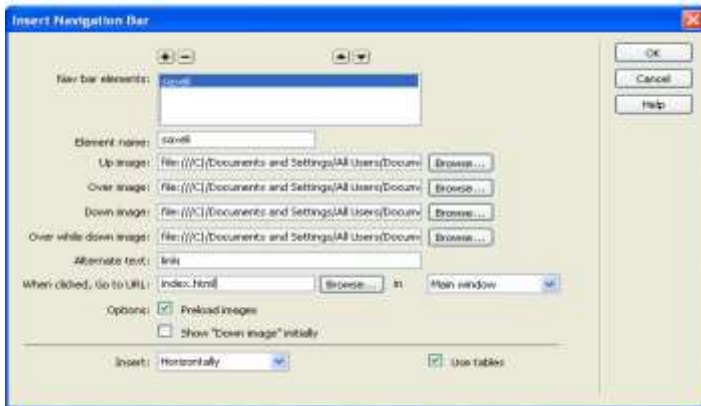
then clicked, go to URL – ამ გრაფაში შეგვაქვს მიმართვა, რომელზეც უნდა მოხდეს გადასვლა, როდესაც ღილაკს დავაჭერთ.

დავალება

1. შექმენით web გვერდი;
2. ჩასვით Rollover Image;
3. ლინკით გადადით მეორე გვერდზე.

2.9. სანავიგაციო ღილაკების ჩასმა

ძირითადი მენიუს **Insert** პუნქტში შევასრულოთ **Navigation Bar** ბრძანება ან ობიექტების ჩასმის ხელსაწყოების ფანჯარაში დავაჭიროთ **Common** ჯგუფის **Navigation Bar** ღილაკს. ამის შემდეგ, გამოსულ ფანჯარში:



სურ. 2.15.

Navigation Bar “+” და “-” ღილაკებით ხდება სანავიგაციო ღილაკების დამატება ან წაშლა.

Element name - ღილაკის სახელი.

Up image - გამოსახულების ფაილის მითითება ღილაკის თავდაპირველი მდგომარეობისათვის.

Over image - ვუთითებთ გამოსახულების ფაილს ღილაკზე მაუსის კურსორის მოთავსებისას.

Down image - ვუთითებთ გამოსახულების ფაილს, დაჭერილი ღილაკის მდგომარეობისათვის.

Over while down image - ვუთითებთ გამოსახულების ფაილს, დაჭერილ ღილაკზე მაუსის კურსორის მოთავსებისას.

Alternate text - შეგვაქვს ტექსტი, რომელიც ღილაკზე მაუსის კურსორის მოთავსებისას გამოვა.

When clicked, go to URL - მიმართვა, რომელზეც მოხდება გადასვლა გრაფაში in უნდა მივუთითოთ, სად უნდა გაიხსნას.

Options preload image - უნდა იყოს ჩართული.

Show “down image” - თავდაპირველ მდგომარეობაში ღილაკის დაჭერით მდგომარეობაში ჩვენება.

Insert - ღილაკების პანელის სიის არჩევა ჰორიზონტალური ან ვერტიკალური.

Use tables - ცხრილის გამოყენება.

Tabular data - ცხრილური მონაცემების ჩასმა. გამოიყენება გარე ცხრილური მონაცემების ჩასამებად. მაგალითად, Excel-ის ფაილი. ძირითადი მენიუს insert პუნქტში ვასრულებთ table objects > Import tabular. გამოსულ ფანჯარაში:

date file – ცხრილური მონაცემების ფაილი.

delimiter – სვეტების გამოყოფის ტიპი სხვადასხვა ვარიანტებით, რომლებიც იმპორტირებულ ფაილშია გამოყენებული. txt გაფართოების ფაილებისათვის ამ ველში აირევა Tab.

Table width - ცხრილის სიგანე.

Cell padding - უჯრაში მოთავსებული მონაცემებისა და უჯრის გვერდებს შორის დაშორება.

Cell spacing - ცხრილის უჯრებს შორის დაშორება.

Format top row - ცხრილის სათაურის გამოყოფა ან არგამოყოფა პირველი სტრიქონის სახით.

Border - ცხრილის ჩარჩოს სისქე.

ცხრილური მონაცემების ჩასმის შემდეგ, ცხრილის თვისებათა ფანჯარაში: Table ID გრაფში მივუთითებთ ცხრილის სახელს, რომელიც გამოიყენება სკრიპტებისათვის (მითითება აუცილებელი არ არის).

დავალება

1. ჩასვით Navigation Bar;
2. ლინკით გადადით მეორე გვერდზე.

2.10. Email link ელექტრონული ფოსტის მისამართის

მოთავსება

ძირითადი მენიუს **Insert** პუნქტში შევასრულოთ **Email link** ბრძანება ან ობიექტების ჩასმის ხელსაწყოების ფანჯარაში დავაჭიროთ **Common**

ჯგუფის **Email link** ღილაკს. გამოსულ დიალოგურ ფანჯარაში:



სურ. 2.16.

Text - ველში შევიტანოთ Email-ის სახელი, რომელიც დოკუმენტში თავსდება. Email ველში შევიტანოთ Email-ის მისამართი.

დავალება

1. ჩასვით Email Link-ის ლინკის ღილაკი.

თქვენ მიერ შესწავლილი ლაბორატორიული სამუშაოს საშუალებით ააგეთ მსგავსი WEB-გვერდი:



სურ. 2.17.

2.11.Named Ancher - ნიშნულის მოთავსება

დოკუმენტის ცალკეულ ნაწილზე ნიშნულის მოსათავსებლად ძირითადი მენიუს **Insert** პუნქტში შევასრულოთ **Named Ancher** ბრძანება ან ობიექტების ჩასმის ხელსაწყოების ფანჯარაში დავაჭიროთ **Common** ჯგუფის **Named Ancher** ღილაკს. გამოსულ ფანჯარაში:



სურ. 2.18.

შევიტანოთ ნიშნულის სახელი, მაგალითად, `aaaa` და დავაჭიროთ OK ღილაკს.

ნიშნულთან დაკავშირების II ხერხი

1. მოვნიშნოთ დოკუმენტში ტექსტი ან გამოსახულება, რომლის დაკავშირება გვინდა ნიშნულთან;
2. ეკრანის ქვედა ნაწილში თვისებათა ფანჯრის Link ველში შევიტანოთ # ნიშანი და ნიშნულის სახელი, მაგალითად:

- თუ ნიშნული იმავე დოკუმენტშია, შეიტანეთ `#aaaa`;
- თუ ნიშნული სხვა დოკუმენტშია, შეიტანეთ `work.html#aaaa`.

ნიშნულთან დაკავშირების II ხერხი

1. გავხსნათ დოკუმენტი, რომელშიც ნიშნულია ჩასმული;
2. მოვნიშნოთ დოკუმენტში ტექსტი ან გამოსახულება, რომლის დაკავშირება გვინდა ნიშნულთან;
3. მაუსის კურსორი მოვათავსოთ ნიშნულზე, მაუსის მარცხენა ღილაკზე და Shift ღილაკზე დაჭერილი მდგომარეობით მაუსის კურსორი განვათავსოთ ნიშნულზე ან მაუსის კურსორი მოვათავსოთ თვისებათა ფანჯარაში Link ველის მარჯვნივ მდგომ ნიშნაკზე და მაუსის მარცხენა ღილაკზე დაჭერილ მდგომარეობით მაუსის კურსორი განვათავსოთ ნიშნულზე.

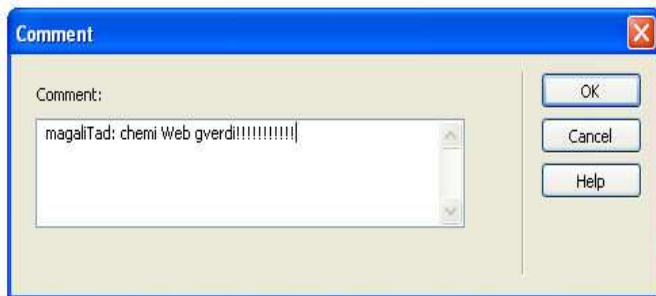
დავალება

1. web გვერდზე მოათავსეთ ნიშნული.

2.12. Comment - კომენტარის ჩასმა

კომენტარი არის განმარტებითი ტექსტი, რომელიც

შეიძლება ჩავსვათ HTML კოდში და ჩანს მთლიანად მხოლოდ HTML კოდში code რეჟიმში; Design რეჟიმში ჩანს მხოლოდ კომენტარის მარკერი. მენიუს ზოლში ვირჩევთ ბრძანებას Comment გამოსულ ფანჯარაში



სურ. 2.19.

შეგვაქვს ტექსტი და ვაჭერთ ok ღილაკს. ჩასმული მარკერი შეგვიძლია ვნახოთ მხოლოდ მაშინ, თუ შევალთ View > Visual Aids და ვირჩევთ Invisible Elements და Preference ავირჩევთ Invisible Elements > Comment.

JavaScript ან VBScript-ის ფუნქციების ნახვისათვის დოკუმენტს ვიპოვით code რეჟიმში, თავისუფალ ადგილზე ვაჭერთ მაუსის მარჯვენა ღილაკს, შემდეგ ვირჩევთ Functions და თუ დოკუმენტი შეიცავს

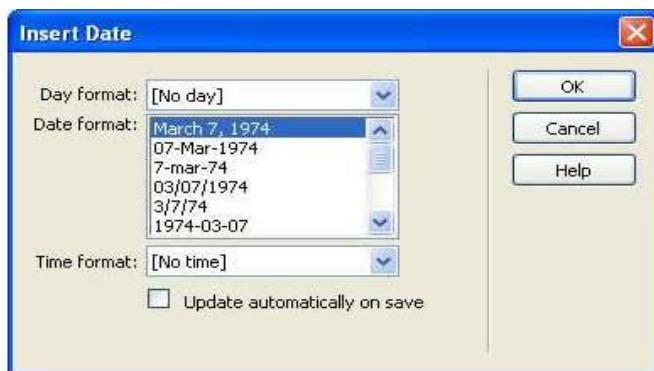
JavaScript ან VBScripts ფუნქციებს, გამოჩნდება ამ Functions ქვემენიუში.

დავალება

1. შექმენით მარტივი web გვერდი და HTML რედაქტორში ჩასვით კომენტარი.

2.13. Insert date თარიღის ჩასმა

მენიუს ზოლში ვირჩევთ ბრძანებას Insert date, ან ობიექტების ჩასმის ხელსაწყოების ფანჯარაში დავაჭიროთ **Common** ჯგუფის Date ლილავს, გამოსულ ფანჯარაში:



სურ. 2.20.

Day Format – ვირჩევთ დღეების ჩაწერის ფორმატს.

Date Format – ვირჩევთ თარიღის ჩაწერის ფორმატს.

Time Format – ვირჩევთ დროის ჩაწერის ფორმატს.

თუ ავირჩევთ update automatically on save, მაშინ დოკუმენტის ყოველი დამახსოვრებისას თარიღი განახლდება.

დავალება

1. web გვერდზე ჩასვით დრო და თარიღი. მოახდინეთ მისი განახლება.

2.14. ახალი css სტილის შემქმნა (css კასკადური სტილების ცხრილი cascading style sheets)

css სტილების ხელსაწყოების ფანჯრის გამოსატანად ძირითად მენიუს Windows პუნქტში ვასრულებთ css styles. ხელსაწყოების ფანჯარა გამოიყენება css სტილების შესაქმნელად, რედაქტირები-

სათვის და სტილების წასაშლელად, აგრეთვე დოკუმენტის სტილის დასანიშნად.

ახალი css სტილის შესაქმნელად css სტილის ხელსაწყოების ფანჯარაში ვაჭერთ New css rule ღილაკს ან ძირითად მენიუში Text > CSS Styles > New. ამის შემდეგ, გამოსულ დამატებით ფანჯარაში: Name გრაფში შეგვაქვს სახელი.

Selector Type – ვირჩევთ Class (შეგვიძლია დავუნიშნოთ ნებისმიერ ტეგს). Define in ვირჩევთ New style Sheet File, თუ გვინდა გარე სტილის შექმნა ან ვირჩევთ This Document only, თუ გვინდა მხოლოდ დოკუმენტისათვის ამ სტილის დანიშვნა. ბოლოს ვაჭერთ ok.

გამოსულ დამატებით ფანჯარაში შეგვაქვს ჩვენთვის სასურველი სტილის პარამეტრები და ბოლოს ვაჭერთ ok.

სტილის დანიშვნა ობიექტისათვის

დოკუმენტში მოვნიშნავთ ტექსტს, რომლისთვისაც გვინდა სტილის დანიშვნა, ამის შემდეგ:

- თვისებათა ფანჯარაში Style ავირჩევთ css სტილის სახელს ან
- CSS სტილის ხელსაწყოების ფანჯარაში დავაჭერთ მაუსის ღილაკს შესაბამის CSS სტილის სახელს და კონკრეტულ მენიუში ვასრულებთ Apply-ს.

2.15. ფორმების შექმნა და გაუქმება

ფორმები წარმოადგენს Web გვერდების სპეციფიკურ ობიექტებს, რომელთა დანიშნულებაა Web გვერდებიდან Web სერვერებზე გარკვეული ინფორმაციის გადაცემა. ფორმის ობიექტებს მიეკუთვნება ტექსტური ველები, გადამრთველები, ალმები, განშლადი სიები და სხვ.

ტექსტური ველის მოთავსება. Text Filed ტექსტური ინფორმაციის შესატანად გამოიყენება. ტექსტური ველის თვისებათა ფანჯარაში:



სურ. 2.21.

Text Filed – ნაჩვენები ფორმის სახელი (Text Filed).

Type > Single Line – ჩვეულებრივი სტრიქონიანი ველი.

Multi Line – მრავალსტრიქონიანი ტექსტური ველი.

Password – პაროლის შეტანისათვის, ამ შემთხვევაში ტექსტურ ველში შეტანის დროს სიმბოლოები გამოისახება X-ით

Char width – მიეთითება მაქსიმალური სიმბოლოების რაოდენობა, რომლებიც ტექსტურ ველში შეიძლება გამოჩნდეს. ტექსტურ ველში

შეიძლება შევიტანოთ უფრო მეტი სიმბოლო, რომელთა რაოდენობა აღემატება ამ პარამეტერს და მთლიანად გადაეცემა Web სერვერს. ტექსტურ ველში მაქსიმალური დასაშვებ სიმბოლოთა შეტანა განისაზღვრება მომდევნო პარამეტრით.

Max chars – სიმბოლოთა მაქსიმალური რაოდენობა, რომელთა შეტანა შეგვიძლია ტექსტურ ველში. გამოიყენება საფოსტო ინდექსის ან პაროლისათვის მაქსიმალური დასაშვები რაოდენობის სიმბოლოების მისათითებლად. მაგალითად, 10 სიმბოლოზე მეტი რომ არ შევიტანოთ. თუ მომხმარებელი მაქსიმალურ სიმბოლოზე მეტს აკრეფს, ამ დროს გაისმის ბგერითი სიგნალი.

Init val – ტექსტურ ველში თავდაპირველად მოსათავსებელი ტექსტის შეტანა, რომელიც არ არის აუცილებელი.

Wrap – თუ არჩეულია ოფფ, მაშინ ტექსტის აკრეფის დროს ტექსტის ველის სიგანეზე უფრო მეტი სიმბოლოთა აკრეფისას, მომდევნო სტრიქონზე

გადასვლა არ მოხდება და გაგრძელდება იმავე სტრიქონში.

თუ არჩეულია Default – ამ შემთხვევაში ტექსტურ ველზე უფრო მეტი სიმბოლოთა რაოდენობაა. აკრეფისას შემთხვევაში მოხდება მომდევნო სტრიქონზე გადასვლა.

Virtual – იგივეა, რაც Default იმ განსხვავებით, რომ Web-სერვერს გადაეცემა ერთი სტრიქონი.

Physical – იგივე, როგორც Default და იმავე სახით გადაეცემა.

2.16. Hidden Filed – ფარული ველის ჩასმა

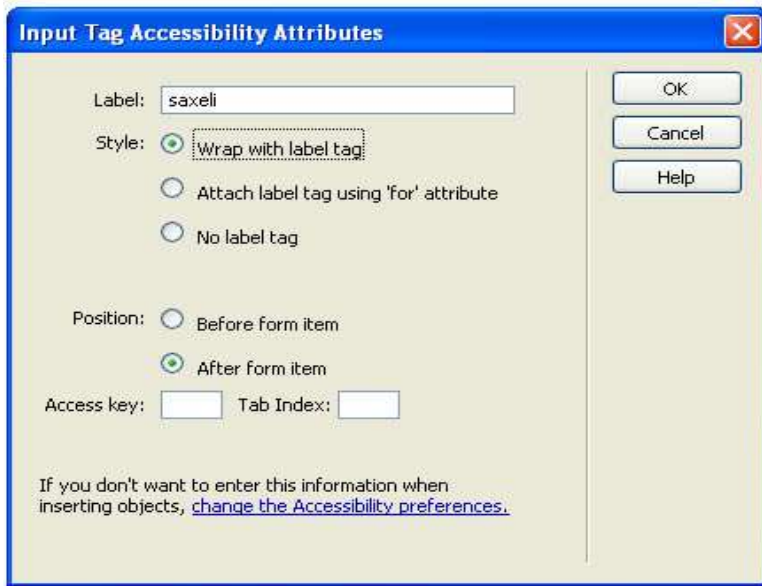
ეს ველი გამოიყენება იმ ინფორმაციის დამახსოვრებისა და შემდეგ Web სერვერზე გადასატანად, რომელიც მომხმარებლისთვის დაფარულია. **Insert** პუნქტში შევასრულოთ **Froms** > Hidden Filed ბრძანება ან ობიექტების ჩასმის ხელსაწყოების ფანჯარაში დავაჭიროთ **Froms** ჯგუფის Hidden Filed ღილაკს. ამ

ველის თვისებათა ფანჯარაში Value გრაფში შეგვაქვს ტექსტი, რომელიც მომხმარებლისთვის დაფარულია. ამის შემდეგ გამოსულ ფანჯარაში: მრავალსტრიქონიანი ტექსტური ველის შექმნა Textarea.

გამოიყენება მრავალსტრიქონიანი ჩვეულებრივი ან პაროლის შემცველი ტექსტების შესაქმნელად, იგი Text Filed ტექსტური ველის ობიექტის მრავალსტრიქონიანი ტიპის კერძო შემთხვევას წარმოადგენს.

2.17. ალმის Checkbox ჩასმა

უმთავრესად გამოიყენება ჯგუფურად და ემსახურება შესაძლო ვარიანტებიდან (მონაცემებიდან) რამდენიმეს ამორჩევას. Checkbox თვისებათა ფანჯარაში:



სურ. 2.22.

Checkbox value – შეგვსძლევს მნიშვნელობა, რომელიც ალმის ამორჩევისას გადაეცემა Web-სერვერს.

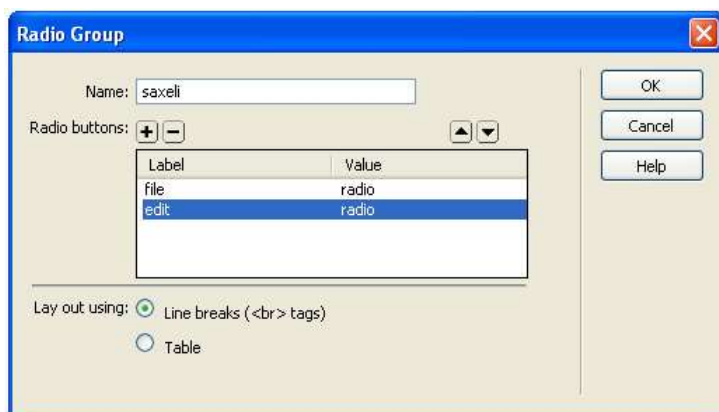
Initial state ალმის თავდაპირველ Checked ან Unchecked არჩეულ მდგომარეობაში.

გადამრთველების Radio Button ჩასმა. გადამრთველები ჯგუფურად გამოიყენება და მისი დანიშნულებაა რამდენიმე შესაძლო ვარიანტებიდან (მონაცემებიდან) ერთ-ერთის ამორჩევა. Radio Button

თვისებათა ფანჯარა ზევით განხილულ Checkbox ობიექტის ანალოგიურია.

2.18. გადამრთველების ჯგუფის Radio Group ჩასმა

Radio Group ქმნის გადამრთველების ჯგუფს, გამოსულ დამატებით ფანჯარაში:



სურ. 2.23.

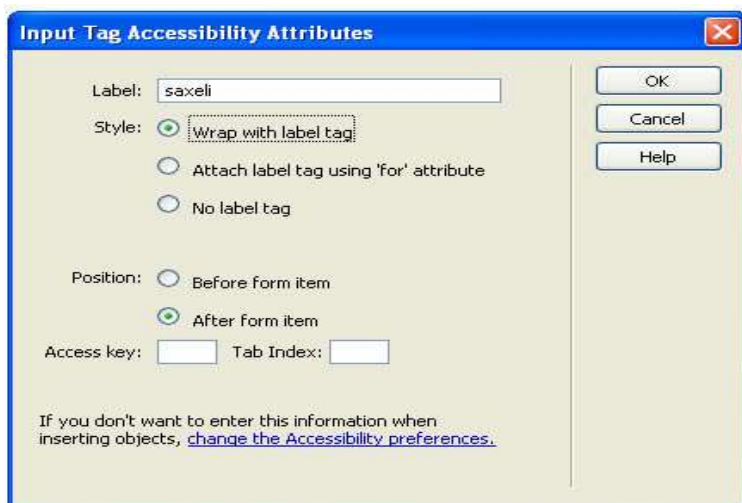
Name ველში შეგვაქვს გადამრთველების ჯგუფის სახელი, “+” გადამრთველი დამატება ჯგუფში, “-“ გადამრთველის მოხსნა.

Lay out using - თუ ავირჩევთ Line break, გადამრთველები განლაგდება ცალკეულ სტრიქონში, ხოლო Table

არჩევისას გადამრთველები მოთავსდება ერთსვეტიან ცხრილში. გადამრთველი იქნება მარცხნივ, ხოლო წარწერა მარჯვნივ.

2.19. განშლადი მენიუს ტიპის List Menu სიის ჩასმა

განშლადი სიიდან შესაძლებელია რამდენიმე სიის ელემენტის, ხოლო მენიუს ტიპის სიიდან ერთი სიის ელემენტის არჩევა, ამ ობიექტის თვისებათა ფანჯარაში:



სურ. 2.24.

Select მენიუს სახელი

Type ვირჩევთ სიის ტიპს.

List Values შეგვავსებს სიის ელემენტები (მნიშვნელობები). ამ დროს გამოდის დამატებითი ფანჯარა, რომელშიც “+” და “-” ღილაკებით ხდება სიაში ელემენტის დამატება და მოხსნა (წაშლა). თითოეული ელემენტისათვის შეგვყავს სახელი (წარწერა) და მნიშვნელობა, რომელიც გადაეცემა ჭებ სერვერს. ამ ელემენტის არჩევის შემთხვევაში, თითოეული ელემენტის სახელს ვკრეფთ Item label სვეტში. შემდეგ ვაჭერთ Value-სვეტს შესაბამის სტრიქონში და ვკრეფთ მნიშვნელობას.

Initially selected ვირჩევთ სიის ელემენტს, რომელიც თავიდანვე იქნება არჩეული.


Height (List რეჟიმისათვის) ვუთითებთ ელემენტის რაოდენობას, რომელიც თავდაპირველად იქნება

გამოტანილი სიაში, დანარჩენების ჩვენებისთვის გამოიყენება გადაფურცლის ზოლები.

Selections თუ მონიშნულია, შეიძლება ერთზე მეტი ელემენტის არჩევა.

2.20. Layer ფენის ჩასმა

უკვე განვიხილეთ ცხრილი, სადაც მისი უჯრების საშუალებით ჭებ-გვერდზე ჩავსვით ტექსტი და გრაფიკული ობიექტები. ცხრილის გარდა, ასევე შეგვიძლია გამოვიყენოთ **Layer** ბრძანება, რის საშუალებითაც შევძლებთ Web-გვერდზე ჩარჩოში მოვათავსოთ ობიექტი.

ამისათვის შევასრულოთ ბრძანება სტანდარტული მენიუს Layout ჩანართში. ავირჩიოთ ბრძანება Draw Layer  არჩევის შემდეგ ვდგებით Web-ფურცელზე და ვხაზავთ ლეიერს. თუ გვინდა მისი ზომების გაზრდა ან დაპატარავება, ვცვლით Properties ჩანართში. განვიხილოთ მაგალითი: Layer-ში

სურათის ჩასმა. ვდგებით ლეიერში და ვასრულებთ ბრძანებას Insert > Image. ვირჩევთ სასურველ სურათს და ვადასტურებთ ok.



სურ. 2.25.

დავალება

1. Web გვერდზე ჩასვით ორი **Layer**;
2. ერთში მოათავსეთ სურათი;
3. მეორეში - ცხრილი.

2.21. საიტის ატვირთვა სერვერზე

როდესაც შევემით web გვერდს (შაბლონს), ამის შემდეგ იგი განთავსდება სერვერზე.

განვიხილოთ web გვერდის ატვირთვა სერვერზე Macromedia dreamweaver-ს მეშვეობით. როდესაც გვინდა დავარეგისტროთ საიტი, შევდივართ სტანდარტული მენიუს სიტე ჩანართში, ვირჩევთ მანაგე **site** პუნქტს, გამოსულ ფანჯარაში ახალს New.



სურ. 2.26.

Site Definition ფანჯარაში ვირჩევთ ჩანართ Advanced-ს. გამოსულ ფანჯარაში Local Info ვავსებთ ბრძანებებს, სადაც

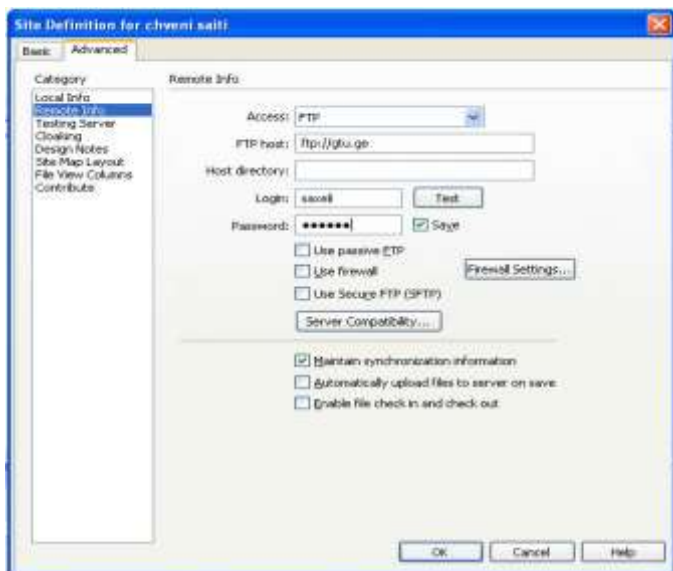
Site name – საიტის სახელია;

Local root folder – ვუთითებთ იმ ფაილს, რომლის ატვირთვაც გვინდა,

Link relative to – გადამრთველს ვაყენებთ Document-ზე,

HTTP address – საიტის Web გვერდის მისამართია.

ამ ფანჯრის შევსების შემდეგ გადავდივართ ჩანართზე Remote Info და გამოსულ ფანჯარაში



სურ. 2.27.

Access ჩამონათვალში ავირჩიოთ FTP, სადაც:

FTP host – შეგვყავს FTP მისამართი,

Login – FTP სახელი,

Password – FTP პაროლი.

გადამრთველს ვაყენებთ Use passive FTP. ამ მონაცემების შეტანის შემდეგ დავაჭერთ ლილაკს Test, დატესტვის შემდეგ კი ბრძანებას ვადასტურებთ ok ლილაკით.

დავალება

1. თქვენ მიერ გაკეთებული web გვერდი ატვირთეთ სერვერზე;
2. განაახლეთ ატვირთული ფაილი;
3. წაშალეთ სერვერიდან ატვირთული ფაილი.

III ტაში

გამოყენებითი ვებ პროგრამები (ვებ აპლიკაციები)

3.1. ზოგადი ინფორმაცია გამოყენებითი ვებ

პროგრამების შესახებ

გამოყენებითი web პროგრამა (აპლიკაცია) წარმოადგენს ვებ საიტს, რომელზეც განთავსებულია ნაწილობრივ ან მთლიანად დაუსრულებელი შიგთავსის მქონე ვებ გვერდები. მათი შიგთავსის საბოლოოდ ფორმირება ხდება მხოლოდ მას შემდეგ, რაც საიტის დამთვალიერებელი მოითხოვს საიტს ვებ სერვერიდან. იმის გამო, რომ ამგვარი ვებ გვერდების საბოლოო შიგთავსი დამოკიდებულია მოთხოვნაზე, რომელიც მომხმარებლის მიერ განხორციელებულ ქმედებებზეა დაფუძნებული, მათ დინამიკურ web გვერდებს უწოდებენ.

აღნიშნული ტიპის ვებ გვერდების გამოყენების სპექტრი ძალზე ფართოა. ქვემოთ გამოყენების ზოგად საკითხებთან ერთად განხილული იქნება მათი მაგალითებიც.

გამოყენებითი ვებ პროგრამის დანიშნულება

გამოყენებითი web პროგრამების დანიშნულება მდგომარეობს, როგორც მათი მომხმარებლების, ისე დამპროექტებლების გარკვეული მოთხოვნილებების უზრუნველყოფაში:

-მომხმარებლებისათვის გამოყენებითი ვებ პროგრამები უზრუნველყოფს საჭირო ინფორმაციის სწრაფად და მარტივად მოძიების შესაძლებლობებს დიდი მოცულობის ინფორმაციის მატარებელ ვებ საიტებზე, ასევე მარტივი ხერხებით უზრუნველყოფს საიტების შიგთავსის მოწესრიგების და მათზე გადაადგილების შესაძლებლობებს.

- გამოყენებითი ვებ პროგრამები უზრუნველყოფენ საიტის მომხმარებლებისაგან მიღებული მონაცემების შეგროვების, შენახვისა და გაანალიზების შესაძლებლობებს;

ადრე, დიდი ხნის განმავლობაში იყენებდნენ მეთოდს, რომლის დროსაც HTML-ფორმებში შეტანილი მონაცემები დასამუშავებლად ეგზავნებოდათ CGI-

გამოყენებით პროგრამებს ან სპეციალურად დანიშნულ მუშაკებს ელექტრონული ფოსტის შეტყობინებების სახით.

- გამოყენებითი ვებპროგრამები, მონაცემთა ბაზებში მონაცემების შენახვის, მათგან მონაცემების მიღებისა და მიღებული მონაცემების საფუძველზე, შემდგომი ანალიზისათვის, ანგარიშების ფორმირების საშუალებას იძლევა. მაგალითის სახით შეიძლება დავასახელოთ ბანკების ინტერაქტიული გვერდები, სასაქონლო მარაგების კონტროლთან და მართვასთან, სოციოლოგიურ კვლევებთან და გამოკითხვებთან დაკავშირებული გვერდები, აგრეთვე მომხმარებლებთან უკუკავშირებისათვის განკუთვნილი ფორმები.

- გამოყენებითი ვებ პროგრამების საშუალებით შესაძლებელია პერიოდულად ცვალებადი შიგთავსის მქონე ვებ საიტების განახლება.

- გამოყენებითი ვებ პროგრამა ათავისუფლებს ვებ დიზაინერს საიტის HTML-გვერდების მუდმივ განახლებასთან დაკავშირებული რუტინული

სამუშაოებისაგან. შიგთავსის მომწოდებლები, მაგალითად, საინფორმაციო წყაროების რედაქტორები, პასუხისმგებელი არიან უკანასკნელი სიახლეების მოწოდებაზე, ხოლო გამოყენებითი ვებ პროგრამა ახორციელებს საიტის ავტომატური განახლების კონტროლს. მაგალითის სახით შეიძლება მოვიყენოთ ჟურნალ «The Economist»-ის ვებ ვერსია (www.economist.com) და უკანასკნელი ამბების სამსახური CNN (www.cnn.com).

გამოყენებითი ვებ პროგრამის მაგალითი

ვთქვათ, ვინმე, კონკრეტული პიროვნება, მუშაობს ვებ დიზაინერად და უკვე დიდი ხანია იყენებს Dreamweaver-ს თავის საქმიანობაში. მის მოვალეობაში შედის შიგა კორპორატიული საიტისა და საშუალო ზომის კომპანიის (1000-მდე თანამშრომელი) ინტერნეტ საიტის მუშაობის უზრუნველყოფა. ვებ დიზაინერს მიმართავს კადრების მენეჯერი თავისი პრობლემით. ის კურირებს თანამშრომელთა გაჯანსაღების პროგრამას. ამ პროგრამის ფარგლებში თანამშრომლებს ერიცხებათ სპეციალური ქულები

ყოველ კილომეტრზე, რომელსაც ისინი გაივლიან ფეხით, სირბილით ან ველოსიპედით. თვის ბოლოს ყოველი თანამშრომელი კადრების მენეჯერს ელექტრონული ფოსტით უგზავნის ანგარიშს მთლიანად გავლილი მანძილის მითითებით. ამის შემდეგ მენეჯერი მიღებული ელექტრონული შეტყობინებების საფუძველზე აჯილდოვებს თანამშრომლებს გარკვეული ფულადი პრემიებით დაგროვებული ქულების შესაბამისად.

პრობლემა მდგომარეობს იმაში, რომ გაჯანსაღების პროგრამა ძალზე პოპულარული გახდა. მოცემული მომენტისათვის მონაწილეთა რაოდენობა ისე გაიზარდა, რომ თვის ბოლოს კადრების მენეჯერი ვერ ასწრებს მასთან შემოსული მთელი ფოსტის დამუშავებას. ამიტომაც, მან მიმართა ვებ დიზაინერს კითხვით: არის თუ, არა შესაძლებელი ამ პრობლემის გადაწყვეტა ვებ ტექნოლოგიების გამოყენებით.

საპასუხოდ დიზაინერმა შესთავაზა გამოყენებითი ვებპროგრამის შექმნა, რომელიც შეძლებდა შემდეგი ამოცანების გადაწყვეტას:

- მომხმარებლების მიერ თავიანთი სპორტული მიღწევების შესახებ ინფორმაციის შეტანას შესაბამის ვებ გვერდზე მარტივი HTML-ფორმების საშუალებით;
- მიღებული მონაცემების შენახვას მონაცემთა ბაზაში;
- ქულების დარიცხვას მიღებული მონაცემების საფუძველზე;
- ყოველი თანამშრომლისათვის თავისი შედეგების შესახებ ინფორმაციის მიღების შესაძლებლობას;
- კადრების მენეჯერისათვის შემაჯამებელი შედეგების მიღების შესაძლებლობას ყოველი თვის ბოლოს;

უმოკლეს დროში დიზაინერმა შექმნა და გაუშვა მოთხოვნილი გამოყენებითი პროგრამა, ვინაიდან Dreamweaver-ს ჰქონდა ყველა აუცილებელი ინსტრუმენტი ასეთი პროგრამების სწრაფად და მარტივად შესაქმნელად.

გამოყენებითი ვებპროგრამების მუშაობის პრინციპები

ნებისმიერი გამოყენებითი ვებ პროგრამა წარმოადგენს სტატიკური და დინამიკური ვებ გვერდების ერთობლიობას. სტატიკური ვებ გვერდი

წარმოადგენს ვებ გვერდს, რომელიც მომხმარებლის წინაშე ყოველთვის წარდგება უცვლელი სახით. ვებ სერვერი აგზავნის ვებ გვერდს ვებ ბრაუზერის მოთხოვნის საპასუხოდ ყოველგვარი ცვლილებების გარეშე. აღნიშნულის საპირისპიროდ, დინამიკურ ვებ გვერდში სერვერს შეაქვს ცვლილებები, ბრაუზერისათვის მისი გაგზავნის წინ. იმის გამო, რომ გაგზავნის წინ ვებ გვერდი ცვლილებას განიცდის, მას დინამიკურს უწოდებენ.

მაგალითად, შესაძლებელია შეიქმნას ვებ გვერდი რომელზეც ასახული იქნება გაჯანსაღების პროგრამის შედეგები. ამასთან, გარკვეული ინფორმაცია (მაგალითად კონკრეტული თანამშრომლის სახელი და მისი შედეგები) განისაზღვრება ამ თანამშრომლის მიერ ვებ გვერდის მოთხოვნის მომენტში.

სტატიკური ვებ გვერდების დამუშავების თავისებურებები

სტატიკური ვებ საიტი შეიცავს შესაბამისი HTML-გვერდების და ფაილების ანაკრეფს, რომლებიც

განთავსებულია იმავე კომპიუტერზე, რომელზეც დაყენებულია ვებ სერვერი.

ვებ სერვერი არის პროგრამული უზრუნველყოფა, რომელიც ვებ დამთვალიერებლის მოთხოვნების საპასუხოდ, წარუდგენს მას შესაბამის ვებ გვერდებს. ზოგადად, ვებ გვერდზე მოთხოვნა წარმოიქმნება ვებ გვერდზე არსებულ მიმართვაზე (მინიშნებაზე) დაწკაპუნებით, ბრაუზერში შესაბამისი ჩანართის არჩევით ან ბრაუზერის სამისამართე სტრიქონში შესაბამისი URL-მისამართის შეტანით.

სტატიკური ვებ გვერდის საბოლოო შიგთავსი განისაზღვრება საიტის შემქმნელის მიერ და უცვლელი რჩება გვერდის მოთხოვნის პროცესში. მაგალითად:

```
<html>
  <head>
    <title>Trio Motors Information Page</title>
  </head>
  <body>
    <h1>About Trio Motors
  </h1>
```

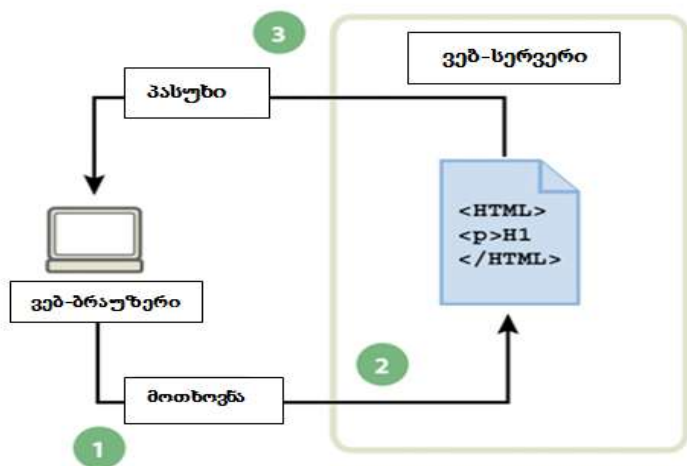
<p>Trio Motors is a leading automobile manufacturer.</p>

</body>

</html>

სერვერზე განთავსებამდე ხდება ვებგვერდის სრული HTML-კოდის შექმნა დამპროექტებლის მიერ. ვინაიდან ვებგვერდის HTML -კოდი არ იცვლება სერვერზე განთავსების შემდეგ, მას **სტატიკურ ვებგვერდს** უწოდებენ.

როდესაც ვებ სერვერი იღებს მოთხოვნას (განაცხადს) მისთვის სტატიკური ვებ გვერდის გაცემაზე, სერვერი ახორციელებს მოთხოვნის ანალიზს, პოულობს შესაბამის ვებგვერდს და უგზავნის მას ბრაუზერს, როგორც ეს ნაჩვენებია ქვემოთ ნახატზე:



1. ვებ ბრაუზერი მოითხოვს სტატიკურ ვებ გვერდს;
2. ვებ სერვერი პოულობს ვებ გვერდს;
3. ვებ სერვერი უგზავნის ვებ გვერდს მის მომთხოვნ ბრაუზერს.

გამოყენებითი ვებ საიტების შემთხვევაში, ვებ გვერდის კოდის ზოგიერთი უბანი არ არსებობს მომხმარებლის მიერ ვებ გვერდის მოთხოვნის მომენტამდე. არ არსებული კოდი ფორმირდება გარკვეული მექანიზმის საშუალებით და მხოლოდ ამის შემდეგ ხდება შესაძლებელი ვებ გვერდის ბრაუზერისათვის გაგზავნა.

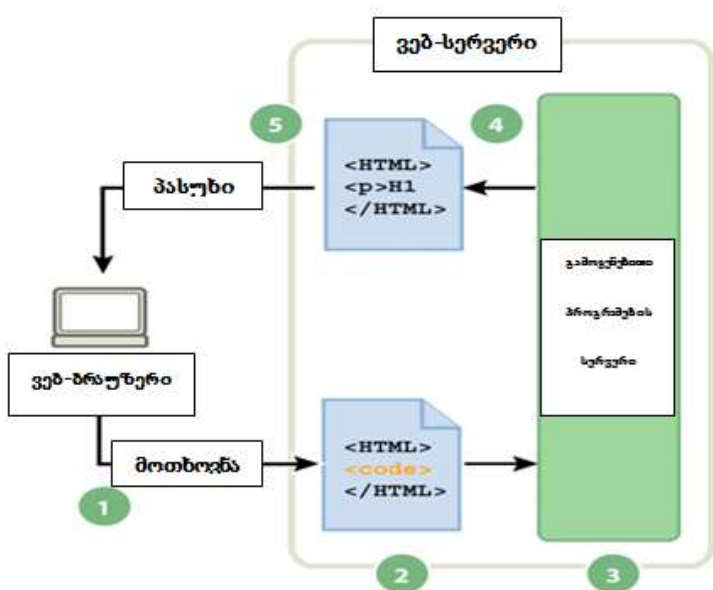
დინამიკური საიტების შემუშავების თავისებურება

ვებ სერვერი სტატიკური ვებ საიტის გადმოცემაზე მიღებული მოთხოვნის საპასუხოდ, უშუალოდ ბრაუზერს უგზავნის შესაბამის ვებ გვერდს. მაგრამ როდესაც ხდება დინამიკური ვებ გვერდის მოთხოვნა, ვებ სერვერის ქმედება არ არის ასეთივე ცალსახა. ამ შემთხვევაში სერვერი გადასცემს ვებ გვერდს სპეციალურ პროგრამას, რომელიც ახდენს საბოლოოდ ვებ გვერდის ფორმირებას. ასეთ პროგრამას ეწოდება **გამოყენებითი პროგრამების სერვერი**.

გამოყენებითი პროგრამების სერვერი ახდენს ვებ-გვერდზე განთავსებული კოდის წაკითხვას, რის შემდეგაც წაკითხული კოდის შესაბამისად აფორმირებს საბოლოო ვებ გვერდს, ხოლო შემდგომ ეტაპზე შლის აღნიშნულ კოდს გვერდიდან. ამ ოპერაციების შედეგად მიიღება სტატიკური ვებ გვერდი, რომელიც გადაეცემა ვებ სერვერს და ეს უკანასკნელი თავის მხრივ უგზავნის მას კლიენტის ბრაუზერს. ყველა გვერდი, რომელსაც იღებს ბრაუზერი, შეიცავს მხოლოდ HTML-კოდს. ქვემოთ მოცემულია აღნიშნული პროცესის სქემატური

გამოსახულება.

1. ვებ ბრაუზერი მოითხოვს დინამიკურ გვერდს;
2. ვებ სერვერი პოულობს ვებ გვერდს და გადასცემს მას გამოყენებითი პროგრამების სერვერს;
3. გამოყენებითი პროგრამების სერვერი ათვალიერებს ვებ გვერდს ინსტრუქციების არსებობაზე და ახორციელებს მის შექმნას;



4. გამოყენებითი პროგრამების სერვერი აბრუნებს მომზადებულ ვებ გვერდს ვებ სერვერზე;

5. ვებ სერვერი უგზავნის მომზადებულ ვებ გვერდს მის მომთხოვნ ბრაუზერს.

ვებ გვერდების კავშირის უზრუნველყოფა მონაცემთა ბაზასთან.

გამოყენებითი პროგრამების სერვერი ისეთი რესურსების გამოყენების საშუალებას იძლევა, როგორც არის **მონაცემთა ბაზები**. მაგალითად, დინამიკური ვებ გვერდი შეიძლება შეიცავდეს პროგრამულ ინსტრუქციებს გამოყენებითი პროგრამების სერვერისათვის, რომელთა თანახმადაც სერვერისათვის აუცილებელია გარკვეული მონაცემების მიღება მონაცემთა ბაზიდან და მათი განთავსება ვებ გვერდის HTML-კოდში.

შიგთავსის შენახვა მონაცემთა ბაზაში საშუალებას გვაძლევს ვებ საიტის გაფორმების პროცესი გამოვყოთ მომხმარებლებისათვის ხილული მისი შიგთავსისაგან. იმის ნაცვლად, რომ თითოეული ვებ გვერდი შეიქმნას ცალკეული HTML-ფაილის სახით, ხდება მხოლოდ ვებ

გვერდების შაბლონების დაწერა წარმოსადგენი ინფორმაციის ყოველი სახეობისათვის. შემდეგში შიგთავსით იტვირთება მონაცემთა ბაზაში, რის მერეც ვებ საიტი მოახდენს მის ამოღებას მომხმარებლების მოთხოვნების შესაბამისად. გარდა ამისა, შესაძლებელია ინფორმაციის განახლება მხოლოდ ერთ წყაროში და შემდეგ ამ ცვლილების დუბლირება მთლიან ვებ საიტზე, ცალკეული გვერდის ხელით რედაქტირების გარეშე. მონაცემთა ბაზაში ინფორმაციის შეტანის, განახლებისა და წაშლისათვის Dreamweaver-ს სათანადო ვებ ფორმების შექმნის საშუალება აქვს.

მონაცემთა ბაზიდან მონაცემების მიღებისათვის განკუთვნილ პროგრამულ ინსტრუქციას ეწოდება მონაცემთა ბაზისადმი **მოთხოვნა**. მოთხოვნა შედგება ძიების კრიტერიუმებისაგან, რომლებიც გამოისახება მონაცემთა ბაზების ენის საშუალებით, რომელსაც ეწოდება **SQL-Structured Query Language** (მოთხოვნების სტრუქტურირებული ენა).

SQL მოთხოვნის ტექსტი თავსდება ვებ გვერდების სცენარებში სერვერის მხარეს ან შესაბამის ტეგებში.

გამოყენებითი პროგრამების სერვერს არ შეუძლია უშუალოდ მიიღოს მონაცემები ბაზიდან, ვინაიდან მონაცემთა ბაზები იყენებენ მონაცემების შენახვის სპეციფიკურ ფორმატს, რის შედაგადაც ასეთი მონაცემების მიღების მცდელობა მოგვარონებს Microsoft Word-ის დოკუმენტის გახსნის მცდელობას Notepad რედაქტორისა და BBEdit-ის საშუალებით. ამიტომ მონაცემთა ბაზასთან მიერთებისათვის გამოყენებითი პროგრამების სერვერი იყენებს შუამავალს - **მონაცემთა ბაზების დრაივერს**. მონაცემთა ბაზების დრაივერი წარმოადგენს პროგრამულ მოდულს, რომლის მეშვეობითაც მყარდება ურთიერთქმედება გამოყენებითი პროგრამების სერვერსა და მონაცემთა ბაზას შორის.

იმის შემდეგ, რაც დრაივერი დაამყარებს კავშირს, სრულდება მოთხოვნა ბაზისადმი, რის შედეგადაც ხდება ჩანაწერების ანაკრების ფორმირება. **ჩანაწერების ანაკრები** წარმოადგენს მონაცემების სიმრავლეს, რომლებიც მიიღებიან მონაცემთა ბაზის ერთი ან რამდენიმე ცხრილიდან. ჩანაწერების ანაკრები

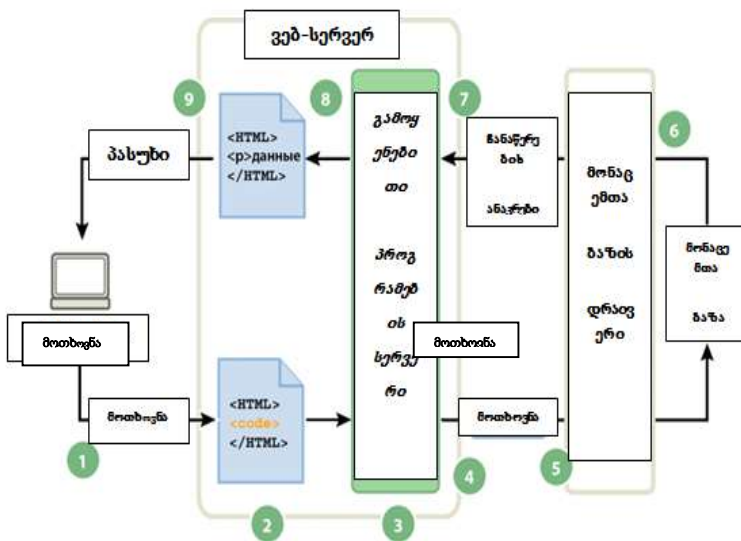
უბრუნდება გამოყენებითი პროგრამების სერვერს, რომელიც მიღებულ მონაცემებს იყენებს ვებგვერდის ფორმირებისათვის.

ქვემოთ მოცემულია მონაცემთა ბაზისადმი მარტივი მოთხოვნის მაგალითი SQL-ენაზე:

```
SELECT lastname, firstname, fitpoints  
FROM employees
```

ამ ინსტრუქციის საშუალებით ხდება ჩანაწერების ანაკრების ფორმირება სამი სვეტისაგან, რომლებიც შეიცავენ გვარს, სახელს და ყველა იმ თანამშრომლების მიერ დაგროვილი ქულების რაოდენობას, რომელთა შესახებაც ცნობები შენახულია მოცემულ ბაზაში.

შემდეგი მაგალითი ახდენს მონაცემთა ბაზისადმი მოთხოვნის შესრულების და მიღებული მონაცემების ბრაუზერისათვის დაბრუნების პროცესების დემონსტრირებას:



1. ვებ ბრაუზერი მოითხოვს დინამიკურ ვებ გვერდს;
2. ვებ სერვერი პოულობს ვებ გვერდს და გადასცემს მას გამოყენებითი პროგრამების სერვერს;

3. გამოყენებითი პროგრამების სერვერი ამოწმებს ვებ გვერდს ინსტრუქციების არსებობაზე და ახორციელებს მათ შესრულებისათვის გვერდის მომზადებას;
4. გამოყენებითი პროგრამების სერვერი უგზავნის მოთხოვნას მონაცემთა ბაზების დრაივერს;
5. დრაივერი ასრულებს მოთხოვნას მონაცემთა ბაზაში;
6. დრაივერს უბრუნდება ჩანაწერების ანაკრები;
7. დრაივერი უბრუნებს ჩანაწერების ანაკრებს გამოყენებითი პროგრამების საერვერს;
8. გამოყენებითი პროგრამების საერვერი სვამს მონაცემებს ვებ გვერდში და გადასცემს გვერდს ვებ სერვერს;
9. ვებ სერვერი უგზავნის მომზადებულ ვებ გვერდს მის მომთხოვნ ბრაუზერს.

გამოყენებით ვებ პროგრამაში გამოსაყენებლად გამოსადეგია ნებისმიერი მონაცემთა ბაზა იმ პირობით, რომ სერვერზე დაყენებულია მონაცემთა ბაზის შესაბამისი დრაივერი.

დაბალ ბიუჯეტთან გამოყენებითი პროგრამების შესაქმნელად შესაძლებელია მონაცემთა ფაილური

ბაზის გამოყენებაც, მაგალითად, Microsoft Access-ის საშუალებით შექმნილი მონაცემთა ბაზის. თუ კი იგეგმება საიმედო კორპორატიული გამოყენებითი პროგრამების შექმნა, მაშინ რეკომენდებულია სერვერული მონაცემთა ბაზის გამოყენება, მაგალითად, Microsoft SQL Server, Oracle 9i ან MySQL სერვერების ბაზაზე.

თუ კი მონაცემთა ბაზა და ვებ სერვერი განთავსებულია სხვადასხვა კომპიუტერზე, საჭირო იქნება სისტემებს შორის ჩქაროსნული უკუკავშირის უზრუნველყოფა, ვინაიდან ამაზე იქნება დამოკიდებული მთელი გამოყენებითი ვებ პროგრამის ეფექტურობა და სისწრაფე.

დინამიკური ვებ გვერდების შემუშავება

დინამიკური გვერდების შემუშავების პროცესი შედგება საბაზო HTML-კოდის დაწერისაგან და შემდგომში HTML - გვერდისათვის სერვერული სცენარების ან ტეგების შექმნისაგან, რომელთა საშუალებითაც ვებ გვერდი ხდება დინამიკური. თუ დავაკვირდებით საბოლოო კოდს დავინახავთ, რომ

სცენარების კოდები ჩაშენებულია ვებ გვერდის HTML-კოდში. შესაბამისად ასეთ ენებს უწოდებენ სცენარების საწერ ან სერვერულ ენებს.

მომდევნო მაგალითში გამოყენებულია სცენარების (სერვერული) ენა PHP (Personal Home Page):

```
1 <html>
2   <head>
3     <title>Trio Motors Information Page</title>
4   </head>
5   <body>
6     <h1>About Trio Motors</h1>
7     <p>Trio Motors isa leading automobile
8 manufacturer.</p>
9     <!-- embedded instructions start here --->
10    <cfset department="Sales">
11    <cfoutput>
12    <p>Be sure to visit our #department# page.</p>
13    </cfoutput>
14    <!-- embedded instructions end here --->
```

15 </body>

</html>

მოცემულ ვებ-გვერდში ჩაშენებული ინსტრუქციები ასრულებენ შემდეგ მოქმედებებს:

1. იქმნება ცვლადი სახელით „department“, რის შემდეგაც მას ენიჭება სიმბოლური მნიშვნელობა „Sales“.

2. მნიშვნელობა „Sales“ თავსდება HTML-კოდში.

გამოყენებითი პროგრამების სერვერი ვებ სერვერში აბრუნებს შემდეგ გვერდს:

1 <html>

2 <head>

3 <title>Trio Motors Information Page</title>

4 </head>

5 <body>

6 <h1>About Trio Motors</h1>

7 <p>Trio Motors isa leading automobile
8 manufacturer.</p>

9 <p>Be sure to visit our Sales page.</p>

10 </body>

11 </html>

ვებ-სერვერი აგზავნის ბრაუზერში ვებ გვერდს, რომელიც მას შემდეგნაირად გამოსახავს:

კომპანია Trio Motors-ის შესახებ

კომპანია Trio Motors წარმოადგენს ავტომობილების ერთ-ერთ წამყვან მწარმოებელს.

ეწვიეთ ჩვენი გაყიდვების განყოფილების ვებ გვერდს.

სცენარების ენის ან ტეგებზე დაფუძნებული ენის არჩევა დამოკიდებულია გამოყენებულ სერვერულ ტექნოლოგიებზე.

Dreamweaver-ს თვითონ შეუძლია შექმნას სერვერული სცენარები ან საჭირო ტეგები ვებ გვერდებისათვის, მაგრამ ამავე დროს თვით ვებ საიტის შემქმნელსაც შეუძლია დამოუკიდებლად დაწეროს საჭირო კოდი **Dreamweaver**-ის კოდირების გარემოს გამოყენებით.

3.2. გამოყენებით ვებ პროგრამებთან დაკავშირებული ტერმინოლოგია

ქვემოთ მოცემულია გამოყენებით ვებ პროგრამებთან დაკავშირებული, ხშირად გამოყენებადი ტერმინების განმარტებები.

გამოყენებითი პროგრამების სერვერი-არის პროგრამული უზრუნველყოფა, რომელიც გამოიყენება ვებ სერვერის მიერ სერვერული სცენარებისა და ტეგების შემცველი ვებ გვერდების დასამუშავებლად. ასეთი გვერდების მოთხოვნისას ვებ სერვერი ჯერ გადასცემს მათ გამოყენებითი პროგრამების სერვერს დასამუშავებლად, ხოლო შემდეგ უგზავნის კლიენტის ბრაუზერს.

გამოყენებითი პროგრამების ყველაზე გავრცელებული სერვერები მუშაობენ ColdFusion და PHP სერვერულ ენებთან.

მონაცემთა ბაზა-არის მონაცემთა ანაკრები, რომელიც ინახება ცხრილებში. ცხრილის ყოველი სტრიქონი წარმოადგენს ერთ ჩანაწერს, ხოლო ყოველი სვეტი-ჩანაწერის ველს;

მონაცემთა ბაზის დრაივერი - არის პროგრამული მოდული, რომლის საშუალებითაც მყარდება ურთიერთ ქმედება გამოყენებით ვებ პროგრამასა და მონაცემთა ბაზას შორის. მონაცემთა ბაზაში მონაცემები ინახება ყოველი ბაზისათვის სპეციფიკურ საკუთარ ფორმატში. მონაცემთა ბაზის დრაივერის საშუალებით გამოყენებითი ვებ პროგრამა იძენს ბაზაში შენახულ მონაცემებთან მუშაობის შესაძლებლობას;

მონაცემთა ბაზების მართვის სისტემა - არის პროგრამული უზრუნველყოფა, რომლის დანიშნულებაც მდგომარეობს მონაცემთა ბაზების შექმნასა და მართვაში. ფართო გამოყენება პოვის მონაცემთა მართვის სისტემებმა Microsoft Access, Oracle 9i и MySQL.

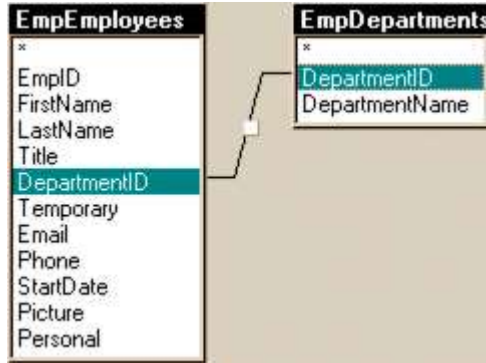
მოთხოვნა მონაცემთა ბაზისადმი - არის მონაცემთა ბაზიდან ჩანაწერების ანაკრების არჩევასთან დაკავშირებული ოპერაცია. მოთხოვნა შედგება ძიების კრიტერიუმებისაგან, რომლებიც გამოისახება მონაცემთა ბაზების ენის საშუალებებით, რომელსაც SQL-ს უწოდებენ. მაგალითად, მოთხოვნა

განსაზღვრავს, თუ რომელი სვეტი ან სტრიქონი უნდა იყოს ჩართული მარეზულტირებელ მონაცემთა ანაკრებში.

დინამიკური ვებ გვერდი - არის გამოყენებითი პროგრამების სერვერის მიერ ბრაუზერისათვის გასაგზავნად მომზადებული ვებგვერდი.

ჩანაწერების ანაკრები - არის მონაცემთა ბაზის ერთი ან რამოდენიმე ცხრილიდან ამორჩეული მონაცემების ანაკრები.

მონაცემთა რელაციური ბაზა-არის ერთი ან რამდენიმე ცხრილიანი მონაცემთა ბაზა, რომელშიც წარმოდგენილია საერთო მონაცემები. ქვემოთ მოყვანილი მონაცემთა ბაზა წარმოადგენს რელაციურ მონაცემთა ბაზას, ვინაიდან ორივე ცხრილში არსებობს ერთნაირი ველი DepartmentID.



სერვერული ტექნოლოგია-არის ტექნოლოგია, რომელსაც იყენებს გამოყენებითი პროგრამების სერვერი დინამიკურ ვებ გვერდებში ცვლილებების განხორციელების პროცესში.

Dreamweaver-ის გარემოში გამოყენებადია შემდეგი სერვერული ტექნოლოგიები:

Adobe® ColdFusion®;

Microsoft Active Server Pages (ASP);

PHP: Personal Home Page (PHP).

აღნიშნულის გარდა Dreamweaver-ის კოდირების გარემოში შესაძლებელია ისეთივე ვებ გვერდების შემუშავება, რომლებშიც გამოყენებული იქნება სხვა სერვერული ტექნოლოგიებიც.

სტატისტიკური ვებ გვერდი - არის ვებ გვერდი, რომელიც ეგზავნება ბრაუზერს გამოყენებითი პროგრამების სერვერის მიერ მასში რაიმე ცვლილების განხორციელების გარეშე.

გამოყენებითი ვებპროგრამა (ვებ აპლიკაცია) - არის ვებსაიტი, რომელზეც განთავსებულია ვებ გვერდები ნაწილობრივ ან სრულად გაურკვეველი შიგთავსით. ასეთი ვებ გვერდების საბოლოო შიგთავსის ფორმირება ხდება მხოლოდ მას შემდეგ, რაც საიტის მომხმარებელი მოითხოვს ვებ გვერდს სერვერიდან. იმასთან დაკავშირებით, რომ ვებ გვერდის საბოლოო შიგთავსი დამოკიდებულია მომხმარებლის მოქმედებების საფუძველზე შექმნილ მოთხოვნაზე, ასეთ ვებგვერდს ეწოდება დინამიკური.

ვებსერვერი - არის პროგრამული უზრუნველყოფა, რომელიც აბრუნებს ვებ გვერდებს ვებ ბრაუზერის მოთხოვნების საპასუხოდ. ჩვეულებრივ მოთხოვნა ვებ გვერდის წარმოდგენაზე ფორმირდება ვებ გვერდზე არსებული მიმართვის (მინიშნების) დაწკაპუნებისას, ბრაუზერში ჩანართის არჩევისას, ან ბრაუზერის

სამისამართე სტრიქონში URL-მისამართის შეტანისას. ყველაზე გავრცელებულ ვებ სერვერებს წარმოადგენენ Microsoft Internet Information Server (IIS) და Apache HTTP Server.

3.3. კომპიუტერის გამართვა გამოყენებითი პროგრამების შესამუშავებლად

გამოყენებითი ვებ პროგრამის შესაქმნელად **Dreamweaver**-თან ერთად საჭიროა შემდეგი პროგრამული უზრუნველყოფის არსებობა:

-ვებ სერვერი;

-გამოყენებითი პროგრამების სერვერი, რომელიც იმუშავებს ვებ სერვერთან ერთად.

შენიშვნა: გამოყენებითი ვებ პროგრამების კონტექსტში ტერმინები ვებ სერვერი და გამოყენებითი პროგრამების სერვერი ეხება პროგრამულ უზრუნველყოფას და არა კომპიუტერულ მოწყობილობას.

თუ გამოყენებით ვებ პროგრამაში აუცილებელია მონაცემთა ბაზის გამოყენება, მაშინ საჭირო ხდება შემდეგი დამატებითი პროგრამული უზრუნველყოფის დაყენება კომპიუტერზე:

-მონაცემთა ბაზების მართვის სისტემის (მზმს);

-მონაცემთა ბაზების შესაბამისი დრაივერის.

ვებ ჰოსტინგის განმახორციელებელი ზოგიერთი კომპანია, გამოყენებითი ვებ პროგრამების შემუშავების, ტესტირებისა და განშლისათვის, თავიანთი პროგრამული უზრუნველყოფის გამოყენებასთან დაკავშირებულ მომსახურებას თავაზობენ. ზოგიერთ შემთხვევაში ვებ ჰოსტინგის მომწოდებელი კომპანიების მომსახურების გამოყენების ნაცვლად, ვებ საიტების შესამუშავებლად შესაძლებელია ყველა საჭირო პროგრამული უზრუნველყოფის დაყენება იმავე ლოკალურ კომპიუტერზე, რომელზეც **Dreamweaver**-ი არის დაყენებული. გარდა ამისა, შესაძლებელია საჭირო პროგრამული უზრუნველყოფის დაყენება ქსელის ერთ-ერთ კომპიუტერზე (ეს, როგორც წესი, არის კომპიუტერი, რომელიც იმართება Windows-ის ოპერაციული სისტემით), რაც სხვა დამპროექტებლებსაც მისცემს პროექტზე ერთობლივად მუშაობის საშუალებას.

გამოყენებით ვებ პროგრამაში მონაცემთა ბაზის გამოყენებისას, პირველ რიგში, აუცილებელია მასთან მიერთება.

დინამიკური საიტების შემუშავებისა და ტესტირებისათვის აუცილებელია მუშა მდგომარეობაში მყოფი ვებ სერვერის არსებობა. როგორც ზემოთ იყო აღნიშნული, ვებ სერვერი - არის პროგრამული უზრუნველყოფა, რომელიც ახორციელებს ვებ გვერდების წარმოდგენას ვებ ბრაუზერში მისივე მოთხოვნების საპასუხოდ. ვებ სერვერს ზოგჯერ HTTP სერვერსაც უწოდებენ. შესაძლებელია ვებ სერვერის გაშვება ლოკალურ კომპიუტერზეც. Macintosh-ის ტიპის კომპიუტერებზე წინასწარ არის ხოლმე დაყენებული ვებსერვერი-Apache.

თუ კი გამოყენებითი ვებ პროგრამის შესამუშავებლად გამოიყენება სერვერი IIS, მაშინ ვებ სერვერის გაჩუმებითი სახელი ემთხვევა კომპიუტერის სახელს. ვებ სერვერის სახელის შეცვლა შესაძლებელია კომპიუტერისათვის სახელის გადარქმევით. თუ კი

კომპიუტერს არ გააჩნია სახელი, ვებ სერვერს ენიჭება სახელი localhost.

სერვერის სახელი შესაბამისობაში უნდა იყოს სერვერის ძირეული საქალაქის სახელთან, რომელსაც Windows-ის ტიპის ოპერაციული სისტემების მქონე კომპიუტერებზე, ყველაზე ხშირად უწოდებენ C:\Inetpub\wwwroot. ძირეულ საქალაქში განთავსებული ნებისმიერი ვებ გვერდის გახსნა, შესაძლებელია ბრაუზერის საშუალებით, შემდეგი URL-მისამართის მითითების გზით:

<http://სერვერის სახელი/ფაილის სახელი>.

ასევე შესაძლებელია ძირეული საქალაქის მიმართ ნებისმიერ ქვესაქალაქში განთავსებული ნებისმიერი ვებ გვერდის გახსნა, მოცემული ქვესაქალაქის მითითებით URL-მისამართში.

მაგალითად: დავუშვათ, რომ ფაილი soleil.html მდებარეობს ქვესაქალაქში gamelan და სრული გზა ფაილამდ ეგამოიყურება შემდეგნაირად:

C:\Inetpub\wwwroot\gamelan\soleil.html

ამ გვერდის გახსნა შესაძლებელია ლოკალური კომპიუტერის ბრაუზერის სამისამართე სტრიქონში შემდეგი URL-მისამართისშეტანით:

http://mer_noire/gamelan/soleil.html

თუ სრულდება მოთხოვნა ვებ სერვერისადმი ლოკალურ კომპიუტერზე, მაშინ შესაძლებელია სერვერის სახელის შეცვლა localhost-ით.

მაგალითად, ქვემოთ მითითებული URL-მისამართები ბრაუზერში გახსნის ერთსა და იმავე ვებ გვერდს:

http://mer_noire/gamelan/soleil.html

<http://localhost/gamelan/soleil.html>

შენიშვნა: სერვერის სახელისა და დარეზერვირებული სიტყვის localhost, ნაცვლად შეიძლება გამოყენებულ იქნას მისამართი: 127.0.0.1

(მაგალითად:<http://127.0.0.1/gamelan/soleil.html>).

ვებ სერვერის არჩევა

გამოყენებითი ვებ პროგრამების შემუშავებისა და ტესტირებისათვის ფართოდ გავრცელებული ვებ სერვერების დიდი არჩევანი არსებობს, რომელთა

რიცხვს განეკუთვნება Microsoft Internet Information Server (IIS) და Apache HTTP Server.

თუ არ ვისარგებლებთ ვებ ჰოსტინგის მომწოდებლის მომსახურებით, უნდა ავირჩიოთ ვებ სერვერი და დავაყენოთ იგი ჩვენს ლოკალურ კომპიუტერზე. Windows-ისდა Macintosh-ის მომხმარებლებს, რომლებიც აპირებენ გამოყენებითი ვებ პროგრამების შემუშავებას ColdFusion ენაზე, გამოადგებათ ვებ სერვერი, რომელიც შედის ColdFusion8 - ის გამოყენებითი პროგრამების სერვერების შემადგენლობაში. შესაძლებელია მისი დაყენება და უფასოდ გამოყენება.

Windows-ის სხვა მომხმარებლებს შეუძლიათ თავიანთ ლოკალურ კომპიუტერებზე დააყენონ და გამოიყენონ ვებ სერვერი IIS. შესაძლებელია, რომ აღნიშნული სერვერი უკვე დაყენებულიც იყოს მომხმარებლის კომპიუტერზე. შეამოწმეთ, არის თუ არა ფაილურ სისტემაში საქაღალდე C:\Inetpub ან D:\Inetpub. აღნიშნული საქაღალდე იქმნება IIS სერვერის დაყენებისას.

Mac OS-ის მომხმარებლებს შეუძლიათ ისარგებლონ ლოკალური ვებ სერვერით Apache, რომელიც ყენდება ოპერაციულ სისტემასთან ერთად. სხვა ვებ სერვერების დაყენებისა და გამართვის შესახებ ცნობების მიღება შესაძლებელია შესაბამისი მომწოდებლების დოკუმენტაციებში, ან შესაბამის სისტემურ ადმინისტრატორებთან.

გამოყენებითი პროგრამების სერვერის არჩევა.

გამოყენებითი პროგრამების სერვერი პროგრამული უზრუნველყოფაა, რომლის საშუალებითაც ვებ სერვერი ახდენს დინამიკური ვებ გვერდების დამუშავებას. გამოყენებითი პროგრამების სერვერის შერჩევა უნდა მოხდეს რიგი ფაქტორების გათვალისწინებით: დაგეგმილი ბიუჯეტის, გამოყენებული სერვერული ტექნოლოგიის (ColdFusion, ASP ან PHP) და ვებ სერვერის ტიპის გათვალისწინებით.

გამოყენებითი პროგრამების სერვერის ბიუჯეტი.

პროგრამული უზრუნველყოფის ზოგიერთი მომწოდებელი თავაზობს, როგორც საყიდლად, ისე საადმინისტრირებოდ, ძვირად ღირებულ, საწარმოო

დანიშნულების გამოყენებითი პროგრამების სერვერებს. სხვა მომწოდებელი კი აწვდის უფრო მარტივ და ეკონომიურ გადაწყვეტილებებს (მაგალითად, ColdFusion). გამოყენებითი პროგრამების ზოგიერთი სერვერი ჩაშენებულია ვებ სერვერებში (მაგალითად, Microsoft IIS), სხვების ჩატვირთვა კი შესაძლებელია უფასოდ ინტერნეტიდან (მაგალითად, PHP).

სერვერული ტექნოლოგიები.

გამოყენებითი პროგრამების სერვერებში გამოიყენება სხვადასხვა ტექნოლოგია. Dreamweaver-ის მიერ მხარდაჭერილია სამი სერვერული ტექნოლოგია: ColdFusion, ASP და PHP. ქვემოთ მოტანილ ცხრილში მოცემულია მეტად გავრცელებული გამოყენებითი პროგრამების სერვერები, რომლებიც ხელმისაწვდომია სერვერული ტექნოლოგიებისათვის.

მონაცემთა ბაზების სერვერის შერჩევა

მონაცემთა ბაზების სერვერის არჩევა დამოკიდებულია იმ ინფორმაციის მოცულობასა და სირთულეზე, რომელიც მასში უნდა იქნას შენახული.

მონაცემთა ბაზის სერვერის არჩევა უნდა მოხდეს დაგეგმილი ბიუჯეტის, ბაზაში წვდომის უფლების მქონე მომხმარებელთა საორიენტაციო რაოდენობისა და სხვა ფაქტორების გათვალისწინებით.

მონაცემთა ბაზების სერვერის ბიუჯეტი.

ზოგიერთი მწარმოებელი თავაზობს მძლავრ მონაცემთა ბაზების სერვერებს, რომლებიც ძვირად ღირებულია და ამავე დროს რთულია მათი ადმინისტრირება. სხვები აწვდის შედარებით მარტივ და ეკონომიურ გადაწყვეტებს, მაგალითად, Microsoft Access ან მონაცემთა ბაზა ღია კოდით MySQL.

მომხმარებელთა რაოდენობა.

თუ ვებ საიტი გათვალისწინებულია დიდი რაოდენობის მომხმარებლებისათვის, უნდა მოხდეს ისეთი მონაცემთა ბაზის შერჩევა, რომელიც გათვლილი იქნება შესაბამის რაოდენობაზე. ისეთი ვებ-საიტებისათვის, რომლებიც ითხოვენ დიდ მოქნილობას მონაცემების მოდელირებისას და მომხმარებელთა დიდი რაოდენობის ერთდროული მუშაობის შესაძლებლობების მხარდაჭერას, შესაძლებელია ისეთი

სერვერული რელაციური მონაცემთა ბაზების გამოყენება (რომელთაც, ზოგადად RDBMS-ს უწოდებენ), როგორცაა Microsoft SQL Server და Oracle.

ვებ-გვერდების შემუშავების ColdFusion გარემოს გამართვა.

ვებ-საიტების შემუშავების ColdFusion გარემოს Dreamweaver-თან სამუშაოდ მომზადების დაწვრილებითი ინსტრუქცია Windows და Mac კომპიუტერებისათვის მოცემულია Adobe-ის ვებ საიტზე:

http://www.adobe.com/devnet/dreamweaver/articles/setup_cf.html.

Windows-ის და Macintosh-ის მომხმარებელს შეუძლია ჩატვირთოს და დააყენოს გამოყენებითი პროგრამების სერვერის-ColdFusion, უფასო და სრულად ფუნქციური, ვებ საიტების დამმუშავებლებისათვის განკუთვნილი ვერსია Adobe-ის ვებ გვერდიდან: www.adobe.com/go/coldfusion_ru.

შენიშვნა: ვებ-საიტების დამმუშავებლებისათვის წარმოდგენილი ვერსია, განკუთვნილია არაკომერციული მოხმარებისათვის და გამოიყენება ვებპროგრამების შემუშავებისა და ტესტირების მიზნებისათვის. მისი გამოყენებისათვის ლიცენზია არ არის საჭირო. მაგრამ მოთხოვნების დამუშავება ხდება მხოლოდ ლოკალური კომპიუტერიდან და ორი ურთიერთდამორებული IP-მისამართიდან. პროგრამას არ გააჩნია შეზღუდვა გამოყენების დროის მიხედვით.

ColdFusion პაკეტის დაყენების პროცესში შესაძლებელია მისი გამართვა საკუთარი, ინტეგრირებული ვებ სერვერის ან კომპიუტერზე დაყენებული სხვა ვებ სერვერის გამოყენებაზე ორიენტირებით. ვებ გვერდების შესამუშავებელი პროგრამული უზრუნველყოფის შერჩევა რეკომენდებულია ისეთნაირად, რომ იგი შეესაბამებოდეს საწარმო-გარემოს პროგრამულ უზრუნველყოფას. ასე რომ, არა აქვს განსაკუთრებული აზრი ColdFusion-ის საკუთარი ვებ სერვერის გამოყენებას, თუ კი ვებ საიტების

შემუშავებისა და ექსპლუატაციისათვის გამოიყენება ვებ სერვერი Microsoft IIS.

ვებ საიტების შემუშავების PHP-გარემოს გამართვა ვებ საიტების შემუშავების გარემოს PHP-ს, Dreamweaver-თან სამუშაოდ მომზადების დაწვრილებითი ინსტრუქცია Windows-ის და Macintosh-ის კომპიუტერებისათვის მოცემულია Adobe-ის ვებ საიტზე [:http://www.adobe.com/devnet/dreamweaver/articles/setup_php.html](http://www.adobe.com/devnet/dreamweaver/articles/setup_php.html).

გამოყენებითი პროგრამების PHP სერვერის სხვადასხვა ვერსია არსებობს, რომელიც განკუთვნილია სხვადასხვა ოპერაციულ სისტემებთან, მათ შორის Windows, Linux, UNIX, HP-UX, Solaris და Mac OS X-თან სამუშაოდ. დამატებითი ცნობების მიღება გამოყენებითი პროგრამების სერვერის PHP-ს შესახებ შესაძლებელია მის დოკუმენტაციაში, რომელიც ხელმისაწვდომია ვებ საიტზე: www.php.net/download-docs.php.

ვებ გვერდების შემუშავების ASP გარემოს გამართვა.

ვებ საიტების შემუშავების გარემოს ASP-ს, **Dreamweaver**-თან სამუშაოდ მომზადების დაწვრილებითი ინსტრუქცია Windows-ის და Macintosh-ის ტიპის კომპიუტერებისათვის მოცემულია Adobe-ის ვებ საიტზე:

http://www.adobe.com/devnet/dreamweaver/articles/setup_a_sp.html.

ASP ვებ-გვერდებთან სამუშაოდ, საჭიროა გამოყენებითი პროგრამების სერვერი, რომელიც უზრუნველყოფს Microsoft Active Server Pages 2.0 ტექნოლოგიის მხარდაჭერას, მაგალითად, Microsoft IIS, რომელიც შედის ოპერაციულ სისტემებში Windows 2000 და Windows XP Professional. OC Windows XP Professional ოპერაციული სისტემის მომხმარებელს შეუძლია თავიანთ ლოკალურ კომპიუტერზე დააყენოს და გაუშვას Microsoft IIS სერვერი. Macintosh-ის მომხმარებელს შეუძლია ისარგებლოს ASP-ტექნოლოგიის მხარდაჭერი ვებ ჰოსტინგის მომსახურებით ან შეუძლია დააყენოს Microsoft IIS სამსახურები დაშორებულ კომპიუტერზე.

გამოყენებითი პროგრამებისათვის ძირეული საქაღალდის შექმნა.

ვებ ჰოსტინგის მომსახურების მომწოდებელთან დარეგისტრირების ან სერვერული პროგრამული უზრუნველყოფის დამოუკიდებლად გამართვის შემდეგ, შექმენით ძირეული საქაღალდე გამოყენებითი ვებ პროგრამისათვის, ვებ სერვერის მართვის ქვეშ მყოფ კომპიუტერზე. ძირეული საქაღალდე შეიძლება იყოს ლოკალური ან დაშორებული, იმის მიხედვით, თუ სად არის დაყენებული ვებ სერვერი.

ვებ სერვერს შეუძლია ამ საქაღალდიდან ან მასში ჩალაგებული საქაღალდეებიდან ნებისმიერი ფაილის წარმოდგენა ვებ ბრაუზერიდან შემოსული HTTP მოთხოვნის საპასუხოდ. მაგალითად: კომპიუტერზე, რომელზეც გაშვებულია გამოყენებითი პროგრამების სერვერი ColdFusion8, ნებისმიერი ფაილი საქაღალდეში \ColdFusion8\wwwroot (ან მასში ჩალაგებულ საქაღალდეში) შეიძლება მიღებულ იქნას ვებ ბრაუზერის საშუალებით.

ქვემოთ ჩამოთვლილია გარკვეული ვებ სერვერებისათვის გაჩუმებით მიღებული ძირეული საქაღალდეები:

ვებ სერვერი	ძირეული საქაღალდე გაჩუმებით
ColdFusion 8	\ColdFusion8\wwwroot
IIS	\Inetpub\wwwroot
Apache (для Windows)	\apache\htdocs
Apache (Macintosh)	Users: მომხმარებლის სახელი: Sites

იმისათვის, რომ შევამოწმოთ ვებ სერვერის მუშაობა, მოვათავსოთ სატესტო HTML-გვერდი გაჩუმებით ძირეულად მიჩნეულ საქაღალდეში და შევეცადოთ მის გახსნას, ბრაუზერში მისი აღმნიშვნელი გვერდის URL-მისამართის მითითებით. URL მისამართი შეიცავს HTML-გვერდის დომენურ სახელს და ფაილის

სახელს და გამოიყურება შემდეგნაირად:

www.example.com/testpage.htm.

თუ ვებსერვერი დაყენებულია ლოკალურ კომპიუტერზე, დომენური სახელის ნაცვლად შესაძლებელია localhost-ის გამოყენება. შევიტანოთ ლოკალური კვანძების შემდეგი URL-მისამართებიდან ერთ-ერთი, გამოყენებული ვებ სერვერის შესაბამისად:

ვებ სერვერი	ლოკალური კვანძის URL მისამართი
ColdFusion 8	http://localhost:8500/testpage.htm
IIS	http://localhost/testpage.htm
Apache (для Windows)	http://localhost:80/testpage.htm

ვებ სერვერი	ლოკალური კვანძის URL მისამართი
Apache (Macintosh)	http://localhost/~მომხმარ ებლის სახელი/testpage.htm (სადაც მომხმარებლის სახელი — Macintosh-ის მომხმარებლის სახელია)

შენიშვნა: გაჩუმებით ვებ სერვერი ColdFusion ემსახურება პორტს 8500, ხოლო Windows-ის ვებ-სერვერი Apach -პორტს 80.

თუ ვებ-გვერდი არ გაიხსნა, ეს იმას ნიშნავს, რომ მოხდა შემდეგი რომელიმე შეცდომა:

- ვებ სერვერი არ იყო გაშვებული. უნდა მიმართოთ ვებ-სერვერის დოკუმენტაციის გაშვებასთან დაკავშირებულ ინსტრუქციებს;
- ფაილს არ გააჩნია გაფართოება html ან htm;

-ბრაუზერის სამისამართე ტექსტურ ველში საჭიროა ფაილამდე ბილიკის შეტანა და არა URL-მისამართის (მაგალიტად:C:\ColdFusion8\wwwroot\testpage.htm და არა <http://localhost:8500/testpage.htm>);

-URL-მისამართი მითითებულია შეცდომით. შეამოწმეთ შეცდომის არსებობა და დარწმუნდით, რომ ფაილის სახელის შემდეგ არ არის შეტანილი დახრილი ხაზი, მაგალიად: <http://localhost:8080/testpage.htm/>.

გამოყენებითი პროგრამისათვის ძირეული საქაღალდის შექმნის შემდეგ საჭიროა **Dreamweaver**-ის საიტის განსაზღვრა ფაილების სამართავად.

Dreamweaver-ის საიტის განსაზღვრის შესახებ.

გამოყენებითი ვებპროგრამების შესამუშავებლად სისტემის გამართვის შემდეგ, საჭიროა **Dreamweaver**საიტის განსაზღვრა:

მუშაობის დაწყების წინ საჭიროა დარწმუნება იმაში, რომ დაკმაყოფილებულია შემდეგი მოთხოვნები:

- არსებობს ვებ სერვერში წვდომის უფლება. ვებ სერვერი შეიძლება დაყენებული იყოს ლოკალურ კომპიუტერზე, დაშორებულ კომპიუტერზე,

მაგალითად, მისი შემმუშავებლების სერვერზე, ან სერვერზე, რომლის მომსახურებაც ხდება ვებ ჰოსტინგის მომსახურებების მომწოდებლების მიერ:

- გამოყენებითი პროგრამების სერვერი დაყენებულია და გაშვებულია იმ სისტემაზე, რომელზეც დაყენებულია ვებსერვერიც;

- გამოყენებითი ვებპროგრამისათვის ძირეული საქაღალდე შექმნილია იმავე სისტემაში, რომელზეც დაყენებულია ვებსერვერი.

გამოყენებითი ვებპროგრამისათვის **Dreamweaver**საიტის განსაზღვრა ხორციელდება სამ ეტაპად:

1. ლოკალური საქაღალდის განსაზღვრა.

ლოკალური საქაღალდე - არის საქაღალდე, რომელიც გამოიყენება საიტის ფაილების სამუშაო ასლების შესანახად მყარ დისკოზე. ლოკალური საქაღალდე შეიძლება განისაზღვროს ყოველი შესაქმნელი გამოყენებითი ვებ პროგრამისათვის. ლოკალური საქაღალდის განსაზღვრა ფაილების მართვისა და მათი ვებ სერვერიდან ვებ სერვერზე მარტივად გადატანის საშუალებას იძლევა;

2. ვებ სერვერის შემცველ კომპიუტერზე განთავსებული საქალაქის განსაზღვრა, როგორც Dreamweaver-ის დაშორებული საქალაქისა. (დაშორებული საქალაქდ-არის საქალაქდე, რომელიც შექმნილია გამოყენებითი ვებ პროგრამისათვის ვებ სერვერზე).

3. **Dreamweaver**-ი დაშორებულ საქალაქდეს იყენებს დინამიკური შიგთავსის წარმოსადგენად და აგრეთვე მომხმარებლის მუშაობის პროცესში მონაცემთა ბაზებთან მისაერთებლად. სატესტო სერვერის სახით შესაძლებელია გამოყენებული იქნას ლოკალური კომპიუტერი, შემუშავების სერვერი, დამხმარე ან საწარმოო სერვერი. შესაძლებელია ნებისმიერი სერვერის არჩევა, მთავარია, რომ მას შეეძლოს საჭირო ტიპის დინამიკური ვებ-გვერდების დამუშავება.

4. **Dreamweaver** საიტის განსაზღვრის შემდეგ შესაძლებელია გამოყენებითი ვებპროგრამის შემუშავების დაწყება.

ლიტერატურა

1. გ.ღვინეფაძე WEB დაპროგრამება, I ნაწილი, HTML. საქართველოს ტექნიკური უნივერსიტეტი. თბ; 2007;
2. თოდუა თ., ვერულავა ლ. WEB-ტექნოლოგიები, HTMLI ნაწილი. საქართველოს ტექნიკური უნივერსიტეტი. თბ; 2006;
3. <http://www.adobe.com/products/dreamweaver/>
4. <http://www.entheosweb.com/dreamweaver/default.asp>
5. რამაზ შამუგია. Internet და Web ტექნოლოგიების საფუძვლები (HTML,CSS,SQL,PHP, Dreamweaver MX), გამომცემლობა „მერიდიანი“, თბ;2013
6. ქ.ნანობაშვილი, ფ.პაატაშვილი, ვ.ტაკაშვილი WEB ტექნოლოგიები (MACROMEDIA DREAMWEAVER) საქართველოს ტექნიკური უნივერსიტეტი. თბ; 2011;

რედაქტორი ბ. ცხადაძე

გადაეცა წარმოებას 28.02.2017. ხელმოწერილია დასაბეჭდად
16.03.2017. ქალაქის ზომა 60X84 1/16. პირობითი ნაბეჭდი თაბახი 8,5.

საგამომცემლო სახლი „ტექნიკური უნივერსიტეტი“, თბილისი,
კოსტავას 77

