

T. U. Utapov, Z. M. Adizova

WEB TEXNOLOGIYALAR (I QISM)
(O'QUV QO'LLANMA)

KAMOLOT nashriyoti
Buxoro-2023

UO‘K: 004.774.6

KBK: 16.263

U92

**T.U.Utapov, Z.M.Adizova, Web texnologiyalar (I qism) /
[Matn]: o‘quv qo‘llanma / T.U.Utapov, Z.M.Adizova - Buxoro :
“BUXORO DETERMINANTI”MCHJning Kamolot nashriyoti,
2023. - 140 b.**

Taqrizchilar:

*T.R.Shafiyev-Buxoro davlat universiteti “Axborot tizmilari va
raqamli texnologiyalar” kafedrasi dotsenti t.f.f.d (PhD)*

*Sh.N.Nosirova-Navoiy davlat pedagogika instituti “Informatika”
kafedrasi professori*

ISBN: 978-9910-734-56-4

*Oliy ta’lim, fan va innovatsiyalar vazirligi Buxoro davlat
universitetining “21” oktabrdagi “556” sonli buyrug‘iga asosan nashrga
ruxsat berildi. Ro‘yxatga olish raqami 556-18.*

© “KAMOLOT” nashriyoti

© T. U. Utapov

© Z. M. Adizova

Mundarija:

Kirish.....	4
1-mavzu. Web texnologiya asosiy tushunchalari. Html hujjat tarkibiy qismlari. Teg tushunchasi.....	5
2-mavzu. Mant bilan ishlash. Blokli va satrli belgilash elementlari.....	18
3-mavzu. Ro‘yxatlar. Jadvallar.....	23
4-mavzu. Grafika va multemediya	41
5-mavzu. Formalar bilan ishlash.	41
6-mavzu. Xarita map bilan ishlash.	53
7-mavzu. Stillar.....	55
8-mavzu. Selektorlar.....	61
9-mavzu. CSSda chegara.....	81
10-mavzu. Psevdosinflar	85
11-mavzu. Front page dasturi	90
12-mavzu. Bootstrap.....	108
13-mavzu. Wpf modeli	112
14-mavzu. Javascript dasturi	123
Foydalanilgan adabiyotlar ro‘yxati.	138

KIRISH

Internetning paydo bo'lishi o'zaro muloqotni amalga oshirish uchun yangi kommunikatsion texnologiyalar hamda ma'lumotlar bazasi bilan ishlovchi texnologiyalarni ishlab chiqishni taqozo etdi. Shu kungacha internetda sahifalar yaratuvchi va ular bilan ishlovchi oddiy texnologiyalar ishlab chiqilgan edi. Java Script webda ishlovchilar uchun html-sahifalarni tez va oson, dinamik holda ishlashi va o'zgartirish uchun imkoniyatlar yaratib beruvchi texnologiyadir. JavaScript texnologiyasining ajoyib xususiyatlaridan biri bu — ochiq dasturiy kodga egaligi va bepul tarqatilishidir. Platformaga bog'liq emasligi, deyarli barcha platformalarda ishlay olish imkoniyati bilan ajralib turadi. Shuning uchun dunyoda ko'plab Web sayt yaratuvchilar ushbu texnologiyani ma'qul ko'rishadi. Boisi JavaScript kuniga milliondan ortiq murojaatlarga xizmat ko'rsata oladi. Ushbu qo'llanma ikki qism dan iborat bo'lib: birinchi qism da HTML, CSS texnologiyasi haqida so'z boradi. Asosiy konstruksiyalar, standart funksiyalar, obyektga yo'naltirilgan dasturlash borasidagi imkoniyatlar haqida ma'lumotlar berilgan. JavaScript yordamida yaratilgan dasturlarda HTML ning ba'zi elementlarini ishlatish haqida ham batafsil to'xtab o'tilgan. Ikkinchi qism ma'lumotlar bazasini boshqarish tizimlarining (MBBT) imkoniyatlari va foydalanish uchun funksiyalar to'plami, JavaScript texnologiyasining MySQL MBBT bilan integratsiyasiga bag'ishlangan. Ushbu bo'limda JavaScript va MySQL MBBT yordamida dinamik WEB saytni loyihalash va dastur kodlari misollar yordamida tushuntirib o'tilgan.

1-MAVZU. WEB TEXNOLOGIYA ASOSIY TUSHUNCHALARI. HTML HUJJAT TARKIBIY QISMLARI. TEG TUSHUNCHASI.

Reja:

1. Web texnologiya asosiy tushunchalari.
2. HTML hujjat tarkibiy qismlari.
3. Teg tushunchasi

WWW — kompyuter tarmoqlarida kerakli ma'lumotni qurishni gipermurojaat deb ataluvchi usul bilan kompyuter tarmoqlarida joylashtirish usuli WWW — Word Wide Web nom Tim Berusers — Lee (CERN laboratoriyasi) tomonidan kiritilgandir. U boshqacha qilib, butun dunyo «o'rgimchaklari» deb ataladi. O'rgimchak yashashi uchun turli yo'llar tashkil qilib, bu yo'llar orqali turli nuqtalarga yurishga o'xshab WWWda ham turli yo'llar orqali tegishli ma'lumotlarga yetib borish va uni qurish imkoniyati borligidir. WWWda nuqtalar rolini kompyuter o'ynaydi. Yo'llar sifatida telefon yo'llari ishlatiladi. Web sahifalar odatda HTML hujjat, ya'ni HTML (Huper Text Marko'p language — gipermatnni belgilash tili) tilida yozilgan hujjat sifatida tayyorlanadi.

Bu holat yozilgan hujjatlarni tabiiy ko'rinishda (keng ommaga tushunarli bo'lgan) kompyuter ekranida tasvirlash uchun maxsus programmalar ishlatiladi. Bunday programmalar Browser (quruvchi, sharhlovchi)lar deb ataladi. Xususan, Windows tarkibida mavjud programmalar sharhlovchi nomi bilan yuritiladi.

Gipermatn va gipermediya. WWW (qisqacha — Web) sistemada ma'lumotlar gipermatniy hujjatlar shaklida olinadi. Gipermatn boshqa matnli hujjatlarga yo'l ko'rsatuvchi matndir. Bu esa boshqa matnlarga (bu matnlar qaysi mamlakatning serverida turishidan qat'i nazar) tezda o'tish imkonini beradi. Matnlar bilan bir qatorda WWW hujjatlarda rangli harakatlanuvchi tasvirlarni, turli video kliplarni, umuman multimedia ma'lumotlarni ko'rish mumkin. Matndan tashqari boshqa shakldagi ma'lumotlarni beruvchi hujjatlar gipermedia hujjatlari deyiladi. Web — Internet tarmoqlarida joylashgan fayllar to'plami bo'lib, ularning soni soat sayin ko'payib bormoqda. Bu fayllarda ma'lumotlarning turli xillarini; matn, grafik, tasvirlar, video, audio ma'lumotlarini uchratish mumkin.

Webning eng asosiy xususiyatlaridan biri unda turli obyektlarga (matn, video, grafik) gipermurojaatning mavjudligidir. Matnlarda kalit soʻzlar deb ataluvchi soʻzlar orqali dunyoning ixtiyoriy burchagida Internet doirasida joylashgan maʼlumotlarga murojaat qilish va u orqali maʼlumotlarni topish gipermurojaat deb ataladi. Ajratilgan soʻz va frazalar — gipermatn aloqalari qisqacha giperaloqalar deb yuritiladi. Bu giperaloqalar orqali boshqa hujjatlarga murojaat qilib, unda yangi giperaloqalarni yaratish mumkin. Shunday qilib, Web — gipermatniy sistema boʻlib, unda maʼlumotlar ixtiyoriy tartibda (chiziqsiz boʻlmagan) joylashadi. Uni na boshi, na oxiri bor. Unda maʼlumotlar ixtiyoriy joyda joylashgan boʻladi. Bunday maʼlumotlar faqat giperaloqalar bilan bogʻlangan xolos. Hozirda giperaloqalar faqat matndagi ajratilgan soʻzlar bilangina emas, hatto tasvirlar, grafiklar, ularning qismlari orqali ham amalga oshirilishi mumkin. Masalan: Webda biror mamlakatning geografik kartasi mavjud boʻlsa, uning bir boʻlagiga sichqoncha koʻrsatkichini oʻrnatib bosilsa, u orqali Web maʼlumotlariga kiriladi. Webda maʼlumotlar Web sahifalari shaklida beriladi. Bu sahifalar maxsus HTML tilida tashkil qilinadi.

Bosh sahifa. Bosh sahifa biror subyektning, shaxs yoki tashkilotlarning borligi belgisi boʻlgan Web sahifadir. Odatda bosh sahifa shaxsning rasmi, uning avtobiografiyasi, mutaxassisligi va boshqa maʼlumotlarni aks ettiradi. Tashkilotlarda esa uning nomi, tuzilishi va faoliyati bilan bogʻliq bosh maʼlumotlar boʻladi.

Internet va Web bir xil narsami? Yoʻq, albatta. Web oʻz sahifalarini saqlash va uzatish uchun Internetdan foydalanadi. Web Internetning imkoniyatlaridan biri deyish mumkin. World Wide Web Internetga oʻxshab har tomonlama uzluksiz oʻzgarib turadi. xar doim yangi serverlar paydo boʻladi, eskilari esa oʻz-oʻzidan yoʻqoladi. Yangi-yangi WWW browserlari yaratiladi, avval maʼlumotlar takomillashtiriladi, yangi imkoniyatlar qoʻshiladi. Internetning yangi servislarida ishlash uchun qaydnomalar ishlab chiqiladi. Uning ajoyib xususiyatlaridan biri Internetda mavjud boshqa sistemalar bilan doʻstona munosabatda boʻlish va ular bilan birgalikda foydalanish mumkinligidir. Bunda gap UseNet, FTP, Telnet va boshqalar kabi Internet xizmatlari ustida ketyapti. Web orqali siz gazetalardagi maʼlumotlarni, turli yangiliklarni, turli sohaga oid maʼlumotlarni, kitob va jurnallarni, kompakt disklarni sotib olish uchun pul sarflamasdan, eng muhimi ortiqcha kuch sarflamay, biror joyga kitob,

gazeta, kompakt disk va hokazolarni izlab bormasdan, ish joyingizda bir zumda olasiz. Bu asrimizning katta mo'jizasi emasmi axir? Shu joyda bir misol keltiraylik.

1989-yili CERN (Evropa elektron zarralar fizikasi laboratoriyasi) tadqiqotchilari o'z oldilariga shunday sistema yaratish masalasini qo'yishadiki, bu sistema turli ilmiy guruhlar o'zaro aloqa qilishlarini ta'minlashi kerak edi. CERN tadqiqotlarida turli shaharlarda faoliyat ko'rsatuvchi ilmiy markazlar va doimiy axborot almashishga qiziqqanlar qatnashdilar. Biroq bu oson kechmadi, matnni qurish yoki grafik tasvirlarni qurishda doimo qidirilayotgan hujjatning joylashgan o'rini qidirishga va bu harakatlarni bajarish uchun bir necha amaliy dasturlardan foydalanishga to'g'ri keladi. TelNet, FTPlarga o'xshash, grafik tasvirlarni ko'ruvchi dasturiga o'xshash dasturlar kerak bo'ladi. Shuning uchun sistemani ishlab chiqishda, maqsadga yetish uchun juda ko'p oraliq qadamlardan foydalanildi. 1990-yil oxirida CERN tadqiqotchilari matn va grafik holatlarda qurish uchun NEXT oilasiga tegishli programma yaratishdi. 1991-yilda WWW sistemasi CERNda keng foydalanila boshladi. WWWning dastlabki foydalanuvchilariga gipermatniy hujjatlar va UseNet telekonferentsiya maqolalariga kirish huquqi berildi. Rivojlanish bosqichida Internet servis turlariga interfeys qo'shildi (WAIS, FTP va boshqalarga o'xshash); 1992-yili CERN WWW loyihasi to'g'risida juda keng ma'lumot tarqatishni boshladi. Internetning butun jahon jamiyati tomonidan tan olinishi turli xil, rang-barang ma'lumotlarga kirish imkoniyati paydo bo'lganidir. Ko'p sonli WWW serverlari yaratildi. Ba'zi jamoalar WWWdan foydalanuvchilar uchun ishlashni osonlashtiruvchi dasturlar yozishga kirishdi. 1993-yildan boshlab WWW Internetning resurslari ichida eng ommaviysiga aylandi.

Gipermatniy aloqalar. Gipermatniy hujjatlarning asosiy ajralib turadigan qismi, bu hujjatlarga quyiladigan giperizohlardir. Giperizohlar «jonli» ravishda namoyon bo'ladi. Ya'ni oddiy matnlarga quyilgan, masalan, quyidagicha izoh «qo'shimcha ma'lumotni ikkinchi varaqdan olasiz» kabi izohda, siz uni ikkinchi varaqqa o'tsangiz olasiz. Gipermatnlarda esa o'sha izohning o'zi ham harakatlanadi. HTML turli buyruqlarni o'z ichiga oladi. Boshqa hujjatlarga yo'l ko'rsatuvchi va olib boruvchi giperizohlar ham gipermatniy aloqalarning asosiy qismi hisoblanadi. Giperaloqalarni

turli obyektlar, hatto rasmlarning bo'laklari orqali ham amalga oshirilishi mumkin.

WWW asosiy konponentsiyalari. Internet tushunchasiga o'xshab World Wide Web muayyan ma'lumotlar resurslarini o'zida jamlagan serverlar to'plamidir. Amalda WWW doimo o'zgarishda, yangi-yangi WWW kontseptsiyalari bilan tanishish «dunyo o'rgimchaklari»ni tushunish imkonini beradi.

WWW loyihasi. WWW haqida to'la ma'lumotlarni quyidagi manzil (ushbu manzil CERN tadqiqot markazida yuzaga kelgan server WWW sahifasiga o'tkazadi) Html bo'yicha olib, unda WWW sistemasi haqida texnik axborotlar va boshqa ko'p ma'lumotlarni qurish mumkin. WWW serveri ro'yxatida mavzular bo'yicha birlashgan guruhlar ro'yxatini, mamlakatlar bo'yicha va axborot servisi turlari bo'yicha turli ma'lumotlar tanlanadi. U yerda WWW server va mijoz ta'minoti haqida ma'lumotlar ham olinadi.

Hujjatlar bilan ishlashni tezlashtirish. Hujjatlar bilan ishlashni tezlashtirish maqsadida Webda ma'lum buyruqlar mavjud. WWW buyruqlar ro'yxati quyidagilardir:

B	avvalgi hujjatga qaytish;
O	hujjatning keyingi sahifasiga o'tish;
Ho	Boshlang'ich hujjatga qaytish;
G	ko'rsatilgan axborot manbaiga bevosita o'tish;
B	sahifani chiqarish (yordam berish yo'li bilan);
I	joriy hujjatda boshqa hujjatlarni murojaatlarini ko'rsatish;
M	programmadan foydalanish haqidagi ma'lumotlarni ekranga chiqarish
N	avvalgi hujjatdan keyingi murojaatga o'tish;
Con	hujjatga murojaat bilan o'tish;
Quit	WWWdan chiqish;
V	ko'rib chiqilgan hujjatlarning ro'yxatini chiqarish;
Vcon	Ko'rsatilgan hujjatga qaytish;
Return	bir sahifa pastga o'tish;
T	hujjatning keyingi sahifasiga qaytish;
U	hujjat ichida bir sahifaga yuqori chiqish.

HTML tili. WWW sistemasidan qandaydir hujjat yoki xabar olsangiz, ekranda yaxshi formatlangan, o'qish uchun qulay matn

paydo bo'lganini ko'rasiz. Bu WWW hujjatlarida qandaydir ma'lumotlarni ekranda boshqarish imkoniyati ham mavjud ekanini anglatadi. Hujjatlar tayyor, siz foydalanuvchining qaysi kompyuterda ishlashini bilmaysiz, hujjatlar aniq bir kompyuter platformalariga mo'ljallangan yoki qaysidir format bilan saqlanishini oldindan ayta olmaysiz. Kompyuterda ishlayotgan foydalanuvchi qaysi terminalda ishlashidan qat'i nazar, yaxshi formatlangan hujjatni olishi kerak. Bu muammoni HTML andoza tili hal qiladi. HTML (Hyper Text Markup Language — gipermatniy belgilash tili) WWW sistemasi uchun hujjat tayyorlashda ishlatiladi. HTML hujjatning tuzilishini ifodalovchi uncha murakkab bo'lmagan buyruqlar majmuidan iborat. HTML buyruqlari orqali matnlarni istagancha shaklini o'zgartirish, ya'ni matnning ma'lum bir qismini ajratib olib boshqa faylga yozish, shuningdek boshqa joydan turli xil rangli tasvirlarni qo'yish mumkin. U boshqa hujjatlar bilan bog'laydigan gipermatniy aloqalarga ega.

HTML — hujjat tuzilishi. HTML tili andozasi bo'yicha hujjatga <HEAD> va <BODY> larini kiritish tavsiya etiladi. Brauzer HTML hujjatni o'qiganida, ularning borligi hujjat bo'limlarini aniq ko'rsatadi. Agar ular bo'lmasa ham brauzer HTML hujjatni to'g'ri o'qiydi, lekin hujjat bo'limlari bir-biridan ajralib turmaydi.

WEB sahifalarini o'qish vositalari (Browserlar). WWW sistemasi bilan ishlashda ma'lumotlarni qulay ko'rinishda tasvirlash uchun kompyuterga maxsus Browser (yo'llovchi) programmasini o'rnatish kerak. WWW browser bu WWW sistemasi bilan o'zaro hamkorlikda ishlovchi amaliy programmalaridir. WWW hujjatlari gipermatn hisoblanadi. Kompyuter imkoniyatlaridan kelib chiqib, gipermatnlar oddiy matnlardan hujjat tuzilishining borligiga karab farq qiladi. Ko'pgina Browserlarda Internetning boshqa servislariga ham kirish imkoni bor. Masalan, bunga FTP, Gopher va WAIS serverlari, telekonferentsiya serveri UseNet hamda Telnet serverlari kiradi.

HTML va boshqa programma vositalari yordamida tayyorlangan Web sahifalarida foydalanuvchiga tushunarli ko'rinishda tasvirlash uchun maxsus programmalar ishlab chiqilgan bo'lib, bunday programmalar brauzer programmalar deb ataladi. Hozirda bir necha shunday programmalar ishlab chiqilgan bo'lib, ular tabiiy ravishda hujjatlarni qurishni turlicha tahrir qiladilar. Bular orasida keng tarqalgan Microsoft Internet Explorer va Netscape Navigator

programmalaridir. Birinchi programma tekinga berilsa (albatta, Windows litsenzion programmi mavjud bo'lsa), ikkinchisi tijorat shaklida (pulli) tarqatiladigan programmadir. Biz asosan Microsoft Internet Explorerga to'xtaymiz, chunki hozirda u Web sahifalarni qurishning yuksak qiroliga aylandi. U Windows98 da brauzer emas, balki, hatto sharhlovchi deb ham yuritiladi. Buning asosiy sababi, HTML va boshqa programma vositalaridan (Java, JavaScript) foydalanib tuzilgan Web sahifalarini uning foydalanuvchiga tushunarli ko'rinishda sharhlab berishidir. Shunday qilib, brauzerning asosiy vazifasi URL adreslarda joylashgan Web sahifalarni kompyuterga yuklash va uni foydalanuvchiga tushunarli ko'rinishda monitor ekranida ko'rsatib berishdir.

Dynamic HTML kengaytirish. HTML vaqt o'tishi bilan murakkablashib boradi. HTML andozalari ham takomillashib boradi. DHTML (Dynamic HTML) — HTML hujjatning yangi andozasidir. U quyidagi yangiliklarni amalga oshiradi. HTML — hujjatni qurishda mumkin bo'lgan hodisalar sonini kengaytirish. HTML — hujjatga uning mos elementining parametri bo'lgan formatlash stilini, ya'ni harf o'lchovi, matn rangi, abzas chekinishlari va hokazo imkoniyatlarini kiritish. Stilli formatlash, HTML hujjat ichida matnni rasmiylashtirish, variantlarini tasvirlash uchun xizmat qiladi. Bu esa HTML ichida mustaqil ravishda shriftlarni va uni o'lchamini, abzas chekinishlarini elementlarning ramkasi, ranglarini va boshqalarni berish imkoniyatini tug'diradi. Brauzer programma ko'rib chiqadigan hodisalar sonini ko'paytirish hujjat dizaynini yaxshilashga olib keladi. Bunday imkoniyatlar, ya'ni DHTML andozani qo'llash faqat MS Internet Explorer 4.0 dan boshlab amalga oshiriladi. Eski brauzer DHTMLni qo'llamasligi ham mumkinligini esda saqlash lozim. Har bir HTML-hujjat hujjat turini ko'rsatish bilan boshlanadi. Hujjat turi esa brauzerda hujjat qanday turda va qaysi versiyada ekanligini aniqlab, foydalanuvchilarga to'g'ri ko'rsatib berish uchun zarur.

<!DOCTYPE html>

Eng oddiy HTML-sahifa kamida quyidagi uchta tegdan iborat bo'ladi: <html>, <head> va <body>. <head> tegi odatda sahifaning sarlavhasi, kalit so'zlar, ta'rifi va boshqa ma'lumotlarni o'z ichiga oladi. Shuningdek, uning ichida tashqi hujjatlarga ya'ni stillarga bog'lanish mumkin. Uning tarkibidagi ma'lumotlar sahifada to'g'ridan-to'g'ri ko'rsatilmaydi. <body> tegi tarkibidagilar esa

brauzer oynasida to‘g‘ridan-to‘g‘ri namoyon bo‘ladi. Sahifaga stillarni bog‘lash uchun maxsus <link> tegidan foydalanamiz. Buning uchun uning href atributiga stil joylashgan manzilni, rel atributiga esa aynan stil ulayotganligimizni bildirish uchun stylesheet qiymatni yozamiz.

```
<head>
```

```
<title>Html Academy — mashg‘ulotlari</title>
```

```
</head>
```

<head> tegi ichida joylashuvchi yana bir muhim teg bu — <meta> tegi. U toq teg hisoblanib, brauzerga va qidiruv tizimlariga sahifa haqida bir nechta ma’lumotlarni beradi. Masalan, sayt kodirovkasi, sahifa ta’rifi va boshqalar. Buning uchun <meta> tegi bir nechta atributlarni yordamga chaqiradi.

- HTML sahifaning kodirovkasi charset atributi yordamida ko‘rsatiladi:

```
<meta charset="kodiroyka nomi">
```

- Eng keng tarqalgan zamonaviy kodirovka — utf-8.
- Kalit so‘zlar zanjiri esa <meta> tegining name atributiga keywords qiymati yordamida tuzilib, maxsus kalit so‘zlar qo‘shimcha content atributiga vergul bilan ajratilgan kalit so‘zlar berish orqali tuzib chiqiladi:

```
<meta name="keywords" content="muhim, kalit, iboralar">
```

sahifa haqidagi qisqacha ta’rif ham xuddi shu tartibda beriladi, ammo name atributi uchun description qiymati beriladi:

```
<meta name="description" content="qisqacha ta’rif">
```

- <body> tegida esa sahifada ko‘rinuvchi barcha teglar joylashadi. Masalan, <main> tegi sahifaning boshqa sahifalarda takrorlanmaydigan asosiy tarkibini o‘z ichiga oladi. Odatda bitta sahifada faqatgina bitta <main> tegi ishlatiladi.
- <header> tegi sahifaga kirish qismi bo‘lib, odatda sahifaning “bosh” qismi deyiladi, <footer> tegi esa o‘z navbatida sahifaning xulosa qismi bo‘lib, uni “oyoq” qismi deb ham yuritiladi.
- <section> tegi yordamida sahifaning katta ma’noga ega bo‘limini ajratib olishimiz mumkin.
- <article> tegi orqali esa, ma’lumotning mustaqil va tugallangan fikrga ega qismini belgilab olishimiz mumkin.
- Sahifadagi navigatsiyani ta’minlash maqsadida bizga <nav> tegi yordamga keladi. Va undan asosan boshqa sahifalarga o‘tish havolalari o‘rin oladi.

- <aside> tegi qo‘shimcha tarkibni belgilashda ishlatilib, bunday bloklar “saydbar”lar yoki yon qism deb yuritiladi.

Matnlarning asosiy tuzilishini tuzish uchun sarlavhalardan foydalanish mumkin. HTML da sarlavhalar oilasi keng foydalaniladi: <h1> dan tortib <h6> gacha. <h1> tegi eng muhim sarlavha hisoblanadi. <h6> esa eng quyi bosqichdagi sarlavha hisoblanadi.

<h1>HTML xususiyatlari</h1>

<h2>1 Bob. Kirish</h2>

<h3>1.1 bo‘lim. Tilning kelib chiqishi</h3>

Matnning asosiy tuzilishi sarlavhalar yordamida tuzilsa, kichik ko‘rinishdagilari esa paragraflar yordamida tuziladi. Sahifaga paragraflar qo‘shish uchun <p> tegidan foydalanamiz. Odatda paragraflar yangi satrdan boshlanib, ular boshqa kontentlardan ma’lum bir kenglikda joylashadi.

HTML hujjatlari – bu matnli fayllar bo‘lib, ularga belgilash teglari deb nomlangan maxsus kodlar kiritilgan. Bu teglar Web-brauzerlarga matn va grafiklarni qanday qilib sharhlash va aks ettirish lozimligini ko‘rsatib turadi. HTML fayl – bu oddiy matnli fayl. Shuning uchun uni istagan matn redaktorida, masalan MS Word yoki oddiy «Bloknot»da yaratish mumkin. **HTML sahifa nima?** - bu oddiy text fayl bo‘lib, **.html** qisqartmasiga ega.

Eslatib o‘tish joiz, hujjat yaratilgach, uni matn formatida saqlash kerak. Lekin, bu ishni bajarishdan oldin uning kengaytmasini o‘zgartirish, ya’ni TXT o‘rniga HTML yoki HTMni qo‘yishni esdan chiqarmaslik kerak. HTML va HTM kengaytmasi HTML fayl uchun **standart** hisoblanadi. Bundan tashqari, bu kengaytmalar kompyuterga faylda matnlardan tashqari HTML kodlari ham mavjudligini ko‘rsatib turadi. HTML tili harflar razmeriga befarqdir, ya’ni bosh va kichik harflar bir xil qabul qilinadi. Lekin teglarni yozishda ko‘pincha bosh harflardan foydalaniladi.

Bundan tashqari, HTML sahifani yaratish uchun maxsus dasturlarni qidirib topib, sotib olish shart emas. Matn tahrirlovchi har qanday dastur orqali HTML sahifa yaratish mumkin.

Masalan: Windows muhitidagi matn muxarrirlari: **Notepad**, **TextPad**, **UltraEdit**, **EdutPlus**. Ana shunday matn tahrirlovchi oddiy dasturlardan biri bo‘lgan **Notepad** (**БЛОКНОТ**), Windows muhitida ishlovchi har bir kompyuterda mavjud.

Ba'zi matn muxarrirlarida HTML hujjatni web brauzerda sinab ko'rish tugmasi mavjud. HTML hujjatni yaratishga mo'ljallangan maxsus dasturlar (HTML muxarrirlar) ham mavjud: **FrontPage**, **Adobe GoLive**, **Macromedia Dreamweaver**, **Nestcape Composer**. Muxarrirlar 2 turga bo'linadi:

- kod muxarrirlari;
- WYSIWYG texnologiyasi (What You See Is What You Get – nimani ko'rsang o'shani olasan) asosida ishlaydigan muxarrirlar. Bu muxarrirlar yordamida foydalanuvchi HTML komandasi va elementlarini yozmaydi, oddiy matn muxarrirlaridek matn yozadi, tasvirlarni kerakli joyga joylashtiradi, formatni o'zgartiradi va h.k. xolos.

Web-sahifa ko'rinishi va aks ettirilayotgan axborotning qanaqaligidan qat'iy nazar, HTML va WWW spetsifikatsiyasiga asosan har bir Web-sahifada ishtirok etishi zarur bo'lgan quyidagi **to'rtta teglar** mavjud:

1. **<HTML>** brauzerga hujjat HTML tilida yozilganligi to'g'risida xabar beradi.
2. **<HEAD>** HTML–hujjatning kirish va bosh qismini belgilaydi.
3. **<BODY>** asosiy matn va axborotni belgilaydi.
4. **<ADDRESS>** bu Web-sahifa to'g'risida ko'proq to'la-to'kis axborot olish uchun kerak bo'ladigan elektron pochta manziliga ega.

Bu teglar Web-brauzerga HTML–hujjatning har xil qismlarini aniqlash uchun juda zarurdir, lekin ular Web-sahifaning tashqi ko'rinishiga to'g'ridan-to'g'ri ta'sir etmaydi. Ular HTMLga kiritilgan navbatdagi yangi ma'lumotlar uy sahifalarida to'g'ri sharhlash, shu bilan birga barcha Web-brauzerlarda bir xil ko'rinishga ega bo'lishi uchun juda zarurdir.

Demak, **HTML tili** andozasi bo'yicha hujjatga **<HEAD>** va **<BODY>** teglarini kiritish tavsiya etiladi. Brauzer HTML hujjatni o'qiganida, ularning borligi hujjat bo'limlarini aniq ko'rsatadi. Agar ular bo'lmasa ham brauzer HTML hujjatni to'g'ri o'qiydi, lekin hujjat bo'limlari bir-biridan ajralib turmaydi.

Shunday qilib, to'g'ri tuzilgan HTML hujjat quyidagi tuzilishga ega:

<HEAD>

Sarlavhaga oid ma'lumot

</HEAD>

<BODY>

Hujjatning mazmuni

</BODY>

Bunda **<HEAD>**, **</HEAD>** orasida joylashgan sarlavhaga oid ma'lumot qismida odatda foydalanuvchiga e'tiborsiz, lekin brauzer uchun lozim ma'lumot beriladi.

<BODY>, **</BODY>** orasiga esa to'raligicha uning operatorlari ketma-ketligi joylashtiriladi.

Masalan, sizning Web-serveringizda barcha HTML — hujjatlarni ko'radigan va ularning ro'yxatini tuzadigan dastur ishga tushirilgan. U **<HEAD>** teglari ichida joylashgan matnlarni ko'radi, xolos (bu erda hujjatlar nomi ham joylashtirilgan bo'ladi). Shunday qilib, agar uy sahifalarida **<HEAD>** va **</HEAD>** teglari bo'lmasa, u holda u ro'yxatga kiritilmaydi. Anchagina nomi chiqqan Web-serverlar – qidiruv vositalarining ko'pchiligi mana shunday ishlaydi. Ular axborotlarni **<HEAD>** teglaridan oladi.

Yuqoridagi fikrlarga asoslangan holda Web-sahifada ishtiroq etishi zarur bo'lgan quyidagi **to'rtta asosiy teglarni** tavsiflashga harakat qilamiz.

<HTML> va **</HTML>** teglari. Bu teglar brauzerlarga ular orasidagi matni xuddi HTML matni kabi sharhlash (izohlash) zarurligi to'g'risida xabar beradi, chunki HTML-hujjatlari faqat matnlidir. **<HTML>** teg esa, faylning gipermatn bog'lanish tilida yozilganligini bildirib turadi.

<HEAD> va **</HEAD>** teglari. Bu teglar Web-sahifalar nomlarini belgilaydi. Buning uchun **<HEAD>** va **</HEAD>** teglar orasida Web-sahifa nomi kiritiladi. Ya'ni HEAD bo'limi sarlavha hisoblanadi va u majburiy teg emas, biroq mukammal tuzilgan sarlavha juda ham foydali bo'lishi mumkin. Sarlavha qismining maqsadi hujjatni tarjima qilayotgan dastur uchun mos axborotni yetkazib berishdan iborat. Hujjat nomini ko'rsatuvchi **<TITLE>** tegidan tashqari bu bo'limning qolgan barcha teglari ekranda aks ettirilmaydi. Odatda **<HEAD>** tegi darhol **<HTML>** tegidan keyin keladi. **<TITLE>** tegi sarlavhaning tegidir va hujjatga nom berish uchun xizmat qiladi. Hujjat nomi **<TITLE>** va **</TITLE>** teglar orasidagi matn qatoridan iborat. Bu nom brauzer oynasining sarlavhasida paydo bo'ladi (bunda sarlavha nomi 60 belgidan ko'p

bo'lmisligi lozim). O'zgartirilmagan holda bu matn hujjatga «zakladka» (**bookmark**) berilganda ishlatiladi. Hujjat nomi uning tarkibini qisqacha ta'riflashi lozim. Bunda umumiy ma'noga ega bo'lgan nomlar (masalan, **Homepage**, **Index** va boshqalar) ni ishlatmaslik lozim. Hujjat ochilayotganda birinchi bo'lib uning nomi aks ettirilishi, so'ngra esa hujjat asosiy tarkibi ko'p vaqt olib, kengayib ketishi mumkin bo'lgan formatlash bilan birga yuklanishini hisobga olgan holda, foydalanuvchi hech bo'lmaganda ushbu axborot qatorini o'qiy olishi uchun hujjatning nomi berilishi lozim. Har bir HTML hujjat faqatgina bitta nomga (sarlavhaga) ega bo'ladi. So'ngra uning oldi va orqa tomonlarini <TITLE> va </TITLE> teglari bilan belgilanadi. U, odatda brauzer darchasi sarlavhasida ko'rsatiladi. Konteyner <TITLE> tegini hujjat faylining nomi bilan adashtirmaslik kerak. Aksincha u fayl nomi va manziliga butunlay bog'liq bo'lmagan matn satridir. Fayl nomi kompyuterning operatsion tizimi (OT) orqali qat'iy ravishda aniqlanadi. Shu bilan birga, hujjatlar nomi (teg <TITLE> bilan birga) ni hujjat ichidagi ko'pincha <H> teglari bilan joylashadigan sarlavhalardan farqlash kerak bo'ladi.

<**BODY**> va </**BODY**> teglari. <BODY> va </BODY> teglari <HEAD> kabi HTML hujjatning maxsus qismlarini belgilashda ishlatiladi. <BODY> teglari egallab olgan matn hujjatning asosiy qismi hisoblanadi.

Matnning katta qismi va boshqa axborotlar ham uning tarkibiga kiritiladi. Quyida <BODY> tegining bir qator parametrlarini keltiramiz.

<**BODY**> tegi parametrlari:

ALINK – faol murojaat (ссылка) ning rangini belgilaydi.

BACKGROUND – fondagi tasvir sifatida foydalaniluvchi tasvirni belgilaydi.

URL - manzilini belgilaydi.

BOTTOMMARGIN – hujjatning quyi chegaralarini piksellarda belgilaydi.

BGCOLOR – hujjat fonining ranglarini belgilaydi.

BGPROPERTIES –agar **FIXED** qiymati o'rnatilmagan bo'lsa, fon tasviri aylantirilmaydi.

LEFTMARGIN – chap chegaralarni piksellarda belgilaydi.

LINK – hali ko'rib chiqilmagan silkasining rangini belgilaydi.

RIGHTMARGIN – hujjat o‘ng chegarasini piksellarda o‘rnatadi.

SCROOL – brauzer darchalari harakatlantirish (prokrutka) yo‘laklarini o‘rnatadi.

TEXT – matn rangini aniqlaydi.

TOPMARGIN – yuqori chegarasini piksellarda o‘rnatadi.

VLINK – ishlatilgan murojaat rangini belgilaydi.

BOTTOMMARGIN, **LEFTMARGIN**, **RIGHTMARGIN** va **TOPMARGIN** parametrlari matn chegarasi va darcha chetlari orasidagi masofani piksellarda belgilaydi.

<ADDRESS> va </ADDRESS> teglari. Bu teglar mazkur sahifaga nisbatan kimdadir savol yoki fikr tug‘ilib qolgan taqdirda kimga murojaat qilish kerakligi to‘g‘risidagi axborotlarni o‘z ichiga oladi. **<ADDRESS>** teglari bu axborotlarni asosiy blokdan ajratib olish uchun ishlatiladi. **Uy sahifasiga** bu teglarni kiritish uchun quyidagi qadamlarni bajaring:

<BODY> va **</BODY>** teglari orasida ismingizni va elektron pochta adresini tering. So‘ngra ismingiz va adresingizga **<ADDRESS>** tegini kiring. Ism (nom) va adresdan so‘ng, yopuvchi **</ADDRESS>** tegni kiring.

<P> bu Web-sahifa Web-dizayner bo‘lishi mumkinligini namoyish qilish maqsadida yaratilganidir. Buning uchun Web-serverga sozlanishi qiyin dastur talab qilinmaydi. Bunda sizning operatsion tizimingiz muvaffaqiyat bilan uning o‘rnini bosa oladi. **</P>** Bu yerda terminologiya to‘g‘risida biroz oldindan kelishib olishimiz kerak. HTML–hujjatda xato bo‘lsa, lekin baribir brauzer tomonidan chidab bo‘larli darajada ko‘rsatilsa, bunday hujjatni **yaxshi rasmiylashtirilgan hujjat** deyiladi. Aksincha, rasmiylashtirishda xatosi bo‘lmagan HTML hujjat **standart hujjat** deyiladi.

Sahifamiz kodiga yana bir bor nigoh tashlaymiz. Barcha **HTML hujjat** juft teglar – **<HTML>** va **</HTML>** ichida joylashganligi ma‘lum bo‘ladi. Bu standart HTML–hujjatlarni rasmiylashtirishning **birinchi qoidasidir. Ikkinchi qoida** bo‘yicha HTML–hujjat ikkita bir-biriga teng bo‘lmagan sektsiyaga bo‘lingan bo‘ladi. **Birinchi** (kichik) **sektsiya** – bu HTML sarlavha. HTML sarlavha juft teglar – **<HEAD>** va **</HEAD>** bilan ajralib turadi. U brauzer darchasida aks etmaydi, lekin brauzer o‘z ehtiyojlari uchun foydalanadigan xizmat axborotlarini o‘z ichiga oladi. **Ikkinchi** (katta) **sektsiya** – bu hujjat jismi deb ataladigan shaxsiy hujjat. Xuddi mana

shu hujjat jismi brauzer darchasida aks ettiriladi. Jism juft teglar – <BODY> va </BODY> bilan ajralib turadi. Bu yerdan standart HTML hujjatlarni rasmiylashtirishning **ikkinchi qoidasi** kelib chiqadi: har bir hujjatda HTML sarlavha va test gipermurojaatlari (ССЫЛКА) bo‘lishi va bu ikkala sektsiyalar to‘g‘ri rasmiylashtirilgan bo‘lishli shart.

Nazorat savollari:

1. WWW nima?
2. Gipermatn va Gipermedia nima?
3. Teg nima?
4. Teg turlarini sanab bering?
5. Body teglari parametrlari qaysilar?

2-MAVZU. MANT BILAN ISHLASH. BLOKLI VA SATRLI BELGILASH ELEMENTLARI.

Reja:

1. Matn bilan ishlash.
2. Blokli belgilash elementlari.
3. Satrli belgilash elementlari.

HTML — matnlar bilan ishlash hujjatning asosiy tanasi ochiluvchi va yopiluvchi qismdan iborat. <body> tegi ichiga olib yoziladi. Bu teg bilan birgalikda quyidagi atributlarni qo'llashimiz mumkin.

- 1.Topmargin-hujjatning peksillardagi inini aniqlaydi.
- 2.Leftmargin-o'ng va chap tomonlari orasidagi masofani aniqlaydi.
- 3.background-hujjatning matni yoziladigan maydonning fonini, rangini aniqlaydi.
- 4.Bgcolor-matn yozilgan sohaning rangini aniqlaydi.
- 5.Text-matnning rangini aniqlaydi.
- 6.Link-hujjatga quyilgan gipermurojaat so'zlarning rangini aniqlaydi.
- 7.Alink-gipermurojaat so'zini ishga tushurish vaqtidagi rangini aniqlaydi.
- 8.Vlink-gipermurojaat so'zining ishdan keyingi holatini, rangini aniqlaydi.

Bu yerda ko'rsatilgan rangni aniqlovchi atributlar bgcolor, text, Alink, vlinklar kompyuterdagi standart RCB modulidagi ranglarning ixtiyoriy bittasini qabul qilishi mumkin. Bu ranglar 16lik sanoq sistemasidagi kodlar orqali aniqlanadi. Agar , rangni nomi bilan bermoqchi bo'lsak u holda, ingliz tilidagi so'zlardan beriladi. Misol uchun hujjatdagi textning rangini quyidagi ikki xil shaklga kiritish mumkin.

- 1.text="red"
- 2.text="#ff0000"

Umuman , Web hujjatda yuqorida ko'rsatilgan atributlardan foydalanishni quyidagi misol orqali ko'rsatish mumkin.

```
<html> <title>. </title> <body background="images/bricks.jpg" Bgcolor="202020"text="#FFFFFF" Link="#FF0000" vlink="505050" Topmargin="30" leftmargin="40"> . . . Hujjat matni . . . </body> </html>
```

Web hujjatni yaratishda Bgcolor va text atributlarini bir vaqtda yozilishi shart .Agarda, birortasida qatnashmay qolsa, ko'rsatish rang qabul qilinmaydi. Uning o'rniga Windows sistemasidagi asosiy rang sifatida qabul qilingan qora rang ishlaydi.

Gipermurojaatlar. Gipermurojaat HTML hujjatda quyidagi teg orqali ko'rsatiladi. A-tegi. Bu teg quyidagi atributlarni o'zida ishlatishda qabul qilingan.

1.Href. – bu atribut giper murojaatning adresini yozish uchun ishlatiladi. Ko‘p hollarda esa hujjatning kerakli sohaga o‘tish joyini ko‘rsatadi. Hujjatdagi umumiy ko‘rinishi quyidagicha.

2. Http ://...Web hujjatdagi URL adresga o‘tish murojaatini bildiradi.

3.Mailto:...elektron pochta adresga murojaatni bildiradi.

4. Telnet://...telekonferensiyalarga qatnashish uchun qo‘yilgan gipermurojaat adreslarini yozish uchun ishlatiladi.

5. Name gipermurojaatda mundareijalar turli xil, ilovalar qo‘yish uchun ishlatiladi.

Ushbu atributlar ishtiroq etgan quyidagi misolni ko‘rib chiqamiz. <html> <title>...</title> <body> . . . tarix </body> </html> Href atributini qo‘llashga misol. <html> <title>...</title> <body> <ahref=’’mailto :Nusrat@mail.ru’’> Xatni jo‘natish </body> </html> Matndagi shriftlar bilan ishlash.

1. h1, h2, h3, h4, h5, h6 teglari, bu teglar hujjatning sarlavhasini yozish uchun ishlatiladi. Har biri alohida o‘lchamga ega bo‘lgan matnni yozishda qo‘llaniladi <h1>matn <h2>. Bu matnlarning o‘lchami tik o‘sish tartibiga qarab kichrayadi, bu teglarning har biri uchun quyidagi atributlardan foydalanishimiz mumkin.

1.1 .Align-sarlavhaning gorizonta bo‘yicha joylashishini belgilaydi.

1.2.Left — sarlavhaning gorizonta bo‘yicha chap tomonga joylashishini belgilaydi.

1.3 .Right — o‘ng tomonga joylashishini belgilaydi.

1.4 Center — o‘rtaga joylashishini belgilaydi.

2. P — hujjatda paragraflarni aniqlash uchun ishlatiladi. Bu tegga ham yuqoridagi atributlardan foydalanib turli xil o‘zgarishlar kiritishimiz mumkin.

2.1. Align — ikkala tomonini to‘g‘irlash.

2.2.Left — chap tomonga to‘g‘irlash.

2.3.Right — o‘ng tomonga to‘g‘irlash.

2.4.Center — o‘rtaga to‘g‘irlash.

3. BR — bu teg matnni alohida satrlga yozish uchun qo‘llaniladi.
matn
matn

4. HR — bu teg hujjat ichida alohida gorizontaal chiziq qo'yish uchun ishlatiladi.
matn <Hr>
 matn. Bu teg bilan birgalikda quyidagi atributlar ham ishlatiladi

4.1.Width — chiziqning uzunligini aniqlash uchun ishlatiladi. Uzunlik o'lchami peksillarda yoki protsentlarda berilishi mumkin

4.2. size — chiziqning qalinligini peksillarda aniqlaydi.

4.3.Align — chiziqning o'ng va chap tomoniga bir xil tekislikda joylashishini ta'minlaydi

4.4.Lift — chap tomonga joylashishini ta'minlaydi.

4.5.Right — o'ng tomonga joylashishini ta'minlaydi.

4.6.Center — o'rtaga joylashishini ta'minlaydi.

4.7.Color — chiziqlar rangini aniqlaydi.

4.8.Noshade — bu atribut chiziqning hujjat fonigat tekis joylashishini ta'minlaydi. Agarda bu atribut qo'llanilmasa chiziq hujjat fonidan ajratilgan holda alohida hajimga ega bo'lgan tarzida tasvirlanadi. Misol: <pre> Matn1 Matn2 Matn3 </pre> Misoldan ko'rinib turibdiki, Matn2 va Matn3 lar satr boshidan ma'lum bo'sh joy qoldirib yozilgan ushbu matnlar hujjatda qanday yozilgan bo'lsa brauzerlar oynasida ham xuddi shunday ko'rinadi. Agar ushbu teg qo'llanilmasa matnlarning joylashishi o'zgarib ketadi .Bu teg yordamida turli xil shakllarni tasvirlash mumkin.

5. Font — bu teg yordamida matn shriftining turini , rangini, o'lchamini aniqlash mumkin. Bu teg bilan birgalikda quyidagi atributlarni ishlatish mumkin.

5.1.Size — bu atribut 1...7gacha qiymat qabul qiladi.

5.2.Color — shrift rangi

5.3.Face — shrift turi,bu atribut ko'p hollarda ishlatilmaydi.Chunki atributlar berilgan shrift turi foydalanuvchi kamyuterning fond bazasida bo'lmasligi mumkin. Misol: Matn matn

7. I — bu tegning ochiluvchi va yopiluvchi qismlari o'rtasiga olib yozilgan matn kursiv shakllarda tasvirlanadi. Misol: <i>matn</i>

8. EM — bu teg orasiga olib yozilgan matn alohida o'lcham yoki rangda tasvirlanadi,ya'ni matn ichida e'tibor berish kerak bo'lgan so'zlar va turli iboralarni asosiy matndan ajratib ko'rsatish uchun ishlatiladi Misol: matn. B — bu teg ichiga olib yozilgan matn qalin shriftlarda tasvirlanadi: matn 10. Strong — bu teg

ichiga olib yozilgan matn `` tegiga o'xshash qalin shriftlarda tasvirlanadi. `matn` U — bu teg ichiga olib yozilgan matnning tagiga chiziq chiziladi. `<u>matn</u>`. BIC bu teg ichiga olib yozilgan matn asosiy matnga nisbatan kattaroq shriftida tasvirlanadi. Bu tegning o'lchami quyidagiga teng kuchli `<big>matn</big>` `matn`. Small — bu teg ichiga olib yozilgan matn asosiy matnga kichikroq o'lchamlar bilan tasvirlanadi. `matn`. Sup — Bu teg ichiga olib yozilgan matn kichik o'lchamlar daraja yoki yuqori indeks shaklida tasvirlanadi. O'lchami asosiy matnga nisbatan bir baravar kichik bo'ladi. Sub — bu teg orasiga olib yozilgan matn quyidagi indeks shaklda tasvirlanadi. HTML — veb-sahifalardagi qurilish bloklari sifatida ishlaydigan turli elementlardan tashkil topgan. Ushbu elementlarning har biri ikkita toifaga kiradi: blok darajadagi elementlar yoki ichki elementlar. Ushbu ikki turdagi elementlar orasidagi farqni tushunish veb-sahifalarni yaratishda muhim qadamdir.

Blok-darajali elementlar. Blok-darajadagi element — bu veb-sahifada yangi satrni boshlaydigan va uning yuqori elementining mavjud gorizontal kengligining kengligini kengaytiruvchi HTML element. Paragraf yoki sahifa bo'linmalari kabi katta bloklar hosil qiladi. Aslida, ko'pchilik HTML elementlari blok darajadagi elementlardir. Blok-darajadagi elementlar HTML hujjat tanasining ichida ishlatiladi. Ular satr elementlari va boshqa blok-daraja elementlarni o'z ichiga olishi mumkin.

Inline Elements. Blok-darajali elementlardan farqli o'laroq, inline element quyidagilarni bajarishi mumkin:

- Bir qatorda boshlaydi;
- Yangi qatorni boshlamaydi;
- Uning kengligi faqat teglar bilan belgilanadigan darajada kengayadi.

Inline elementning namunasi teg, ya'ni matn tarkibining shriftini qora rangda joylashtiradi. Bir satr ichidagi element odatda faqat boshqa satr ichidagi elementlarni o'z ichiga oladi yoki tanaffus yorlig'i kabi hech narsaga ega bo'lmaydi. HTMLda elementning uchinchi turi ham bor: ular hech qachon ko'rsatilmaydi. Ushbu elementlar sahifa haqida ma'lumot beradi, lekin veb-brauzerida ko'rsatilmaydi.

Nazorat savollari:

1. Inline Elements nima?
2. Blokli belgilash elementlari nima?
3. Font atributi nima?
4. Size atributi nima?

3-MAVZU. RO‘YXATLAR. JADVALLAR.

Reja:

1. HTMLda ro‘yxatlar va ularning turlari.
2. HTMLda jadvallar.
3. HTMLda ro‘yxatlar va jadvallar bilan ishlash.

HTML da matnni yaxshi shakllangan, tartiblangan va semantik tarzda taqdim etish uchun ro‘yxatlardan foydalaniladi. HTML da uch xil ro‘yxatlar mavjud: tartiblangan (raqamli yoki harfli) va tartiblanmagan (belgili) ro‘yxat hamda atamalarni tariflash ro‘yxati.

` ... ` — tartiblangan (raqamli va harfli) ro‘yxatni aniqlash tegi. Teg atributlari: **type**=«1, A, a, I, i — ro‘yxat markerlarini aniqlash»

start=«ro‘yxatning birinchi raqamini aniqlash». Tartiblangan ro‘yxat ichidagi har bir elementi quyidagi teglar yordamida aniqlanadi:

- `<lh> ... </lh>` — ro‘yxat sarlavha elementini aniqlash tegi.
- ` ... ` — ro‘yxat oddiy elementini aniqlash tegi.

Tartiblangan ro‘yxat elementlarini aniqlash ushbu ikkita teglari quyidagi atributlarga ega:

- **align** = elementni hujjatdagi joylashuvi (chap «left» , o‘ng — «right» , o‘rta — «center» yoki ikkala tomon bo‘yicha — «justify»)
- **title** = «sichqoncha bilan elementga ko‘rsatilganda, ekranda chiquvchi yordamchi izoh matni»

Misol:

```
<!DOCTYPE html>
<html>
<head>
  <!-- Sahifaning ekranga chiqmaydigan qismi -->
  <title> ..: Mening birinchi sahifam .. </title>
  <link rel="shortcut icon" href="picture/icon.png"
type="images/png">
</head>
<body>
  <!-- Tartiblangan ro‘yxat -->
  <ol type="I" start="1">
  <lh> IT-ko‘rs rejasi: </ln>
```

```

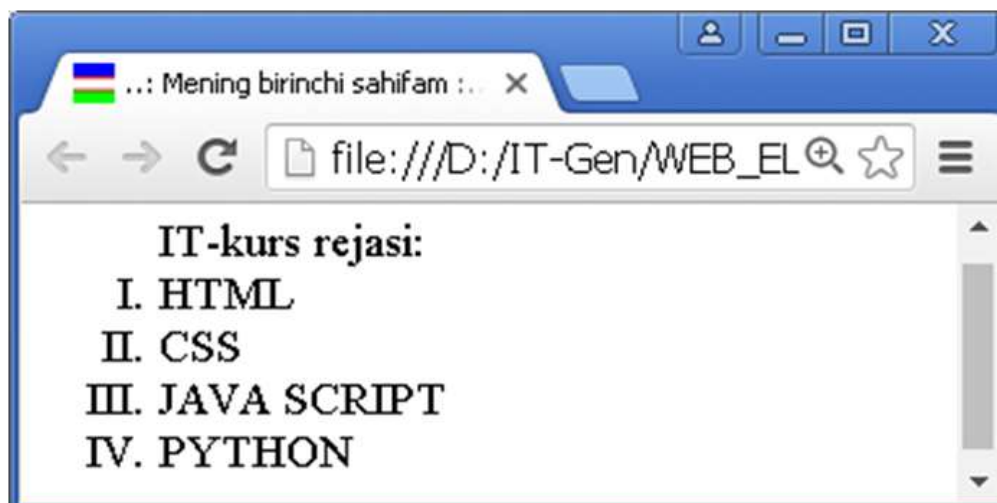
<li> HTML </li>
<li> CSS</li>
<li> JAVA SCRIPT</li>
<li> PYTHON</li>
</ol>

```

```

</body>
</html>

```



Bu misolda web sahifamizda rim raqamlar bilan boshlangan 4-ta elementdan iborat tartiblangan ro'yxat yaratilgan.

** ... ** — **tartiblanmagan (belgili) ro'yxatni** aniqlash tegi.

Teg atributlari:
type=«disk — doira, circle -aylana yoki square -to'rtburchak kabi ro'yxat markerlarini aniqlash».

Tartiblanmagan ro'yxat ichidagi har bir elementi quyidagi teglar yordamida aniqlanadi:

<lh> ... </lh> — ro'yxat sarlavha elementini aniqlash tegi.

** ... ** — ro'yxat oddiy elementini aniqlash tegi.

Tartiblanmagan ro'yxat elementini aniqlash ushbu ikkita teglari quyidagi atributlarga ega:
align = elementni hujjatdagi joylashuvi (chap -«left» , o'ng — «right» , o'rta — «center» yoki ikkala tomon bo'yicha — «justify»).
title = «sichqoncha bilan elementga ko'rsatilganda, ekranda chiquvchi yordamchi izoh matni».

Misol:

```

<!DOCTYPE html>
<html>

```

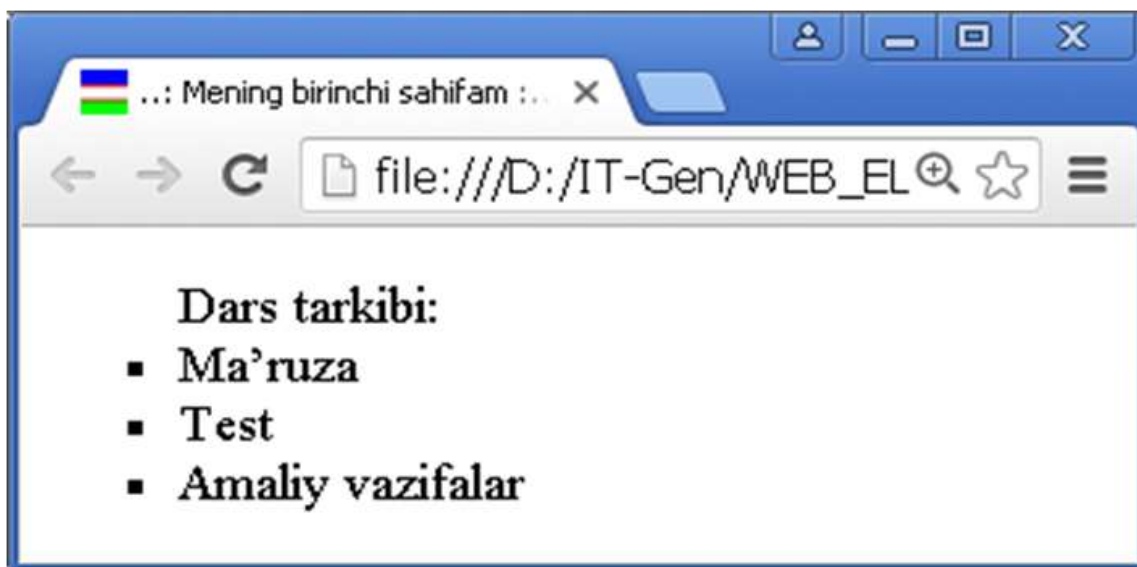


```

<head>
  <!-- Sahifaning ekranga chiqmaydigan qism -->
  <title> ..: Mening birinchi sahifam .. </title>
  <link rel="shortcut icon" href="picture/icon.png"
type="images/png">
</head>

<body>
  <!-- Tartiblanmagan ro'yxat -->
<ul type="square">
<lh> Dars tarkibi: </ln>
<li> Ma'ruza </li>
<li> Test </li>
<li> Amaliy vazifa </li>
</ol>
</body>
</html>

```



Bu misolda web sahifamizda to'rtburchak belgisi bilan boshlangan 3-ta elementdan iborat tartiblanmagan ro'yxat yaratiladi. Yana bir ro'yxat turi bu atamalarni (terminlarni) ta'riflash ro'yxati.

<dl> ... </dl> — **atamalarni (terminlarni) ta'riflash ro'yxatni** aniqlash tegi. Ushbu ro'yxat ichida quyidagi teglar yordamida juftlik elementlar aniqlanadi:

<dt> ... </dt> — asosiy atamani (terminni) aniqlash tegi.

<dd> ... </dd> — atamaning (terminning) ta'rifini aniqlash tegi.

Ushbu ikkita — atama va ta'riflar ro'yxat teglari quyidagi atributlarga ega:

align = elementni hujjatdagi joylashuvi (chap -«left» , o'ng — «right» , o'rta — «center» yoki ikkala tomon bo'yicha — «justify»)

title = «sichqoncha bilan elementga ko'rsatilganda, ekranda chiquvchi yordamchi izoh matni»

Misol:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<!-- Sahifaning ekranga chiqmaydigan qism -->
```

```
<title> ..: Mening birinchi sahifam .. </title>
```

```
<link rel="shortcut icon" href="picture/icon.png" type="images/png">
```

```
</head>
```

```
<body>
```

```
<!-- Atamalarni ta'riflash ro'yxati -->
```

```
<dl>
```

```
<dt> HTML </dt>
```

```
<dd> Hyper Text Marko'p Language — Giper Matnli Markerlash Tili yani web-sahifalarni yaratish tili. Ammo dasturlash tili emas.</dd>
```

```
<dt> CSS </dt>
```

```
<dd> Web -sahifaning stilini (elementlar va bloklarning tashki ko'rinishini) aniqlash fayllari</dd>
```

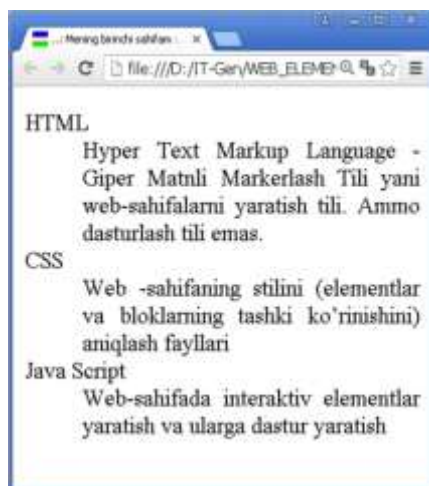
```
<dt> Java Script </dt>
```

```
<dd> Web-sahifada interaktiv elementlar yaratish va ularga dastur yaratish</dd>
```

```
</dl>
```

```
</body>
```

```
</html>
```



Bu misolda web sahifamizda 3 ta atamalar ta'rifini aniqlash ro'yxat yaratiladi.

Jadvallar-bu Web sahifalarda ma'lumotlarni (vizual) tashkil qilishning muhim vositalaridan biridir. Bizga ma'lumki HTMLda, bezatish elementlarini aniq koordinatalar bo'yicha joylashtirish vositalari mavjud emas. Shuning uchun jadvallardan shu maqsadda foydalanish maqsadga muvofiq. Jadvallarni Web sahifaga joylashtirib, ularning kataklariga bezatish elementlarini joylashtirish mumkin. Bundan tashqari yana freymlar deb ataluvchi vositalar yordamida ham bu ishlarni amalga oshirish mumkin. Lekin bu vosita oxirgi vaqtlarda Web masterlar orasida o'z ommaviyligini yo'qota boshladi. Biz freym'larga keyingi ma'ruzamizda batafsil to'xtalamiz.

Jadvallardan to'raligicha foydalanish qoidalarini bilish uchun ularning tuzilishini yaxshi bilish talab qilinadi. Jadval tuzishda ishlatiladigan teglarni tasniflashdan avval, oddiy jadval tuzish sxemasi bilan tanishib chiqsak. Har qanday jadval quyidagi sxema bo'yicha tuziladi:

<JADVALNI OCHUVCHI TEG>

<JADVAL QATORINI OCHUVCHI TEG>

<JADVAL BO'LAGINI OCHUVCHI TEG>

Matn, tasvir yoki jadval

</JADVAL BO'LAGINI YOPUVCHI TEG>

</JADVAL QATORINI YOPUVCHI TEG>

</JADVALNI YOPUVCHI TEG>

Har qanday jadvalda qatorlar va ustunlar mavjud bo'ladi. Jadvalning bo'lakchalarida (yacheyka) esa axborot kiritilishi mumkin. Yuqorida keltirilgan misoldan ko'rinib turibdiki, bitta

ustun, bitta qator va bitta bo‘lakchadan iborat jadvalni tuzish uchun jadval ochiladi va yopiladi. Agar bitta ustundan iborat va bir necha qatorni o‘z ichiga olgan jadvalni tuzish uchun, jadval qatorini ochuvchi va yopuvchi teglar orasiga bir nechta jadval bo‘lakchalarini ochuvchi va yopuvchi teglarini kiritish lozim.

Demak, HTML da jadvallar satrlardan tashkil topadi, satrlar esa o‘z navbatida yacheyka (katak)lardan tashkil topadi. Shunday ekan bu obyektlarning har birining o‘z parametrlari mavjud. Bu parametrlar yordamida ularning o‘lchamlarini o‘rnatish mumkin. Agar biz jadvalning kengligini oldindan aniqlab qo‘ygan bo‘lsak, masalan 100 piksel deb aniqlagan bo‘laylik va har bir satrda 4 ta yacheyka joylashgan bo‘lib bu yacheykalarining har birining uzunligi 30 pikseldan iborat bo‘lsin, u holda jadval kengligi 100 piksel bo‘lmasdan balki 120 piksel bo‘ladi. Agar birorta yacheyka kengligi 30 pikseldan ko‘p bo‘lgan grafik tasvir joylashgan bo‘lsa, u holda yacheykaning kengligi tasvir kengligiga mos ravishda kengayadi, buning natijasida esa butun jadvalning kengligi oshadi. Bu imkoniyat shuning uchun yaratilgan, buning natijasida har bir yacheykadagi ma‘lumot aniq va hech qanday yo‘qotishlarsiz to‘laligicha tasvirlanadi.

Shuni e‘tiborga olish kerakki jadvalning ustunlarini oldindan aniqlash imkoniyati yo‘q. Ustunlarni brauzerning o‘zi jadval satrlarini tahlil qilib aniqlaydi va jadvalni to‘laligicha tasvirlaydi.

Endi biz jadval teglari bilan tanishishni boshlaymiz. Jadval va uning tashkil etuvchi hamma elementlari **<table>** va **</table>** teglari orasida sarlavhasi esa **<caption>** va **</caption>** teglari orasida joylashtiriladi. Jadvallarda bundan tashqari ustunlar guruhini **<col>** va **<colgroup>** teglari yordamida aniqlash mumkin, hamda **<thead>** va **<tfoot>** teglar bilan mos ravishda jadvalning yuqorigi va pastgi sarlavhalarini (shapkalarini) tashkil qilish mumkin. Satrlar guruhini esa **<tbody>** tegi yordamida hosil qilamiz. Shunday qilib jadval asosan yuqorida ko‘rilgan teglardan tashkil topadi, qolgan boshqa jadval elementlari esa shu obyektlar ichida joylashadi. **<table>** tegida jadvalning barcha xususiyatlarini o‘rnatish uchun yyetarli bo‘lgan parametrlar mavjud. Jadvallarda balandlikni o‘rnatish parametri yo‘q, kenglikni esa **width** shart bo‘lmagan parametr bilan o‘rnatish mumkin. Jadvaldagi

yacheykaning balandligi yacheykadagi obyektning o'lchamiga mos ravishda hisoblanadi. Jadval chegaralarining kengligini o'rnatish uchun **border** parametridan foydalanamiz. Bu parametrqa manfiy bo'lmagan butun qiymat beriladi. Chunki u chegaradagi chiziqlarning kengligini (piksellarda) o'rnatadi. Agar biz bu parametrqa "nol" qiymat o'rnatib jadval chegaralari ko'rinmas holatga o'tadi. Bu esa bizga yacheykalarida Web sahifaning elementlari joylashgan ko'rinmas jadval tuzish imkoniyatini yaratadi. Yacheykalar o'rtasidagi masofani **cellspacing** parametri yordamida piksellarda o'rnatish mumkin. Xuddi shunga o'xshash **cellpadding** parametri yacheyka ichidagi obyekt bilan uning chegaralari orasidagi masofani piksellarda o'rnatadi. Shunday qilib **cellspacing** yacheykalar o'rtasidagi ochiq (bo'sh) masofa, **cellpadding** esa yacheyka ichidagi bo'sh qoldirilgan joy. Jadval tuzish usullarini o'rganishni, eng oddiy jadval yaratishdan boshlaymiz. Biz tuzmoqchi bo'lgan jadval uchta ustundan iborat bo'ladi va har bir ustunda 3 tadan yacheykalar o'rin oladi. Jadvalning birinchi ustunini "Mahsulot", ikkinchisini esa — "Rangi", uchinchisini- "Narhi" deb nomlaymiz. Biz tuzgan jadval mana bunday ko'rinishda bo'ladi:

Mahsulot	Rangi	Narhi (so'm.)
Qalam	Qora	100
Qalam ochgich	Qora	500

Endi ushbu jadvalni tuzishda qanday HTML teglardan foydalanilganligini ko'rib chiqamiz.

```

<TABLE>
  <TR>
    <TD>Mahsulot</TD>
    <TD>Rangi</TD>
    <TD>Narhi (so'm.)</TD>
  </TR>
  <TR>
    <TD>Qalam</TD>
    <TD>Qora</TD>
    <TD>100</TD>

```

```

</TR>
<TR>
    <TD>Qalam ochgich</TD>
    <TD>Qora</TD>
    <TD>500</TD>
</TR>
</TABLE>

```

Misolda ko‘rinib turibdiki, jadvalni `<TABLE>` va `</TABLE>` teglari yordamida ochiladi va yopiladi. Qatorlar uchun `<TR>` va `</TR>` teglari qo‘llaniladi, `<TD>` va `</TD>` teglari esa jadval ustunlarini yacheykalarini (bo‘lakchalarini) belgilash uchun ishlatiladi.

Jadvallarning qo‘llanilishi va uni qurish. Ro‘yxatlardagi bitta kamchilik – bu ularning **bir o‘lchovlilikidir**. Bu degani siz ro‘yxatda ma‘lumotlarni ketma-ketlik bilan satrma-satr joylashtirishingiz mumkin deganidir. Jadvallar esa ma‘lumotlarni faqat satr bo‘ylab emas balki ustun bo‘yicha ham joylashtirish imkonini beradi. Jadvallarni shunday ishlatish kerakki ular Web sahifada shunchaki joy egallamasligi kerak. Jadvalning satr va ustunlari ma‘lumotlarni taqqoslash, qarama-qarshi qo‘yish imkonini beradi. Har bir satr va ustundagi ma‘lumotlarni elektron jadval singari tasvirlash mumkin. Yaxshi tuzilgan jadval bilan Web sahifa ma‘lumotga boy va tartibli bo‘ladi, ammo ko‘rimsiz jadval sizning ma‘lumotlaringizni chalkashtirib yuboradi.

Jadval masalasi murakkab ko‘rinishi mumkin, chunki buning uchun butun bir teglar ketma-ketligi tuziladi.

`<table>` va `</table>` teglar jadvalini butunligicha o‘z ichiga oladi. Ma‘lumot chiqarilishini boshqa bir teglar ketma-ketligi aniqlaydi. Quyida jadvalning hamma teglari ifoda qilingan:

`<table>`, `</table>` bu teglar jadvalni egallaydi. `<table>` tegi brauzerga o‘zidan keyin jadvalning berilishi kelishini xabar qiladi. Agar siz satr va ustunlarni ajratib turuvchi (chiziqlari) ko‘rinishini o‘zgartirishni xoxlasangiz **border** kalit so‘zini kiriting (`<table border>` hosil bo‘ladi);

`<caption>` va `</caption>` bu teglar bilan belgilangan matn sarlavha ko‘rinishini oladi. Sarlavhani berish masalasi `<tc>` va `</tc>` yordamida ham bajarilishi mumkin;

`<th>` va `</th>` bu teglar matnni satr yoki ustun sarlavhasi qilib, biroz qalin shriftda tasvirlaydi;

`<tr>` va `</tr>` jadvalning har bir satrini aniqlaydi. `</tr>` tegi zarur emas, biroq u sizning HTML kodingizni yanada to‘la tushunarli qiladi;

`<td>` va `</td>` bu teglar juftligi jadvalning har bir yacheykasi uchun matn ajratadi.

Sizning birinchi qadamingiz – ona teg `<table>` ni kiritish. Jadvalning har bir yacheykasi **nozik chiziqlar** bilan bo‘linishi uchun `<table>` tegiga **border=1** kalit so‘zini kiriting. Agar bu bo‘lmasa yacheykaning so‘zlari **avtomatik tarzda tekislanadi**, ammo jadvalda to‘r (ajratuvchi chiziqlar) bo‘lmaydi.

Endi yacheykalarni birin-ketin yaratib boring. Dastlab satrni bering.

```
<table> <tr> </tr> </table>
```

Bundan so‘ng ustunning sarlavhasi bo‘ladigan yacheykani bering. Brauzerlar sarlavhani qolgan matnlarga nisbatan qalinroq shriftda tasvirlaydi.

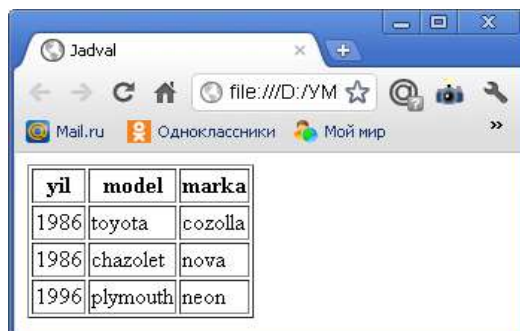
```
<table>
<tr>
<th> yil </th>
<th> model </th>
<th> marka </th>
</tr>
</table>
```

Hozir sizning jadvalingiz ustunlarining sarlavhasi hisoblangan bitta satrdan iborat. Qolgan satrlarni ham kiritib boshlaymiz. Har bir satrda ustunlar soniga razm solib boring, hozirgi holatda 3 ta ustun mavjud.

Misol 1.

```
<html><head><title>Jadval</title></head>
<table border=1>
<tr> <th> yil </th> <th> model </th><th> marka
</th> </tr>
<tr> <td> 1986 </td> <td> toyota </td> <td> cozolla
</td> </tr>
<tr> <td> 1986 </td> <td> chazolet </td> <td> nova
</td> </tr>
<tr><td> 1996 </td><td> plymouth </td><td> neon </td></tr>
</table>
```

</html>



yil	model	marka
1986	toyota	cozolla
1986	chazolet	nova
1996	plymouth	neon

Rasm 1.

Web sahifalarda juda ko'p satr va ustundan iborat jadvallarni ham yaratish mumkin. Quyidagi misolga e'tibor bering:

Misol 2.

```
<html>
<head>
  <title>Jadval</title>      Sarlavha
</head>
<body> <TABLE BORDER="10" WIDTH="100%">
  <CAPTION ALIGN="TOP"> <B> <H1> Telefon nomerlar
ro'yxati </H1> </B>
</CAPTION>
  <TR BGCOLOR="YELLOW" ALIGN="CENTER">
  <TH><H3>Familiyasi</H3>
  <TH><H3>Telefon nomeri</H3>
  <TH><H3>Adresi</H3>
  <TR>
  <TD><H5><A
HREF="Avazov.htm">Avazov</A></H5></td>
  <td>5-14-57</td>
  <td>A Navoiy</td>
  </TR>
  <TR><TD><H5><A
HREF="Karimov.htm">Karimov</A></H5></td>
  <td>3-16-49</td>
  <td>Behbudiy 5</td>
  </TR>
  <TR><TD>Norov</td>
```



```

<td>4-23-56</td>
<td>Temur Malik</td>
</TR>
<TR><TD>Minarov</td>
<td>6-10-42</td>
<td>A Avloniy</td>
</TR>
<TR><TD>Salimov</td>
<td>3-56-56</td>
<td>Fitrat</td>
</TR>
<TR><TD>Jalilov</td>
<td>2-74-50</td>
<td>Gulxaniy 10</td>
</TR>
<TR><TD ALIGN="CENTER" COLSPAN="3">Binoning birinchi
qavatida bepul telefon avtomat mavjud</TABLE>
</body>
</html>

```

Familyasi	Telefon nomeri	Adresi
Avloniy	5-14-57	A Avloniy
Karimov	3-16-49	Behbodiy 5
Minarov	4-23-56	Temur Malik
Minarov	6-10-42	A Avloniy
Salimov	3-56-56	Fitrat
Jalilov	2-74-50	Gulxaniy 10

Binoning birinchi qavatida bepul telefon avtomat mavjud.

Rasm 2.

Bir yacheykaga bir necha satrlarni birlashtirish. Agar siz tez-tez jadvallardan foydalanib tursangiz, qaysidir bir yacheyka jadvalning bir necha satrlarini birlashtirish lozim bo‘lib qolgan vaziyat bilan albatta tanishsiz. Bu yerda sizga **<td>** tegi ichiga qo‘shiladigan **rowspan** kalit so‘zi asqotadi.

Misol 3.

```

<html>
<head>
<title>Birlashtirish</title> Sarlavha
</head>

```

```

<body>
<table border="8" width="94%">
  <CAPTION ALIGN="TOP"> <B> <H1> Olimpiada </H1>
</B> </CAPTION>
  <TR BGCOLOR="YELLOW" ALIGN="CENTER">
    <TH><H3>Ismi</H3>
    <TH><H3>Viloyati</H3>
    <TH><H3>Egallagan orin</H3>
    <TH><H3>Musobaqa turi</H3>
    <TH><H3>Bo‘yi</H3>
  <tr>
    <td> Dilshod </td>
    <td rowspan=2> Samarqand </td>
    <td> 1 </td>
    <td> 100 metrga yugurish </td>
    <td> 194 </td>
  </tr>
  <tr>
    <td> Sarvar </td>
    <td> 2 </td>

    <td> So‘zish </td>
    <td> 196 </td>
  </tr>
  <tr>
    <td> Javlon </td>
    <td rowspan=3> Termez </td>
    <td> 1 </td>
    <td> Tosh ko‘tarish </td>
    <td> 194 </td>
  </TR>
  </tr>
  <tr>
    <td> Zuhro </td>
    <td> 1 </td>

    <td> Vilosiped poygasi </td>
    <td> 192 </td>

```

```

</tr> <tr>
  <td> Adizova </td>
    <td> 3 </td>
    <td> So‘zish</td>
    <td> 196 </td>
  </tr>
</table>
</body>
</html>

```

Ismi	Viloyati	Egallagan orin	Musobaqa turi	Bo'yi
Dilshod	Samarqand	1	100 metrga yugurish	194
Sarvar		2	Suzish	196
Javlon	Termez	1	Tosh ko'tarish	194
Suzrob		1	Vilosped poygasi	192
Nodir		3	Suzish	196

Rasm 3.

Bir necha ustunlarni bir yacheykaga birlashtirish va ro‘yxat kirgizish.

Yuqoridagi misolda **rowspan** bir necha satrlarni birlashtirgani guvohi bo‘ldingiz, endi e‘tiboringizni, **colspan** kalit so‘ziga qaratishingizni so‘raymiz. **Colspan** kalit so‘zi bir necha ustunlarni birlashtiradi. **Colspan** kalit so‘zi yordamida siz istagan miqdordagi ustunlarni bir yacheykaga birlashtirishingiz mumkin. Quyidagi misolga e‘tibor bering:

Misol 4.

```

<html>
<head><title>Ustunlarni birlashtirish</title>
</head>
<body>
<table border="8" width="100%">
<CAPTION ALIGN="TOP"> <B> <H2> Ma'lumot </H2>
</B> </CAPTION>
<TR BGCOLOR="#00FFFF" ALIGN="CENTER">
<TH colspan=3><H3>F.I.SH.</H3>
<TH><H3>Viloyati</H3>
<TH colspan=3><H3>Tug‘ilgan yili</H3>
<tr>

```

<th> Zaripova </th>
<th> Muqaddas </th>
<th> Jumaniyozovna </th>

<th> Samarqand </th>
<th> 1972 yil </th>
<th> 7 </th>
<th> aprel </th>
</tr>

<tr>
<th> Yusupova </th>
<th> Dilfuza </th>
<th> Aminovna </th>
<th> Farg'ona </th>
<th> 1968 yil </th>
<th> 7 </th>
<th> may </th>
</tr>

<tr>
<th> Karimova </th>
<th> Nodira </th>
<th> Salimovna </th>
<th> Termiz </th>
<th> 1972 yil </th>
<th> 14 </th>
<th> mart </th>
</tr>

</table>
</body>



Ma'lumot						
F.I.S.H.			Viloyati	Tug'ilgan yili		
Zaripova	Muqaddas	Jumaniyozovna	Samarqand	1972 yil	7	aprel
Yusupova	Dilfuza	Aminovna	Farg'ona	1968 yil	7	may
Karimova	Nodira	Salimovna	Termiz	1972 yil	14	mart

</html>

Rasm 4.

rowspan (yoki **colspan**) lardan foydalanganda mos holda satr yoki ustunlarning miqdorini kamaytirishni esdan chiqarmang. Jadvalga HTML ning boshqa elementlarini ham qo'shish mumkin. Shunday ekan unga ixtiyoriy turdagi ro'yxat kiritilishi mumkin. Qat'iy ravishda yoyilgan teglardan foydalanganlikni kuzatib borish kerak, aks holda ro'yxat kodlarini ajratib ko'rish qiyin bo'ladi.

Jadval yacheykalarining vertikal yoki gorizontal ravishda matnni tekislash masalasini berish mumkin. Tekislash usullari maxsus **align** va **valign** kalit so'zlari yordamida beriladi. Yana siz umumiy ravishda jadval uchun tekislashda (**<table>** tegida), joriy satr uchun (**<tr>** tegda) yoki bir yacheyka uchun (**<td>** tegda) berish mumkin. **align** va **valign** kalit so'zlarining har biri uchun 3 ta mumkin bo'lgan ifodalari bor:

align=left matnni yacheykaning chap tarafi bo'yicha tekislaydi (jimlik holatida)

align=center matnni yacheykaning markazi bo'ylab tekislaydi

align=right matnni yacheykaning o'ng tarafi bo'yicha tekislaydi

valign=top matnni yuqori qismi bo'yicha tekislaydi (agarda ma'lumot bir necha satrdan iborat bo'lganda foydalaniladi)

valign=middle matnni yacheykaning markazidan vertikal bo'ylab tekislaydi (jimlik holatida)

valign=bottom matnni yacheykaning past qirrasini bo'ylab tekislaydi.

Jadvaldagi ranglar. So'ngi vaqtlargacha Web sahifaning jadvaldagi ranglarni boshqarish imkoni mavjud emas edi. Jadval turi qora rangda, fon esa sahifa foni rangi bilan mos tushar edi. Biror bir yacheykaning rangini boshqasidan ajratish va har xil ranglardan foydalanish juda qiyin bo'lgan.

Hozirda fon va tur rangini berish mumkin bo'lgan yangi teglar qatori mavjud. Birinchi holda **bgcolor** kalit so'zi **<table>** tegida quyidagi ravishda qo'yiladi.

```
<table border bgcolor=yellow></table>
```

Endi esa jadval fonining rangi sariq bo'ladi. Yuqoridagi 4 misolda **<table border="8" width="100%">** o'rniga **<table border bgcolor=yellow>** ni kiriting. Quyidagi jadval hosil bo'ladi.

Ma'lumot				
F.I.S.H.		Viloyati	Tug'lgan yili	
Zaripova	Muqaddas	Jumaniyozovna	Samarqand	1972 yil 7 aprel
Yusupova	Dilfuza	Aminovna	Farg'ona	1968 yil 7 may
Karimova	Nodira	Salimovna	Termiz	1972 yil 14 mart

Rasm 4.1.

Quyidagi 16 rangdan ixtiyoriy birini uning nomini ko'rsatgan holda berishingiz mumkin: Black-qora, Treen- to'q yashil, Navy-to'q ko'k, Tial-ko'k, Silver-ochiq kulrang, Lime-yashil, Blue-ko'k, Aqua-berюзовый, Maroon-малиновый, Olive-оливковый, Purple-сиреневый, Troy-темно-серый, Red-qizil, Yellow -sariq, Fuchsia-розовый, White-oq.

bgcolor kalit so'zidan boshqa yana ranglarni boshqarish uchun quyidagi kalit so'zlari ham ishlatiladi:

bordercolor - jadval hoshiyasining (turining) rangini o'zgartirish.

bordercolordark/bordercolorlight - jadval hoshiyasining (turining) rangini qo'shimcha 3 o'lchamli ko'rinishda hosil qilish uchun ishlatiladi.

Quyida yangi kalit so'zlarning barchasini ishlatib tuzilgan oddiy namuna berilgan:

```
<table border=4 bgcolor=yellow bordercolor=red
bordercolordark=blue bordercolorlight=grey width=100%>
</table>
```

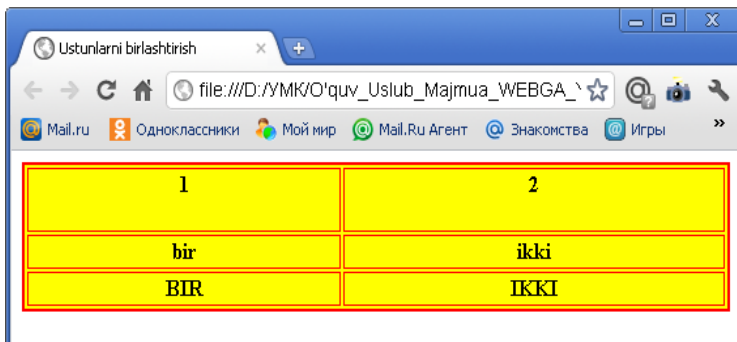
Misol 5.

```
<html>
<head><title>Jadval hoshiyasini o'zgartirish</title>
</head>
<body>
<table border=2 bgcolor=yellow bordercolor=red
bordercolordark=blue bordercolorlight=grey width=100%>
<TR ALIGN="CENTER">
<TH><H4>1</H4>
<TH><H4>2</H4>
</TR>
<tr>
</tr>
<tr>
```

```

</tr>
</table>
</body>
<th> bir </th>
<th> ikki </th>
<th> BIR </th>
<th> IKKI </th>
</html>

```



Rasm 5.

Izoh: “Rasm 5” Misol 5 ning brauzer oynasida tasvirlanishi.



Rasm 5.1.

Rasm 5 va rasm 5.1 ga e'tibor bering, ko'rib to'rganingizdek, siz bu kalit so'zlar yordamida jadvalning faqatgina tashqi ko'rinishini o'zgartirishni o'rgandingiz xolos. Bu hali hammasi emas. Bu kalit so'zlarni har bir satr va yacheyka uchun berishimiz mumkin. Quyida keltirilgan HTML kodda har bir yacheykaning rangi alohida beriladi.

Misol 6.

```

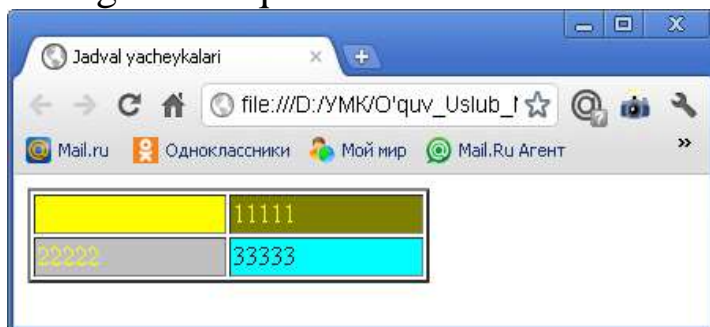
<html>
<head><title>Jadval yacheykalari</title></head>
<body>
<table border=2 width=300>
<tr>
</tr>
<tr>

```

```

</tr>
</table>
</body>
</html>
<td bgcolor=yellow></td>
<td bgcolor=olive><font color=black>1111</font></td>
<td bgcolor=silver><font color=yellow>2222</font></td>
<td bgcolor=aqua>3333</td>

```



Rasm 6.

Bu kalit soʻzlarni **tr** tegida ham berish mumkin. Bunda satrning rangi beriladi. Jadval matnining rangi uning foni rangi bilan mos tushmaganligiga eʼtiboringizni qarating. Matn rangi **font** tegi yordamida oʻzgartirilishi kerak.

Nazorat savollar:

1. Jadvallarda ustunlar guruhini qanday teglar yordamida aniqlash mumkin?
2. Jadvalning yuqorigi va pastgi sarlavhalarini (shapkalarini) qanday teglar bilan mos ravishda tashkil qilish mumkin?
3. Qanday teg yordamida satrlar guruhi hosil qilinadi?
4. Jadval va uning tashkil etuvchi elementlari qanday teglar orasida joylashtiriladi? Jadval sarlavhasichi?

4-MARUZA. GRAFIKA VA MULTEMEDIYA

Reja:

1. HTMLda grafika.
2. HTMLda multimedia
3. HTMLda grafika va multimediani qo‘llash

Tarmoqda **tasvirlar** haqida gapirilganda aniqlangan ma’lum bir turdagi fayllar haqida gap ketadi. U «цифровой» kamera yoki skaner yordamida olingan rasm, tasvir, hattoki, grafik formatda ifodalangan matn bo‘lishi ham mumkin. Bu yerda faylning **aniqlangan formati** muhim rol o‘ynaydi. Ko‘pchilik matn muharrirlari (**Notepad, SimpleText**) grafik fayllarni o‘zlarida tasvirlay (ifodalay) olmaydilar. Buning uchun maxsus tasvirlarni ko‘rish programmasi, grafik muharrirlar mavjud. Tasvirlarni web-sahifaga joylashtirish juda oddiy. Buning uchun **** elementi kerak bo‘ladi. Shuni tushunish kerakki, rasm sahifaga joylashtirilmaydi, balki rasimga sahifadan turib **murojaat**(rasimga yo‘l) ko‘rsatiladi. Rasm fayli esa kompyuterning qattiq diskida yoki tarmoqda joylashgan bo‘lishi mumkin. Tarmoqda tasvirlarning asosan **JPEG (Joint Photographic Experts Group «Dji-peg»** deb o‘qiladi), **GIF (Graphics Interchange Format «gif»** deb o‘qiladi) va **PNG (Portable Network Graphics)** kabi formatlari qo‘llaniladi. Bundan tashqari **PCX, TIFF** va **PICT** kabi formatlar ham internetda uchraydi. Agar foydalanuvchida boshqa formatdagi tasvirlar bo‘lib, uni web-sahifaga joylashtirmoqchi bo‘lsa, u holda uni yuqoridagi formatlardan birida ifodalab olishi maqsadga muvofiqdir. Web-sahifa uchun tasvirni qo‘yayotganda quyidagi **qoidalarga** e’tibor berish lozim: **birinchidan**, «tushlik uchun yaxshi qoshiq» tanlagan kabi, ya’ni tasvir o‘z joyida qo‘llanilishi kerak; **ikkinchidan**, tasvir faylini yuklash ko‘p vaqtni olmasligi kerak.

Tasvirlar, jadvallar, grafiklar web-sahifani bezaydi, uning ko‘rinishi va informativlik xususiyatini oshiradi. Faqat matndan iborat Web-sahifa foydalanuvchini zeriktirib qo‘yishi mumkin. Web-sahifaga tasvirlarni qoyish oddiy vazifa. Murakkabi, yaxshi tasvirni hosil qilish hisoblanadi. U ko‘plab xususiyatlarga ega bo‘lishi lozim: qiziqarli, foydali, o‘ziga tortuvchi va juda ham katta hajmli bo‘lmasligi lozim. Tasvirlarni yaratishing ba’zi bir qoidalari:

- Tasvirlar va grafikalar berilayotgan informatsiyaga tegishli bo'lishi lozim. Sahifaning o'lchamini oshirish uchungina tasvirlarni qo'yish kerakmas. Foydalanuvchilar Internetdan ma'lumot qidiradi va faqat keraklisini o'qiydi, sizning yaxshi ko'rgan rasmingizni ko'rish u uchun qiziq emas.
- Tasvirlar tez yuklanishi kerak, bu yutuqqa olib keladi. Buning uchun esa fayl katta hajmda bo'lmasligi kerak. Shuning uchun sahifani rasmlar bilan «to'ldirib» tashlamaslik kerak.
- Oxirgi yillarda ishlab chiqarilayotgan yuqori sifatli raqamli fotoapparatlar xizmatidan foydalaning. Bu sizga kerakli format va o'lchamni tanlashda qulaylik tug'diradi.

Agar siz sahifangizga tasvir qo'yishga qaror qilgan bo'lsangiz tasvirlarni yaratishga mo'ljallangan grafik muxarrirlar xizmatidan foydalanishingiz mumkin. Ular: Paint Shop Pro, Graphic Converter, Adobe Photoshop, Adobe Image Ready, Corel Draw, Corel Xara.

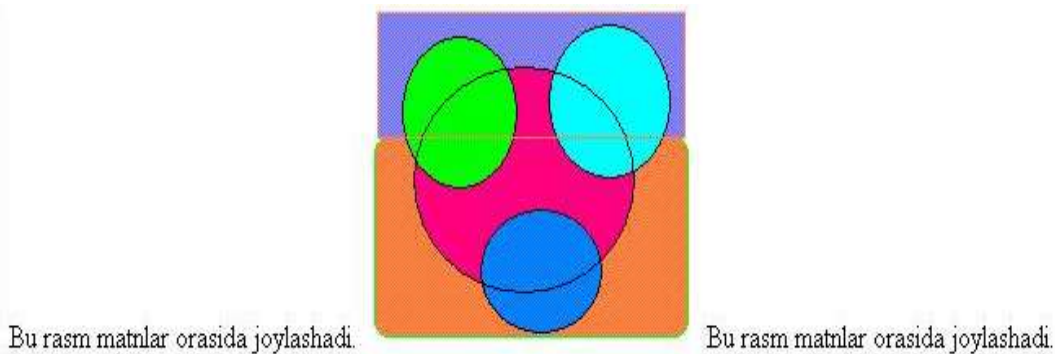
Grafik obyektlar yuqorida aytib o'tganimizdek Web sahifani bezashda juda muhim o'rin tutadi. **Grafik obyekt** deganda biz har xil tipdagi rasmlar, fotorasmlar, tovushli fayllar va videokliplarni tushinishimiz kerak.

Brauzer **GIF**, **JPEG** va **PNG** formatdagi grafik fayllarni tasvirlaydi. GIF formatidagi fayllar animatsion tasvirlarni yaratish imkonini beradi. JPEG fayllar fototasvirlarni saqlash uchun ishlatiladi. **PNG** formati esa tasvirning yuqori sifati va grafik faylning kichik hajmda bo'lishini ta'minlaydi. Web sahifada grafik fayllarni tasvirlash uchun **** tegi o'zining bir qancha parametrlari bilan qo'llaniladi. Bu teg o'zining yopiluvchi tegigi ega emas, ya'ni biror soha uchun emas balki ko'rsatilgan joyga grafik obyekt joylashtirilishini bildiradi. **** tegining asosiy va majburiy parametrlaridan biri **src parametridir**. Bu parametr qiymati sifatida tasvirlanishi kerak bo'lgan obyekt manzili yoki aniqroq qilib aytganda uning URL manzili ko'rsatiladi.

Masalan: Web sahifaning HTML fayllari joylashgan **IMAGES** papkasi o'zining grafik fayllari bilan joylashgan bo'lsa bu papkadagi rasm1.gif faylini tasvirlash uchun quyidagi yozuvni yozishimiz kerak: ****. Hozirgi kunda grafik obyektlarni tasvirlay olmaydigan brauzerlar umuman ishlatilmaydi, lekin brauzerlarda grafik obyektlarni ekranga chiqarmaslik imkoniyati mavjud. Shuning uchun grafik obyekt

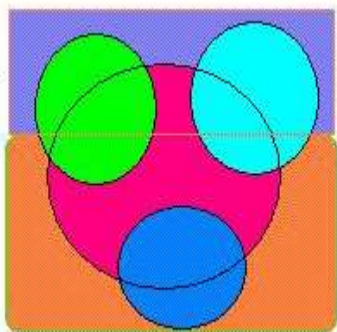
chiqarilmagan vaqtda uning o‘rnida shu obyekt haqida paydo bo‘luvchi alternativ matnli ma’lumot bo‘lishi zarur. Bu alternativ matnli ma’lumot **** tegiga **alt parametri** yordamida qo‘shiladi va matnli qiymat qabul qiladi. **Misol:******. Agar grafik tasvir brauzer tomonidan ekranga chiqarilsa u holda alternativ matn sichqoncha ko‘rsatkichi tasvir ustiga olib borilganda ko‘ratkich yonida paydo bo‘ladi. Grafik tasvir haqidagi to‘liq ma’lumotni beruvchi internet manba manzilini ko‘rsatuvchi **longdesc** parametri ham sichqoncha ko‘rsatkichi yonida paydo bo‘luvchi matnli ma’lumotni yaratishda ishlatiladi. Bu parametr qiymati sifatida tasvirning **URL** manzili ko‘rsatiladi. Jimlik holati bo‘yicha grafik tasvir saqlangan paytda qanday **gorizontal** va **vertikal** o‘lchamlarda saqlangan bo‘lsa ekranda ham xuddi shunday tasvirlanadi. Grafik tasvirni foydalanuvchi o‘zi xohlagan o‘lchamda tasvirlash imkoniyatiga ega, buning uchun **height** va **width** parametrlaridan foydalaniladi. Web sahifada grafik tasvirni boshqa obyektlardan bo‘sh joy bilan ajratib turuvchi **hspace** va **vspace** parametrlari ham mavjud. **hspace** parametri bo‘sh joyni gorizontal bo‘yicha piksellarda o‘rnatadi, **vspace** parametri esa vertikal bo‘yicha o‘rnatadi.

Bu parametrlar qiymati faqat **sonlarda** beriladi. **Border** parametri bilan tasvir atrofidagi chegara qalinligi o‘rnatiladi. Parametr qiymati sifatida pikselda o‘lchangan son beriladi. Jimlik bo‘yicha bu parametr **0** qiymat qabul qiladi va chegara hoshiyasi ko‘rinmaydi. Tasvirni o‘zing atrofidagi matnga nisbatan tekislashni ko‘rib chiqamiz. Buning uchun **align** parametri ishlatiladi. Bu parametr qiymati sifatida quyidagi so‘zlar ishlatiladi: **bottom, middle, top, left, right**. Endi yuqoridagi parametrlar qo‘llanilgan quyidagi misollarni ko‘ramiz:



Rasm 1.

Endi shu misolda **** tegiga gorizontaal bo'yicha tekislash parametrini qo'shib ko'raylik.



Bu rasm matndan chapda joylashadi. Bu rasm matndan chapda joylashadi. Bu rasm matndan chapda joylashadi. Bu rasm matndan chapda joylashadi. Bu rasm matndan chapda joylashadi. Bu rasm matndan chapda joylashadi.

Rasm 2.

Misol1.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head> <title>Rasm va matn</title> </head>
<body>
<p> Bu rasm matnlar orasida joylashadi. <IMG SRC=rasm1.GIF>
Bu rasm matnlar orasida joylashadi. </p>
</body>
</html>
```

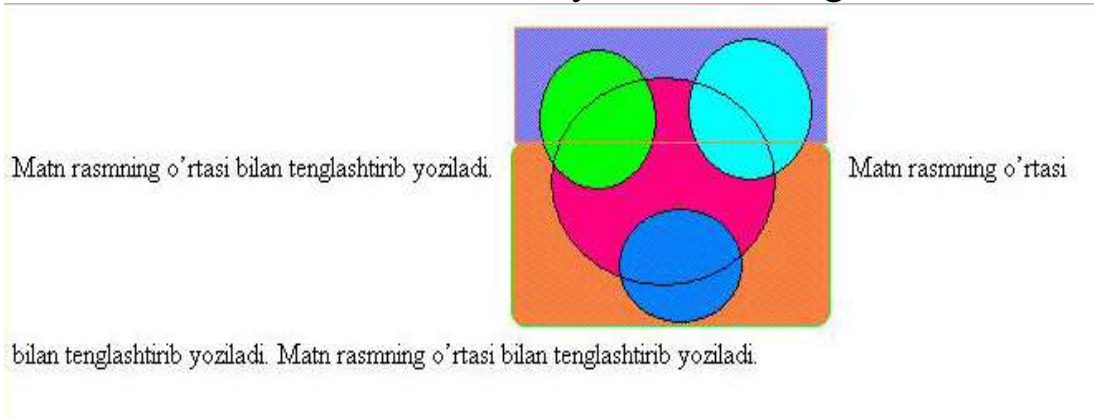
Misol 2.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html><head> <title>Rasm va matn</title> </head>
<body>
<p>
Bu rasm matndan chapda joylashadi. <img src=rasm1.gif align=left > Bu rasm matndan chapda joylashadi. Bu rasm matndan
```

chapda joylashadi. Bu rasm matndan chapda joylashadi. Bu rasm matndan chapda joylashadi. Bu rasm matndan chapda joylashadi. Bu rasm matndan chapda joylashadi.

```
</p>
</body>
</html>
```

Endi xuddi shu misolni vertikal bo'yicha tekislashga o'tkazamiz.



Rasm 3.

Misol3.

```
<html>
<head><title>Rasm va matn</title></head>
<body>
<p> Matn rasmning o'rtasi bilan tenglashtirib yoziladi.  Matn rasmning o'rtasi bilan
tenglashtirib yoziladi. Matn rasmning o'rtasi bilan tenglashtirib
yoziladi. </p>
</body> </html>
```

Videofayllar. Bizda Web sahifani bezash uchun unga videoroliklar joylashtirish imkoniyati ham mavjud. Foydalanuvchi o'z brauzerida videoma'lumotni ko'rishi uchun videoma'lumot yozilgan faylni o'z kompyuteriga to'liq yuklab olishi lozim. Video fayllar katta hajm egallaganligi sababli uni tarmoq orqali kompyuterga yuklash uchun ko'proq vaqt sarflanadi. Brauzerlar odatda **AVI**, **Real**, **Video** va **Windows Media** formatlaridagi videofayllarni tasvirlash imkoniyatiga ega. Ularni Web sahifa tarkibiga qo'shish uchun ham **** tegi ishlatiladi. Videofayl joylashgan manzilni ko'rsatish uchun esa **dynsrc** parametri ishlatiladi. Jimlik bo'yicha Web sahifa tarkibiga kiritilgan videoma'lumot Web sahifa yuklangandan keyin

avtomatik ravishda bir marta tasvirlanadi, lekin bizda tasvirlanish jarayonini boshqarish yoki bu jarayonni foydalanuvchi tomonidan boshqariladigan qilish imkoniyati mavjud. **** tegiga **<start>** parametrini qo‘shish orqali biz yuklangan videoklipni tasvirlash jarayonini bimalol boshqarishimiz mumkin. **<start>** parametrining qiymatlari sifatida **mouseover** va **fileopen** kalit so‘zlari ishlatiladi. **mouseover** parametri foydalanuvchi sichqoncha ko‘rsatkichini videotasvir joylashgan soha ustiga olib kelishi bilan videotasvirning tasvirlanish jarayoni boshlanishini bildirsa, **fileopen** parametri esa Web sahifa kompyuterga to‘liq yuklanib bo‘lgandan keyin darhol tasvirlanish jarayoni boshlanishini bildiradi.

Misol:

```

```

Biz **loop** parametri orqali videorolikning necha marta tasvirlanishini boshqarishimiz mumkin. Buning uchun tasvirlanishlar sonini **loop** parametriga qiymat sifatida berishimiz zarur. Agar biz videorolikning takroriy tasvirlanishini doimiy ravishda takrorlanuvchi qilib qo‘ymoqchi bo‘lsak u holda **loop** parametriga **infinite** qiymatini berishimiz kerak bo‘ladi. **Loopdelay** parametri takroriy tasvirlanish jarayonida tasvirlanishlar orasidagi vaqtni belgilaydi va qiymati millisekundlarda beriladi.

Misol:

```

```

Yuqoridagi misolda Web sahifa to‘liq yuklanishi bilan videoklip ikki marta orasida 5 sekundlik tanaffus bilan tasvirlanadi. **** tegiga **controls** parametrini qo‘yish orqali biz videofaylni to‘liq o‘zimiz boshqarishimiz mumkin, ya’ni Web sahifada videoklip bilan birgalikda uni boshqarish elementlari ham paydo bo‘ladi. Boshqarish elementlariga tasvirni ishga tushirish, orqaga qaytarish, oldinga o‘tkazish yoki to‘xtatish kabi ishlarni bajaruvchi tugmalar kiradi.

Misol:

```
<html>  
<head><title>Video</title></head>  
<body>
```

```
<p>Oddiy matn.  </p>
</body>
</html>
```

Videotasvirlarni Web sahifaga joylashtirish bu grafik obyektlarni joylashtirishning bir qismi bo‘lganligi uchun **** tegining barcha parametrlarini videotasvirlar uchun ham qo‘llash mumkin.

Tovushli fayllar. Web sahifaga tovushli fayllarni joylashtirish ham HTML ning multimedia imkoniyatlari doirasiga kiradi. Tovushli fayllar Web sahifada fon hisobida qatnashadi. **Tovushli fayl** Web sahifa to‘liq yuklangandan keyin bajariladi. Foydalanuvchi kompyuteriga Web sahifa yuklanganda birinchi navbatda uning HTML tilida yozilgan fayli, keyin esa shu sahifada ishlatilgan grafik, video va audiofayllar yuklanadi. Demak Web sahifaga tovushli fayl qo‘yishimiz uchun biz sahifaning sarlavhasiga, ya’ni **<head>** va **</head>** teglari orasiga **<bgsound>** tegini joylashtirishimiz kerak bo‘ladi. Bu tegning src parametri qiymat sifatida tovushli faylning **URL** manzilini qabul qiladi. Brauzerlar **MIDI** va **WAV** formatdagi audiofayllarni taniydi va bemalol aytirish imkoniga ega. Boshqa formatdagi, ya’ni **MP3**, **RealAudio** kabi audiofayllarni aytirish uchun brauzerlarga qo‘shimcha modullar qo‘shish kerak bo‘ladi. Xuddi videokliplar bilan ishlaganimizdek audiofayllar bilan ishlaganimizda ham faylning necha marta takrorlanishini **loop** parametri bilan boshqarishimiz mumkin. Qiymat sifatida esa takrorlanishlar soni beriladi. Agar audiofaylning to‘xtovsiz takrorlanishi zarur bo‘lsa **loop** parametriga **“-1”** qiymat berish kerak.

Misol: `<bgsound src="song.mid" loop="-1">`

Nazorat savollari:

1. Web sahifaga videofayl qanday joylashtiriladi?
2. Grafik obyektlar nima?
3. Grafik obyektlarga nimalar kiradi?

5-MAVZU. FORMALAR BILAN ISHLASH.

Reja:

1. HTMLda formalar.
2. HTMLda formalar bilan ishlash.

Biz veb-ilova yoki veb-sayt ustida ishlayotganimizda, biz ko‘pincha foydalanuvchilardan Login va ro‘yhatdan o‘tish orqali ma’lumot olishimizga to‘g‘ri keladi. Veb-sahifada formani yaratish HTML orqali amalga oshirilsa, PHP veb-sahifadan olingan ma’lumotlarni serverga yuborish va ularni qayta ishlashda transport sifatida xizmat qiladi. PHPda to‘plangan forma ma’lumotlarini qayta ishlashda ikkita superglobal: \$_GET va \$_POST metodlari ishlatiladi.

HTML formaning ishlashi bo‘yicha tushuncha. Keling oddiy HTML forma yaratamiz va uni qanday ishlashini tushunishga harakat qilamiz. <form> tegi ichida ishlatilgan atributlarning vazifalarini o‘rganamiz.

```
<html>
  <body>
    <form action="forma.php" method="POST">
      Ism: <input type="text" name="name"> <br/>
      Email: <input type="text" name="email"> <br/>
      <input type="submit" value="Yubor">
    </form>

  </body>
</html>
```

Yuqoridagi kodda biz HTML forma yaratish uchun <form> tegidan foydalandik va ma’lumot kiritish uchun Ism va Email maydonchalarini ochib oldik va ma’lumotni yuborish uchun submitdan foydalandik.

Form teg ichida ikkita atribut mavjud, **action** va **method**. Ular nima uchunligini bilasizmi?

1. action: Bu atributdan foydalanish bilan biz forma ma’lumotlarini ishlab va to‘playdigan fayl nomini kirita olamiz. Yuqorida biz yozgan php fayl nomi.
2. method: Bu atribut forma ma’lumotini yuborilish usulini ko‘rsatadi, u POST metodi yoki GET metodi orqali yuborilishi mumkin.

Quyida GET metodi uchun forma namunasi:

```
<html>
  <body>
    <form action="forma.php" method="POST">
      Ism: <input type="text" name="name"> <br/>
      Email: <input type="text" name="email"> <br/>
      <input type="submit" value="Yubor">
    </form>

  </body>
</html>
```

Formani POST orqali yuborish. Agar biz forma metodi POST bo'lishini ko'rsatsak, unda forma ma'lumotlari serverga HTTP POST metodi dan foydalanib yuboradi. Quyida bizda HTML formamizning action atributida ko'rsatilgan Php fayldagi forma ma'lumotini olish kodi bor.

```
<?php
// ism maydonidagi qiymatni olish
$ism= $_POST["name"];
// email maydonidagi qiymatni olish
$email = $_POST["email"];
echo "Assalomu alaykum, ". $name . "<br>";
echo "Sizning email manzilingiz: ". $email . "<br>";
?>
```

Va quyidagi natijani olamiz:

Ismingiz:

Pochtangiz:

Formani GET orqali yuborish. Agar biz forma metodi POST bo'lishini ko'rsatsak, unda forma ma'lumotlari serverga HTTP GET metodi dan foydalanib yuboradi. Quyida bizda HTML formamizning action atributida ko'rsatilgan Php fayldagi forma ma'lumotini olish kodi bor. Bu safar biz GET superglobalidan foydalanmoqdamiz.

```
<?php
// ism maydonidagi qiymatni olish
$ism= $_GET["name"];
```

```
// email maydonidagi qiymatni olish
$email = $_GET["email"];
```

```
echo "Assalomu alaykum, ". $name . "<br>";
echo "Sizning email manzilingiz: ". $email . "<br>";
?>
```

Natija bir xil:



Ismingiz:
Pochtangiz:

GET metodidan foydalanish. GET metodi usulida, forma ma'lumotlari URL parameterlar ko'rinishida yuboriladi. Misol uchun, foydalanuvchi tomonidan to'ldirilgan forma ma'lumotlari ish skriptga yuboriladi, URL ga qo'shiladi.

```
<html>
  <body>
    <form action="forma.php" method="GET">
      Ism: <input type="text" name="name"> <br/>
      Yosh: <input type="text" name="age"> <br/>
      <input type="submit" value="Yubor">
    </form>
  </body>
</html>
```

```
<?php
// ism maydonidagi qiymatni olish
$ism= $_GET["name"];
// email maydonidagi qiymatni olish
$sage = $_GET["age"];
echo "Assalomu alaykum, ". $ism . "<br>";
echo "Sizning yoshingiz: ". $sage . "<br>"; ?>
```

Bizda ikkita maydon bor: ism va yosh. Ushbu maydonga kiritilgan ma'lumotlar forma yuborilgandan so'ng, brauzerimizning URL kiritiladigan maydonida aks etadi. Misol uchun: <https://tutorials.uz/index.php?name=Sanjar&age=18> .



tutorials.uz/index.php?name=Sanjar&age=18

- *GET metodi orqali kiritilgan forma ma'lumotlari brauzerning URL maydonida ko'rinadi.*

- *GET metodi ma'lumotlarni 2000 belgidan oshmagan holda yuborish bilan cheklangan.*
- *GET metodi serverga ikkilamchi ma'lumotlarni yuborolmaydi, jumladan: rasm, hujjat va boshqalar.*
- *GET metodidan ro'yxatdan o'tish va login(kirish) holatlarida aslo foydalanmang.*
- *GET metodidan yuborilgan ma'lumotga QUERY_STRING muhit o'zgaruvchisidan foydalanib kirishimiz mumkin.*

POST metodidan foydalanish. POST metodi orqali yuborilgan forma ma'lumotlari boshqalarga ko'rinmas bo'ladi(barcha nom/qiymatlar HTTP so'rovining tana qismi ichida qamrab olinadi) va ma'lumotni yuborishda hech qanday chegara mavjud emas. Bundan tashqari POST metodi serverga ko'p-qismli ikkilamchi ma'lumot yuklashni qo'llaydi. Shuningdek, URL maydonida formadagi o'zgaruvchi va qiymatlar ko'rinmaydi.

```
<html>
  <body>
    <form action="forma.php" method="POST">
      Ism: <input type="text" name="name"> <br/>
      Yosh: <input type="text" name="age"> <br/>
      <input type="submit" value="Yubor">
    </form>
  </body>
</html>
<?php
// ism maydonidagi qiymatni olish
$ism= $_POST["name"];
// email maydonidagi qiymatni olish
$age = $_POST["age"];
echo "Assalomu alaykum, ". $ism . "<br>";
echo "Sizning yoshingiz: ". $age . "<br>";
?>
```

URL maydonda:



- POST metodida ma'lumot yuborish uchun hech qanday belgi cheklovi yo'q

- POST metodi ASCII va ikkilamchi ma'lumotni yuborishda foydalanish mumkin
- Ma'lumotlar xavfsizligiga kafolat

Ko'pgina dasturchilar ma'lumot bilan ishlashda POST metodidan foydalanishni afzal ko'rishadi.

Nazorat savollari

1. Forma qanday teg yordamida hosil qilinadi?
2. Sahifaga 1 ta input, 1 ta button hosil qiling.
3. 4 ta amaldan iborat kalkulator hosil qiling.
4. Mevalar ro'yxatini checkbox yordamida hosil qiling.

6-MAVZU. XARITA MAP BILAN ISHLASH.

Reja:

1. HTMLda Map tegi.

2. HTMLda Map bilan ishlash.

<map> tegi tasvir xaritalarini yaratish uchun ishlatiladi. Tasvir xaritasining faol joylarini (koordinatalar, shakllar, o'lchamlar va boshqalar) aniqlash uchun <area> tegidan foydalaniladi. Rasmning faol maydonini bosganingizda, ma'lum bir harakat sodir bo'ladi, masalan, batafsil ma'lumotga ega sahifaga o'tish. <area> tegi har doim <map> tegi ichida joylashgan. Hotspotlar bir-birining ustiga qatlamlanishi mumkin, bu holda bosilganda HTML kodida yuqoriroq joy faollashadi. <map> tegi elementi bilan birga mijozning rasm xaritasi bilan bog'lash uchun ham ishlatiladi. Buning uchun tegida usemap atributi, <map> tegida esa nom atributi tomonidan o'rnatiladigan yagona identifikatordan foydalaniladi. HTML5 da, agar <map> elementi global id atributiga ega bo'lsa, u nom atributi bilan bir xil qiymatga ega bo'lishi kerak. HTML <map> tegi juftlashtiriladi, mazmun ochilish (<map>) va yopish (</map>) teglari orasiga yoziladi.

Misol

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Заголовок документа</title>
```

```
</head>
```

```
<body>
```

```
<p>Нажмите на логотип или на один из элементов логотипа, чтобы посмотреть его ближе:</p>
```

```

```

```
<map name="blockmap">
```

```
<area shape="circle" coords="45,58,30" alt="html-book" href="https://www.w3docs.com/learn-html.html">
```

```
<area shape="circle" coords="88,64,15" alt="html-quiz" href="https://www.w3docs.com/quiz-start/html-basic">
```

```

    <area shape="circle" coords="114,67,14" alt="html-snippets"
href="https://www.w3docs.com/snippets/html.html">
  </map>
</body>
</html>
<!DOCTYPE html>
<html>
<body>
<h1>Елементи map та area</h1>
<p>Натисніть на комп'ютер, телефон або чашку кави, щоб
перейти на нову сторінку і дізнатись більше по темі:</p>


<map name="workmap">
  <area shape="rect" coords="34,44,270,350" alt="Комп'ютер"
href="computer.html">
  <area shape="rect" coords="290,172,333,250" alt="Телефон"
href="phone.html">
  <area shape="circle" coords="337,300,44" alt="Чашка кави"
href="coffee.html">
</map>

</body>
</html>

```

Nazorat savollari.

1. HTMLda Map tegi.
2. Map tegi yordamida amallar bajaring.

7-MAVZU. STILLAR

Reja:

1. CSS haqida tushuncha

2. CSS ni HTML hujjati ichida ishlatilishi

CSS — bu stillar bilan ishlash uchun mo'ljallangan kodlar majmuasidir. Hozirgi kunga kelib yangi CSS3 versiyasi ishlab chiqilmoqda, lekin hali ko'p brauzerlar bu stillarni aks ettira olmayapti. Shuning uchun CSS3 yaratuvchilari brauzer o'rtasida kelishmovchiliklarni oldini olish maqsadida brauzer turiga qarab har xil prefikslar ishlab chiqishdi va bu prefikslardan CSS3 to'liq yaratilmaguncha ishlatish tavsiya etilgan, CSS3 to'liq yaratilgandan so'ng bu prefikslar olib tashlanadi. Bu prefikslar quydagilar:

Safari va Chrome brauzerlari uchun -webkitOpera brauzeri uchun -o-Firefox brauzeri uchun -moz- Internet Explorer(IE) brauzeri uchun esa -ms-

Man bu prefikslarni ishlatmagan holda misollar keltirdim, agar sizda bu misollar ishlamas oldiga prefikslarni qo'yib ishlatib ko'ring.

CSS3 da eng asosiy qo'shilgan stillardan biri bu animastiyalardir, ya'ni stillar almashinish jarayoni birdaniga emas, balki sekin astalik bilan sodir bo'lishidir. Quyida CSS3 versiyasiga yangi qo'shilgan ba'zi stillarni misol tariqasida ko'rsatib o'taman.

- Elementlarni(matn, rasm, fon,..) och(прозрачный) ko'rinishda aks ettirish. `background-color: rgba(10,0,255,0.7);`

Bu yerda sahifa foniga rang berib, shu rangni 0.7 darajali och tusga keltirilmoqda, tanishing CSS3 ning yangi elementi `rgba`.

- Elementlarning istalgan burchagiga(qismiga) bir vaqtning o'zida har xil rasmlarni joylashtirish. Bu saytni verstka qilish jarayonini yengillashtiradi.

`background:`

`url(top.gif) top left no-repeat, url(center.png) top 11px no-repeat, url(bottom.png) bottom left no-repeat, url(middle.png) left no-repeat;`

- `Resize` buyrug'i. Bu buyruq orqali foydalanuvchi istalgan elementning o'lchamini o'zgartirish mumkin bo'ladi.

```
div.resize { width: 25px; height: 35px; resize: both;
}
```

- CSS3 ning yana bir yangiligi bu — elementlar(shakllar, bloklar) burchagini istalgan radiusda burish mumkinligidir. Ko'p web saytlar forma yaratish jarayonida, formaning burchaklarini

burishni verstka paytida biror rasm orqali amalga oshirishadi, bu esa qo‘shimcha ish va vaqt yo‘qotishga olib keladi, CSS3 da bu muammo bartaraf etilgan va quyidagicha amalga oshirish mumkin bo‘ladi.

```
#forma {  
border-bottom-right-radius: 2em; border-bottom-left-radius: 1em;  
border-top-left-radius: 5em;  
border-top-right-radius: 3em;  
}
```

- CSS3ning soyalar bilan ishlash qismi. Barcha “p” teglari uchun soyalar hosil qilish.

```
P {  
text-shadow: #003471 /* soya rangi */ 2px /* o‘ng tomonga surilishi */  
5px /*pastga surilish */ 2px /* размытие*/;  
}
```

- Internetda ko‘p foydalanuvchilar verdana shrifti bilan ishlaydi. Nega? Chunki bu shrift barcha kompyuterlarda mavjud va brauzerda chiroyli ko‘rinishga ega. Agar stillarda qo‘llanilgan shrift foydalanuvchi kompyuterida mavjud bo‘lmasa, brauzer matnni istalgan boshqa shriftga ko‘rsatishi mumkin. Bu esa shriftlar rang barangligiga olib keladi. CSS3 da shriftlar bilan ishlash uchun yangi komanda @font-face.

```
@font-face {  
font-family: shrift_akm;  
src: url('http://blabla.uz/fonts/shrift_akm.ttf');  
}
```

```
h1 {  
font-family: shrift_akm;  
}
```

- Web sahifada kolonkalar. Bunisiga nima deysiz? Element ichidagi matnlarni bir necha ustunli kolonkalarda chiqarishingiz mumkin.

```
div {  
column-width: 15em; column-gap: 2em; /* yashil rangda */ column-  
rule: 4px solid red; /* qizil rangda */ padding: 5px; /* qora rangda */  
}-----
```

```
< div>
```

```
42242 422 42 422 422 4565 56 5454 54565 456
```


</div>

CSS — bu stillar bilan ishlay oladigan kaskadli stillar majmuasidir. Web sahifalarga har xil stillar berish uchun ishlatiladi. HTML hujjat ichida teglar bilan birga foydalaniladi. CSS kodlarini kompilyastiya qilish uchun qandaydir kompilyatorlar, qandaydir dasturlar, va bu kod tushuna oladigan qandaydir redaktorlar kerak bo‘lmaydi. Bu kodlarni HTML singari web brauzerning o‘zi kompilyastiya qiladi va natijani chiqarib beradi.

Bu stillar jadvali(CSS) to‘liq holatda 1997-yilda tashkil topib, “WWW Consorcium” da qo‘llab quvvatlangan va foydalanishga taqdim etilgan. O‘sha paytlarda HTML ning 3 versiyasi ommabop edi. CSS ni dastlab Netscape Navigator 4.0 va Internet Explorer 4.0 brauzerlarida ko‘rish mumkin bo‘lgan. Hozirda barcha brauzerlar CSS da yozilgan kodlarni tushuna oladi. Bu stillarga oid dasturlash tilini o‘rganish sodda bo‘lib, yangi stillar qo‘shilishi natijasida kengayib bormoqda.

CSS ni HTML hujjati ichida ishlatilishini birin-ketin ko‘rib o‘tamiz:

1. Web sahifada ishlatiladigan CSS kodlari orqali hosil qilinadigan stillarni hammasini bitta faylga yozib, uni serverda saqlaymiz. So‘ng, sahifadan, serverdagi saqlab qo‘yilgan faylga murojaat qilib, kerakli stilni olamiz. Bu usul katta hajmdagi web sahifalar yaratishda foydali hisoblanadi. Faylga murojaat, < head>tegi ichida amalga oshiriladi va quyidagi ko‘rinishga ega bo‘ladi.

```
< LINK REL=STYLESHEET TYPE = "text/CSS" HREF="URL">
```

< link> teги sahifa biror faylga murojaatni amalga oshirmoqda degani, REL- fayl qandaydir stillardan iborat degani, TYPE — stillar CSS kodlaridan iborat va fayl joylashgan manzil yoziladi.

1. CSS kodlar veb sahifa ichidagi teги orasida beriladi. Bunda yaratilgan stillar shu sahifaga tegishli bo‘ladi, bu degani faqat shu sahifada ishlatilish mumkindir.

```
< head>< STYLE TYPE="text/CSS">
```

```
<!--
```

```
A {text-decoration:none; }
```

```
-->
```

```
</--STYLE></head>
```

2. Navbatdagi turi, bu har bir element uchun alohida teg ichida stil berib chiqishdan iborat. Misol uchun, “p” teги ichida biror matn yozilsa, shu matn uchun teg ichida alohida stil beriladi va bu stil shu teg uchun xususiy bo‘ladi. Bu usuldan foydalanishni maslahat

bermayman, chunki bu usul kodlarni chalkash va tushunarsiz bo‘lishiga olib keladi. Shoshilinch vaziyatlarda yoki stil qay tartibda namoyon bo‘lishni ko‘rib olish uchun bu usuldan foydalanishni maslahat beraman.

Undan tashqari < style> tegi orqali CSS faylni import qilish mumkin bo‘ladi.

```
@import: url(mystyles.CSS);
```

CSS stillarini ishlatishda, quyidagi guruhlash usullaridan foydalanish, tartibli web sahifa tuzishga olib keladi:

- Kodlarni ixchamlash uchun, xar xil elementlar uchun bir xil tipdagi stillarni guruhlab ishlatish maqsadga muvofiqdir.

```
H1 {font-family: Verdana}
```

```
H2 {font-family: Verdana}
```

Yuqoridagi stilni guruhlab quyidagicha yozamiz.

```
H1, H2 {font-family: Verdana}
```

- Guruhlashda bir xil element uchun bir necha stillarni birlashtirish mumkin.

```
H2 {font-weight: bold}
```

```
H2 {font-size: 14pt}
```

```
H2 {font-family: Verdana}
```

Barcha stillarni bittaga yozamiz.

```
H2 {font-weight: bold; font-size: 14pt; font-family: Verdana;}
```

- Ba’zi stillar bitta stil ichida berilishi ham mumkin.

```
H2 {font: bold 14pt Verdana}
```

CSS stillari bilan ishlaganda, element ichidagi elementlar ham o‘zidan yuqori to‘rgan element stilini qabul qiladi, ya’ni o‘zidan oldingi element ”ota” vazifasini bajaradi va o‘z “farzandlari”ga ham o‘zida borini beradi. Misol uchun, < p> tegi ichidagi matn ko‘k rangda yozilishi lozim bo‘lsa(P {color: blue}), < p> tegi ichidagi < em> tegiga tegishli bo‘lgan matn ham ko‘k rangda yoziladi.

Ba’zi stillar faqat yakka tartibda ishlaydi, ya’ni ichki teglarga stillari o‘tmaydi(misol uchun, background), shuning uchun barcha stillarni birma bir ishlatib, o‘rganib olish lozim. Tegiga stil berib, dastlabki o‘rnatilish (po umolchaniyu) jarayonini hosil qilish mumkin. Shunda biror elementga stil berish esdan chiqib qolsa, <body> tegida ta’luqli stil esdan chiqqan elementga o‘rnatiladi.

```
BODY {color: green; font-family: "Verdana"; background: url(joke.gif) white; }
```

CSS fayl biriktirmasi. CSS faylini ulashning bir necha yo‘li mavjud. Saytlarni yaratishda eng ko‘p qo‘llaniladigan ikkita usul haqida gaplashamiz.

1. Bog‘lash. Ushbu usul bitta faylda barcha sahifalar uchun uslublarni o‘rnatishingiz kerak bo‘lganda ishlatiladi. Ushbu usul sayt yaratishda ko‘pincha ishlatiladi. Uslublar jadvalini ulash uchun teg tanasida joylashtirilishi kerak bo‘lgan buyruqdan foydalaning.

Dastlabki ikkita xususiyat brauzerda sayt CSS ishlatilishini bildiradi, so‘ngra uslublar bilan faylning manzili ko‘rsatiladi.

2. Hujjat teglariga joylashtirish. Ushbu usul bilan ma‘lum bir sahifa elementi uchun uslub to‘g‘ridan-to‘g‘ri HTML-kodda o‘rnatiladi

CSS menyu parametrlari

CSS menyularining bir nechta turlari mavjud:

1. Asosiy menyu . Bu veb-saytning asosiy bo‘limlariga aniq havolalar ro‘yxati.
2. Qo‘shimcha menyu. Bunday menyu ierarxik tuzilishga ega va uch turlidir:
 - Menyuni ochish Element ustida kursorni ochganda kengayadi,
 - Saytning qalqib chiqadigan menyusi. Bu ma‘lum bir narsani bosganingizda paydo bo‘ladi,
 - CSS uchun statik qo‘shimcha menyu. Daraxt tuzilishi, allaqachon ochilgan shaklda.

CSS fayli html kabi oson. Istalgan matn muharririga, masalan Notepadga kirish va shu bilan fayl yaratish kifoya (**CSS kengaytmasi**). Keling, unga standart nom beraylik —**uslub.CSS**.

Tashqi uslublar jadvali. Ehtimol tashqi **CSS**— Bu sayt uchun eng to‘g‘ri va maqbul yechim. Buni faqat sahifada ko‘rsatish kifoya **havola yorlig‘i** (boshqa fayllarni ulash uchun yaratilgan) uslublar jadvaliga havola va uslublar ulangan!

Bog‘lanish yorlig‘idagi satr style.CSS bizning faylimiz bilan bir xil katalogda (papkada) joylashganligini taxmin qiladi (masalan, index.html). Agar CSS faylining joylashuvi o‘zgargan bo‘lsa, unda href atributida tegishli o‘zgarishlar amalga oshirilishi kerak. Masalan, ko‘pincha CSS ostida ular alohida papka yaratadilar (bu qulay va mantiqiy). Ammo keyin href atributi boshqacha ko‘rinadi:

Ichki uslublar jadvallari. Ba‘zan siz to‘g‘ridan-to‘g‘ri kiritilgan boshqa CSS tuzilishini ko‘rishingiz mumkin. Plyus shundan

iboratki, siz darhol bitta faylda uslublarni o‘rnatishingiz mumkin. Muhim minus — o‘rnatilgan xususiyatlar faqat ma’lum bir elementga tayinlanadi. Shunday qilib, masofadan ulash xususiyatlarining barcha foydalari yo‘qoladi.

Ichki uslublar jadvallari. Oson yo‘l bor. CSS qo‘shimchalari. Agar siz alohida CSS faylini yaratishni xohlamasangiz va siz faqatgina ikkita xususiyatdan foydalanishingiz kerak bo‘lsa, unda ushbu parametr mos keladi. Keling, darhol misol bilan boshlaylik. Misoldan ko‘rinib turibdiki, biz yopilishdan oldin uslublarni o‘rnatdik(bosh yorlig‘i ichidagi ochish va yopish yorliqlari uslubi). Aslida teg uslubi ichida buyurilishi mumkin tanasi, lekin agar sizning eslatmalarangiz tuzilgan ko‘rinishda bo‘lishini va siz ushbu kodni ko‘rganingizdan keyin uning boshini ushlamasligini xohlasangiz, uslublarni bir joyda yozganingiz ma’qul.

Nazorat savollari.

1. CSS nima?
2. CSS da jadvallar hosil qiling.
3. CSS da ichki uslublar jadvallari?

8-MAVZU. SELEKTORLAR

Reja:

1. Selektorlar haqida tushuncha

2. Selektorlarining o'ziga xosligi

CSS selektorlari maxsus HTML elementlarini CSS uslublari uchun maqsad sifatida aniqlaydi. Ushbu mavzu CSS selektorlari HTML elementlarini qanday nishonga olishini o'z ichiga oladi. Selektorlar elementlar, sinflar, identifikatorlar, psevdoelementlar va psevdosinflar va naqshlarni o'z ichiga olgan CSS tili tomonidan taqdim etilgan 50 dan ortiq tanlov usullaridan keng foydalanadi.

#Atribut selektorlari

#Umumiy ko'rinish

Atribut selektorlari tanlash mezonlarini mos ravishda o'zgartiradigan har xil turdagi operatorlar bilan ishlatilishi mumkin. Ular berilgan atribut yoki atribut qiymati mavjudligidan foydalanib, elementni tanlaydilar.

Selektor	Mos keladigan element	Elementlarni tanlaydi...	CSS versiyasi
[attr]	<div attr>	Atribut bilan attr	2
[attr='val']	<div attr="val">	Atribut attr qiymatga ega bo'lgan joyda val	2
[attr~='val']	<div attr="val val2 val3">	Bo'shliqlar bilan ajratilgan ro'yxatda qayerda val paydo bo'ladi attr	2
[attr^='val']	<div attr="val1 val2">	Attr qiymati qayerdan boshlanadi _val	3
[attr\$='val']	<div attr="sth aval">	ning attr qiymati qayerda tugaydi val	3
[attr*='val']	<div attr="somevalhere">	Qayerda har qanday joy attr mavjud val	3
[attr = 'val']	<div attr="val-sth etc">	Qiymati qayerda attr aynan val,	2

		yoki bilan boshlanadi val va darhol (U+002D) bilan boshlanadi	
[attr='val' i]	<div attr="val">	Qayerda attr qiymat bor val, harf kattaligi e'tiborga olinmaydi .val	4 ⁽²⁾

Eslatmalar:

- Atribut qiymati bitta tirnoq yoki ikkita tirnoq bilan o'ralgan bo'lishi mumkin. Hech qanday tirnoq ham ishlamasligi mumkin, lekin u CSS standartiga ko'ra haqiqiy emas va tavsiya etilmaydi.
- Yagona, birlashtirilgan CSS4 spetsifikatsiyasi yo'q, chunki u alohida modullarga bo'lingan. Biroq, "4-darajali" modullar mavjud. [Brauzer yordamiga qarang](<http://caniuse.com/#feat=CSS-case-insensitive>).

#Tafsilotlar

#[attribute]

Berilgan atributga ega elementlarni tanlaydi.

```
div[data-color] {
  color: red;
}
<div data-color="red">This will be red</div>
<div data-color="green">This will be red</div>
<div data-background="red">This will NOT be red</div>
```

#[attribute="value"]

Berilgan atribut va qiymatga ega elementlarni tanlaydi.

```
div[data-color="red"] {
  color: red;
}
<div data-color="red">This will be red</div>
<div data-color="green">This will NOT be red</div>
<div data-color="blue">This will NOT be red</div>
```

#[attribute*="value"]

Berilgan atribut va qiymatga ega elementlarni tanlaydi, bunda berilgan atribut istalgan joyda berilgan qiymatni o‘z ichiga oladi (pastki qator sifatida).

```
[class*="foo"] {
  color: red;
}
<div class="foo-123">This will be red</div>
<div class="foo123">This will be red</div>
<div class="bar123foo">This will be red</div>
<div class="barfoo123">This will be red</div>
<div class="barfo0">This will NOT be red</div>
```

#[attribute~="value"]

Belgilangan atribut va qiymatga ega bo‘lgan elementlarni tanlaydi, bu yerda berilgan qiymat bo‘sh joy bilan ajratilgan ro‘yxatda paydo bo‘ladi.

```
[class~="color-red"] {
  color: red;
}
<div class="color-red foo-bar the-div">This will be red</div>
<div class="color-blue foo-bar the-div">This will NOT be red</div>
```

#[attribute^="value"]

Berilgan atribut va qiymatga ega elementlarni, bu atribut qiymatdan boshlanadigan joyni tanlaydi.

```
[class^="foo-"] {
  color: red;
}
<div class="foo-123">This will be red</div>
<div class="foo-234">This will be red</div>
<div class="bar-123">This will NOT be red</div>
```

#[attribute\$="value"]

Berilgan atribut va qiymat berilgan atribut berilgan qiymat bilan tugaydigan elementlarni tanlaydi.

```
[class$="file"] {  
  color: red;  
}
```

```
<div class="foobar-file">This will be red</div>  
<div class="foobar-file">This will be red</div>  
<div class="foobar-input">This will NOT be red</div>
```

#[attribute]="value"]

Belgilangan atribut va qiymatga ega elementlarni tanlaydi, bunda atributning qiymati aynan berilgan qiymat yoki aynan berilgan qiymatdan keyin -(U+002D)

```
[lang|"EN"] {  
  color: red;  
}
```

```
<div lang="EN-us">This will be red</div>  
<div lang="EN-gb">This will be red</div>  
<div lang="PT-pt">This will NOT be red</div>
```

#[attribute="value" i]

Berilgan atribut va qiymatga ega elementlarni tanlaydi, bunda atribut qiymati Value, VALUE, vAlUeyoki katta-kichik harflarga sezgir bo‘lmagan boshqa imkoniyat sifatida ko‘rsatilishi mumkin.

```
[lang="EN" i] {  
  color: red;  
}
```

```
<div lang="EN">This will be red</div>  
<div lang="en">This will be red</div>  
<div lang="PT">This will NOT be red</div>
```

Sinf selektori va psevdoklass bilan bir xil.

Esda tutingki, bu atribut selektoridan elementni identifikatori bo‘yicha tanlab olish uchun identifikator selektori bilan tanlanganidan

ko‘ra pastroq o‘ziga xoslik darajasida foydalanish mumkin. : [id="my-ID"]bir xil elementni nishonga oladi, #my-ID lekin o‘ziga xosligi pastroq.

Sintaksis bo‘limiga qarang batafsil ma’lumot uchun.

#Asosiy selektorlar

Selektor	Tavsif
*	Universal selektor (barcha elementlar)
div	Teg selektori (barcha <div>elementlar)
.blue	Sinf selektori (sinfga ega barcha elementlar blue)
.blue.red	Sinfga ega barcha elementlar blue va red (Murakkab selektor turi)
#headline	ID selektori ("id" atributiga ega element headline)
:pseudo-class	<u>Pseudo-klassga ega barcha elementlar</u>
::pseudo-element	<u>Pseudo-elementga mos keladigan element</u>
:lang(en)	Masalan, :lang deklaratsiyasiga mos keladigan element
div > p	bolalar tanlovchisi

Eslatma: ID qiymati veb-sahifada yagona bo‘lishi kerak. Bu HTML standartining buzilishi identifikatorning qiymatni bir xil hujjat daraxtida bir necha marta ishlatish.

Selektorlarning to‘liq ro‘yxatini CSS Selectors Level 3 spetsifikatsiyasida topish mumkin.

#Kombinatorlar. #Umumiy ko‘rinish

Selektor	Tavsif
div span	Avlod selektori (a ning avlodlari bo‘lgan barcha lar <div>)
div > span	Child selektor (to‘g‘ridan-to‘g‘ri bolasi bo‘lgan barcha <div> lar)
a ~ span	Umumiy aka-uka selektori (dan keyin aka-uka bo‘lgan barcha <a>)
a + span	Qo‘shni Sibling selektori (birdan keyin bo‘lgan barcha <a>)

Eslatma: Sibling selektorlari dastlabki hujjatda ulardan keyin keladigan elementlarni nishonga oladi. CSS o‘z tabiatiga ko‘ra (u kaskadli) **oldingi** yoki **asosiy** elementlarni nishonga olmaydi. Biroq,

flex order xususiyatidan foydalanib, oldingi birodar selektorini vizual muhitda simulyatsiya qilish mumkin.

#Avlod birlashtiruvchi:selector selector

Kamida bitta bo'sh joy belgisi (") bilan ifodalangan avlod kombinatori belgilangan elementning avlodi bo'lgan elementlarni tanlaydi. Bu kombinator elementning **barcha** avlodlarini tanlaydi (bola elementlardan pastga).

```
div p {
  color:red;
}
<div>
  <p>My text is red</p>
<section>
  <p>My text is red</p>
</section>
</div>
<p>My text is not red</p>
```

Yuqoridagi misolda birinchi ikkita <p>element tanlangan, chunki ularning ikkalasi ham <div>.

#Bolalar kombinatori:selector > selector

Bola () birlashtiruvchi ko'rsatilgan elementning **bolalari** yoki **to'g'ridan-to'g'ri avlodlari** bo'lgan elementlarni tanlash uchun ishlatiladi .

```
div > p {
  color:red;
}

<div>
  <p>My text is red</p>
<section>
  <p>My text is not red</p>
</section>
</div>
```

Yuqoridagi CSS faqat birinchi <p>elementni tanlaydi, chunki u to'g'ridan-to'g'ri a dan olingan yagona paragrafdir <div>.

Ikkinchi <p>element tanlanmagan, chunki u ning bevosita bolasi emas <div>.

#Qo'shni aka-uka birlashtiruvchi:selector + selector

Qo‘shni birodar (+) kombinatori ko‘rsatilgan elementga darhol ergashuvchi birodar elementni tanlaydi.

```
p + p {
  color:red;
}
<p>My text is not red</p>
<p>My text is red</p>
<p>My text is red</p>
<hr>
<p>My text is not red</p>
```

Yuqoridagi misol faqat to‘g‘ridan-to‘g‘ri boshqa elementdan **oldin** bo‘lgan elementlarni tanlaydi .<p>

#Umumiy aka-uka kombinati:selector ~ selector

Umumiy birodar() kombinatori belgilangan elementga amal qiladigan **barcha**~ birodarlarni tanlaydi .

```
p ~ p {
  color:red;
}
<p>My text is not red</p>
<p>My text is red</p>
<hr>
<h1>And now a title</h1>
<p>My text is red</p>
```

Yuqoridagi misol, ular darhol qo‘shni bo‘ladimi yoki yo‘qmi, boshqa elementdan **oldin**<p> barcha elementlarni tanlaydi.<p>

#Pseudo-sinflar

Pseudo-sinflar Hujjat daraxtidan tashqarida joylashgan yoki boshqa selektor yoki kombinatorlar tomonidan ifodalab bo‘lmaydigan ma’lumotlar asosida tanlash imkonini beruvchi **kalit so‘zlar** . Ushbu ma’lumot ma’lum bir davlat bilan bog‘lanishi mumkin (shtat va dinamik psevdosinflar), joylarga (tizimli va maqsad psevdosinflar), birinchisining inkorlariga (inkor psevdosinf) yoki tillarga (lang psevdosinf). Masalan, havolaga amal qilingan yoki bajarilmagan (:visited), sichqoncha element ustida joylashgan (:hover), belgilash katakchasi belgilangan (:checked) va hokazo.

#Sintaksis

```
selector:pseudo-class {
```

property: value;

}

#Pseudo-sinflar ro'yxati:

Ism	Tavsif
<u>:active</u>	Foydalanuvchi tomonidan faollashtirilgan (ya'ni bosilgan) har qanday element uchun amal qiladi.
<u>:any</u>	Kiritilgan elementlar mos keladigan guruhlarni yaratish orqali tegishli selektorlar to'plamini yaratishga imkon beradi Bu butun selektorni takrorlashga muqobildir.
<u>:target</u>	Joriy faol #news elementini tanlaydi (ushbu langar nomini o'z ichiga olgan URL manziliga bosing)
<u>:checked</u>	Belgilangan yoki "yoqilgan" holatiga o'tgan radio, belgilash qutisi yoki parametr elementlariga qo'llaniladi.
<u>:default</u>	O'xshash elementlar guruhi orasida standart bo'lgan har qanday foydalanuvchi interfeysi elementini ifodalaydi.
<u>:disabled</u>	O'chirilgan holatda bo'lgan har qanday UI elementiga qo'llaniladi.
<u>:empty</u>	Bolalari bo'lmagan har qanday element uchun amal qiladi.
<u>:enabled</u>	Yoqilgan holatda bo'lgan har qanday UI elementiga qo'llaniladi.
<u>:first</u>	Qoida bilan birgalikda foydalaniladi @page, bu chop etilgan hujjatning birinchi sahifasini tanlaydi.
<u>:first-child</u>	Ota-onasining birinchi yordamchi elementi bo'lgan har qanday elementni ifodalaydi.
<u>:first-of-type</u>	Element o'zining ota-onasi ichida tanlangan element turidan birinchi bo'lsa qo'llaniladi. Bu birinchi farzand bo'lishi mumkin yoki bo'lmasligi mumkin.
<u>:focus</u>	Foydalanuvchi diqqat markazida bo'lgan har qanday element uchun amal qiladi. Bu foydalanuvchi klaviaturasi, sichqoncha hodisalari

	yoki boshqa kiritish shakllari orqali berilishi mumkin .
<u>:focus-within</u>	Undan butun qismni ta'kidlash uchun foydalanish mumkin, agar uning ichidagi bir element fokuslangan bo'lsa. U :focus psevdosinfiga mos keladigan yoki avlodi fokuslangan har qanday elementga mos keladi.
<u>:full-screen</u>	To'liq ekran rejimida ko'rsatilgan har qanday element uchun amal qiladi. U faqat yuqori darajadagi elementni emas, balki butun elementlar to'plamini tanlaydi .
<u>:hover</u>	Foydalanuvchining ko'rsatuvchi qurilmasi ko'rsoriga o'tayotgan, lekin faollashtirilmagan har qanday elementga taalluqlidir.
<u>:indeterminate</u>	Belgilanmagan yoki belgilanmagan, lekin noaniq holatda bo'lgan radio yoki belgilash katakchasi UI elementlarini qo'llaydi. Bu elementning atributi yoki DOM manipulyatsiyasi bilan bog'liq bo'lishi mumkin
<u>:in-range</u>	CSS :in-range psevdosinfi, agar element ushbu element uchun belgilangan diapazon chegaralari ichida qiymat atributiga ega bo'lsa, mos keladi. Bu sahifaga element yordamida hozirda aniqlangan qiymat diapazon chegaralari ichida ekanligi haqida fikr bildirishga imkon beradi .
<u>:invalid</u>	Atributda ko'rsatilgan turga <input>ko'ra qiymatlari yaroqsiz bo'lgan elementlarga nisbatan qo'llaniladi . type=
<u>:lang</u>	<body>O'rash elementi to'g'ri belgilangan atributga ega bo'lgan har qanday elementga qo'llaniladi lang=. Pseudo-sinf haqiqiy bo'lishi uchun u ikki yoki uch harfli haqiqiy til kodini o'z ichiga olishi kerak.
<u>:last-child</u>	Ota-onasining oxirgi elementi bo'lgan har qanday elementni ifodalaydi.
<u>:last-of-type</u>	Element uning ota-onasi ichida tanlangan element turining oxirgisi bo'lganda qo'llaniladi . Bu oxirgi bola bo'lishi mumkin yoki bo'lmasligi mumkin.

<u>:left</u>	Qoida bilan birgalikda foydalaniladi , bu chop etilgan hujjatning @pagebarcha chap sahifalarini tanlaydi .
<u>:link</u>	Foydalanuvchi tashrif buyurmagan havolalar uchun amal qiladi.
<u>:not()</u>	O‘tkazilgan qiymatga mos kelmaydigan barcha elementlarga qo‘llaniladi (:not(p)yoki :not(.class-name)masalan. U to‘g‘ri bo‘lishi uchun qiymatga ega bo‘lishi kerak va u faqat bitta selektorni o‘z ichiga olishi mumkin. Biroq, siz bir nechta :not selektorlarni bir-biriga bog‘lashingiz mumkin.
<u>:nth-child</u>	Element nota-onasining --chi elementi bo‘lganda qo‘llaniladi, bu yerda n butun son, matematik ifoda (masalan ,) yoki kalit so‘zlar yoki n+3kalit so‘zlar bo‘lishi mumkin . oddeven
<u>:nth-of-type</u>	Element bir xil element n turidagi ota-onasining --chi elementi bo‘lganda qo‘llaniladi , bu yerda n butun son, matematik ifoda (masalan ,) yoki kalit so‘zlar yoki n+3kalit so‘zlar bo‘lishi mumkin .oddeven
<u>:only-child</u>	CSS psevdosinfi ota-onasining yagona farzandi bo‘lgan :only-child har qanday elementni ifodalaydi . Bu yoki bilan bir xil , lekin o‘ziga xosligi pastroq. :first-child:last-child:nth-child(1):nth-last-child(1)
<u>:optional</u>	CSS pseudo-klassi kerakli atributga ega bo‘lmagan :optionalhar qanday elementni ifodalaydi . Bu shakllarga ixtiyoriy maydonlarni osongina ko‘rsatishga va ularni mos ravishda uslublashga imkon beradi.
<u>:out-of-range</u>	CSS :out-of-rangepsevdosinfi elementning qiymat atributi ushbu element uchun belgilangan diapazon cheklovlaridan tashqarida bo‘lganda mos keladi.

	Bu sahifaga element yordamida hozirda aniqlangan qiymat diapazon chegaralaridan tashqarida ekanligi haqida fikr bildirishga imkon beradi. Qiymat maksimal va minimal belgilangan qiymatlardan kichikroq yoki kattaroq bo'lsa, diapazondan tashqarida bo'lishi mumkin .
<u>:placeholder-shown</u>	Eksperimental. To'ldiruvchi matnni ko'rsatayotgan har qanday shakl elementiga qo'llaniladi.
<u>:read-only</u>	Foydalanuvchi tomonidan tahrirlab bo'lmaydigan har qanday element uchun amal qiladi.
<u>:read-write</u>	Elementlar kabi foydalanuvchi tomonidan tahrirlanadigan har qanday elementga taalluqlidir <input>.
<u>:right</u>	Qoida bilan birgalikda foydalaniladi @page, bu chop etilgan hujjatning barcha to'g'ri sahifalarini tanlaydi
<u>:root</u>	hujjatni ifodalovchi daraxtning ildiz elementiga mos keladi.
<u>:scope</u>	CSS psevdosinfi selektorlar mos keladigan mos yozuvlar nuqtasi bo'lgan elementlarga mos keladi .
<u>:target</u>	Joriy faol #news elementini tanlaydi (ushbu langar nomini o'z ichiga olgan URL manziliga bosing)
<u>:visited</u>	Foydalanuvchi tashrif buyo'rgan har qanday havolalar uchun amal qiladi.

Psevdoklassni :visitedendi ko'plab zamonaviy brauzerlarda ko'pchilik uslublar uchun ishlatib bo'lmaydi, chunki bu xavfsizlik teshigi. Ushbu havolaga qarang ma'lumot uchun.

#Bolalar psevdosinfi

CSS psevdosinfi n uchun berilgan musbat **yoki nol qiymati** uchun hujjat daraxtida o'zidan oldin +b-1 aka-uka bo'lgan elementga mos keladi" — [MDN :nth-child](https://developer.mozilla.org/en-US/docs/Web/CSS/:nth-child)

psevdoselektor	1	2	3	4	5	6	7	8	9	10
:first-child	✓									
:nth-child(3)			✓							

:nth-child(n+3)			✓	✓	✓	✓	✓	✓	✓	✓
:nth-child(3n)			✓			✓			✓	
:nth-child(3n+1)	✓			✓			✓			✓
:nth-child(-n+3)	✓	✓	✓							
:nth-child(odd)	✓		✓		✓		✓		✓	
:nth-child(even)		✓		✓		✓		✓		✓
:last-child										✓
:nth-last-child(3)								✓		

#Sinf nomini tanlash vositalari

Sinf nomi selektori maqsadli sinf nomi bilan barcha elementlarni tanlaydi. Masalan, sinf nomi .warning quyidagi elementni tanlaydi <div>:

```
<div class="warning">
```

```
  <p>This would be some warning copy.</p>
```

```
</div>
```

Bundan tashqari, aniqroq maqsadli elementlar uchun sinf nomlarini birlashtirishingiz mumkin. Keling, murakkabroq sinf tanlovini namoyish qilish uchun yuqoridagi misolga asoslanaylik.

CSS

```
.important {
```

```
  color: orange;
```

```
}
```

```
.warning {
```

```
  color: blue;
```

```
}
```

```
.warning.important {
```

```
  color: red;
```

```
}
```

HTML

```
<div class="warning">
```

```
  <p>This would be some warning copy.</p>
```

```
</div>
```

```
<div class="important warning">
```



```
<p class="important">This is some really important warning copy.</p>
</div>
```

Ushbu misolda sinfga ega bo'lgan barcha elementlar ko'k .warning rangli matn rangiga ega bo'ladi, .important sinfga ega bo'lgan elementlar to'q sariq rangga ega bo'ladi va **ikkala**.important va .warningsinf nomiga ega bo'lgan barcha elementlar qizil matn rangiga ega bo'ladi.

E'tibor bering, CSS ichida .warning.important deklaratsiyada ikkita sinf nomi o'rtasida bo'sh joy yo'q edi. Bu shuni anglatadiki, u faqat ikkala sinf nomini va important ularning atributini o'z ichiga olgan elementlarni topadi. Ushbu sinf nomlari elementda istalgan tartibda bo'lishi mumkin.

.warning. Agar CSS deklaratsiyasida ikkita sinf o'rtasida bo'sh joy qo'shilgan bo'lsa, u faqat sinf nomlari bo'lgan ota-ona elementlari va sinf nomlari bo'lgan pastki elementlarga ega bo'lgan elementlarni tanlaydi.

#ID selektorining yuqori o'ziga xosligisiz uning identifikatoridan foydalanib elementni tanlang

Ushbu hiyla sizga ID selektorining yuqori o'ziga xosligini oldini olish uchun atribut selektori uchun qiymat sifatida ID yordamida elementni tanlashga yordam beradi.

HTML:

```
<div id="element">...</div>
```

CSS

```
#element { ... } /* High specificity will override many selectors */
[id="element"] { ... } /* Low specificity, can be overridden easily */
```

#Belgilangan katakchali global mantiqiy: belgilangan va ~ (umumiy aka-uka kombinatori)

~ selektori yordamida siz JavaScriptni ishlatmasdan global kirish mumkin bo'lgan mantiqiy tilni osongina amalga oshirishingiz mumkin.

#Booleanni tasdiqlash qutisi sifatida qo'shing

Hujjatingizning eng boshida noyob id va hidden atributlar to'plami bilan xohlaganingizcha mantiqiy qiymatlarni qo'shing:

```
<input type="checkbox" id="sidebarShown" hidden />
```

```

<input type="checkbox" id="darkThemeUsed" hidden />
<!-- here begins actual content, for ehample: -->
<div id="container">
  <div id="sidebar">
    <!-- Menu, Search, ... -->
  </div>
  <!-- Some more content ... -->
</div>
<div id="footer">
  <!-- ... -->
</div>

```

#Boolean qiymatni o‘zgartiring

Label Atributlar to‘plamiga a qo‘shish orqali mantiqiy qiymatni almashtirishingiz mumkin :

```
<label for="sidebarShown">Show/Hide the sidebar!</label>
```

#CSS yordamida mantiqiy qiymatga kirish

Oddiy selektor (kabi .color-red) standart xususiyatlarni belgilaydi. Ular quyidagi / false selektorlar orqali bekor qilinishi mumkin :

```
/* true: */
```

```
<checkbox>:checked ~ [sibling of checkbox & parent of target]
<target>
```

```
/* false: */
```

```
<checkbox>:not(:checked) ~ [sibling of checkbox & parent of target]
<target>
```

Shuni esda tutingki <checkbox>, [sibling ...]va ni <target>tegishli selektorlar bilan almashtirish kerak. [sibling ...]ma’lum bir selektor bo‘lishi mumkin, (ko‘pincha dangasa bo‘lsangiz) oddiy *yoki hech narsa, agar maqsad allaqachon tasdiqlash qutisining ukasi bo‘lsa.

Yuqoridagi HTML tuzilishiga misollar:

```

#sidebarShown:checked ~ #container #sidebar {
  margin-left: 300px;
}
#darkThemeUsed:checked ~ #container,
#darkThemeUsed:checked ~ #footer {
  background: #333;
}

```

```
}
```

Bu skripkani ko‘ring_bu global booleanlarni amalga oshirish uchun.

#CSS3: diapazon ichidagi selektor misoli

```
<style>
```

```
input:in-range {  
    border: 1px solid blue;
```

```
}
```

```
</style>
```

```
<input type="number" min="10" max="20" value="15">
```

```
<p>The border for this value will be blue</p>
```

CSS :in-rangepsevdo-sinfi, agar element ushbu element uchun belgilangan diapazon chegaralari ichida qiymat atributiga ega bo‘lsa, mos keladi. Bu sahifaga element yordamida hozirda aniqlangan qiymat diapazon chegaralari ichida ekanligi haqida fikr bildirishga imkon beradi.

#Javob: psevdosinf emas va B.: CSS psevdosinfida fokuslash

A. Sintaksis yuqorida keltirilgan.

Quyidagi selektor <input>HTML hujjatidagi o‘chirib qo‘yilmagan va sinfga ega bo‘lmagan barcha elementlarga mos keladi .ehample:

HTML:

```
<form>
```

```
    Phone: <input type="tel" class="ehample">
```

```
    E-mail: <input type="email" disabled="disabled">
```

```
    Password: <input type="password">
```

```
</form>
```

CSS:

```
input:not([disabled]):not(.ehample){
```

```
    background-color: #ccc;
```

```
}
```

Pseudo :not()-sinf, shuningdek, 4-darajali Selektorlarda vergul bilan ajratilgan selektorlarni qo‘llab-quvvatlaydi:

CSS:

```
input:not([disabled], .ehample){
```

```
    background-color: #ccc;
```

```
}
```

Bu erda fon sintaksisini ko‘ring_.

B. :focus-within CSS pseudo-selector

HTML:

```
<h3>Background is blue if the input is focused .</h3>
<div>
  <input type="text" />
</div>
```

CSS:

```
div {
  height: 80px;
}
input {
  margin: 30px;
}
div:focus-within {
  background-color: #1565C0;
}
```

```
div {
  height: 80px;
}
input {
  margin: 30px;
}
div:focus-within {
  background-color: #1565C0;
}
```

Background is blue if the input is focused .



:focus-within CSS pseudo-class Global 13.62%

The `:focus-within` pseudo-class matches elements that either themselves match `:focus` or that have descendants which match `:focus`.

Current aligned Usage relative Date relative Show all

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
		52	49			93		4.4	
	14	53	58		45	10.2		4.4.4	
11	15	54	59	10.1	46	10.3	all	56	59
	16	55	60	11	47	11			
		56	61	TP	48				
		57	62						

Notes Known issues (0) Resources (11) Feedback

Can be enabled via the "Experimental Web Platform Features" flag

#:oxirgi turdagi selektor

`:last-of-type` Muayyan turdagi ota-onasining oxirgi bolasi bo'lgan elementni tanlaydi. Quyidagi misolda CSS oxirgi paragraf va oxirgi sarlavhani tanlaydi h1.

```
p:last-of-type {
  background: #C5CAE9;
}
h1:last-of-type {
  background: #CDDC39;
}
<div class="container">
  <p>First paragraph</p>
  <p>Second paragraph</p>
  <p>Last paragraph</p>
  <h1>Heading 1</h1>
  <h2>First heading 2</h2>
  <h2>Last heading 2</h2>
</div>
```

First paragraph
Second paragraph
Last paragraph
Heading 1
First heading 2
Last heading 2

[jsFiddle](#)

#ID selektorlari

ID selektorlari maqsadli ID bilan DOM elementlarini tanlaydi. CSS-da ma'lum bir identifikator bo'yicha elementni tanlash uchun #prefiks ishlatiladi.

Masalan, quyidagi HTML divelementi...

```
<div id="ehampleID">
  <p>Ehample</p>
</div>
```

#ehampleID... quyida ko'rsatilganidek CSS-da tanlanishi mumkin :

```
#ehampleID {
  width: 20px;
}
```

Eslatma : HTML spetsifikatsiyalari bir xil identifikatorga ega bir nechta elementlarga ruxsat bermaydi.

#Rangli kiritishni qanday uslublash mumkin

HTML

```
<input type="range"></input>
```

CSS

Effekt	Pseudoselektor
Bosh barmog'i	input[type=range]::-webkit-slider-thumb, input[type=range]::-moz-range-thumb, input[type=range]::-ms-thumb
Trek	input[type=range]::-webkit-slider-runnable-track, input[type=range]::-moz-range-track, input[type=range]::-ms-track
OnFocus	input[type=range]:focus
Trekning pastki qismi	input[type=range]::-moz-range-progress, input[type=range]::-ms-fill-lower(hozirda brauzerlarida mumkin emas — JS kerak)

WebKit

#:only-child psevdosinf selektori misoli

Pseudo :only-child CSS-sinf ota-onasining yagona farzandi boʻlgan har qanday elementni ifodalaydi.

HTML:

```
<div>
```

```
  <p>This paragraph is the only child of the div, it will have the color blue</p>
```

```
</div>
```

```
<div>
```

```
  <p>This paragraph is one of the two children of the div</p>
```

```
  <p>This paragraph is one of the two children of its parent</p>
```

```
</div>
```

CSS:

```
p:only-child {  
  color: blue;  
}
```

Yuqoridagi misol <p>ota-onasidan yagona bola boʻlgan elementni tanlaydi, bu holda a <div>.

#Sintaksis

- # identifikator
- . sinf nomi
- : psevdo-sinf nomi
- :: psevdo-element nomi
- [attr] /* attr atributiga ega. */
- [attr = " qiymat "] /* attr atributiga ega va uning qiymati aynan " qiymat ". */
- [attr ~=" value "] /* attr atributiga ega va uning qiymati boʻsh joyga boʻlinganda " qiymat " ni oʻz ichiga oladi. */
- [attr |= " value "] /* attr atributiga ega va uning qiymati aynan " qiymat " yoki uning qiymati " qiymat *-*" bilan boshlanadi. */
- [attr ^= " value "] /* attr atributiga ega va uning qiymati " qiymat " bilan boshlanadi. */
- [attr \$= " qiymat "] /* attr atributiga ega va uning qiymati " qiymat " bilan tugaydi. */
- [attr *= " value "] /* attr atributiga ega va uning qiymati " qiymat " ni oʻz ichiga oladi. */

- element nomi
- *

#Izohlar

- ::Ba'zan bitta () o'rniga qo'sh nuqta () ni ko'rasiz :. Bu psevdosinflarni ajratishning bir usuli psevdo elementlardan.
- Internet Explorer 8 kabi eski brauzerlar psevdoelementlarni aniqlash uchun **faqat** bitta ikkita nuqta (:) qo'llab-quvvatlaydi.
- Psevdo-sinflardan farqli o'laroq, har bir selektor uchun faqat bitta psevdoelement ishlatilishi mumkin, agar mavjud bo'lsa, u selektor subyektlarini ifodalovchi oddiy selektorlar ketma-ketligidan keyin paydo bo'lishi kerak (W3C spetsifikatsiyasining kelajakdagi versiyasi har bir selektor uchun bir nechta psevdoelementlarga ruxsat berishi mumkin).

Nazorat savollari.

1. Psevdo-snif nima?
2. Oxirgi turdagi selektor.

9-MAVZU. CSSDA CHEGARA

Reja:

1. CSS chegaralari

2. Mos kodlarni ko‘rib chiqish

CSS va HTML jadvallar aralashmaganligini eshitgan bo‘lishingiz mumkin. Bu shunchaki haqiqat emas. Ha, jadval uchun HTML jadvallarni ishlatib, endi veb-dizaynning eng yaxshi amaliyoti emas, CSS-ning tartib-qoidalari bilan almashtirildi, ammo jadvallar veb-sahifaga jadvalli ma’lumotlarni qo‘shish uchun foydalanish uchun to‘g‘ri belgilar. Afsuski, juda ko‘p veb-mutaxassislar zahar ekanligini o‘ylab, jadvallardan uzoqlashib ketganligi sababli, ko‘plab professionallar ushbu umumiy HTML element bilan ishlashda va ularni veb-sahifasida ko‘rib chiqish kerak bo‘lgan ko‘rashda juda kam tajribaga ega. Misol uchun, agar siz sahifada jadvalga ega bo‘lsangiz va jadval chizig‘iga ichki satrlarni qo‘shishni xohlasangiz. CSS stoli chegaralari. Jadvallarga chegara qo‘shish uchun CSS-dan foydalanganda, u faqat jadvalning tashqarisidagi chegarani qo‘shadi. Jadvalning alohida hujayralariga ichki chiziqlar qo‘shish zarur bo‘lsa, ichki CSS elementlariga chegaralarni kiritishingiz kerak. Bundan tashqari, har bir hujayralar ichidagi qatorlarni qo‘shish uchun HR yorlig‘idan foydalanishingiz mumkin.

Ushbu maqolada keltirilgan uslublarni qo‘llash uchun veb-saytingizda jadval bo‘lishi kerak. Keyinchalik hujjatning boshida ichki uslublarni jadvalini yaratishingiz kerak bo‘ladi (ehtimol, agar siz "sayt" bitta sahifa bo‘lsa) yoki hujjatga tashqi stil sahifasi sifatida qo‘shilasiz (bu sizning agar saytingiz bir nechta sahifalar bo‘lsa, barchasini bitta tashqi varaqdan uslublashingiz mumkin). Uslublar jadvalini ichki jadvallarni ushbu uslub sahifasiga qo‘shish uchun joylashtirasiz.

Boshlashdan oldin

Birinchidan, chiziqlarni jadvalingizda ko‘rishni istagan joyni tanlashingiz kerak. Sizda turli xil variantlar mavjud:

- Barcha hujayralarni panjara hosil qilish uchun o‘rab qo‘yish;
- Chiziqlarni faqat ustunlar orasidagi joylashtirish;
- Faqat qatorlar orasidagi; yoki
- Muayyan ustunlar yoki qatorlar orasida.

Bundan tashqari, alohida xujayralar atrofidagi chiziqlarni yoki alohida kameralar ichida joylashtirishingiz mumkin.

Jadvaldagi barcha hujayralar atrofida chiziqlar qo‘shish
Jadvaldagi barcha xujayralar atrofidagi satrlarni qo‘shish uchun ushbu gridga o‘xshash effekt yaratish uchun uslub sahifasiga quyidagilarni qo‘shing:

```
td, son {  
chegarasi: qattiq 1px qora;  
}
```

Jadvaldagi faqat ustunlar orasidagi chiziqlar qanday qo‘shiladi
Ustunlar orasidagi satrlarni qo‘shish uchun (jadvalning ustunlaridan yuqoridan pastgacha bo‘lgan vertikal chiziqlar hosil qiladi), uslub sahifasiga quyidagilarni qo‘shing:

```
td, son {  
chegara-chap: qattiq 1px qora;  
}
```

So‘ngra, agar siz ularning birinchi ustunida paydo bo‘lishini xohlamasangiz, u va **td** hujayralariga sinf qo‘shishingiz kerak bo‘ladi. Ushbu misolda, biz ushbu hujayralardagi **chegara chegarasi mavjud** deb hisoblaymiz va chegarani ushbu aniqroq CSS qoidasi bilan olib tashlaymiz. Shunday qilib biz foydalanadigan HTML sinf:

```
class = "no-border">
```

Keyin uslublar jadvalimizga quyidagi uslubni qo‘shishimiz mumkin:

```
cheksiz {  
chegara-chap: yo‘q;  
}
```

Jadvalda faqat qatorlar orasidagi satrlarni qo‘shish
Ustunlar orasidagi chiziqlar qo‘shilgandek, uslublar jadvaliga qo‘shilgan oddiy uslub bilan buni qilishingiz mumkin. Quyidagi CSS jadvalimizning har bir qatori o‘rtasida vertikal chiziqlar qo‘shib beradi:

```
tr {  
pastki taglik: qattiq 1px qora;  
}
```

Chegarani stol ostidan olib tashlash uchun, siz yana bir marta ushbu tre tagiga sinf qo‘shasiz:

```
class = "no-border">
```

Uslublar jadvalingizga quyidagi uslubni qo‘shing:

```
cheksiz {  
chegara pastki: yo‘q;  
}
```

Jadvalda muayyan ustunlar yoki qatorlar orasidagi chiziqlar qo‘shish faqat ma’lum satrlar yoki ustunlar orasidagi chiziqlarni xohlasangiz, u hujayralar yoki satrlarda sinfdan foydalanishingiz kerak. Ustunlar orasidagi satr qo‘shish satrlar orasidagi biroz qiyinroqdir, chunki siz bu ustundagi har bir xonaga sinf qo‘shishingiz kerak. Agar sizning jadvalingiz avtomatik ravishda CSSdan yaratilgan bo‘lsa, bu mumkin bo‘lmasa ham, agar siz yo‘lni kodlashni qo‘llyotgan bo‘lsangiz, ushbu ta’sirga erishish uchun kerakli sinflarni qo‘shishingiz mumkin.

```
class = "yon chegaradagi">
```

Satrlar orasidagi satrlarni qo‘shish juda oson, chunki siz faqat satrni kerakli satrga qo‘shishingiz mumkin.

```
class = "border-bottom">
```

Keyin CSS uslublar jadvaliga qo‘shing:

```
.border-yon {  
chegara-chap: qattiq 1px qora;  
}
```

```
.border-bottom {  
pastki taglik: qattiq 1px qora;  
}
```

Jadvaldagi alohida hujayralar atrofida chiziqlar qo‘shish

Alohida hujayralar atrofidagi satrlarni qo‘shish uchun siz chekka atrofida kerakli hujayralarga sinf qo‘shasiz:

```
sinf = "chegara">
```

Keyin uslublar jadvaliga quyidagi CSS-ni qo‘shing:

```
.border {  
chegarasi: qattiq 1px qora;  
}
```

Jadvaldagi alohida hujayralar ichidagi qatorlarni qanday qo‘shish kerak?

Agar siz hujayra tarkibidagi satrlarni qo‘shishni xohlasangiz, buni eng oson yo‘li gorizontal qoidalar yorlig‘i bilan amalga oshiriladi.

Siz yuqoridagi CSSlardan butunlay qochishingiz va jadval yorlig‘ida chegara atributini ishlatishingiz mumkin. Ammo uning xususiyatiga ega emas, ammo eskirmasa-da, CSSdan ancha moslashuvchan emas, chunki siz chegarani kengligi bilan belgilashingiz mumkin va u

faqatgina jadvalning barcha hujayralari atrofida bo'lishi yoki hech kimga tegishli bo'lishi mumkin emas.

Nazorat savollari.

1. CSS stoli chegaralari?
2. CSS da chegara turlari?

10-MAVZU. PSEVDOSINFLAR

Reja:

1.Psevdosinf selektorlar

2. Psevdosinflarda kombinatsiya

Psevdosinf selektorlar. Psevdosinf selektorlarni ko'pchilik brauzerlar o'qiydi va shuning uchun amalda keng qo'llaniladi.

CSS 1 da havola (ssilka) lar bilan ishlash uchun to'rt xil Psevdosinf mavjud:

- link — oddiy havola (ssilka) ni belgilash uchun ishlatiladi;
- active — aktiv (ko'rib chiqilayotgan) havola (ssilka) ni belgilash;
- visited — ko'rib chiqib bo'lingan havola (ssilka) ni belgilash;
- hover-sichqoncha ko'rsatkichi bilan aktivlashtirilgan (ko'rib chiqilmoqchi bo'lgan) havola (ssilka) ni belgilash.

CSS yordamida havola (ssilka) larni tashqi ko'rinishini berish uchun quyidagicha qoidalar yozish mumkin:

```
A: link { ... oddiy (ko'rib chiqilmagan) havola (ssilka) ni bezash stili... }  
A:active { ... aktiv havola (ssilka) ni bezash stili ... }
```

```
A:visited { ... ko'rib chiqilgan havola (ssilka) ni bezash stili... }
```

va nihoyat HTML vositasi bilan bajarish mumkin bo'lmagan ish, sichqoncha ko'rsatkichi bilan aktivlashtirilgan havola (ssilka) ni berish.

```
A:hover { ... ko'rib chiqilmoqchi bo'lgan havola (ssilka) ni bezash stili... }
```

Sinfs selektori orqali havola (ssilka) yaratsak, u quyidagi ko'rinishda bo'ladi:

```
visited {color: red;} <a href="http://www.w3.org/"  
class="visited">W3C Web site</a> Xuddi shu ishni Psevdosinf  
selektori orqali ko'rsatamiz:
```

```
a:visited {color: red;}
```

Bu holda ko'rib chiqilgan har qanday havola (ssilka) qizil rangda bo'ladi, va unga sinfs atributini qo'shish shart emas .a va visited ni ajratuvchi ikki nuqta (:) aynan Psevdo-sinfdan foydalanilayotganini bildiradi. Psevdosinf va psevdoelementni hamma kalit so'zlari ikki nuqtadan boshlanadi.

Psevdo-sinflardan, oddiy sinf va identifikatorlar bilan birgalikda turli kombinatsiya tuzib ham foydalanish mumkin.

```
<A class="pagelink" href="page.html">... tekst...</A>  
A.pagelink:hover { ... stilni ifodalash ... }
```

Yoki identifikator bilan birgalikda foydalanish:

```
<A id="mylink" href="page.html">...tekst...</A> A#mylink:visited  
{...stilni ifodalash ...}
```

Ko‘pchilik web sahifalardagi stil quyidagi ko‘rinishda bo‘ladi:

a:link {color: navy;} a:visited {color:gray;}	Bu ikkita qoidada statik psevdosinfdan foydalanilgan
a:hover {color: red;} a:active {color:yellow;}	Bu ikkita qoidada dinamik psevdosinfdan foydalanilgan

Dinamik Psevdosinlar foydalanuvchining ta’sirida hujjatning tashqi ko‘rinishini o‘zgartirish imkonini beradi. CSS2.1. uchta dinamik Psevdosinfnani aniqlaydi:

7.2-jadval. Dinamik psevdosinlar turi.

Nomi	Tushuntirish
:focus	Joriy vaqtda diqqat-e’tiborni tortayotgan har qanday elementga qo‘llash mumkin (klaviaturadan axborot kiritish jarayoni)
:hover	Sichqoncha ko‘rsatkichi bilan ustiga ta’sir etilayotgan har qanday elementga qo‘llash mumkin
:active	Foydalanuvchi tomonidan aktivlashtirilgan har qanday elementga qo‘llash mumkin

Misol:

```
input:focus {background: silver; font-weight: bold;}
```

Bu misol formaga axborot kiritish jarayonida, axborot kiritilayotgan oynaning fon rangi va shriftni qalinligi o‘zgarishi effektini beradi:

Misol:

```
body *:hover {background: yellow;}
```

Bu qoidaga ko‘ra, body bo‘limida joylashgan har qanday elementlar, sichqoncha ko‘rsatkichini ustiga qoysak, sariq fonli rangni oladi. Agar hujjatning tiliga e’tibor bermoqchi bolsak, :lang() Psevdosinfiga murojaat qilamiz. Masalan fransuz tilidagi ixtiyoriy elementni og‘ma shriftida ko‘rsatish uchun quyidagicha qoida yozamiz:

*:lang(fr) {font-style: italic;}

CSS2.1 da Pseudosinflarni kombinatsiyasidan foydalanish mumkin. Masalan hemis tilida yozilgan hamda ko‘rib chiqilmagan va ko‘rib chiqilgan havola (ssilka) larni ifodasi quyidagi ko‘rinishda bo‘ladi:

```
a:link:hover:lang(de) {color: gray;} a:visited:hover:lang(de) {color: silver;}
```

Psevdoelement. HTML tilida xat boshining birinchi qatori yoki birinchi harfini ifodalash uchun element yo‘q. Demak bu ishni bajarish uchun bezash stilini qo‘llab bo‘lmaydi. Shuning uchun ham CSS da psevdoelement tushunchasi kiritilgan.

CSS2.1 da to‘rtta psevdoelement mavjud:

Birinchi harf — :first-letter Birinchi qator — :first-line Elementdan oldin — :before Elementdan keyin — :after

Sinfs yordamida katta harf yaratish usulini ko‘ramiz:

```
<P><SPAN class="firstletter">K</SPAN>Katta harf yaratish</P>
```

 stil qoidalarini yozamiz:

```
SPAN.firstletter { color: blue; font: bold large Verdana, sansserif }
```

 Bu holda har bir xat boshini birinchi harfini

```
<SPAN class= "firstletter">...</SPAN>
```

, teglar orasiga yozish kerak.

Bu ish noqulaylik yaratadi.

Pseudosinfdan foydalansak ish ancha soddalashadi, quyidagicha kod yozish yyetarli:

```
<P> Psevdoelement first-letter yordamida katta harf yaratamiz</P>
```

 stil yozish qoidasi:

```
P:first-letter{ color: blue; font: bold large Verdana, sansserif }
```

Agar matn birinchi harfdan ajralgan holda aks etsin desak, qo‘shimcha xususiyat flloat:left beramiz. U holda katta harf chap tomonga tekslanadi, Matn katta harfdan ajralib, uni atrofida joylashadi.

```
P:firstletter{ color: blue; font: bold 24pt Verdana. sansserif; float:left }
```

```
p:first-letter {color: red;} – xat boshining birinchi harfi qizil rangda aks
```

etadi.

```
h2:first-letter {font-size: 200%;} — sarlavhaning birinchi harfi ikki baravar
```

kattalashgan holda aks etadi.

Matnning birinchi qatorini, boshqa qatorlardan farqli tarzda ifodalash uchun quyidagicha qoida yozamiz:

P:first-line{ font-style: italic }

bu holda ixtiyoriy xat boshining birinchi qatoridagi matn ko‘rsiv shrift bilan aks etadi. p:first-line {color: purple;} – xat boshining birinchi qatori siyoh rangda aks

etadi. :first-letter va :first-line psevdoelementlarni faqatgina blok darajasidagi

elementlarga qo‘llash mumkin. Psevdoelementlarda qo‘llash mumkin bo‘lgan xususiyatlarni quyidagi jadvalda keltiramiz:

7.3-jadval. Psevdoelementlarda qo‘llash mumkin bo‘lgan xususiyatlar.

:first-letter	:first-line
Hamma xususiyatlar: font color background margin padding border text-decoration text- transform line-height	Hamma xususiyatlar: font color background word-spacing letter-spacing vertical-align text- decoration text-transform line- height

Elementdan oldin qo‘yiladigan maxsus stilni qo‘llash. Masalan har bir h2 elementning oldiga ikkita qizil kvadrat qavs qo‘yish kerak bo‘lsin:

```
h2:before {content: "[ ]"; color: red;}
```

Elementdan keyin qo‘yiladigan maxsus stilni qo‘llash. Masalan hujjatni yakunlashda quyidagicha qoidadan foydalanish mumkin:

```
body:after {content: " The End.";}
```

Psevd elementlardan foydalanishga misollar keltiramiz:

```
<html>
```

```
<head>
```

```
<meta http-equiv="content-type" content="text/html;  
charset=windows-1251">
```

```
<title>Harfning rangi</title>
```

```
<style type="text/CSS">
```

```
P:first-letter { color: red; /* Har bir xat boshining birinchi harfi qizil  
rangda
```

```
*/ } P:first-line{ font-style: italic } h2:first-letter{color:blue;  
font:30pt; font-family:Verdana. san-serif;}
```

```
</style>
```

```
</head>
```

```
<body>
```



```

<h2>Hikmatlar</h2>
<p>Abu Rayhon Beruniy hikmatlari</p>
<p>Har bir insonning qadr-qiymati o‘z ishini qoyil qilib
bajarishida.</p>
</body>
</html> 2 — misol
<head>
<meta http-equiv="content-type" content="text/html;
charset=windows-1251">
<title>sarlavha ostidagi fon rangi</title>
<style type="text/CSS"> H2 {
background: #D9FFAD; /* sarlavha ostidagi fon */ p:first-
letter{font-size:200% }
</style>
</head>
<body>
<h2>Beruniy hikmatlari</h2>
<p>Rozilik ko‘zi hech bir aybni ko‘rmaydi,</p>
<p>Lekin norozilik ko‘zi barcha ayblarni oshkor qiladi...</p>
</body>
</html>

```

Nazorat savollari

1. HTML hujjatga CSS stillar majmuasini ulash va uning usullari?
2. CSS stillar majmuasida selektorlar va ularning vazifasi?
3. Pseudosinf nima va uning qanday turlari mavjud?
4. Psevdoelement nima va uning qanday turlari mavjud?

11-MAVZU. FRONT PAGE DASTURI

Reja:

1. Front Page dasturi haqida umumiy tushunchalar
2. Front Page dasturning asosiy oynalari, menyusi, komponentlari.
3. Front Pagening ish rejimlari.
4. Master va shablonlar bilan ishlash va web-sahifa yaratish.

1. Microsoft Front Page dasturi haqida umumiy tushunchalar

Faraz qilamizki, siz biror bir tashkilotning mas'ul xodimisiz va sizning asosiy vazifangiz ushbu tashkilot haqida Internet tizimiga ma'lumot yetkazib berish hamda shu yo'l bilan tashkilotingizga keladigan mijozlarning sonini iloji boricha ko'paytirish. Demak, sizning oldingizda Internetda tashkilotingiz haqidagi qiziqarli ma'lumotlarni aks ettiradigan sahifa yoki boshqacha aytganda Web-sayt yaratish masalasi ko'ndalang bo'lib turibdi. Sizda esa bunday ishlarni qilish uchun bilim ham, tajriba ham yetarli emas. Lekin rahbariyat sizning oldingizga bunday sahifani tezda yaratish va uni Internetga joylashtirish kabi vazifani ko'ndalang qilib qo'ygan. Nima qilish kerak va qanday qilib tegishli Web-saytni qisqa muddatlarda ishga tushirish mumkin? Xuddi shunday holatlarda sizga Microsoft Front Page amaliy dasturi yordam beradi.

Bu masalani hal qilishdan avval World Wide Web (yoki Web) nima ekanligini bilib olishimiz kerak bo'ladi. World Wide Web Internet tizimi bilan ishlashni ancha osonlashtiradi va u Tim Berners Li tomonidan 1992-yilda Internet resurslarini ko'rib chiqishni osonlashtirish maqsadida yaratilgan edi. Internet resurslarini ko'rib chiqish uchun mo'ljallangan programmalar Web-brauzerlar deb ataladi. Shuni ham aytib o'tish kerakki, ko'pincha Web deganda World Wide Web tushuniladi, lekin Microsoft Front Page bilan ishlayotganda Web tushunchasi Web-sayt uchun ishlab chiqilgan Web-sahifalar to'plami sifatida tushuniladi.

World Wide Web hosil qilish texnologiyasi asosiga gipermatnlar tuzish asos qilib olingan. Gipermatn deganda bir-biri bilan bog'liq bo'lgan sahifalar majmuasi tushuniladi. Bunday tuzilishdagi sahifalar to'plamini uzatish, qabul qilish va ularni boshqarish uchun HTTP (Hyper Text Transfer Protocol- Giper matnlarni uzatish protokoli) tushunchasi ishlatiladi va World Wide Web ham xuddi shu protokol vositasida faoliyat ko'rsatadi. Agarda World Wide Web asosini HTTP

tashkil qilsa, Web sahifaning o‘zi HTML (Hyper Text Marko‘p Language — Gipermatnli hujjatlarni belgilaydigan programma tili) ga asoslangan. HTML tilida gipermatnli hujjatlarning tuzilishi va formatni aniqlash uchun TEG deb nomlangan buyruqlar majmuasi ishlatiladi. Web-brauzerlar ushbu buyruqlarni tegishli matnlarga yoki grafik tasvirlarga aylantiradilar hamda ularning kompyuter ekranidagi tasvirlarini hosil qiladilar.

Bundan tashqari HTML tili giperko‘rsatgichlar orqali Web-brauzerga boshqa sahifalarga o‘tishni hamda ularning ma’nosini ekranda ko‘rsatishni ham tashkillashtirib berishi mumkin. Microsoft Front Page dasturining asosiy oynasida HTML tugmachasi bosilgan holatda hosil bo‘ladigan quyidagi rasmda HTML tilining bir qancha buyruqlarini ko‘rib chiqishimiz mumkin:

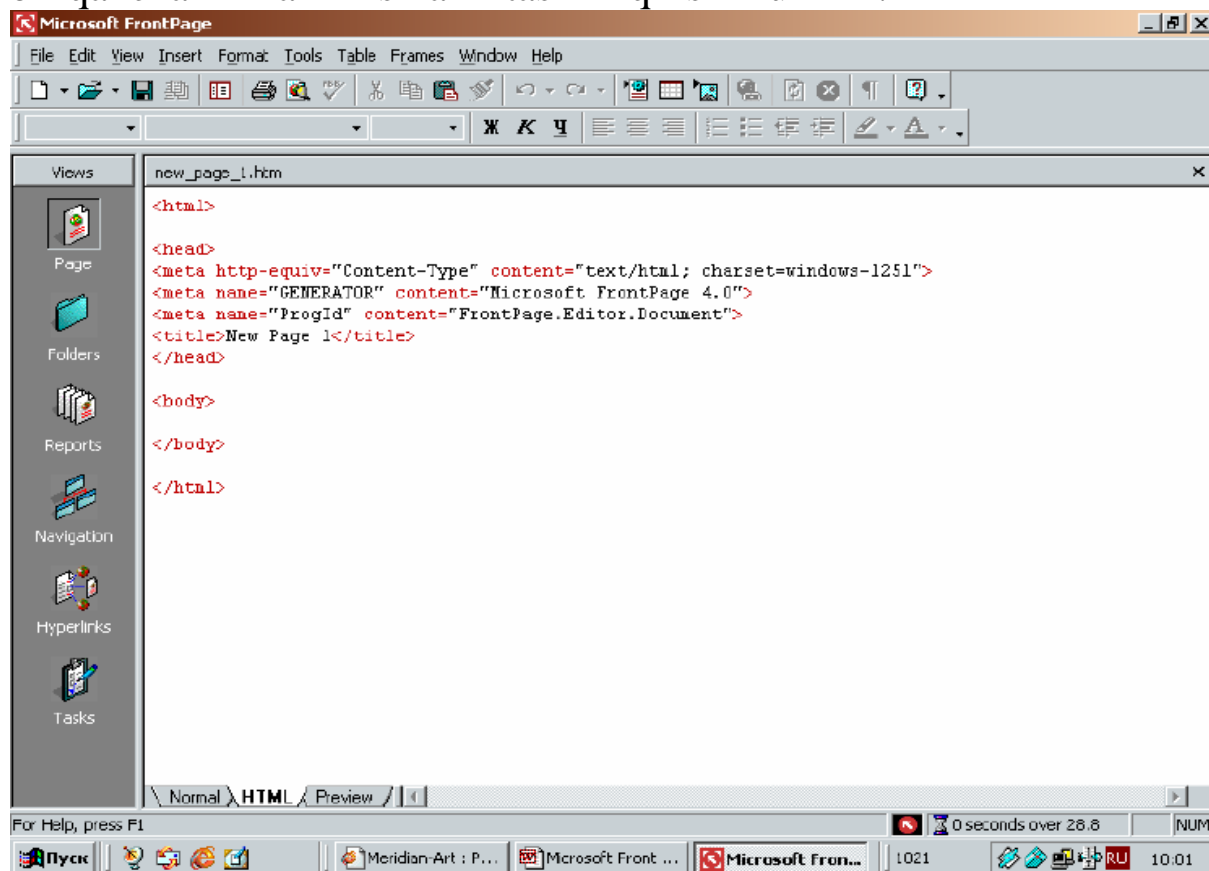
Bu yerda **<html>** buyrug‘i HTML — hujjatning boshlanishini bildiradi, **<head>** buyrug‘i esa Web-sahifa boshlanganini bildiradi, uning tugallanishi ham xuddi shu simvol bilan aniqlanadi. Body buyrug‘i gipermatnini o‘z ichiga qamrab oladi, **<body>**

</body> buyruqlari esa Web –sahifani aniqlab beradigan gipermatnini o‘z ichiga qamrab oladi. Hujjatning xuddi shu qismi brauzer tomonidan kompyuter ekranida ko‘rsatiladi.

<body> elementining boshlang‘ich va oxirgi teglari orasida Web –sahifaning dizaynini aniqlab beradigan barcha elementlarni joylashtirish mumkin.

Agarda fonning rangini o‘zgartirish kerak bo‘lsa, uning rangi quyidagi buyruq orqali beriladi **bgcolor="#FFD700"** – bu misolda sahifa foni rangi sifatida tillarang tanlanadi. Sahifada gorizontal chiziq chizish **<HR>** buyrug‘i orqali amalga oshiriladi. Ko‘rinib turibdiki, HTML tilining bir qancha buyruqlari bo‘lib, ular yordamida xohlangan turdagi sahifalarni osonlik bilan yaratish mumkin. HTML tilining yana bir qancha buyruqlari bo‘lib, ular xilma-xil ishlarni amalga oshirishga va Web-sahifani chiroyli ifodalashga yordam beradi. Hozirgi paytda bir qancha Web-sahifalarda skript deb nomlangan maxsus programmalar ishlatiladi. Ular Microsoft VBScript yoki JavaScript tillarida yozilishlari mumkin. Web-sahifalarning kodlariga joylashtirilgan bunday «mini-programmalar» orqali formatlashtirishni, multimediani ko‘rishni yoki tasvirlar chiqarishni osongina boshqarish mumkin. Hozirgi paytda skriptlarni boshqarishning eng kuchli vositasi sifatida DHTML (Dinamik

HTML) xizmat qiladi va u yordamida animatsiya va shunga o‘xshash bir qancha xilma-xil ishlarni tashkil qilish mumkin.



Ko‘rib to‘rganingizdek, HTML tilidan foydalangan holda Web-sahifa hosil qilish uchun ancha malaka, vaqt va mehnat talab qilinadi. Bu ishni osonlashtirish uchun Front Page amaliy programmasini ishlatish mumkin, chunki bu holda sizga HTML tilini o‘rganishga, skriptlardan foydalanishni bilishga yoki DHTML tilini o‘rganishga hojat qolmaydi.

Front Page amaliy programmasi yordamida Web-sahifaga joylashtirilishi kerak bo‘lgan matnni osonlik bilan terishingiz, rasmlarni joylashtirishingiz, ovoz effektlarini o‘rnatishingiz, skriptlar yoki DHTML tili programmalari yordamida amalga oshirilishi mumkin bo‘lgan xilma-xil effektlarni amalga oshirishingiz mumkin bo‘ladi. Demak, Front Page amaliy programmasi vositasida World Wide Web uchun mo‘ljallangan turli xil Web-sahifalar hosil qilishingiz va ularni Internet tarmog‘iga joylashtirishingiz mumkin. Endi Web-sahifa va Web-sayt degan atamalarning ma’nosiga va bir-biridan farqiga biroz to‘xtalib o‘tamiz. Web-sayt (yoki Web-tugun) bir-biri bilan ma’no va mohiyati jihatidan bog‘liq bo‘lgan Web-sahifa va fayllar to‘plamidir.

Ko‘pincha Web-saytlar biror bir mavzuga bag‘ishlangan bo‘lib, firma, kompaniya, tashkilot yoki biror bir insonga tegishli ma‘lumotlarni o‘z ichiga qamrab oladi. Front Page programmasida masterlar deb nomlangan dasturlar bo‘lib, ular Web-saytlarni va Web-sahifalar ko‘rinishidagi shablonlarni qadamba-qadam tuzishga hamda sozlashtirishga imkon beradilar. Bu xildagi masterlar va shablonlar xilma-xil turdagi Web-saytlarni hosil qilish uchun juda qulay.

Har bir Web-saytda asosiy yoki uy sahifasi (homepage) deb atalgan bir Web-sahifa bo‘ladi va ushbu saytga kiruvchi barcha insonlar eng oldin xuddi shu sahifaga kiradilar. Keyinchalik ushbu sahifada joylashgan giperko‘rsatgichlar va ko‘rsatmalardan foydalanib, ular bu saytning boshqa sahifalariga yoki butunlay boshqa saytlarning sahifalariga ham kirishlari mumkin. Giperko‘rsatgichlar orqali aniqlanadigan Web- sahifalar ushbu kompyuterda yoki Internet bilan ulangan istalgan kompyuterda bo‘lishi mumkin.

Gipermatn Web-sahifada joylashgan biror bir matn bo‘lagi yoki rasm bo‘lib, agar u turtilsa, boshqa Web-sahifa ga yoki saytga o‘tish amalga oshiriladi.

Uy sahifasi (homepage) eng yuqorida joylashgan bo‘lib, undagi giperko‘rsatgichlar istalgan Web-sahifaga o‘tishni amalga oshirib beradi. O‘z navbatida ushbu Web- sahifalardan boshqa sahifalarga o‘tish ham xuddi shunday usuldan foydalangan holda osonlik bilan amalga oshirilishi mumkin. Bunday sahifalarning har birida boshqa sahifalarga va saytlarga bo‘lgan murojaatlardan tashqari orqaga, bundan oldingi sahifaga yoki uy sahifasi (homepage) ga qaytishga imkon beradigan giperko‘rsatgichlar ham bo‘ladi.

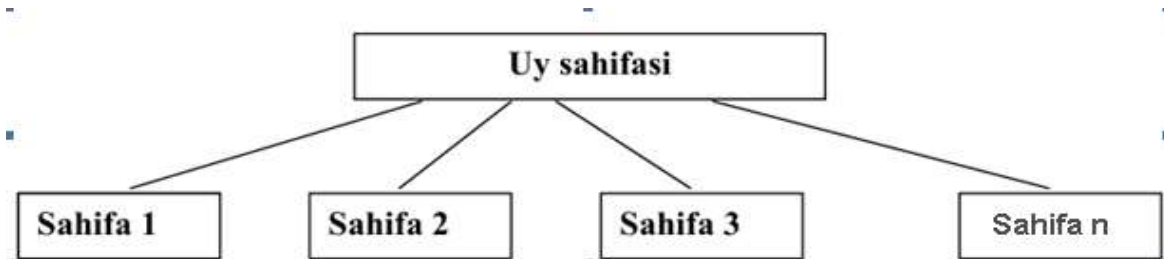
Front Page programmasi yordamida osonlik bilan Internetda yoki intranetda joylashgan Web-saytlarni hosil qilish mumkin. Bunday Web-saytlar saytga kiruvchilar uchun uning sahifalarini ko‘rishga imkon beruvchi kompyuterda ya‘ni Web-serverda (Web-server) joylashadi. Lekin Web-saytni Front Page programmasi yordamida o‘z kompyuteriingizda ham tashkil qilishingiz va u tayyor bo‘lgandan so‘ng, uni Web- serverda joylashtirishingiz (nashr qilishingiz) mumkin.

Saytni joylashtirish uchun u joylashtiriladigan Web-serverni tanlash kerak bo‘ladi. Ko‘pchilik Internet-provayderlar (Internet Service Provider – ISP) Web-saytlar ni joylashtirish bo‘yicha xizmat ko‘rsatadilar (Web-xosting -Web-hosting). Bunda provayder Web-

fayllarni saqlash uchun Web-serverda diskdan ma'lum miqdorda joy ajratadi va tegishli xizmatni ko'rsatadi. Kichik va oddiy Web-saytlar uchun bu albatta yetarli, lekin katta va murakkab Web-saytlar uchun alohida server (dedicated server) ajratgan maqsadga muvofiq, chunki bunda katta hajmdagi disk hajmini ishlatish mumkin bo'ladi. Bu esa ko'proq miqdordagi ma'lumotlarni joylashtirish va tezroq ishlaydigan qurilmalarni ishlatish imkonini beradi. Lekin saytni joylashtirishning qanday usulini tanlaganingizga qaramay bir muhim savolga javob topish kerak bo'ladi: Server Front Page Server Extentionsni ishlatish imkoniyatiga egami yoki yo'qmi? Chunki Front Page programmasining imkoniyatlarini amalga oshiruvchi dasturlar xuddi shu Front Page Server Extentions dasturiga mansub bo'ladi. Hosil qilinadigan Web-saytni tuzishdan avval u nimalardan iborat bo'lishligini aniqlash masalasini hal qilish uni rejalashtirish deb ataladi. Buning uchun avvalo bu Web-sayt nima uchun va qanday maqsadda tashkil qilinayotganini aniqlab olinadi. Masalan:

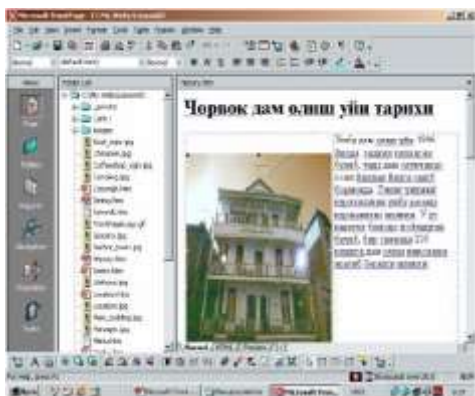
- Tashkilot ishlab chiqarayotgan mahsulotlar yoki xizmatlar bilan ko'pchilikni tanishtirish uchun;
- Mahsulot yoki xizmatlarni sotish uchun mijozlar topish uchun;
- Mahsulot yoki xizmatlarni bevosita sotish uchun;
- Yangi bozorlarni o'zlashtirish uchun;
- Eski bozorlarni kengaytirish uchun;
- Masofaviy ta'lim tizimini tashkil qilish uchun;
- Elektron kommertiya faoliyatini tashkil qilish uchun va xakozolar.

Har qanday sayt uy sahifasidan boshlanadi va unda sayt bilan bog'liq asosiy ma'lumotlar saqlanilishi kerak yoki ularni olishga imkon beradigan giperilovalar bo'lishi lozim. Uy sahifasi esa sayt mavzusiga tegishli bir qancha boshqa sahifalar bilan bog'liq bo'lib, ularga o'tishni amalga oshirib berishi lozim. Buni quyidagi sxema shaklida keltirishimiz mumkin:



O‘z navbatida sahifalar boshqa quyi darajadagi sahifalar hamda boshqa saytlarning giperko‘rsatgichlari bilan bog‘langan ham bo‘lishi mumkin. Yuqorida aytib o‘tganimizdek, bu sahifalardan boshlang‘ich uy sahifasiga orqaga) va uy sahifasidan istalgan sahifaga o‘tish imkonini yaratib berish juda ham muxim. Bu ishni amalga oshirishni sahifalararo navigatsiyani amalga oshirish deb tushuniladi. Bunday sahifalar ro‘yxatini oldindan aniqlab olish juda muhim va foydali bo‘ladi. Shundan so‘ng, yaratilayotgan sayt Web-sahifalarining bir-biri bilan qanday tartibda bog‘liq bo‘lishini aniqlab olish kerak bo‘ladi. Bu ko‘pincha giperilovalar orqali amalga oshiriladi. Ammo Front Page programmasida bunday giperilova (yoki sahifalararo bog‘lanishlarni) boshqa usullarda ham amalga oshirish mumkin. Masalan, barcha giperilovalarni uy sahifasida joylashtirish mumkin yoki navigatsiya paneli (navigation bar) tashkil qilib, u orqali sahifalararo bog‘lanishlarni boshqarishi mumkin. Bundan tashqari ekrandagi faol (aktiv) bo‘laklar tasvirlari (hotspots) ko‘rinishidagi interaktiv karta (image map) hosil qilgan holda, boshqa sahifalarga ham o‘tish mumkin. Bu usulda kompyuter ekranidagi rasmning o‘sha bo‘laklarni sichqoncha bilan tanlansa, kerakli sahifaga o‘tish amalga oshiriladi. Ba‘zi paytlarda freym ko‘rinishidagi menyu yordamida kerakli sahifalarga o‘tishni amalga oshirish ham mumkin. Freym ekranning shunday bir qismiki, uning tarkibiy qismlari uy sahifasiga bog‘liq bo‘lmagan holda ekranda ko‘rinib turadi. Masalan, siz bir sahifadan ikkinchisiga menyu punktlarini tanlagan holda o‘tganingizda, sahifalar ma‘nosi o‘ng freymda ko‘rinadi, giperilovali matnlar ko‘rinishidagi chap freymning tarkibi esa o‘zgarmasdan qolaveradi.

Front Page programmasi orqali nafaqat Web-sayt va Web-sahifalar tuzish, balki ularni maqsadga muvofiq ravishda boshqarish ham mumkin. Bu dasturning asosiy oynasi quyida ko‘rsatilganidek uch tarkibiy qismga bo‘lingan bo‘lib, ular Web-saytni xilma-xil usulda boshqarishga yordam beradi.



Birinchi qismning nomi Views bar (ko‘rish rejimlari paneli) deb ataladi va u Web- saytni ko‘rish rejimlarini o‘zgartirishga yordam beradi. Ular bilan quyida tanishishingiz mumkin:

Page (Sahifa) – Web-sahifa larni ko‘rish va taxrirlashga imkon beradi.

Folders (Papka) - Joriy Web-saytlar ning papkalari ro‘yxati va fayllarini ko‘rishga imkon beradi.

Reports (Hisobot) – Joriy Web-sayt ning holati haqida jadval shaklidagi hisobotni ko‘rsatadi, shu jumladan, ushbu rejimda noto‘g‘ri ilovalar va sekin ishlaydigan sahifalar sonini ko‘rib chiqish mumkin.

Navigation (Navigatsiya) – Joriy Web-sayt ning daraxtsimon tuzilishini ko‘rish imkonini yaratadi. Folder list (Papkalar ro‘yxati) ro‘yxatida ushbu saytning papka va fayllari ko‘rinadi.

Tasks (Masalalar) - Joriy Web-sayt bilan ishlagan davrda bajarilishi lozim bo‘lgan masalalar ro‘yxatini aks ettiradi.

Web-sayt yaratish jarayonida masterlar ishlatilgan taqdirda, masalalar ro‘yxati to‘ldirilishi mumkin yoki siz o‘zingiz ham o‘z masalalaringizni bu yerga qo‘shib qo‘yishingiz mumkin.

Hyperlinks (Giperilovalar) - Joriy Web-saytga va undan bo‘lgan giperilovalarning o‘zaro bog‘lanishining grafik tuzilishini sxema tarzida ko‘rsatadi.

Agarda birorta Web-sahifani ko‘rish kerak bo‘lsa, uni sichqoncha ko‘rsatgichi orqali turtish zarur bo‘ladi va o‘ng tomondagi darchada yoki maxsus oynada sahifaning mazmuni paydo bo‘ladi:



Ekraning pastki qismida joylashgan holatlar paneli (**Status bar**) da ushbu sahifani uch xil holatda ko‘rish mumkin bo‘ladi. Ularning biri **Normal** deb atalib, unda Web- sahifani tahrirlash mumkin, ikkinchisida esa sahifani **HTML kodlari** ko‘rinishida ko‘rishimiz va **Preview** deb atalgan uchinchisida esa sahifa Internetda qanday ko‘rinishda bo‘lishini ko‘rishimiz mumkin. Papkalar ro‘yxatida joriy Web-saytga tegishli barcha fayllar va papkalarni, joriy sahifani esa sahifalarni tasvirlovchi oynasida ko‘rishimiz mumkin.

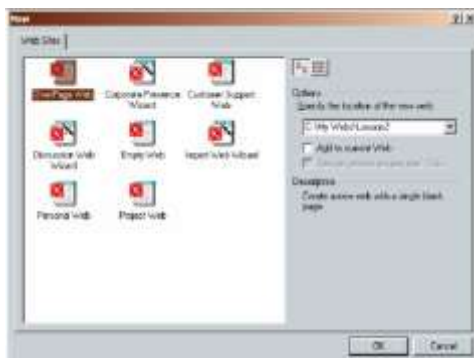
2. Master va shablonlar bilan ishlash va web-sahifa yaratish.

O‘rganilayotgan dastur yordamida Web-saytlarni uch xil usuldan foydalangan holda yaratish mumkin:

- Maxsus programmalar – Masterlar (Wizard) yordamida;
- Shablonlar yordamida;
- Oldindan tayyorlab qo‘yilgan fayllar yordamida.

Ularning qaysi birini tanlab olish shart-sharoitga bog‘liq ravishda va yaratilayotgan Web-saytning qanday bo‘lishiga qo‘yilgan talablar asosida aniqlaniladi. Quyida biz masterlar yordamida Web-sayt tashkil qilishni ko‘rib chiqamiz. Masterlar bir necha xil bo‘lib, ular xilma-xil turga mansub oddiy yoki murakkab Web-saytlarni maxsus dialog oynachalari orqali tashkil qilishga imkon beradi. Xar bir dialog oynasidagi savollarga aniq va to‘g‘ri javob berilsa, foydalanuvchi talablariga javob beradigan Web-saytlarni osonlik bilan yaratish mumkin bo‘ladi. Bu usulni ko‘rib chiqish uchun Windows ning asosiy oynasidagi Start (Pusk) tugmachasi bosilib, undan Programs (Programmi) punkti tanlab olinadi va undan Microsoft Front Page belgisi turtuladi. Ishga tushgan Front Page dasturi oynasidan File (Fayl) buyrug‘ining New (Cozdat) imkoniyati tanlansa, unda Web buyrug‘i ko‘rinadi. Agarda u tanlangan taqdirda ekranda New (Cozdat) dialog oynasi hosil bo‘ladi. Bu oynadan Web-saytlarning qanday turga mansub bo‘lishini tanlab olish mumkin. Ya’ni, undan

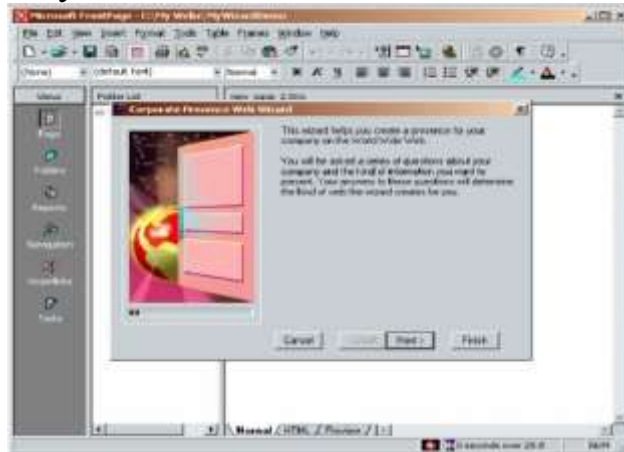
foydalanib, Web-sayt hosil qilish uchun ishlatilishi mumkin bo'lgan bir qancha masterlarni ishga tushirish mumkin. Ularning biri Master (Wizard) Discussion Web bo'lib, uning yordamida muhokamalar tashkil qilishga mo'ljallangan Web-saytlar tashkil qilish mumkin. Bunday saytning foydalanuvchilari boshqalarning ma'lumotlarini ko'rishlari, ularga javoblar yuborishlari yoki o'z ma'lumotlarini saytda joylashtirishlari mumkin. Ikkinchisi Master (Wizard) Import Web deb atalib, kompyuteringizdagi yoki Web — serverdagi fayllar asosida yangi Web-saytlar qilish imkonini beradi. Saytni yaratib bo'lgandan so'ng Web – sahifalarni yoki sayt tuzilishini o'z ehtiyojingga mos ravishda o'zgartirishingiz yoki to'ldirishingiz mumkin.



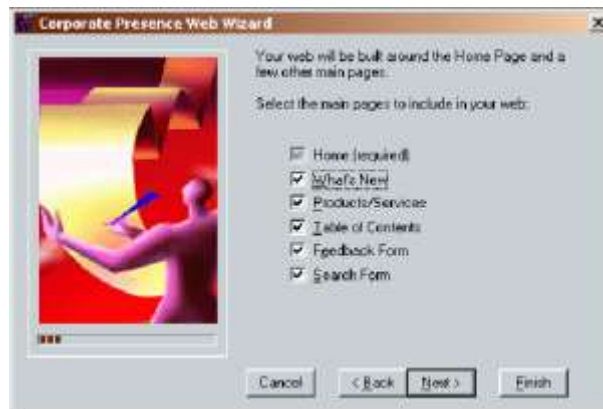
Ushbu master oldindan mavjud bo'lgan Web-saytlaringizni yangilash va tekshirish uchun juda qo'l kelishi mumkin. Keyingi master Master (Wizard) Corporate Presence deb nomlangan va u tashkilotlarning yoki kompaniyalarning Web-saytlarini yaratishga imkon beradi. Yuqoridagi oynadan boshqa turdagi masterlarning bajaradigan ishlarini ham ko'rib chiqishingiz mumkin. Quyida Corporate Presence Wizard deb nomlangan master yordamida Web-sayt hosil qilishni ko'rib chiqamiz. Bunday sayt uy sahifasini, mundarijalar jadvalini, yangiliklar sahifasini, mahsulotlar va xizmatlarni o'z ichiga olgan direktoriyni, har bir mahsulot yoki xizmat turiga bag'ishlangan Web – sahifalarni, foydalanuvchilar bilan muloqotni tashkil qilishga mo'ljallangan sahifani hamda tashkilot saytida ma'lumotlar axtarishga imkon beradigan Web – sahifani o'z ichiga qamrab oladi.

Endigi ishimiz yuqoridagi oynadan Corporate Presence Wizard nomli belgini sichqoncha ko'rsatgichi bilan bir marta turtish (belgilash) bo'ladi. Natijada Front Page dasturi sayt tashkil qilish uchun Corporate Presence masterini ishlatish kerakligi haqida

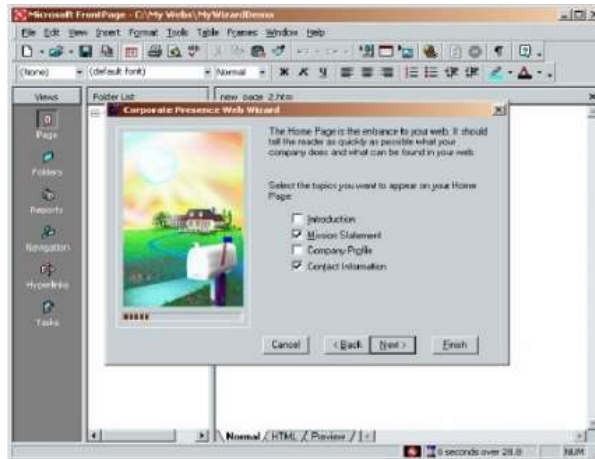
topshiriq oladi. Bu buyruq yordamida Front Page dasturi xuddi shu nomli papka hosil qiladi va barcha Web fayllarni xuddi shu papkaga joylashtiradi. Agarda siz boshqa disk bilan ishlayotgan bo‘lsangiz, yuqoridagi S harfini boshqasiga almashtirishingiz kifoya. Endi OK tugmachasini turing. Bunda Corporate Presence masterining quyida ko‘rsatilgan birinchi oynasi ochiladi:



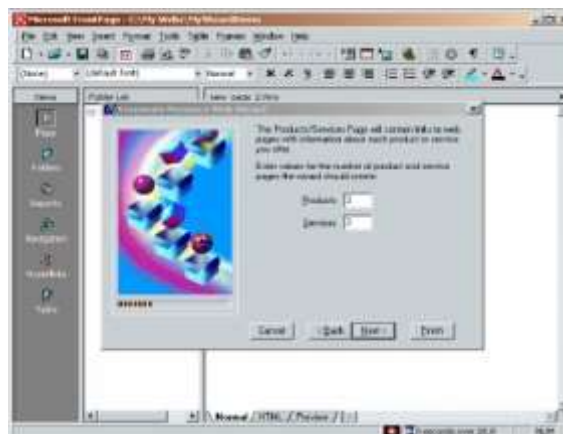
Ushbu oyna masterning nima ekanligini va qanday ishlar bajarishini tushuntiradi. Endi bu oynadagi Next (Dalee) tugmasi bosilsa, Web – sahifaga joylashtirilishi lozim bo‘lgan sahifalar ro‘yxati chiqadi va siz undan keraklilarini tanlab olishingiz mumkin:



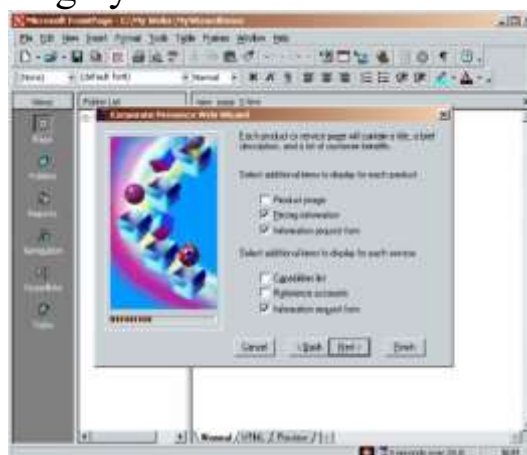
Sizning nimalarni tanlaganingizga bog‘liq ravishda Master yaratilayotgan saytda xilma-xil dialog oynalarini aks ettiradi. What’s New (Yangiliklar) va Search Form (Qidiruv formasi) maydonlariga sichqoncha ko‘rsatgichini keltirib, uning chap tugmasi bosilsa, darchalardagi tegishli belgilar yo‘qoladi. Demak, bu ishni qilsangiz, sizning saytingizda xuddi shu bo‘limlar bo‘lmaydi. Tanlov nihoyasiga etgandan so‘ng, Next (Dalee) tugmachasi turtilsa, ekranda sizning saytingizda aks ettirilishi lozim bo‘lgan ma‘lumotlarni tanlash imkoniyatini yaratadigan quyidagi oyna tasviri hosil bo‘ladi:



Ushbu oynadagi Introduction (Kirish) maydoniga belgi qo‘yilsa, saytda yana bir bo‘lim – Kirish hosil qilinadi. Yana Next (Dalee) tugmachasi turtilsa, ekranda mahsulot va xizmatlar (Products – mahsulotlar, Services – xizmatlar) uchun qancha miqdordagi sahifalar ajratilishi kerakligini aniqlashga imkon beradigan dialog oynasi hosil bo‘ladi:



Ularning kerakli bo‘lgan sonini kiritgandan so‘ng, yana Next (Dalee) tugmachasi turtilsa, ekranda yaratilayotgan saytda mahsulot va xizmatlar haqida qanday ma‘lumotlar bo‘lishini aniqlashga imkon beradigan quyidagi dialog oyna hosil bo‘ladi:



Ularning ichidan keraklilarini belgilab bo'lgandan so'ng yana bir marta Next (Dalee) tugmachasi turtulsa, ekranda Web – saytga kiruvchilar haqida qanday ma'lumotlar olishingiz kerakligini aniqlashga imkon beruvchi dialog oynasi hosil bo'ladi:



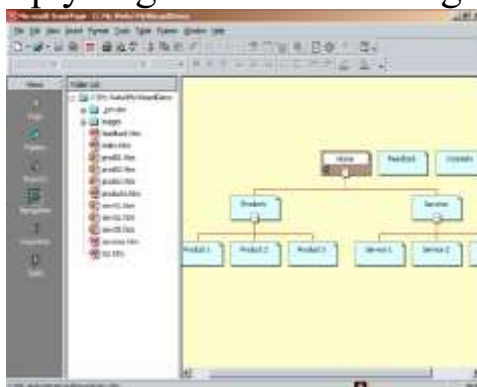
Ushbu ma'lumotlarni beruvchi sahifaning nomi Feedback (Teskari aloqa) deb ataladi va uning qanday ko'rinishda bo'lishi keyingi dialog oynalari orqali aniqlanadi. Keyingi qadamlarda Web – sayt mundariyasi qanday ko'rinishda bo'lishi, sahifalar dizayni o'z ichiga nimalarni olishi, tugallanmagan sahifalar qanday simvol orqali belgilanishini tanlab olinadi. Bu ishlarni bajarish oqibatida ekranda quyidagi dialog oynasi hosil bo'ladi:



Ushbu oynadan foydalanib, tashkilot to'liq nomi, uning qisqa nomi va manzili yaratilayotgan Web – saytga kiritiladi. Yana bir marta Next (Dalee) tugmachasini bosish qo'shimcha ma'lumotlar kiritilishini so'raydi. Uni kiritib bo'lgandan so'ng sahifalarda ishlatiladigan ranglar, grafikalar va shriftlar haqida ma'lumot so'raladi. Bu savollarning barchasiga javob berib bo'linganidan so'ng, oxirgi oynada Finish (Gotovo, Tayyor) tugmasi turtuladi. Biroz vaqtdan so'ng, sizning saytingizning boshlang'ich versiyasi tayyor bo'ladi. Uning uy sahifasi (Home Page) quyidagi ko'rinishga ega bo'ladi:



Uning tuzilishini ko'rish uchun chap tomondagi Navigation (Navigatsiya) belgisini turtish kifoya. Bunda sahifalar orasidagi o'zaro aloqani ko'rsatadigan quyidagi sxema shaklidagi tasvir hosil bo'ladi:



Hosil bo'lgan sxemadagi to'rtburchaklar ko'rinishida ifodalangan istalgan sahifani ko'rish uchun esa undagi tegishli belgini sichqoncha ko'rsatgichi bilan ikki marta turtildi, masalan uy sahifasini ko'rish uchun Home Page belgisini ikki marta turtish kerak bo'ladi. Bu ishni bir qancha sahifalar uchun bajarib ko'rishingizni tavsiya qilamiz, chunki bunda yaratilayotgan sahifalarning qanday bo'layotganini ko'rib va o'rganib chiqishingiz mumkin. Bunda har bir sahifani o'zgartirish va unga xilma-xil qo'shimcha ma'lumotlar qo'shish imkoniyati mavjud. Bu ishlar bajarilgandan so'ng, sayt tayyor bo'ldi desak ham bo'laveradi. Agar keyinchalik saytga hamda uning istalgan sahifasiga o'zgarishlar kiritish yoki uni rivojlantirish va mukammallashtirish kerak bo'lsa, ular ko'rsatilgan tartibda amalga oshiriladi. Barcha kerakli o'zgarishlarni amalga oshirib hamda tegishli ma'lumotlarni kiritib bo'lgandan so'ng, ushbu saytni saqlab qo'yish kerak bo'ladi. Buning uchun File (Fayl) menyusidagi Close Web (Web-sayt yopilsin) buyrug'i tanlab olinadi. Ushbu buyruq bajarilishi natijasida hosil qilingan saytning yopilishi amalga oshadi. Web-saytlarga nom berganda so'zlar orasida probellar (bo'sh joylar) bo'lmasligiga e'tibor berish kerak, chunki ko'pchilik Web-serverlar fayllar nomidagi probellarga tushunmaydilar. Agar Web-fayllarning

nomlari bir necha soʻzlardan iborat boʻlsa, u holda bu soʻzlarni bir-biridan ajratish uchun tagiga chizish «_» (podcherkivanie) belgisidan foydalanish mumkin. Agarda yaratgan Web-saytingizga biror bir yangi sahifa qoʻshish kerak boʻlib qolsa, u holda bu sahifalarni mustaqil ravishda Web-sahifalar shablonlaridan foydalangan holda yaratishingiz mumkin. Shablon yordamida sahifalar hosil qilish uchun programmaning asosiy oynasidagi Page (Sahifa) tugmasini belgilaysiz va soʻngra File (Fayl) menyusidan New (Sozdat) punktini tanlab, undan Page (Stranitsa) buyrugʻini tanlaymiz. Buning natijasida New (Sozdat) dialog darchasi hosil boʻladi va undan foydalangan holda bizga kerak boʻlgan sahifa shablonini tanlab olishimiz mumkin:



Shablonlar yordamida Web-sayt tashkil qilish. Maʼlumot kiritishni soʻrab, ular asosida saytlar tashkil qiladigan masterlardan farqli ravishda shablonlar tayyor Web-saytlar asosida ishlashga imkon yaratadi. Bunda oldindan tayyorlab qoʻyilgan Web-saytlarni tanlagan holda, ularni oʻz talablaringizga moslashtirishingiz yoki oʻzgartirishingiz mumkin. Front Page programmasi quyidagi shablonlar bilan ishlashga imkon yaratadi:

Personal Web (Shaxsiy sayt) – bunday sayt oʻz ichiga shaxsiy maʼlumotlarni qamrab oladi. Unda fotoalbom, muallif qiziqishlarini koʻrsatadigan sahifa, boshqa saytlarga boʻlgan murojaatlar boʻlishi mumkin.

Project Web (Loyiha sayti) – bir loyiha ustida ishlayotgan bir qancha mutaxassislar orasida maʼlumot almashinishni taʼminlab beradigan sayt boʻlib, u guruh aʼzolari roʻyxati sahifasini, ish jadvali sahifasini, loyiha haqidagi maʼlumotlarni aks ettiradigan sahifani, maʼlumot almashinish darchasini hamda saytda maʼlumot axtarishni taʼminlab beradigan sahifani oʻz ichiga oladi.

Customer Support Web (Iste'molchilarga xizmat sayti) – ushbu sayt orqali tashkilot iste'molchilarning savollariga javoblar berishi va ular bilan muloqotni amalga oshirishi mumkin. Bunday sayt ma'lumotlar almashinishga imkon beradigan sahifani, tez-tez berilib turadigan savollarga javoblar sahifasini (FAQ – Frequently Asked Questions), saytga kiruvchilar o'z takliflarini bera oladigan forma sahifasini o'z ichiga olib, saytdan ma'lumotlar va programmaviy ta'minot olish imkoniyatini ham yaratib beradi.

One Page Web (Bir sahifali sayt) – faqat birgina sahifadan iborat sayt tuzishga imkon beradi.

Empty Web (Bo'sh sayt) – Mustaqil ravishda tashkil qilinadigan Web-sayt bo'lib, o'z tarkibigi hech qanday tayyor Web-sahifalarni olmaydi.

Endi **Customer Support** shabloni yordamida Web-sayt yaratishni ko'rib chiqamiz. Buning uchun File (Fayl) menyusidan New (Sozdat) punktini tanlab, undan Web punktini tanlaymiz. Hosil bo'lgan oynadan Customer Support Web (Iste'molchilarga xizmat sayti) belgisi tanlanib turtilsa, firma mahsulotlaridan foydalanuvchi iste'molchilarining muammolarini hal qilish uchun mo'ljallangan Web-sayt yaratish uchun mo'ljallangan shablon hosil bo'ladi. Lekin belgini turtishdan oldin Specify the Locations Of the New File (Yangi Web-saytning joylashishi) deb nomlangan maydonda yaratilmoqchi bo'lingan saytning nomini yozish kerak bo'ladi.

Agarda Folder List (Papkalar ro'yxati) deb nomlangan panelda **index.htm** fayli turtilsa, yaratiladigan Web-saytning uy sahifasi ekranda namoyon bo'ladi:



Web-sayt yaratish uchun foydalanuvchiga bir qancha savollar beradigan Masterdan farqli ravishda Shablon biror-bir turdagi Web-sayt hosil qilish uchun mo'ljallanilgan Loyiha deb tushunilishi kerak. Uning, ya'ni loyihaning yordamida Web-saytni bir marta yaratgandan

so‘ng, uni iste‘molchi talabiga muvofiq ravishda xohlagancha o‘zgartirish yoki moslashtirish mumkin bo‘ladi.

Navbatdagi ishimiz Web-sahifadagi ma‘lumotlarni o‘zgartirish bo‘lgani uchun

«Company name» (Kompaniya nomi) matnini o‘z firmamiz nomiga o‘zgartiramiz. Shundan so‘ng kiritilgan o‘zgarishlarni saqlab qolish uchun Save (Caqlansin) tugmachasini bosamiz va buning natijasida Front Page programmasi Web-sahifadagi o‘zgarishlarni saqlab qoladi. Agarda papkalar ro‘yxatidagi **bugrep.htm** faylini ikki marta turtsak, iste‘molchilarga firma mahsulotlari va xizmatlarini ishlatishda paydo bo‘lgan muammo va qiyinchiliklar haqida ma‘lumot berish uchun mo‘ljallanilgan Web-sahifa hosil bo‘ladi:

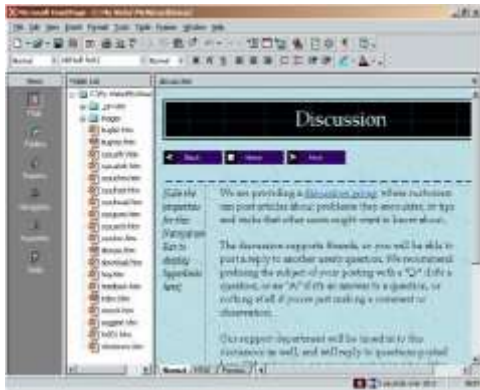


Agarda Folder List papkalar ro‘yxatidagi discuss.htm faylini ikki marta turtsak, foydalanuvchilarga muhokama oynasidan ma‘lumotlarni o‘qish va ularga javob berish imkoniyati yaratiladigan Web-sahifa tasviri hosil bo‘ladi. Hosil qilingan Web-saytni saqlash va yopib qo‘yish uchun yana File (Fayl) menyusidan Close Web (Web-sayt yopilsin) buyrug‘ini tanlaymiz. Bu holda hosil qilingan saytning yopilishi amalga oshadi. Shunday qilib, Web-sayt yaratish jarayonida masterlar ishlatildimi yoki shablonlardan foydalanildimi, bundan qat‘iy nazar, Web-sayt yaratishning ikki asosiy usuli mavjud:

- Sayt hosil qilingandan so‘ng sahifa va boshqa kerakli fayllarni hosil qilish;
- Yangi saytni mavjud fayllar asosida tashkil qilish.

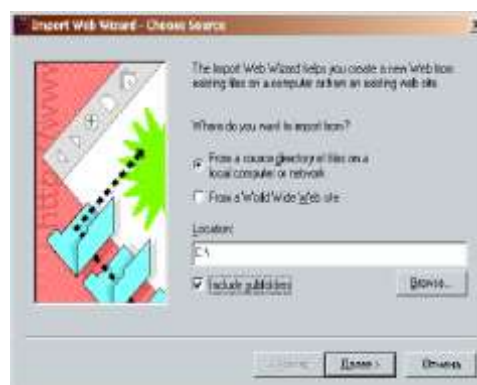
Agar sizda yaratilishi kerak bo‘lgan Web-saytning oldingi versiyasi kompyuteringizda bo‘lsa yoki u boshqa biror bir tashkilotning kompyuterida mavjud bo‘lsa va siz Web-saytning yangilangan versiyasini yaratishni istasangiz, ikkinchi usul juda ham qo‘l keladi. Agarda siz o‘zingiz uchun ma‘qul real sayt yaratish uchun

bo'sh sayt hosil qilish kerak bo'lsa, u holda File (Fayl) menyusidan New (Sozdat) punktining Web buyrug'i tanlanishi natijasida ishga to'shadigan Empty Web (Bo'sh sayt) belgichasini turtishingiz kerak bo'ladi.

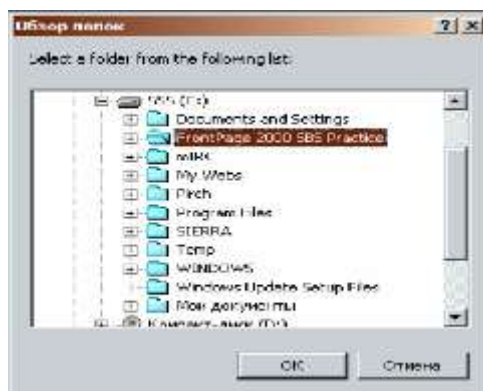


Agarda Sizga biror bir saytni nusxalashtirish kerak bo'lsa, u holda quyidagi ishlarni bajarishingiz lozim bo'ladi:

1. File (Fayl) menyusidan New (Sozdat) punktining Web buyrug'i tanlanadi va buning natijasida New (Sozdat) dialog oynasi hosil bo'ladi;
2. Ushbu oynadagi Import Web Wizard (Web-saytlarni import qilish) deb nomlangan tugmacha turtiladi va undagi Specify the Locations Of the New File (Yangi Web-saytning joylashishi) deb nomlangan maydonga yangi Web-saytning qayerda joylashishi kerakligi ko'rsatiladi va OK tugmachasi bosiladi. Natijada ekranda Import Web Wizard (Web-sayt importi masteri) ning birinchi dialog oynasi hosil bo'ladi:



Undagi From A Source of Files (Direktoriydan) va Include Subfolders (Ichidagi papkalar bilan birga) maydonlariga sichqoncha bilan belgilar qo'ying. So'ngra Browse (Ko'rib chiqish) tugmachasini bosing. Bunda ekranda quyidagi tasvirni ko'rasiz:



Navbatdagi ishingiz bu oynada nusxalashtirilayotgan Web-saytga tegishli papkani tanlab olish va OK tugmachasini bosish bo‘ladi. Bunda quyida ko‘rsatilgan dialog oynasi hosil bo‘ladi va siz undagi Finish (Tayyor) tugmasini bosishingiz kerak bo‘ladi:



Yuqorida bajarilgan ishlar natijasida tanlangan Web-sayt ko‘rsatilgan joyga to‘la nusxalashtiriladi va siz uni xohlagancha o‘zgartirishingiz yoki to‘ldirishingiz mumkin bo‘ladi.

Nazorat savollar:

1. Front Page ahamiyati qanday?
2. Front Pagening asosiy vazifalari nimalardan iborat?
3. Front Page ish rejimi qanday?
4. Tasks ishchi rejimining asosiy jadvali nimalardan iborat?

12-MAVZU. BOOTSTRAP

Reja:

- 1. Bootstrap haqida tushuncha**
- 2. Bootstrapning afzalliklari:**
- 3. Bootstrapning kamchiliklari:**

Bootstrap — bu sezgir, birinchi mobil veb-saytlar va veb-ilovalarni loyihalash uchun bepul va ochiq manbali old dastur. 2010-yilda Twitter tomonidan ishlab chiqilgan Bootstrap HTML, CSS va JavaScript-ning ustiga qurilgan bo‘lib, u sezgir, mobil qurilmalarga mos veb-saytlarni yaratishni osonlashtirish uchun mo‘ljallangan.

Bootstrap-dan foydalanishning asosiy afzalliklaridan biri uning sezgir tarmoq tizimidir. Bu tizim ishlab chiquvchilarga foydalanuvchi ekrani o‘lchamiga avtomatik moslashadigan maketlarni yaratish imkonini beradi, ya’ni Bootstrap yordamida yaratilgan veb-saytlar va veb-ilovalar har qanday qurilmada ajoyib ko‘rinadi. Bundan tashqari, Bootstrap shuningdek, veb-saytingizga funksiyalarni tez va oson qo‘shish uchun ishlatilishi mumkin bo‘lgan shakllar, tugmalar va navigatsiya menyo‘lari kabi bir qator oldindan ishlab chiqilgan UI komponentlarini taqdim etadi. Bootstrap shuningdek, katta va faol hamjamiyatga ega, ya’ni ishlab chiquvchilarga o‘rganish va so‘nggi ishlanmalardan xabardor bo‘lishga yordam beradigan ko‘plab resurslar va o‘quv qo‘llanmalari mavjud. Bundan tashqari, Bootstrap keng ko‘lamli brauzerlar bilan mos keladi, ya’ni ishlab chiquvchilar turli platformalarda muammosiz ishlaydigan kod yozishlari mumkin. Bootstrapning yana bir muhim afzalligi — bu sizning veb-saytingizni minimal harakat bilan ajoyib qilish qobiliyatidir. Bootstrap veb-saytingizga funksiyalarni tez va oson qo‘shish uchun ishlatilishi mumkin bo‘lgan bir qator oldindan ishlab chiqilgan UI komponentlarini taqdim etadi. Shuningdek, u veb-saytingizni tez va oson uslublash uchun ishlatilishi mumkin bo‘lgan bir qator oldindan ishlab chiqilgan CSS sinflarini taqdim etadi. Bootstrap shuningdek, veb-saytingizga qo‘shimcha funksiyalarni qo‘shish uchun ishlatilishi mumkin bo‘lgan bir qator JavaScript plaginlarini taqdim etadi. Ushbu plaginlar interaktiv va qiziqarli foydalanuvchi interfeyslarini yaratish uchun ishlatilishi mumkin bo‘lgan modallar, maslahatlar va karusellarni o‘z ichiga oladi.

Xulosa qilib aytganda, Bootstrap sezgir, birinchi mobil veb-saytlar va veb-ilovalarni loyihalash uchun mashhur va keng qoʻllaniladigan front-end ramka hisoblanadi. U kuchli sezgir tarmoq tizimiga, oldindan ishlab chiqilgan UI komponentlariga, oldindan ishlab chiqilgan CSS sinflariga, JavaScript plaginlariga va katta va faol hamjamiyatga ega. Minimal kuch sarflagan holda veb-saytingizni ajoyib qilish qobiliyati bilan Bootstrap millionlab odamlarning veb-ilovalarni yaratish usulini oʻzgartirdi. Bundan tashqari, Bootstrap keng va faol hamjamiyatga ega, yaʼni ishlab chiquvchilarga eng soʻnggi ishlanmalardan xabardor boʻlish va oʻrganishga yordam beradigan koʻplab resurslar va oʻquv qoʻllanmalari mavjud. Bootstrap oʻzining kuchli xususiyatlari bilan ishlab chiquvchilar uchun ajoyib vositadir.

Bootstrap-ning afzalliklari:

1) Responsive dizayn: Bootstrap-ning sezgir grid tizimi ishlab chiquvchilarga foydalanuvchi ekranining oʻlchamiga avtomatik ravishda moslashadigan maketlarni yaratishga imkon beradi. Bu Bootstrap bilan yaratilgan veb-saytlar va veb-ilovalar har qanday qurilmada ajoyib koʻrinishini anglatadi.

2) Oldindan ishlab chiqilgan UI komponentlari: Bootstrap veb-saytingizga funksiyalarni tez va oson qoʻshish uchun ishlatilishi mumkin boʻlgan shakllar, tugmalar va navigatsiya menyoʻlari kabi bir qator oldindan ishlab chiqilgan UI komponentlarini taqdim etadi.

3) Oldindan ishlab chiqilgan CSS sinflari: Bootstrap veb-saytingizni tez va oson uslublash uchun ishlatilishi mumkin boʻlgan bir qator oldindan ishlab chiqilgan CSS sinflarini taqdim etadi.

4) Katta va faol hamjamiyat: Bootstrap katta va faol hamjamiyatga ega, yaʼni ishlab chiquvchilarga eng soʻnggi ishlanmalardan xabardor boʻlish va oʻrganishga yordam beradigan koʻplab resurslar va oʻquv qoʻllanmalari mavjud.

5) Brauzerlararo muvofiqligi: Bootstrap keng koʻlamli brauzerlar bilan mos keladi, yaʼni ishlab chiquvchilar turli platformalarda muammosiz ishlaydigan kod yozishlari mumkin.

6) Moslashuvchan: Bootstrap juda moslashtiriladi va ishlab chiquvchilarga veb-sayt yoki veb-ilovaning koʻrinishini va hissiyotini osongina oʻzgartirishga imkon beradi.

7) JavaScript plaginlari: Bootstrap veb-saytingizga qoʻshimcha funksiyalar qoʻshish uchun ishlatilishi mumkin boʻlgan bir qator JavaScript plaginlarini taqdim etadi.

Bootstrap-ning kamchiliklari:

1) Cheklangan sozlash: Bootstrap juda moslashtirilgan bo'lsa-da, u bilan chinakam noyob dizaynni yaratish qiyin bo'lishi mumkin.

2) Katta fayl hajmi: Bootstrap ramkasi nisbatan katta va uni yuklash uchun vaqt kerak bo'lishi mumkin, bu sizning veb-saytingiz yoki veb-ilovangizning ishlashiga ta'sir qilishi mumkin, ayniqsa sekinroq internet ulanishlarida.

3) Brauzer mosligi bilan bog'liq muammolar: Bootstrap brauzerining ba'zi eski versiyalari bilan moslik bilan bog'liq muammolar bo'lishi mumkin, bu esa qo'shimcha vaqtinchalik echimlar yoki polifilllarni talab qilishi mumkin.

4) Sukut bo'yicha kirish imkoni yo'q: Bootstrap foydalanish imkoniyatini taqdim etmaydi, ishlab chiquvchilar o'z veb-saytlarini nogironlar uchun ochiq qilish uchun qo'shimcha choralar ko'rishlari kerak.

5) Eski brauzerlar uchun cheklangan qo'llab-quvvatlash: Bootstrap eski brauzerlar uchun cheklangan qo'llab-quvvatlanadi, ya'ni ishlab chiquvchilar o'zlarining ilovalari barcha brauzerlarda ishlashini ta'minlash uchun polifilllardan yoki boshqa vaqtinchalik echimlardan foydalanishlari kerak bo'lishi mumkin.

Xulosa qilib aytganda, Bootstrap sezgir, birinchi mobil veb-saytlar va veb-ilovalarni loyihalash uchun mashhur va keng qo'llaniladigan front-end ramka hisoblanadi. U kuchli sezgir tarmoq tizimiga, oldindan ishlab chiqilgan UI komponentlariga, oldindan ishlab chiqilgan CSS sinflariga, JavaScript plaginlariga, keng va faol hamjamiyatga va brauzerlararo muvofiqligiga ega. U sezgir dizayn, oldindan ishlab chiqilgan UI komponentlari, oldindan ishlab chiqilgan CSS sinflari, katta va faol hamjamiyat va brauzerlararo muvofiqligi kabi ko'plab afzalliklarga ega bo'lsa-da, u cheklangan sozlash, katta fayl hajmi, brauzer mosligi muammolari kabi ba'zi kamchiliklarga ega. sukut bo'yicha mavjud emas va eski brauzerlar uchun cheklangan qo'llab-quvvatlash. Loyihangizda Bootstrap-dan foydalanish yoki foydalanmaslik haqida qaror qabul qilishda ushbu omillarni hisobga olish muhimdir. Bundan tashqari, ishlab chiquvchilar ramkani o'rganish va so'nggi ishlanmalar va o'zgarishlardan xabardor bo'lish uchun biroz vaqt sarflashga tayyor bo'lishlari kerak. O'zining kuchli xususiyatlari va imkoniyatlari bilan Bootstrap sezgir va mobil qurilmalarga mos veb-ilovalarni yaratish uchun ajoyib vositadir.

Bootstrap bilan ishlashda quyidagi narsalarni yodda to‘tish kerak:

1) Grid tizimi: Bootstrap-ning sezgir tarmoq tizimi uning eng kuchli xususiyatlaridan biridir va har qanday qurilmada ajoyib ko‘rinadigan sezgir tartiblarni yaratish uchun undan qanday samarali foydalanishni tushunish muhimdir.

2) Moslashtirish: Bootstrap juda moslashtiriladi va ishlab chiquvchilarga veb-sayt yoki veb-ilovanning ko‘rinishini va hissiyotini osongina o‘zgartirishga imkon beradi. Noyob va moslashtirilgan dizaynni yaratish uchun Bootstrap-ni qanday sozlashni tushunish muhimdir.

3) Foydalanish imkoniyati: Bootstrap maxsus foydalanish imkoniyatlarini taqdim etmaydi, ishlab chiquvchilar o‘z veb-saytlarini nogironlar uchun ochiq qilish uchun qo‘shimcha choralar ko‘rishlari kerak.

4) Brauzer mosligi: Bootstrap brauzerining ba’zi eski versiyalari bilan moslik bilan bog‘liq muammolar bo‘lishi mumkin, bu esa qo‘shimcha vaqtinchalik echimlar yoki polifillarni talab qilishi mumkin. Sizing veb-saytingiz yoki veb-ilovangiz barcha asosiy brauzerlar bilan mos kelishini ta’minlash muhimdir.

5) Ishlash: Bootstrap ishlashga mo‘ljallangan, ammo dastur o‘sib borishi bilan unumdorlikka e’tibor berish va HTTP so‘rovlari sonini minimallashtirish hamda ilovangiz ish faoliyatini optimallashtirish uchun minimallashtirish va birlashtirish kabi usullardan foydalanish muhim ahamiyatga ega.

6) JavaScript plaginlari: Bootstrap veb-saytingizga qo‘shimcha funksiyalar qo‘shish uchun ishlatilishi mumkin bo‘lgan bir qator JavaScript plaginlarini taqdim etadi. Ushbu plaginlardan qanday foydalanishni va ularni ehtiyojlaringizga moslashtirish uchun qanday sozlashni tushunish muhimdir.

7) Mobil optimallashtirish: Mobil qurilmalardan foydalanish ortib borishi bilan veb-saytingiz yoki veb-ilovangiz mobil qurilmalar uchun optimallashtirilganligini ta’minlash muhimdir. Bootstrap mobil qurilmalarda veb-saytingiz ish faoliyatini yaxshilash uchun ishlatilishi mumkin bo‘lgan bir qator xususiyatlar va usullarni taqdim etadi.

Nazorat savollari.

1. Bootstrap nima uchun qo‘llaniladi?
2. Bootstrap kamchiliklari?
3. Bootstrap afzalliklari?

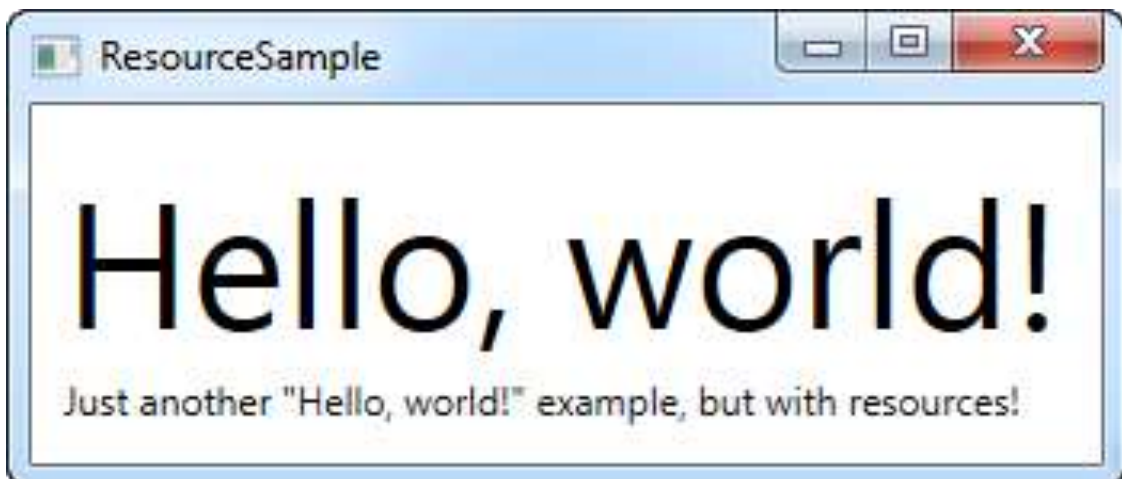
13-MAVZU. WPF MODELI

Reja:

- 1.WPF modeli haqida tushuncha
2. Statik va Dinamik resurslar.
3. CodeBehinddagi manbalar

WPF modeli juda qulay funksiyalarni taqdim etadi, ma'lumotlarni boshqarish sifatida butun oyna uchun yoki butun dastur uchun global miqyosda resurs sifatida saqlay oladi. Ma'lumotlar hozirgi ma'lumotdan tortib, WPF element ierarxiyasiga qadar har qanday narsa bo'lishi mumkin. Bu sizga ma'lumotlarni bir joyga qo'yishga, so'ngra ularni turli joylarda ishlatishga imkon beradi.

```
<Window
x:Class="WpfTutorialSamples.WPF_Application.ResourceSample"
xmlns="http://schemas.microsoft.com/winfx/2006/haml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/haml"
xmlns:sys="clr-namespace:System;assembly=mmscorlib"
Title="ResourceSample" Height="150" Width="350">
<Window.Resources>
<sys:String x:Key="strHelloWorld">Hello, world!</sys:String>
</Window.Resources>
<StackPanel Margin="10">
<TextBlock Text="{StaticResource strHelloWorld}" FontSize="56"
/>
<TextBlock>Just another " <TextBlock Text="{StaticResource
strHelloWorld}" />" ehample, but with resources!</TextBlock>
</StackPanel>
</Window>
```



WPF-dagi resurslar kalitga ega (x: Key atributi), bu dasturning istalgan qismidan StaticResource ifodasi bilan kalit yordamida murojaat qilishga imkon beradi. Ushbu misolda biz faqat satrni resurslarda saqlashni va uni ikki xil TextBlock elementlarida ishlatishni ko‘rib chiqdik.

Statik va Dinamik resurslar. Yuqoridagi misollarda men manbalarga(resursga) murojaat qilish uchun StaticResource marko‘p kengaytmasidan foydalandim, lekin DynamicResource ko‘rinishida boshqa alternativ(zaxira manba) mavjud. Bu resurslarning asosiy farqi shundaki, statik resurslar (StaticResource) faqat bir marta, HAML fayli yuklanganda aniqlanadi. Agar keyinchalik dastur ishlayotganda resurslar o‘zgartirilsa va statik bo‘lsa, bu o‘zgarishlar resurslarda aks etmaydi.

Boshqa tomondan, Dynamic Resources. Ular zarur bo‘lgan vaqtda aniqlanishi mumkin, so‘ngra ish vaqtida bekor qilinishi mumkin. Buni statik qiymatga bog‘lash va qiymatni kuzatadigan va qiymat o‘zgarganda sizga yuboradigan funksiya uchun majburiy deb o‘ylang. Shuningdek, dinamik manbalar dasturni ishlab chiqishda hali mavjud bo‘lmagan manbalardan foydalanishga imkon beradi, masalan: dasturning boshida ushbu resurslarni CodeBehind-dan qo‘shasiz.

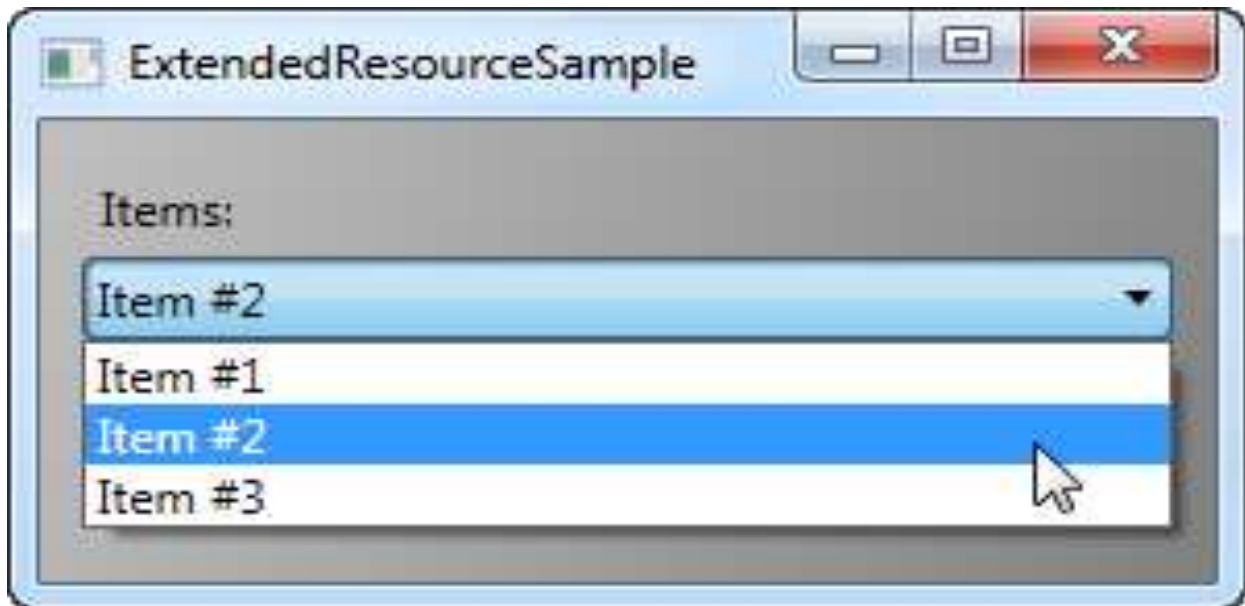
Resurslarning boshqa turlari.

```
<Windowx:Class="WpfTutorialSamples.WPF_Application.Extended
ResourceSample"
xmlns="http://schemas.microsoft.com/winfx/2006/haml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/haml"
xmlns:sys="clr-namespace:System;assembly=mscorlib"
Title="ExtendedResourceSample" Height="160" Width="300"
Background="{DynamicResource WindowBackgroundBrush}">
<Window.Resources>
<sys:String x:Key="ComboBoxTitle">Items:</sys:String>
<x:Array x:Key="ComboBoxItems" Type="sys:String">
<sys:String>Item #1</sys:String>
<sys:String>Item #2</sys:String>
<sys:String>Item #3</sys:String>
</x:Array>
<LinearGradientBrush x:Key="WindowBackgroundBrush">
<GradientStop Offset="0" Color="Silver"/>
```

```

<GradientStop Offset="1" Color="Gray"/>
</LinearGradientBrush>
</Window.Resources>
<StackPanel Margin="10">
<Label Content="{StaticResource ComboBoxTitle}" />
<ComboBox ItemsSource="{StaticResource ComboBoxItems}" />
</StackPanel>
</Window>

```



Yuqorida biz bir nechta qo‘shimcha manbalarni qo‘shdik, shunda bizning oynamiz endi qatorni, satrlar massivi va gradient cho‘tkasini o‘z ichiga oladi. String yorliq uchun ishlatiladi, satrlar massivi ComboBox elementlari va gradient cho‘tkasi oynaning fonida ishlatiladi. Ko‘rib to‘rganingizdek, juda ko‘p qiziqarli va turli xil narsalar manbalar sifatida saqlanishi mumkin.

CodeBehind-dagi manbalar

Oldingi misolda biz qanday qilib turli xil joylarda resurslarni saqlashimiz mumkinligini ko‘rdik, shuning uchun ushbu misolda biz har biri boshqa doirada saqlangan Code-back-dan uchta turli xil manbalarga murojaat qilamiz:

App.haml:

```

<Application x:Class="WpfTutorialSamples.App"
xmlns="http://schemas.microsoft.com/winfx/2006/haml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/haml"
xmlns:sys="clr-namespace:System;assembly=mcorlib"

```

```

StartupUri="WPF
application/ResourcesFromCodeBehindSample.haml">
<Application.Resources>
<sys:String x:Key="strApp">Hello, Application world!</sys:String>
</Application.Resources>
</Application>

```

Oyna:

```

<Window
x:Class="WpfTutorialSamples.WPF_Application.ResourcesFromCodeBehindSample"
xmlns="http://schemas.microsoft.com/winfx/2006/haml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/haml"
xmlns:sys="clr-namespace:System;assembly=mcorlib"
Title="ResourcesFromCodeBehindSample"           Height="175"
Width="250">
<Window.Resources>
<sys:String x:Key="strWindow">Hello, Window world!</sys:String>
</Window.Resources>
<DockPanel Margin="10" Name="pnlMain">
<DockPanel.Resources>
<sys:String x:Key="strPanel">Hello, Panel world!</sys:String>
</DockPanel.Resources>
<WrapPanel DockPanel.Dock="Top" HorizontalAlignment="Center"
Margin="10">
<Button Name="btnClickMe" Click="btnClickMe_Click">Click
me!</Button>
</WrapPanel>
<ListBox Name="lbResult" />
</DockPanel>
</Window>

```

Code-back (back-end)

```

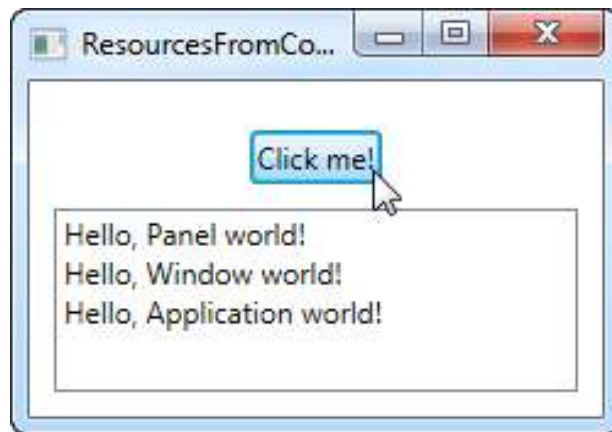
using System;
using System.Windows;
namespace WpfTutorialSamples.WPF_Application
{
public partial class ResourcesFromCodeBehindSample : Window
{
public ResourcesFromCodeBehindSample()

```

```

{
InitializeComponent();
}
private void btnClickMe_Click(object sender, RoutedEventArgs e)
{
lbResult.Items.Add(pnlMain.FindResource("strPanel").ToString());
lbResult.Items.Add(this.FindResource("strWindow").ToString());
lbResult.Items.Add(Application.Current.FindResource("strApp").ToString());
}
}
}

```



Shunday qilib, siz ko‘rib to‘rganingizdek, biz uch xil "Salom, dunyo!" xabarlar yozib chiqdik: Bittasi App.haml-da, biri oyna ichida, bittasi esa asosiy panel uchun. Interfeys tugma va ListBox dan iborat. Kod orqasida biz tugmachani bosish hodisasini boshqaramiz, unda har bir matn satrini ko‘rinib to‘rganidek, ListBox-ga qo‘shamiz. Biz resursni ob‘ekt sifatida qaytaradigan **FindResource()** usulidan foydalanamiz (agar topilgan bo‘lsa), keyin **ToString()** usuli yordamida o‘zimiz bilgan qatorga aylantiramiz.

FindResource() usulini turli xil sohalarda qanday ishlatayotganimizga e‘tibor bering — avval panelda, so‘ngra oynada va keyin amaldagi Application ob‘ektida. Resursni biz bilgan joydan qidirish mantiqan to‘g‘ri keladi, ammo yuqorida aytib o‘tilganidek, agar manba topilmasa, qidiruv ierarxiyani rivojlantiradi, shuning uchun biz asosan paneldagi **FindResource()** usulidan foydalanishimiz mumkin edi uchala holat ham, chunki agar u topilmasa, oynaga (Window) qadar va keyinchalik dastur darajasiga qadar davom etishi mumkin edi.

JavaScript ga kirish . **Java Script** bu Web-sahifalarni yanada interaktiv va funksional qilish imkonini beruvchi ssenariylar tayyorlovchi tildir. Web-sahifalarni kuzatgan bo'lsangiz xar bir sahifada biror matn, bir nechta rasmlar, vakti ko'rsatib turuvchi soat yoki brauzerning holatlar satrida yuguruvchi matnni qurishingiz mumkin. Yoki biror bir tuldirlilishi kerak bo'lgan forma bo'lishi mumkin. Agan anketaning biror bir qismini tuldirlishni unutib koldirsangiz xatolik haqida xabar paydo bo'ladi. Ayrim bir sahifalarda ekran bo'ylab harakat qiluvchi rasm yoki mantnni uchratishingiz mumkin.

Shu kabi harakatlarni **Java Script da** kuzatishingiz mumkin, shunga o'xshash effektlarni esa ushbu tilni biluvchi ixtiyoriy kishi yaratishi mumkin.

Java Script ni qo'llab-kuvvatlovchi brauzer Netscape2.0 , bo'lgan.

Netscape brauzerlarida Java Script ning qullab-quvvatlanishi

Brauzer	Versiya
Netscape 2.0	1.0
Netscape 3.0	1.1
Netscape 4.0	1.2
Netscape 4.5	1.3/1.4
Netscape 6.0	1.5

Internet Explorer brauzerlarida Java Script ning qo'llab-quvvatlanishi

Brauzer	Versiya
Internet Explorer 3.0	1.0/2.0
Internet Explorer 4.0	3.0
Internet Explorer 5.0	5.0
Internet Explorer 5.5	5.5

Java Script 1.5 va JScript5.5 Web-sahifangizning ko'rinishi va uning funksiyasini boshqarish imkonini beruvchi ssenariylar yozish tillaridir. Java Script ni hujjatning obyektli modeli (Document Object Model- DOM) va stillarning kaskadli jadvali (Cascading Style Sheets -CSS) kabi yangi texnologiyalar bilan birgalikda qo'llash bilan web-sahifalaringizdagi barcha hodisalarni nazorat qilishingiz mumkin. Java Scrip va obyektli modeldan foydalanib, sichqonchanning bosilishi va klaviaturadan ma'lumotning kiritilishiga ta'sir qiluvchi sahifalar

yaratish mumkin. Kaskadli stillar jadvali yordamida sahifadagi matnlarni ko'rsatish va berkitish, brauzer oynasi o'lchamlarini boshqarish va shu kabilarni bajarish mumkin.

Ushbu texnologiyalarni qo'llash bilan siz nafaqat Web-sahifa yaratasiz balki boshqa kompyuter dasturlari kabi harakat qiluvchi foydalanuvchi uchun muhit yaratasiz.

Ssenariya (script) – bu buyruqlar ketma-ketligi (ayrim xollarda hattoki dastur) bo'lib, boshqa dastur tomonidan kayta ishlanadi va interpretatsiya qilinadi. Bu esa ssenariy yozish uchun matn redaktori yyetarli ekanligini anglatadi, dastur yaratish uchun esa boshqa bir dastur (hech bo'lmaganda kompilyator) talab qilinadi. Boshqa so'z bilan aytganda ssenariy C va C++ kabi murakkab tarkibli tillarda yaratiluvchi dasturlardan ko'ra oson va tez yoziladi.

Biroq ssenariy bajarilishi uchun kompilyatsiya qilingan dasturning bajarilishidan ko'ra ko'p vakt talab qilinadi. Chunki ssenariyning xar bir buyrugi protsessor ostida emas, balki boshqa dastur yordamida kayta ishlanadi. Ushbu kamchilik ssenariyni yaratish uchun bupul matn redaktorining yyetarlilik va ayni paytda C++ tili uchun kompilyatorning bir necha yuz dollar turishi bilan izohlanadi.

Java Script ssenariysi yozish uchun instrumentlar. Oddiy matn redaktorini (masalan Windows Notepad) tanlash va unga kodni yozish oson bo'lsa kerak. Balki bu Java Script da ssenariy yozishning eng oson yo'li bo'lmasligi mumkin, biroq unda quyidagi ishonchli afzalliklar mavjud:

- Bepulligi
- Murakkab dasturni o'rganish talab qilinmaydi
- dasturiy ta'minot komplekti emas Java Script o'rganiladi

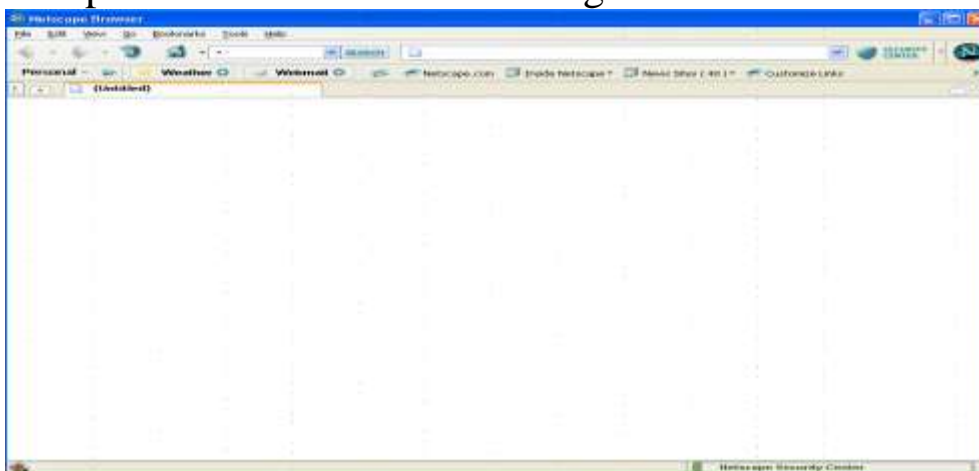
Vizual taxrirlash vositalaridan foydalanish. Java Script ssenariylarini yozishda vizual taxrirlash vositalari sifatida odatda HTML redaktorlaridan foydalaniladi. Bu turdagi vositalarga Microsoft Front Page, Macromedia Dreamweaver va Adobe GoLive larni kiritish mumkin.

- Bunda ko'pincha Java Script kodining kulda kiritilishi va tekshirilishi talab qilinmaydi va bu dastlab juda ahamiyatli bo'lib tuyo'lishi mumkin. Biroq ishni yengillashtirish bilan ixchamlikni yukotish mumkin.
- Biroq vizual taxrirlash vositalarining ham bir nechta afzalliklari

ham mavjud:

- Tezlik-kulda terilgandan ko‘ra natijaga tezrok erishiladi;
- Samaradorlik- vizual taxrirlash vositalari yordamida ishlaydigan ssenariyni olish mumkin;
- Maxsus funksiyalar- masalan rang bilan belgilash kodlarni o‘qish imkonini osonlashtiradi.

Brauzerning so‘ngi versiyasi. Siz Java Script ning zamonaviy versiyasini o‘rganar ekansiz uning barcha imkoniyatlaridan foydalanish uchun brauzerlarning eng so‘ngi versiyasiga ega bo‘lishingiz zarur. Minimum Netscape Navigator 4.0 va Microsoft Internet Explorer 5.5 brauzerlari o‘rnatilgan bo‘lishi kerak.



Boshqa instrumentlar. Java Script bilan ishni osonlashtirish uchun HTML sahifa shablonlaridan foydalanish va shaxsiy ssenariylar kutubxonasi yaratish kerak.

HTML shablon

```
<html>
<head>
<title>Html sahifa shabloni</title>
<script language="javascript">
<!--
// — — >
</script>
</head>
<body>
</body>
```

</html>



Rasm 1.3 Windows Notepad redaktori yordamida tayyorlangan HTML shablon

Java Script ni HTML sahifalarga tadbiq qilish. HTML va Java Script mutlako boshqa-boshqa texnologiya bo'lganligi sababli HTML hujjatga Java Script ni tadbiq qilishning uz qoidalari mavjuddir. Web-sahifalarga JavaScript ni kiritishning bir nechta usullari mavjud, biroq dastlab eng ko'p tarqalgan va ixcham turini ko'rib chiqamiz, ya'ni JavaScript ni bloklab <head> tegiga qo'yish usuli.

HTML hujjati. Web – sahifaning oddiy HTML tarkibini ko'rib chikaylik. Natija 2.1 rasmda keltirilgan:

```
<html>
<head>
<title>Oddiy sahifa</title>
</head>
<body>
<p> Judayam oddiy Web sahifa! </p>
</body>
</html>
```

Bu yerda e'tiboringizni HTML kodning <head> tegi bilan boshlanuvchi va </head> tegi bilan yopiluvchi blokka karating. Bu HEAD bloki yoki Web sahifaning sarlavxasi bo'lib, aynan shu yerga siz uz Java Script ssenariylaringizni joylashtirasiz.

Script bloki. Head blokiga Java Script ssenariysini kiritishdan oldin unga yangi HTML blokini kiritish shart. Bu blok SCRIPT bo'lib, uning vazifasi brauzerga mazkur blokning ochiluvchi va yopiluvchi

teglari orasida maxsus ssenariy joylashganligini ko‘rsatishdir. Bu blok HEAD blokning ixtiyoriy joyida quyilishi mumkin, biroq odatda u </head> yopiluvchi tegi oldidan quyiladi:

```
<html>
<head>
<title>Oddiy sahifa</title>
<script>
</script>
</head>
<body>
<p> Judayam oddiy Web sahifa! </p>
</body>
</html>
```

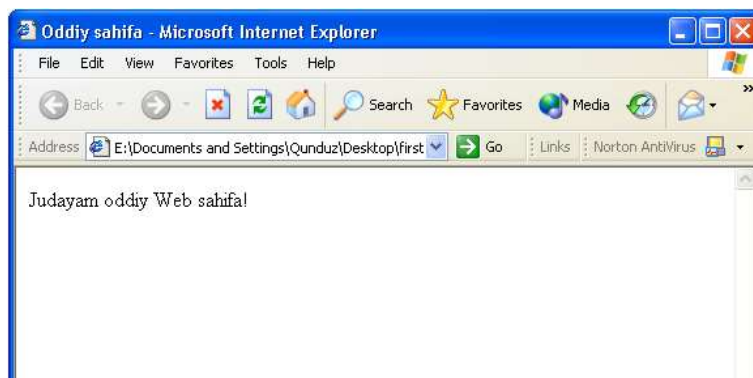
SCRIPT blokiga Java Script ssenariysini qo‘yishdan oldin yana bir necha harakatlar bajarilishi kerak. Birinchidan brauzerga SCRIPT blokida qanday turdagi ssenariy joylashishini ko‘rsatish kerak. Chunki Java Script ssenariy tayyorlovchi yagona til emas. Microsoft kompaniyasining VBScript tili ham yana bir ssenariy tayyorlovchi til hisoblanadi. Demak, Java Script ni izohlash uchun <script> tegiga language atributini qo‘shiladi:

```
<html>
<head>
<title>Oddiy sahifa</title>
<script language="" "">
</script>
</head>
<body>
<p> Judayam oddiy Web sahifa! </p>
</body>
</html>
```

Va Java Script dan foydalanayotganligingiz haqida Java Script so‘zini kushtirnok ichida ko‘rsatib utiladi.

```
<html>
<head>
<title>Oddiy sahifa</title>
<script language="JavaScript" "">
</script>
</head>
```

```
<body>
<p> Judayam oddiy Web sahifa! </p>
</body>
</html>
```



Nazorat savollari

1. JavaScript dasturlash tilining vazifasi ?
2. Brauzerning vazifasi ?
3. Brauzerlar turlari ?
4. Ssenariya nima ?

14-MAVZU. JAVASCRIPT DASTURI

Reja:

1. JavaScript da ma'lumotlar tiplari
2. JavaScript da o'zgaruvchilar
3. Ifodalar va arifmetik operatorlar

JavaScript da ma'lumotlar tiplari Olamdagi eng kadrli narsa bu informatsiyadir. JavaScript da informatsiyaning xar bir bo'lagi qiymat (value) misolida ko'riladi. Informatsiya xar xil bo'lgani kabi qiymatlarning ham turli kategoriyalari mavjuddir. JavaScript da oddiy ma'lumotlar turlari asosiy turlar (primitive types) deb nomlanadi.

3 ta asosiy ma'lumotlar turi mavjud:

satr (string); son (number); mantiqiy (boolean).

Satrlar. Satr turdagi ma'lumotlar eng ko'p tarqalgan bo'lsa kerak. Satr – harflar, sonlar va tinish belgilarni uz ichiga oluvchi belgilar ketma-ketligidir. JavaScript da satrlar ko'pincha matnlar ifoda qilish uchun ishlatiladi.:

Assalomu alaykum, hush kelibsiz!

Siz kim?

JavaScript ssenariysidagi satrlar apostrof yoki qo'shtirnoqka olinadi, masalan:

“Assalomu alaykum!”

Ikkita to'ring ishlatilishining sababi shundaki, qo'shtirnoq ichida apostrofli satr ishlatish va aksincha mumkin. Masalan:

“Mening o'g'irligim 60 kg”

”Siz kimsiz?”- deb hayron buldi u'

Son. JavaScript 2 ta turdagi sonlarni kabul qiladi: butun (integer) va kuzgaluvchi nuqtali son (floating-point number).

Butun sonlar. Butun sonlar uzida musbat masalan, 1, 2, 3 va manfiy -1, -2, -3 va sonlarni va 0 ni namoyon qiladi.

Kuzgaluvchi vergulli sonlar. Qo'zg'aluvchi vergulli sonlar

Katta va kichik sonlar. JavaScript da ishlatiladigan sonlar judayam katta yoki judayam kichik bo'lishi mumkin. Eng katta son sifatida 10^{308} va eng kichik son sifatida 10^{-308} qiymat karaladi.

Mantiqiy ifodalar. Mantiqiy ifodalar satr va sonlardan faqat 2 ta qiymat kabul qilishi bilan farqlanadi: true (rost) va false (yolgon).

Alohida turdagi ma'lumotlar: sonlar, noaniq va aniqlanmaydigan ifodalar. Ko'rib chiqilgan ma'lumotlar turidan

boshqa bir nechta turlar mavjud bo‘lib quyida sonli qiymatlarning turlari keltiriladi:

- Musbat cheksizlik
- Manfiy cheksizlik
- Musbat va manfiy nul
- Mavjud bo‘lmagan son (not a number -NaN)

Yana 2 ta maxsus ma’lumotlar turi mavjud:

- Noaniq (Null)
- Aniqlanmaydigan (Undefined)

Dastlab keltirilgan 4 ta tur haqida gapiradigan bo‘lsak, siz ular bilan ishlashingizga to‘g‘ri kelmaydi (albatta siz dasturingizda cheksiz sondan foydalanmaysiz). Mavjud bo‘lmagan son deganda ma’nosiz matematik operatsiyalar natijasi tushuniladi (masalan, nolga bo‘lish). Cheksiz qiymat deganda 10^{308} dan katta son tushuniladi.

Noaniq turdagi ma’lumot faqat bitta null qiymatni olishi mumkin.

O‘zgaruvchilar nafaqat Java Script tilida balki barcha dasturlash tillarida muzim ahamiyatga egadir. Ular yordamida barcha turdagi ma’lumotlarni boshqarish mumkin.

O‘zgaruvchi- Java Script ssenariysi bajarilayotgan paytda ma’lum turdagi ma’lumotlarni saklovchi kompyuter xotirasining nomlangan katakchasi hisoblanadi. Bu ta’rif biroz murakkab tuyo‘lishi mumkin, lekin o‘zgaruvchilardan foydalanish judayam osondir.

Java script da o‘zgaruvchilar O‘zgaruvchi yaratish. Java Script tilida o‘zgaruvchi yaratish judayam osondir. Yana o‘sha HTML shabloniga murojaat kilamiz.

```
<html>
<head>
<title>Oddiy sahifa</title>
<script language="JavaScript">
</script>
</head>
<body>
</body>
</html>
```

Dastlab o‘zgaruvchini yaratish kerak. Buni 2 yo‘l bilan oldindan va zarur paytda e’lon qili shorqali yaratish mumkin. Dastlab

oldindan e'lon qilinadigan to'rni ko'rib chiqamiz. Ushbu bo'lim oxirida "zarur paytdagi" o'zgaruvchilarning yaratilishini ko'rib chiqamiz.

Misol: Java Script tilida o'zgaruvchini e'lon qilish (yaratish) uchun var operator iva undan so'ng o'zgaruvchi nomi ko'rsatiladi. Ushbu misolda msg nomli o'zgaruvchi e'lon qilingan:

```
<html>
<head>
<title>Oddiy sahifa</title>
<script language="JavaScript ">
var msg;
</script>
</head>
<body>
</body>
</html>
```

O'zgaruvchi nomlari haqida quyidagilarni bilishingiz kerak:

- O'zgaruvchilar nomida katta va kichik registr belgilarini va ularni birgalikda ishlatish mumkin;
- O'zgaruvchini e'lon qilgunga qadar uning qiymati undefined hisoblanadi;
- O'zgaruvchi nomi sonlardan boshlana olmaydi;
- O'zgaruvchilar nomida probellar ishlatilishi mumkin emas, agar ularni ajratish kerak bo'lsa, tagiga chizish belgisidan (_) foydalaning;
- O'zgaruvchilar nomida dollar (\$) belgisini ishlatishdan extiyot bo'ling chunki, Internet Explorer3.02 va Netscape Navigator 2.02 brauzerida tanilmaydi.

Quyida to'g'ri nomlangan o'zgaruvchilar keltirilgan:

```
Msg
Hello_world
Msg1
Msg_1
```

Quyida noto'g'ri nomlangan o'zgaruvchilar keltirilgan:

```
1msg – son bilan boshlangan
Hello world – probelga ega
Var – Java Script ning xizmatchi operatori
```

Dollar\$ — nomda \$ belgisi mavjud

Msg va msg Ushbu nomlarni bitta ssenariyda ishlatmang, chunki ular faqat registr bilan farqlanadi.

Misol. Agar siz e'lon qilingan o'zgaruvchiga shu zaxoti qiymat bermokchi bo'lsangiz buni o'sha satrning uzida qilish mumkin:

```
<html>
<head>
<title>Oddiy sahifa</title>
<script language="JavaScript ">
var msg="Java Script ning o'zgaruvchilar olamiga hush kelibsiz!";
</script>
</head>
<body>
</body>
</html>
```

Misol. O'zgaruvchini e'lon qilish va unga qiymat berish alohida satrlarda ham amalga oshirilishi mumkin:

```
<html>
<head>
<title>Oddiy sahifa</title>
<script language="JavaScript ">
var msg;
msg="Java Script ning o'zgaruvchilar olamiga hush kelibsiz!";
</script>
</head>
<body>
</body>
</html>
```

Misol. Agar siz o'zgaruvchingizning qiymati satr emas, son bo'lishini xoxlasangiz uni quyidagicha qilishingiz mumkin:

```
<html>
<head>
<title>Oddiy sahifa</title>
<script language="JavaScript ">
var msg=22;
</script>
</head>
<body>
```

```
</body>
</html>
```

Bir nechta o‘zgaruvchilarni e’lon qilish. Bitta emas balki bir nechta o‘zgaruvchilarni e’lon qilish kerak bo‘lsa, ularni xar birini alohida satrda yoki bitta satrda e’lon qilish mumkin.

```
var msg1; var msg2; var num1; var num2;
yoki var msg1, msg2, num1, num2;
```

Bir nechta o‘zgaruvchini bitta satrda e’lon qilganda ular vergul bilan ajratilishi va oxirida nuqta -vergul bilan tugashi kerak. Agar vergul bilan ajratilmasa, xatolik yuz beradi.

Misol. Bitta satrda o‘zgaruvchilarni e’lon qilishingiz va qiymat berishingiz mumkin.

```
<html>
<head>
<title>Oddiy sahifa</title>
<script language="JavaScript ">
var msg1="Barchaga", msg2="salom", num1=6, num2=52;
</script>
</head>
<body>
</body>
</html>
```

O‘zgaruvchilar qiymatini ekranga chiqarish. Demak, siz bir nechta o‘zgaruvchini e’lon qildingiz va ularga satr va son qiymatni berdingiz. Endi ushbu o‘zgaruvchilarni amalda ishlatib ko‘ramiz. Quyidagi kichik misolda alert() metodi o‘zgaruvchi qiymatni ekranga chiqarish uchun ishlatiladi (Rasm 4.1).

Misol. 2 ta o‘zgaruvchi msg1 va num1 ni e’lon qiling va ularga “Barchaga salom” (satr) va 22 (son) qiymatlarni bering:

```
<html>
<head>
<title>Oddiy sahifa</title>
<script language="JavaScript ">
var msg1="Barchaga salom", num1=22;
</script>
</head>
<body>
</body>
```

```
</html>
```

Keyin alert() metodidan foydalanib o‘zgaruvchini metodning qavslari orasiga joylashtiring.

```
<html>
```

```
<head>
```

```
<title>Oddiy sahifa</title>
```

```
<script language="JavaScript ">
```

```
var msg1="Barchaga salom", num1=22;
```

```
alert(msg1);
```

```
</script>
```

```
</head>
```

```
<body>
```

```
</body>
```

```
</html>
```



Endi siz ikkinchi o‘zgaruvchining qiymatni boshqa darchada chiqarishingiz mumkin (rasm 4.2).

```
<html>
```

```
<head>
```

```
<title>Oddiy sahifa</title>
```

```
<script language="JavaScript ">
```

```
var msg1="Barchaga salom", num1=22;
```

```
  alert(msg1);
```

```
  alert(num1);
```

```
</script>
```

```
</head>
```

```
<body>
```

```
</body>
```

```
</html>
```

Sahifangizni yana saqlang va kiritilgan o‘zgarishlarni qurish uchun uni brauzeringizda qaytadan ishlatib. Bu safar ogohlantiruvchi ikkita darcha ketma-ket namoyon bo‘ladi va ikkinchi darchada 22 soni paydo bo‘ladi (rasmga karang).



Ifoda va shart. O‘zgaruvchi va massivlar yordamida siz o‘zingizga kerakli bo‘lgan informatsiyani turli ko‘rinishda kompyuter xotirasida saqlaysiz. Biroq bu informatsiyani siz faqatgina ekranga chiqarish yoki ogohlantiruvchi darchada joylashtirish uchun emas, balki ko‘proq amallar bajarishingiz mumkin. Siz uni o‘zgartirishingiz, uni boshqarishingiz yoki tekshirishingiz mumkin. Ushbu maqsadlar uchun ifoda va shartlar keltirilgan.

Ifodalar(expressions) ikki yoki undan ortiq qiymatlarni birlashtirish uchun va natijada yangi qiymatga ega bo‘lish uchun ishlatiladi. Quyidagi yig‘indili ifoda bunga misol bula oladi:

$$1+2=3$$

Shartlar (conditions) qiymatlarni solishtirish va mantiqiy qiymatni (true yoki false) aniqlash imkonini beradi.

$$3+3=6 ? \text{ true}$$

Amallar bilan tanishuv

Shart va ifodalarda ma’lumotlar bir-biri bilan amallar bilan birlashtiriladi. Agar harakat bitta kattalik bilan amalga oshirilsa u holda bu amal **bitta operandli amal** (unary operator) deb ataladi. Agar bunaqa kattaliklar 2 ta bo‘lsa, u holda bunday amallar **2 ta operandli amallar** (Binary operator), agar 3 ta bo‘lsa- **3 ta operandli amallar** (ternary operator) deyiladi.

Arifmetik amallar

Arifmetik amallar bu barchaga ma’lum bo‘lgan matematik amallar hisoblanadi:

- qo‘shish (+) $1+3=4$
- ayirish (-) $2-1=1$
- bo‘lish (/) $4/2=2$
- ko‘paytirish (*) $2*2=4$
- bo‘lishdan kolgan koldik (%) $9\%5=4$

Bular 2 ta operandli amallarga misolllar edi.

Java Script ssenariylariga misollar:

Quyida arifmetik amallardan foydalanilgan bir nechta misollar keltiriladi. Bu yerda SCRIPT blokining o‘zi berilgan.

Misol. Qo‘shish amaliga misol 6.1- rasm

```
<script language="JavaScript ">  
var a=6, b=4;  
alert(a+b);  
</script>
```



Misol. Ayirish amaliga misol 6.2- rasm

```
<script language="JavaScript ">  
var a=6, b=4;  
alert(a-b);  
</script>
```



Misol. Bo‘lish amaliga misol 6.3- rasm

```
<script language="JavaScript ">  
var a=6, b=3;  
alert(a/b);  
</script>
```



Misol. Ko‘paytirish amaliga misol 6.4- rasm

```
<script language="JavaScript ">  
var a=6, b=3;  
alert(a*b);  
</script>
```



Misol. Bo‘lishdan kolgan koldik amaliga misol -6.5 rasm

```
<script language="JavaScript ">  
var a=6, b=4;  
alert(a%b);  
</script>
```



6.5 rasm

Taqqoslash amallari ifodalarni bir-biri bilan solishtirish ishlatiladi. Ularga quyidagi amallar kiradi:

- kichik (<) $5 < 6$
- kichik yoki teng (<=) $6 <= 6$ yoki $6 <= 7$
- katta (>) $7 > 4$
- katta yoki teng (>=) $5 >= 5$ yoki $5 >= 4$
- teng (==) $5 == 5$
- teng emas (!=) $5 != 3$

Bularning barchasi 2 ta operandli amallarga kiradi. Bu misollarda sonli ma’lumotlar taqqoslandi, biroq satrlarda ham ushbu amallar ishlatilishi mumkin (bu haqida keyinroq so‘z yuritamiz). Eng asosiysi, bitta turdagi ma’lumotlar taqqoslanilishi mumkin. Aks holda Java Script bir turdagi ma’lumotlarni boshqa turga o‘tkazishga harakat qiladi va bu har doim ham amalga oshirilaverilmaydi. Xatolarga yo‘l qo‘ymaslik uchun bitta turdagi ma’lumotlarni taqqoslang.

Mantiqiy amallar

Quyida 3 ta mantiqiy amallar keltirilgan:

- mantiqiy XA (and) **&&**
- mantiqiy YOKI (or) **||**
- mantiqiy YUK (not) **!**

Bitta operantli amallar

Nomidan ham ko‘rinib turibdi bu amal bitta kattalik bilan amalga oshiriladi. Bularga:

- prefiksli va postfiksli o‘shish (increment) — ++
- prefiksli va postfiksli kamayish (decrement) — --
- unar plyus +
- unar minus –

Unar minus ifoda ishorasini qarama-qarshisiga aylantiradi. 4 ta amal ichida bu eng oddiysi hisoblanadi. Bundan tashqari unar plyus ishorani almashtirish uchun emas, balki operandni songa aylantirish uchun ishlatiladi (masalan, bu satr bo‘lgan holda).

Prefiks/postfiks o‘shish va kamayish amali o‘zgaruvchi qiymatni bittaga o‘shishi va kamayishi uchun xizmat qiladi. Biroq bu amallar natijasi bu amalning preyaiks yoki postfiksli ekanligiga bog‘liqdir.

Prefiksli amalda agar $a=5$ bo‘lsa, u holda $++a+2=8$ bo‘ladi, chunki o‘zgaruvchi qiymatiga 2 soni qo‘shilmasidag oldin u bittaga orttirilgan edi. Shu vaktning uzida $-a+2=6$, chunki o‘zgaruvchi qiymati 2 ga kamaytirilmasdan oldin u bittaga kamaytirilgan edi.

Postfiksli amalda esa aksincha, ifodada o‘zgaruvchining eski qiymati ishlatilib, so‘ngra uning qiymati bittaga kamaytiriladi yoki orttiriladi. Shu tariqa agar $a=5$ bo‘lsa, u holda $a++ +2=7$, chunki o‘zgaruvchining qiymati ifoda hisoblangandan so‘ng bittaga orttiriladi. Bu yerda $a-- +2=7$ ham o‘rinlidir, chunki o‘zgaruvchining qiymati ifoda hisoblangandan so‘ngina bittaga kamaytiriladi.

Misol. Prefiksli o‘shish – 6.6 rasm

```
<script language="JavaScript" >
var a=6, b=2;
alert(++a+b);
</script>
```



O‘zlashtirish amali. Oddiy o‘zlashtirish amali (=) bo‘lib, u o‘zgaruvchiga qiymatni yuklash uchun xizmat qiladi. Siz bu amal bilan tanishsiz, buni a o‘zgaruvchiga 3 sonini yuklashni misol qilish mumkin $a=3$.

Bu amaldan foydalanib bitta qiymatni bir nechta o‘zgaruvchiga yuklashni ham bajarish mumkin. Quyidagi misolda 5 qiymatni bir nechta (a,b,c va d) o‘zgaruvchilariga o‘zlashtirish ko‘rsatilgan:

$a=b=c=d=5$

Boshqa o‘zlashtirish amallari kushma amallar hisoblanib quyida ular keltiriladi:

- $x+=3$ bu amal $x=x+3$ bilan teng kuchlidir
- $x-=3$ bu amal $x=x-3$ bilan teng kuchlidir
- $x*=3$ bu amal $x=x*3$ bilan teng kuchlidir
- $x/=3$ bu amal $x=x/3$ bilan teng kuchlidir

Misol. += kushma amali – 6.10 rasm

```
<script language="JavaScript" >
```

```
var x=6;
```

```
alert(x+=2);
```

```
</script>
```



Misol. -= kushma amali –6.11 rasm

```
<script language="JavaScript">
```

```
var x=6;
```

```
alert(x -=2);
```

```
</script>
```



Java Script tilida operatorlardan foydalanish. Operatorlar (statements) Java Script tilining kuchli instrumenti hisoblanadi. Operatorlarni o‘rganib olganingizdan so‘ng sizning ssenariylaringiz

yuqori darajaga ko‘tariladi va murakkab vazifalarni bajarish imkoniga ega bo‘lasiz.

Ushbu bo‘limda quyidagi mavzular ko‘rib chiqiladi:

- operatorlar vazifalari
- variable operatori
- if operatori
- sikl tashkil qiluvchi operator
- continue operatori
- break operatori
- switch operatori

Operator nima?

Java Script ning barcha ssenariylari uning hajmi va murakkabligidan qat’iy nazar operatorlar ketma-ketligidan tashkil topgan bo‘ladi.

Misol. Ekranga ogohlantiruvchi darchani chiqarish operatorning bajarilishi natijasi hisoblanadi:

```
<script language="JavaScript ">  
alert("Bu ham operator!");  
</script>
```

Misol. Quyida operatorga yana bir misol keltiriladi:

```
<script language="JavaScript ">  
var ourname="Talaba va domla";  
</script>
```

Java Script da nuqtali vergul operatorning tugashini bildiradi. Ularni qo‘yish shart emas, agar siz qo‘ymasangiz JavaScript ning o‘zi siz uchun qo‘yadi. Biroq ulardan foydalanish quyidagi sabablar uchun kerak:

- sizning ssenariyingiz oson o‘qiladi
- nuqtali vergulni qo‘ymaslik ayrim hollarda xatolikka olib keladi, chunki Java Script xar doim ham ularni to‘g‘ri qo‘ymaydi.

variable operatori

Misol. Variable operatori bilan tanishsiz. Ya’ni

```
Var this_year=2001;
```

Variable operatori this_year o‘zgaruvchisi yaratilganligi va unga 2001 qiymati berilganligini bildiradi.

If operatori

If operatoridan buyruqlar ketma-ketligini boshqarish va turli maqsadlarda foydalaniladi. Java Script ssenariysida bu operator 2 ta guruh operatorlardan bittasini tanlash va bajarish imkonini beradi. Tanlash Bool ifodalar yordamida amalga oshiriladi(true va false).

Masalan, tasdiklash darchasining if operatori yordamida qanday ish qurishi bilan tanishaylik.

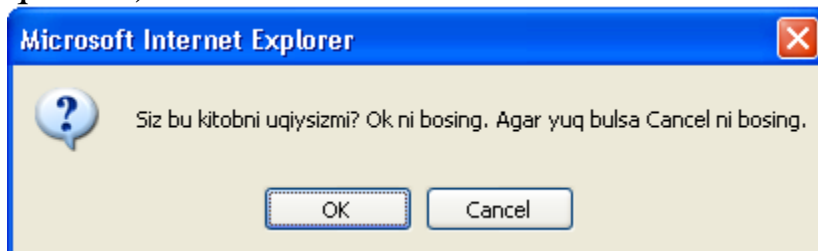
Misol. Dastlab confirm() metodi yordamida oddiy tasdiqlash darchasini hosil qilamiz:

```
<script language="JavaScript ">
```

```
confirm("Siz bu kitobni uqiysizmi? Ok ni bosing. Agar yuq bo'lsa  
Cancel ni bosing.");
```

```
</script>
```

Foydalanuvchiga 2ta tugmachadan bittasini tanlash imkoni beriladi(7.1-rasm). Keling, OK tugmasini bosgan foydalanuvchilar uchun kod yozamiz (Cancel tugmasini hech kim bosmaydi deb faraz qilamiz).



Foydalanuvchi OK tugmachasini bosgan paytda Bul ifoda true qiymatni oladi va siz tanlov amalga oshirilganini bilishingiz mumkin. Buning uchun o'zgaruvchi e'lon qilishingiz va unga confirm() metodi qiymatni yuklashingiz kerak bo'ladi:

```
<script language="JavaScript ">
```

```
var response=confirm("Siz bu kitobni uqiysizmi? Ok ni bosing. Agar  
yuq bo'lsa Cancel ni bosing.");
```

```
</script>
```

response o'zgaruvchisi savolga javobni uzida saqlab qoladi. prompt() metodi yordamida foydalanuvchi kiritgan satrni ham o'zgaruvchi saqlashi mumkin. Boshqacha qilib aytsak, agar foydalanuvchi Ok tugmachasini bossa, u holda response o'zgaruvchisi true qiymatni o'zlashtiradi. Bunga ishonch hosil qilish uchun if operatoridan foydalanamiz.

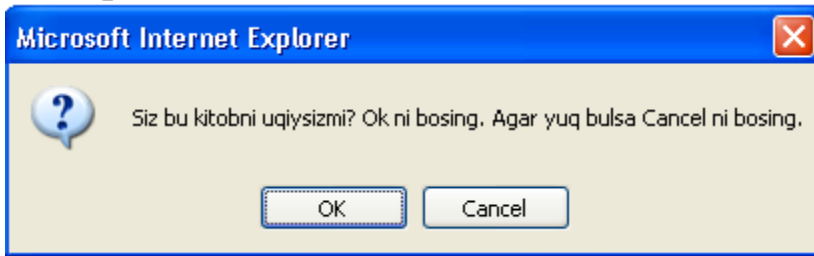
Misol. Bu Java Script da quyidagicha amalga oshiriladi:

```
<script language="JavaScript ">
```

```

var response=confirm(“Siz bu kitobni o‘qiysizmi? Ok ni bosing. Agar
yo‘q bo‘lsa Cancel ni bosing.”);
if (response==true)
{
alert(“Zur tanlov”);
}
</script>

```



Shu tariqa if operatori quyidagi sxema orqali ifodalanadi:

```

If (shart){
Operatorlar
}

```

Shart doimo bul ifoda talab qiladi. Agar shart rost (true) qiymatni olsa figurali qavs ichidagi operatorlar bajariladi.

Agar shart natijasi Null yoki undefined bo‘lsa, bu false (yolg‘on) qiymati bilan teng kuchli hisoblanadi.

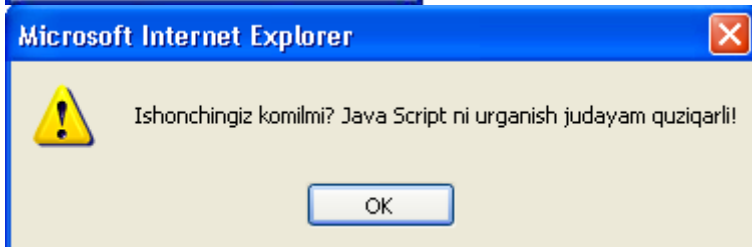
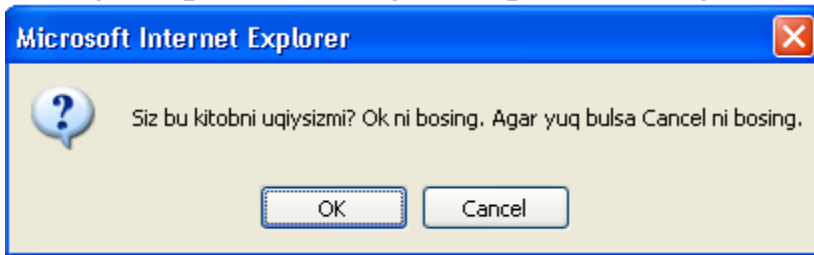
Misol. Endi sizning ssenariyngiz ok tugmasini bosgandan so‘ng aniq bir amallarni bajaradi. Cancel tugmasini bosgandan so‘ng if operatorining 2-qismi if...else ishga to‘shadi.

```

<script language="JavaScript ">
var response=confirm(“Siz bu kitobni uqiysizmi? Ok ni bosing. Agar
yuq bo‘lsa Cancel ni bosing.”);
if (response==true)
{
alert(“Zur tanlov”);
}
else
{
alert(“Ishonchingiz komilmi? Java Script ni o‘rganish judayam
qiziqarli!”)
}
</script>

```


Else qismi yopiluvchi figurali qavsdan keyin quyiladi va u uz ichiga oladigan operatorlar figurali qavslar ichiga olinadi.



Shu tarika if...else operatori quyidagi sxema asosida ko‘riladi:

```
If ()  
{  
operatorlar  
}  
else  
{  
operatorlar  
}
```

Nazorat savollari

1. Java Script qanday ma'lumotlar tiplari ?
2. Java Script O'zgaruvchilar turlari ?
3. Java Script Ifodalar turlari va arifmetik operator ?
4. Java Script da qanday ugartirilgan operatorlari turlari ishlatiladi ?

FOYDALANILGAN ADABIYOTLAR RO‘YHATI.

1. Ayupov R.X, Ismoilov S.I, Ayupov X.R. Xususiy kompyuterda ishlash/ “Microsoft Word 2000 matn muharriri”, T., TMI, 2002.
2. Yuldashev U, Boqiev R., Mamarajabov M. EXSEL-97. O‘quv qo‘llanma. T.: 2000.
3. Yuldashev U, Mamarajabov M., Mirvalieva K. Power Point 97/ Uquv qo‘llanma. T.: 2001.
4. Braydo V.A. Ofis texnikasi: Boshqarish va ish yuritish uchun. T.: Mehnat, 2001.
5. Nasretdinova Sh. Excel 7.0 jadval hisoblagichlari/ Moliya va bank akademiyasi nashriyoti, T.: “Moliya”: 2000.
6. Ayupov R.X., Ilxomova E. Kompyuter tarmoqlari va internet tizimi. T.: TMI, 2003. Microsoft FrontPage-2000. Shag za shagom: Prakticheskoe posobie. Herbert Schildt – Java A Beginner Guide Sixthm Edition – 2014
7. Java 8 Rukovodstvo dlya nachinayuschikh-Schildt_G
8. Шилдт Г. - Java 8. Полное руководство – 2015
9. Зыков С.В. Программирование. Объектно-ориентированный подход. Учебник и практикум для академического бакалавриата. Издательство Юрайт.2018.-155.
10. Гниденко И.Г..Технологии и методы программирования.Учебное пособие для прикладного бакалавриата.Издательство Юрайт.2018.-235.
11. Тузовский А.Ф. Объектно-ориентированное программирование. Учебное пособие для прикладного бакалавриата.Издательство Юрайт. 2018.- 206с.
12. Левин М.П. Самоучитель разработки web-сайтов: HTML, CSS, графика, анимация, раскрутка+видеокурс DVD. М.:Изд.:Триумф.,2007., - 400с.
13. Ташков П.А. Веб мастеринг на 100%. Учебное пособие. СПб:Питер 2008г.
14. Орлов Л.В. Web - сайт без секретов. 2-е изд. М.: ЗАО —Новый
15. М.: Izdatelstvo EKOM, 2000

T. U. Utapov, Z. M. Adizova

WEB TEXNOLOGIYALAR (I QISM)
(O'QUV QO'LLANMA)

Muharrir:

E.Eshov

Tex. muharrir:

D.Abduraxmonova

Musahhih:

M.Shodiyeva

Badiiy rahbar:

M.Sattorov

Nashriyot litsenziyasi № 022853. 08.03.2022.

Original maketdan bosishga ruxsat etildi: 12.12.2023. Bichimi 60x84. Kegli 16 shponli. "Times New Roman" garnitura 1/16.

Ofset bosma usulida. Ofset bosma qog'oz.

Bosma tabog'i 8,75. Adadi 100. Buyurtma № 232



KAMOLOT

**"BUXORO DETERMINANTI" MCHJ
bosmaxonasida chop etildi.**

Buxoro shahar Namozgoh ko'chasi 24 uy

Tel.: + 998 91 310 27 22