

**O‘ZBEKISTON RESPUBLIKASI O‘LIY VA O‘RTA MAXSUS
TA‘LIM VAZIRLIGI
TOSHKENT DAVLAT SHARQSHUNOSLIK INSTITUTI**

**«INFORMATIKA VA AXBOROT TEXNOLOGIYALAR»
fanidan**

MA‘RUZA MATN

1 – mavzu. CSS (cascading style sheets) texnologiya asoslari.

Maqsad: CSS ning maqsad va imkoniyatlarini tushuntirish

Kalit soʻzlar: Content, dizayn, inline, embedding, Linking, Import

Asosiy savollar:

1. Hujjatning mazmuni va uning dizayni
2. Kaskad jadvallar stilining yaratilishi va rivojlanishi tarixi
3. CSS ni HTML hujjatda qoʻllash.

Hujjatning mazmuni va uning dizayni

Taʼrif: Content- bu hujjatning mazmunini ifodalovchi axborotlardan tashkil topgan qismi boʻlib, unda tekst, rasm, jadval, grafika boʻlishi mumkin.

Dizayn – bu hujjatning axborotli qismini biror- bir qurilmada aks ettirish usuli.

Odatda hujjatning mazmuni brauzerda aks ettirish uchun HTML teglaridan foydalaniladi. Hujjatning dizaynini ham HTML orqali ifodalash mumkin. Lekin buni ikkita manfiy tomoni bor. Birinchidan hujjatning hajmi kattalashib ketadi. Ikkinchidan hujjatning dizaynini oʻzgartirish uchun katta hajmdagi ishini bajarishga roʻgʻri keladi. Yani formatlash ishi qiyinlashadi.

CSS kaskad jadvallar stilidan foydalanish quyidagicha qulayliklarni beradi:

1. Koʻp hollarda kodning hajmi kamayadi;
2. Sahifalarni printerga yuborishni boshqarish imkoniyati yuzaga keladi;
3. Kodni logiklik xususiyati ortadi;
4. Ehtiyoj tugʻilganda kodni oʻzgartirish osonlashadi;
5. Hujjatning formatlash ishini boshqarish sezilarli ortadi.

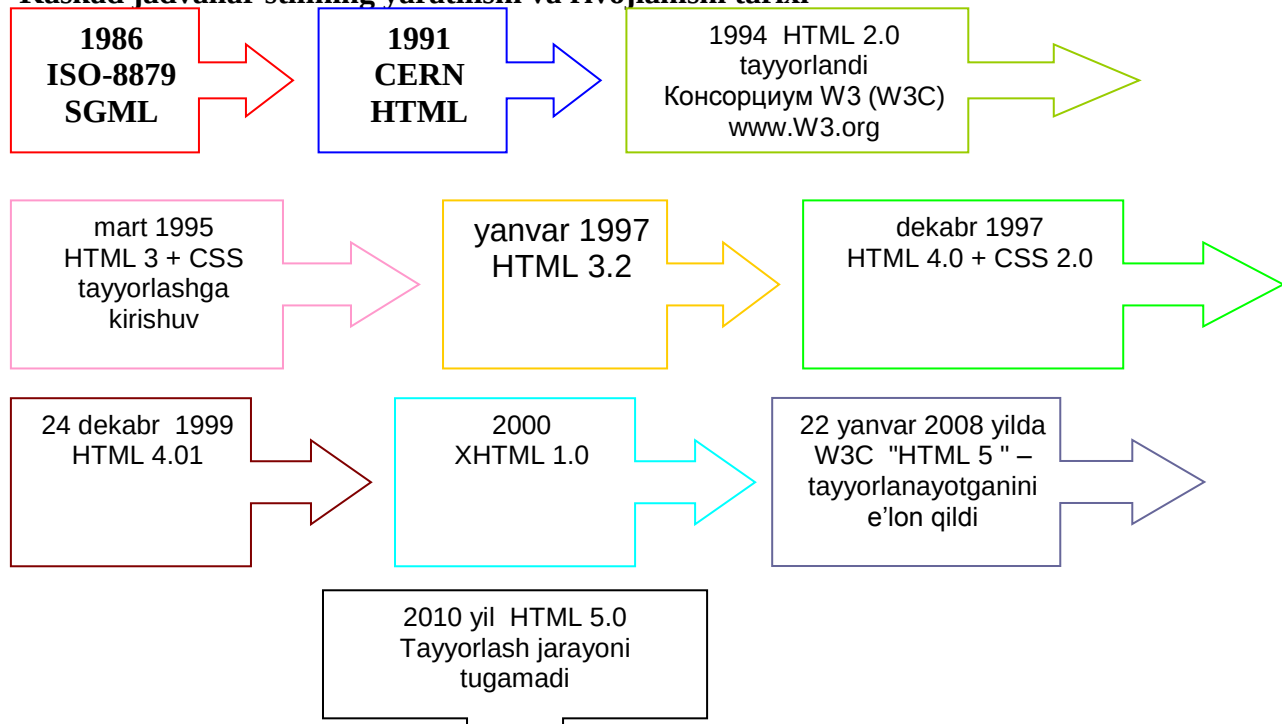
Shunday qilib, CSS web-dizaynerlar uchun yangi imkoniyatlar beradi.

Kaskad jadvallar stili - bu shunday qoidalar tarzida yozilgan xususiyatlar toʻplamiki, HTML elementlarini brauzer oynasida yoki boshqa qurilmada qanday aks etishi kerakligini aniqlab beruvchi vosita.

Aniqlab berish usullari quyidagicha:

- Bogʻlash (linking)-Tashqi jadvallar stili. Web-sahifani bezash stili alohida fayl sifatida beriladi. Bu faylning nomi *.css kengaytmaga ega boʻladi.
- Hujjatning ichiga joylashtirish (embedding)-hujjatning sarlavha sohasida Style elementi kiritilib, web –sahifani bezash stili shu yerda koʻrsatiladi;
- Hujjatning ichidagi html tegning oʻzida bevosita bezash stili (inline) –bunda Style atributidan foydalaniladi;
- Import (import)-IE.

Kaskad jadvallar stilining yaratilishi va rivojlanishi tarixi



CSS ni birinchi versiyasi 1996 yil 17 dekabrda standart sifatida tasdiqlandi.

<http://www.w3.org/TR/CSS21/cover.html>

CSS ni o'qiy oladigan birinchi brauzer Internet Explorer ni uchinchi versiyasi edi, bu narsa Internet Explorer ni ommaviyligini ortishiga sabab bo'ldi.

Netscape Navigator ni to'rtinchi versiyasidan boshlab, keyingi versiyalari CSS ni o'qiy olgan. Opera 5.0 va keyingi versiyalari CSS ni o'qiy oladi.

Web sahifa yaratuvchilar har doim brauzerlarni imkoniyatlarini hisobga olib ish qiladilar.

CSS 1 ni imkoniyatlari.

Kaskad jadvallar stilini qo'llash quyidagi alomatlarga qarab belgilanadi:

Berilgan tipdagi har qanday elementga;

- Elementning aniqlangan klasi bo'yicha. Klas HTML elementining ichida atribut **class** ni kiritish bilan aniqlanadi;

`<P class="blue">Bu abzatsdagi Matnko'k rangda bo'ladi</P>`

- HTML elementining ichida atribut id yordamida unikal(yagona) element aniqlanadi;

`<H1 id="unique">Unikal sarlavha</H1>`

- Boshqa elementni ichida, biror elementni qo'llash;

`<P>Abzatsni ichidagi Matnni <STRONG>ajratib</STRONG> yozish mumkin</P>`

- Elementni birinchi harfiga qo'llash, Bukvitsa yaratish;

- Elementni birinchi qatoriga qo'llash;

- Aktiv, ko'rib chiqilgan, ko'rib chiqilmagan ssilkalarga;

Bloklar modeli shunday tashkil qilinganki, u quyidagi imkoniyatlarni beradi:

- bloklarni o'lchami;
- hoshiya, chegara chiziqlari(ramka), otstuplar o'lchami;
- ramkaning tipi va rangi;
- bloklarni joylashtirish,

Shriftni boshqarish uchun quyidagi parametrlar o'rinli:

- Tashqi ko'rinishi va qalinlik darajasi;
- Har xil o'lchov birliklarida o'lchami;
- Garnituras.

Standart CSS2 CSS1ga asoslanib yaratilgan va u CSS1ni ba'zi xususiyatlar, mexanizmlarni to'ldiradi.

CSS ni HTML hujjatda qo'llash

Kaskad jadvallar stilini faqatgina CSS da emas, balki XSL da ham yozish mumkin. Odatda stillar jadvali tilini o'rnatish uchun META elementidan foydalaniladi. Bu elementdan hujjatning HEAD bolimiga joylashuvchi turli tuman xizmatchi axborotlarni berishda foydalaniladi:

`<head>`

`<META http-equiv="Content-Style-Type" content="text/css">`

`</head>`

http-equiv="Content-Style-Type" atributi, bu META element kaskad jadvallar stiliga tegishli ekanini bildiradi, **content="text/css"** atributi esa jadvallar stili CSS tilida yozilganini ko'rsatadi.

Inline stilidan foydalanish

Bu usul hujjatni istalgan joyida style atributini qo'llash bilan yoziladi. Masalan:

<H1 style="color: green"> Yashil sarlavha</H1>

Bu holda hujjatni ichidagi faqat bitta sarlavha yashil rangda bo'ladi.

Embeding stilidan foydalanish

Hujjatning HEAD bo'limiga stil qoidalar kiritiladi, bu qoidalar <style type="text/css"> va </style> elementlar ichiga yoziladi.

<STYLE type="text/css">

H1{ color:green }

</STYLE>

Bu holda berilgan hujjatdagi hamma birinchi darajali (H1) sarlavhalar yashil rangda bo'ladi.

STYLE, elementini yana bir atributi, media quyidagi qiymatlarni qabul qilishi mumkin.

- screen - kompyuter ekranlari uchun;
- tty – simvollarni aks ettirish cheklangan qurilmalar,terminal,teletayp va xokazo;
- tv –televizor tipidagi qurilmalar;
- projection –projektor tipidagi qurilmalar uchun.
- handheld – mobil telefon, cho'ntak kompyuterlari uchun;
- print – pechatga chiqaruvchi qurilma yoki pechatni oldindan ko'rib chiqish rejimi;
- braille –Brayl alfavitli qurilmalar uchun(ko'rish qobiliyati pastlar);
- aural –ovozni sintezlash qurilmalari uchun;
- all – hamma qurilmalar uchun.

Misol keltiramiz:

<html>

<head>

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

<title>@media</title>

<style type="text/css">

@media screen { /* brauzerda aks ettirish uchun stil */

BODY {

font-family: Arial, Verdana, sans-serif; /* shriftni turi

*/

font-size: 90%; /* shriftni o'lchami */

color: #000080; /* Matnni rangi */

}

H1 {

background: #faf0e6; /* Matnni tagidagi fon rangi */

border: 2px dashed #800000; /* sarlavha atrofidagi ramka */

color: #a0522d; /* Matnni rangi */

padding: 7px; /* Matnatrofidagi hoshiya */

}

H2 {

color: #556b2f; /* Matnni rangi */

margin: 0; /* otstuplarni yo'qotamiz */

}

P {

margin-top: 0.5em; /* yuqori otstup */

}

}

@media print { /* pechat uchun stil */

BODY {

font-family: Times, 'Times New Roman', serif;

}

H1, H2, P {

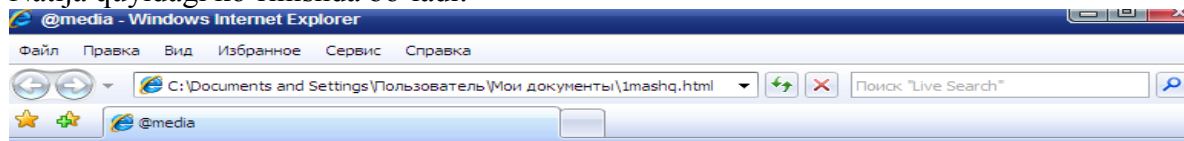
color: #000; /* Matnqora rangda */

```

    }
  }
</style>
</head>
<body>
<h1> Hujjatning mazmuni va uning dizayni </h1>
  <h2>CSS texnologiyadan foydalanish</h2>
  <p> Content- bu hujjatning mazmunini ifodalovchi axborotlardan tashkil topgan qismi
bo'lib, unda tekst, rasm,jadval,grafika bo'lishi mumkin.
</p>
  <p> Dizayn – bu hujjatning axborotli qismini biror- bir qurilmada aks ettirish usuli.
</p>
</body>
</html>

```

Natija quyidagi ko'rinishda bo'ladi:



Hujjatning mazmuni va uning dizayni

CSS texnologiyadan foydalanish

Content- bu hujjatning mazmunini ifodalovchi axborotlardan tashkil topgan qismi bo'lib, unda tekst, rasm,jadval,grafika bo'lishi mumkin.

Dizayn – bu hujjatning axborotli qismini biror- bir qurilmada aks ettirish usuli.

Tashqi stildan foydalanush

Faraz qilaylik sizning hujjatingiz bir nechta sahifadan iborat. Har bir sahifada bir xil `<style>...</style>` elementini takrorlamaslik uchun, kerakli stil qoidalari yozilgan alohida fayl tayyorlanadi. Bu faylning nomi *.css kengaytmaga ega bo'ladi. Bu fayl LINK elementi bilan har bir sahifaning HEAD bo'limida, quyidagi tarzda yoziladi:

```
<LINK href="style.css" rel="stylesheet" type="text/css">
```

bu yerda "style.css" – stil qoidalari yozilgan alohida tashqi faylning nomi.

LINK elementning href atributi bilan fayl joylashgan URL adres va faylni nomi ko'rsatiladi.

Atribut **rel** esa bu fayl HTML hujjatga qanday munosabatda ekanini bildiradi. (relation – bog'lanish ma'nosini beradi).

Import usuli. Bu usulda ham HTML hujjat tashqaridagi alohida fayl bilan bog'lanadi.Bu ish maxsus direktiva `@import` bilan quyidagi tarzda amalga oshiriladi:

```
<STYLE TYPE="text/css">
```

elementlar uchun stilni ifodalash

```
@import url (style.css)
```

stilni ifodasini davomi

```
</STYLE>
```

Bu usul faqat internet explorer uchun o'rinli. Bu usuldan kam foydalaniladi.

Misol keltiramiz:

```
<html>
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html;
```

```
charset=windows-1251">
```

```
<title>stilni importi</title>
```

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

```
  @import "/style/main.css" screen; /* monitorga chiqarish uchun stil */
```

```
  @import "/style/palm.css" handheld, print; /* pechat uchun stil */
```

```
</style>
</head>
<body>
<p>...</p>
</body>
</html>
```

Kaskad. Nima uchun kaskad jadvallar degan termin ishlatiladi. Kaskad –bu stillarni qo'llash ketma-ketligi (yoki iyerarxiyasi) deb tushunmoq kerak. Yuqorida keltirilgan bitta hujjatning o'ziga to'rt hil stilni qo'llash mumkinligiga e'tibor beramiz. Kaskad stil qoidalarni qo'llash usulini aniqlaydi. Bir nechta stil qo'llanganda, brauzer qaysi stilni o'qisa, sahifa shu stilda ko'rsatilgan qoidalar bo'yicha kompyuter ekranida aks etadi.

[Nazorat uchun savollar:](#)

1. Nima uchun CSS dan foydalaniladi?
2. CSS ni HTML hijjatga qo'llash qanday usullar bilan amalgam oshadi?

Uyga vazifa.

Shunday stil yaratingki, hujjat brayzer oynasida quyidagi tarzda aks etsin:

- Asosiy ranglar: oq fonda qora tekst;
- Abzatslar- o'ng tomonga tekislansin;
- Sarlavhalar- o'rtaga tekislanadi, Arial shriftida yoziladi va qizil rangda bo'lsin.
- Kursiv bilan ajratilgan terminlar yashil rangda bo'lsin.

Adabiyotlar.

1. Мальчук, Е.В. HTML и CSS. Самоучитель. М. Издательский дом "Вильямс", 2008.
2. Михаил Дубаков Web-МАСТЕРИНГ Средствами CSS .Санкт-Петербург, 2002
3. Мейер Э. CSS – каскадные таблицы стилей. Подробное руководство, 33е издание. – Пер. с англ. – СПб: Символ Плюс, 2008. – 576 с., ил.
4. Шмит К. CSS. Рецепты программирования. 3-е изд.: Пер. с англ. — М.: Издательство «Русская редакция»; СПб.: «БХВ-Петербург», 2011. — 672 с: ил.
5. Хольцшлаг, Молли Е. Языки HTML и CSS : для создания Web-сайтов : [учеб. пособие] / М. Хольцшлаг; [пер. с англ. А. Климович]. — М.: ТРИУМФ, 2007. — 304 с.: ил. — (Официальный учебный курс).

2- mavzu. Selektorlarning tiplari

Maqsad: Selektorlarning tiplari, o'lchov birliklari bilan tanishtirish

Kalit so'zlar:Selektor, Hususiyat, qiymat,stilni e'lon qilish,klass,ID,konteksr selector,

[Asosiy savollar:](#)

1. CSS qoidalari
2. Selektorlarning tiplari
3. O'lchov birliklari

CSS qoidalari. CSS, ham boshqa programmalashtirish tillari kabi qat'iy aniqlangan sintaksisga, ya'ni jadvallar stili yaratish bo'yicha qoidalar jamlanmasiga ega. CSS da elementlar, atributlar mavjud emas. Unda qoidalar bor. Har bir qoida HTML ni bir yoki bir nechta elementini tashqi ko'rinishini ifodalaydi.

CSS qoidalari – bu jadvallar stilining shunday strukturalashgan birligiki, unda biror HTML elementning stili ifodalangan bo'ladi.

CSS qoidalar, selector va stilni e'lon qilish blokidan tashkil topadi.Selektor har doim chap tomonda joylashadi va undan keyin figurali qavsni ichida stilni e'lon qilish bloki joylashadi. Har bir e'lon,hususiyat va uni qiymatidan tashkil topadi. E'lonlar bir –biridan nuqtali vergul bilan ajratiladi. Aynan hususiyat stilni tashqi ko'rinishini aniqlaydi.



Ta'rif. Stil qo'llash mumkin bo'lgan elementning konstruktsiyasini selector deb ataymiz.

Selektorlarning tiplari

- **Element bo'yicha selector**

Selektor sifatida elementning nomidan ham foydalanish mumkin (p, td, body).

Masalan quyidagicha konstruktsiya yaratamiz:

```
P {
color: #CCC;
font-size: 14px}
```

Bu abzats ichidagi Matnkulrang rangda, 14 piksel o'lchamli shriftda yozilganini ko'rsatilgan.

HTML tilidagi ixtiyoriy elementdan shu tarzda selector sifatida foydalanish mumkin. Selektorga qo'llanilgan stil, butun sahifa uchun o'rinli bo'ladi. Baza shunday holat yuzaga keladiki, H1 sarlavha ba'zi joyda ko'k rangli, 18 px o'lchamli bo'lsin, ba'zi joyda esa H1 sarlavha yashil rangda, 16 px o'lchamli bo'lsin. Agar biz selector sifatida H1 ni olsak, bunday differentsiallashtirishga erisha olmaymiz. Bu muammoni hal qilish uchun CSS1 da ikkita mexanizm, **klass va unikal identifikatorlar(ID)** tushunchasi kiritilgan.

- **Klass bo'yicha selector**

Nomidan ko'rinib turibdiki, bu mexanizm, berilgan elementlarni klasslarga ajratadi.

Bu ish quyidagicha amalga oshiriladi:

```
<P>Bu abzatsda biror Matnkeltirilgan</P>
```

```
<P CLASS="smaller"> Bu abzatsdagi Matnesa kichraytirilgan shrift bilan
yoziladi </P>
```

Ko'rsatilgan klassga stillar quyidagi tartibda yoziladi:

```
P { color: #CCC; font-size: 14px}
```

```
P.smaller {font-size: 12px}
```

Bu holda birinchi abzats 14 piksel o'lchamli shrift bilan yoziladi. Ikkinchi abzats esa 12 piksel o'lchamli shrift bilan yoziladi. Sintaksis ikkala holda ham o'xshash, lekin klassga murojaat nuqta orqali amalga oshiriladi, nuqtadan keyin klassni nomi yoziladi.

Agar ko'rsatilgan klassni faqatgina abzats uchun emas, balki ixtiyoriy element uchun qo'llasak, stil quyidagicha yoziladi:

```
.smaller {color: #CCC; font-size: 14px}
```

Agar ikkita element <P> va <PRE> bir xil klass bilan berilsa, ularni differentsiallashtirish mumkin:

```
P.smaller {color: #CCC; font-size: 14px}
```

```
PRE.smaller {color: #F00; font-size: 12px}
```

Bu holda <PRE> element bilan ko'rsatilgan Matnqizil rangda, shrift 12 piksel boladi.

<P> element bilan ko'rsatilgan Matnkulrang rangda, shrift 14 piksel olchamda boladi.

- **ID bo'yicha selector:**

Unikal identifikatorlarni qisqa dilib, ID deb ataymiz. Berilgan hujjatda faqat bir martagina ID selektoridan foydalanish mumkin.

```
<H1 ID="first"> Unikal sarlavha</H1>
```

ID unikal sarlavha uchun CSS stil yozganda ajratuvchi sifatida (#) belgisidan foydalaniladi:

```
H1#first {
```

```
color: #00F;
```

```
font-family: Verdana, sans-serif;
```

```
font-weight: bold;
```

```
text-align: center}
```

- **KonMatnselektor**

Web-sahifalar yaratishda bazan, bir tegni ichiga boshqa tegni joylashtirishga extiyoj paydo bo'ladi. Bu holda stildan to'g'ri foydalanish uchun, konMatnselektorlar yordam beradi. Masalan. uchun stil belgilaylik, qachonki u <P> konteynerni ichida joylashgan bo'lsa. Sintaksis quyidagicha bo'ladi:

EF { Stil qoidalarini ifodalash }

Bu yerda **E** konteyner teg, **F esa** — <E> konteynerni ichidagi teg. Bu holda stil <F> tegi uchun qo'llaniladi va quyidagi kodga amal qilinadi <E><F> ...</F></E>. Faqatgina ikkita teg emas, balki undan yuqori qatlamlar ishlatish mumkin. Bu holda konstruktsiya quyidagicha bo'lishi mumkin: div div ul li {...}.

Misol

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
  <title>KonMatnselektor</title>
  <style type="text/css">
    P B {
      font-family: Times, serif; /* shriftlar oilasi */
      font-weight: bold; /* qalin chiziqli */
      color: red; /* Matnqizil rangda */
    }
  </style>
</head>
<body>
  <div><b>Qalin chiziqli tekst</b></div>
  <p><b>Bir vaqttni o'zida ham qalin chiziqli va qizil rangli Matn</b></p>
</body>
</html>
```

- **Universal selector**

Ba'zi hollarda web-sahifaning hamma elementlari uchun bir vaqtda bitta stil o'rnatishga to'g'ri keladi. Bu holda universal selektordan foydalaniladi.

Sintaksis quyidagicha bo'ladi:

*{Stil qoidalarini ifodalash}

Universal selektordan foydalanish uchun yulduzcha (*) belgisi ishlatiladi. Bazi hollarda universal selektorni ko'rsatish shart emas. Masalan *.class va .class yozuvlar bir xil natijani chiqaradi. Universal selektordan foydalanish uchun yulduzcha (*) belgisi ishlatiladi. Masalan.

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
  <title>Universal selektor</title>
  <style type="text/css">
    *{
      margin: 0;
      padding: 0; /*Hamma elementlar uchun hoshiya va otstup nolga teng */
      font-family: Arial;
      font-size: 14px;
    }
  </style>
</head>
<body>
```

<p> Universal selektordan foydalanish uchun yulduzcha (*) belgisi ishlatiladi. Bazi hollarda universal selektorni ko'rsatish shart emas. Masalan *.class va .class yozuvlar bir xil natijani chiqaradi.</p>

</body>

</html>

O'lchov birliklari.

CSS da hususiyatlarni qiymatini berish uchun turli o'lchov birliklari ishlatiladi. Bular uzunlik o'lchov birligi, ranglarni berish birligi va URL.

Uzunlik birliklari

- **em** - **ems**, foydalanilayotgan shriftning balandligi
- **ex** - **x-height**, foydalanilayotgan yozma harfning balandligi ("x")
- **px** - **pixels**, piksellar (monitor ekranidagi bitta nuqta)
- **in** - **inches**, dyumlar (1in=2,54 sm)
- **cm** - **centimeters**, santimetrlar
- **mm** - **millimeters**, millimetrlar
- **pt** - **points**, punktlar (1pt = 1/72in = 0,35mm)
- **pc** - **picas**, piklar (1pc = 12pt)

Rang	RGB	Kod	Qisqacha
Oq	255,255,255	#ffffff	#fff
Qora	0,0,0	#000000	#000
Qizil	255,0,0	#ff0000	#f00
Yashil	0,255,0	#00ff00	#0f0
Ko'k	0,0,255	#0000ff	#00f

- **%** - nisbiy qiymat (masalan +20%)

Ranglarni belgilash. CSS da ranglar uch xil usulda beriladi: kalitli so'z orqali (blue, red va hokazo), o'n oltilik sistemadagi kodi va RGB modelidagi kod.

Nazorat uchun savollar:

1. Selektor nima?
2. Selektorning hususiyati va qiymati nima?
3. Selektorlarning qanday turlari mavjud?
4. Qanday o'lchov birliklari mavjud?

Uyga vazifa.

Shunday stil yaratingki, hujjat brayzer oynasida quyidagi tarzda aks etsin:

- Sahifaning hamma tomondan hoshiyasi 20px bolsin;
- Abzatslar chap tomonga tekislansin;
- Sarlavha uchun ID selector belgilang va uning hossalari va qiymatini ko'rsating;
- Abzatsni ichidagi kursiv bilan ajratilgan terminlar yashil rangda aks etsin.

Adabiyotlar

1. Мальчук Е.В. HTML и CSS Самоучитель. М.Издательский дом "Вильямс", 2008.
2. Михаил Дубаков ВЕБ-МАСТЕРИНГ Средствами CSS .Санкт-Петербург, 2002

3 – mavzu. Pseudoklaslar va psevdoelementlar

Maqsad: Pseudoklas va psevdoelement tushunchasi va undan foydalanish usullarini ko'rsatish

Kalit so'zlar: pseudo-class selectors, pseudo-element selectors

Asosiy savollar:

1. Pseudoklasslar
2. Psevdoelement

Psevdoklas va psevdoelementlarni qiziqarli, chiroyli sahifa tayyorlashda ahamiyati beqiyos.

Psevdoklas selektorlar

Psevdoklas selektorlarni ko'pchilik brouzerlar o'qiydi va shuning uchun amalda keng qo'llaniladi.

CSS 1 da ssilkalar bilan ishlash uchun tort xil psevdoklas mavjud:

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

CSS yordanida ssilkalarni tashqi ko'rinishini berish uchun quyidagicha qoidalar yozish mumkin:

A: link { ... oddiy (ko'rib chiqilmagan) ssilkani bezash stili... }

A:active { ... aktiv ssilkani bezash stili ... }

A:visited { ... ko'rib chiqilgan ssilkani bezash stili... }

va nixoyat HTML vositasi bilan bajarish mumkin bo'lmagan ish , sichqoncha ko'rsatkichi bilan aktivlashtirilgan ssilkani berish

A: hover { ... ko'rib chiqilmoqchi bo'lgan ssilkani bezash stili... }

Klas selektori orqali ssilka yaratsak, u quyidagi ko'rinishda bo'ladi:

a:visited {color: red;}

W3C Web site

Xuddi shu ishni psevdoklas selektori orqali ko'rsatamiz:

a:visited {color: red;}

Bu holda ko'rib chiqilgan har qanday ssilka qizil rangda bo'ladi, va unga klass atributini qo'shish shart emas.a va visited ni ajratuvchi ikki nuqta (:) aynan psevdoklasdan foydalanilayotganini bildiradi.Psevdoklas va psevdoelementni hamma kalit so'zlari ikki niqtadan boshlanadi.

Psevdoklaslardan, oddiy klas va identifikatorlar bilan birgalikda turli kombinatsiy tuzib ham foydalanish munkin.

... tekst...

A.pagelink: hover {... stilni ifodalash ... }

Yoki identifikator bilan birgalikda foydalanish:

...tekst...

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 psevdoklasdan foydalanilgan

a: hover {color: red;}

a: active {color: yellow;}



Bu ikkita qoidada dinamik psevdoklasdan foydalanilgan

Dinamik psevdoklaslar foydalanuvchining ta'sirida hujjatning tashqi ko'rinishini o'zgartirish imkonini beradi. CSS2.1. uchta dinamik psevdoklasni aniqlaydi:

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 etilayotganhar qanday elementga qo'llash mumkin
:active	Foydalanuvchi tomonidan aktivlashtirilgan har qanday elementga qo'llash mumkin

Misol 1.

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:

Name	Eric Meyer
Title	Standards Evang
E-mail	

Misol 2.

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 hujjatni tiliga e'tibor bermiqchi bolsak, :lang() psevdoklasiga murojaat qilamiz. Masalan frantsuz tilidagi ixtiyoriy elementni og'ma shriftda ko'rsatish uchun quyidagicha qoida yozamiz:

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

CSS2.1 da psevdoklaslarni kombinatsiyasidan foydalanish mumkin. Masalan hemis tilida yozilgan hamda korib chiqilmagan va ko'rib chiqilgan ssilkalarni ifodasi quyidagi ko'rinishda bo'ladi:

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

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

Psevdoelement. HTML tilida abzatsning birinchi qatori yoki birinchi harfini ifodalash uchun element yo'q. Demak bu ishni bajarish uchun bezash stilini qo'llab bo'lmaydi. Shuning uchun han 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**

Klass yordamida bukbitsa yaratish usulini ko'ramiz:

<P>Katta harf (bukbitsa) yaratish</P>

stil qoidalarini yozamiz:

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

Bu holda har bir abzatsni birinchi harfini **,,,**, teglar orasiga yozish kerak. Bu ish noqulaylik yaratadi.

Psevdoklasdan foydalansak ish ancha soddalashadi, quyidagicha kod yozish yetarli:

<P> Psevdoelement first-letter yordamida bukbitsa yaratamiz</P>

stil yozish qoidasi:

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

Agar Matnbirinchi harfdan ajralgan holda aks etsin desak, qo'shimcha hususiyat **float:left** beramiz. U holda bukbitsa chap tomonga tekislanadi, Matnbukbitsadan ajralib, uni atrofida joylashadi.

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

p:first-letter {color: red;} - abzatsni birinchi harfi qizil rangda aks etadi.

h2:first-letter {font-size: 200%;} - sarlavhaning birinchi harfi ikki baravar kattalashgan holda aks etadi.

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

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

bu holda ixtiyoriy abzatsning birinchi qatoridagi Matnkursiv shrift bilan aks etadi.

p:first-line {color: purple;} - abzatsning 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 hususiyatlarni quyidagi jadvalda keltiramiz:

:first-letter	:first-line
Hamma hususiyatlar: font color background margin padding border text-decoration text-transform line-height float	Hamma hususiyatlar: 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.";}

Psevdoelementlardan 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 abzatsning 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-qiyamati 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 kozi hech bir aybni ko'rmaydi,</p>
<p>Lekin norozilik ko'zi barcha ayblarni oshkor qiladi...</p>
</body>
</html>
```

Nazorat uchun savollar:

1. Psevdoklas nima va uni qanday turlari mavjud?
2. Psevdoelement nima va uning qanday turlari mavjud?

Uyga vazifa.

Shunday stil yaratingki, hujjat brayzer oynasida quyidagi tarzda aks etsin:

- Sahifaning hamma tomondan hoshiyasi 10px bolsint;
- Abzatslarning birinchi harfi 150 % kattalashsin va kursiv shaklda bo'lsin;
- Sarlavha uchun yashil fon rangi belgilang ;

Adabiyotlar

1. Мальчук, Е.В. HTML и CSS. Самоучитель. М. Издательский дом "Вильямс", 2008.
2. Михаил Дубаков ВЕБ-МАСТЕРИНГ Средствами CSS .Санкт-Петербург, 2002
3. Eric A. Meyer, Cascading Style Sheets. The Definitive Guide. O'Reilly 2008

4- mavzu. CSSning hususiyatlari

Maqsad: CSS da hususiyatlarning berish usullari tushuntiriladi

Kalit so'zlar: color, background, font-family, font-style, text-decoration, text-transform

Asosiy savollar:

1. Rang va fon
2. Shrift
3. Matnva uning hususiyati
4. Intervallar

Color

Bu hususiyat elementning ichidagi Matnni rangini berish vositasidir. Bu hususiyatning qiymati uch xil ko'rinishdan birida beriladi:

1. kalit so'zlar bo'yicha (red, yellow, Silver,...);
2. o'n oltilik kod bo'yicha (#ff0099, #006688,...);
3. rgb modelidagi o'nlik kod bo'yicha (color:rgb(25, 205, 172)).

Background-color

Bu hususiyatdan web sahifaning yoki ixtiyoriy elementning fon rangini berishda foydalaniladi.

Background – image

Bu hususiyatdan web sahifaning yoki ixtiyoriy elementning fon rasmini berishda foydalaniladi. Rasmi fayl aloxida papkada saqlanadi.

Background-repeat

Bu hususiyat sahifaning fon rasmini takrorlanish stilini beradi. U quyidagi qiymatlardan birini qabul qiladi:

1. Repeat -tasvir vertikal va gorizontal yo'nalish bo'yicha takrorlanadi va butun sahifani qoplaydi.
2. Repeat-x- tasvir gorizontal yo'nalish bo'yicha bir qator bo'lib takrorlanadi.
3. |repeat-y- tasvir vertikal yo'nalish bo'yicha bir ustun bo'lib takrorlanadi.
4. No-repeat- tasvir takrorlanmaydi.

Misol :

```
body {
background-image:url (images/bg.gif);
background-repeat:no-repeat;
background-attachment:fixed;
background-position:50%,50%.}
```

bu misolda fon rasmining nomi, “bg.gif”, u images papkasida joylashgan, fon rasmi takrorlanmaydi, fon tasmi Matn bilan birga harakatlanmaydi, fon rasmi ekranning oʻrtasiga joylashgan.

Css yordamida fon rasmini, ekranning ixtiyoriy joyiga qoʻyish, vertical va gorizontal yoʻnalish boʻyicha takrorlanishini taʼminlash mumkin. Quyidagi jadvalda, CSSda qoʻllaniladigan rang va fonning hususiyatini keltiramiz.

Hususiyat	Tushuntirish	Misol
color	Element ichidagi Matnning rangi	Strong{color:red;}
Background-color	Elementning fon rangi	H4{background-color:silver}
ackground-image	Fon rasmning URL adresini aniqlaydi	Body{background-image: url(bg10.gif) ;}
Background-repeat	Fon rasmning takrorlanish stilini beradi; Repeat repeat-x repeat-y no-repeat qiymatlardan birini oladi.	Body{background-repeat: no-repeat’}
Background-attachment	Fon rasmini element bilan birgalikda harakatlanishi yoki harakatlanmasligini oʻrnatadi Scroll yoki fixed qiymatlardan birini oladi	Body{background-attachment:scroll;}
Background-position	Fon rasmini boshlangʻich pozitsiyasini oʻrnatadi, prosentlarda, absolyut qiymatda, yoki kalit soʻzlar orqali beriladi. Top center bottom left right	Body{background-position: top center;}
background	Fonning hususiyatini ifodalovchi qisqacha yozuv	P{background: url(ches.png) repeat fixed 50% ;}

Shrift

CSS da shriftlarni boshqarish imkoniyati HTML ga qaraganda keng. Matnli bloklar uchun CSS qoidalaridan foydalanish HTML kodni hajmini anchagina kamaytiradi.

font-family hususiyati - shriftni garniturasini berish uchun ushlatiladi. Qabul qilishu mumkin boʻlgan qiymatlar:

- **Serif**
- **Sans-serif**
- **Monospace**
- **Cursive**
- **Fantasy** dekorativ shriftlar.

Shriftning hususiyatlari

Hususiyat	Tushuntirish	Misol
Font-family	Shriftlar oilasini belgilash	p{font-family:Helvetica,sans-serif;}
Font-style	Kursiv yoki ogʻma shriftni oʻrnatadi italic oblique normal	em{font-style:normal;}
Font-variant	Registrni oʻrnatadi small-caps normal	h3{font-variant:small-caps;}
Font-weight	Shriftni qalinlik darajasini belgilaydi. Bold normal 100 ...900	B{font-weight:700;}
Font -size	Shriftni oʻlchamini oʻrnatadi, bunda absolyut, nisbiy oʻlchamlar, % , xizmatchi soʻzlar ishlatilishi mumkin. Xx-smal, x-smal smal medium	h2 {font-size:200%;} h3 {font-size:36 pt;}

	large x-large xx-large	
--	----------------------------	--

Matn va uning hususiyatlari

Matnni hususiyati so'zlar va jumalarning o'zari joylashuvini aniqlaydi. Ya'ni Matnprotessori bilan taqqoslasak, abzayslar va uni ichini formatlash jarayonidagi ishlar bajariladi.

Text - decoration

Bu quyidagi qiymatlardan birini qabul qilishi mumkin:

- **none** — qo'shimcha bezash ishlatilmaudi;
- **underline** — takstni har bir qatorini tagiga chiziladi;
- **overline** - Matnni har bir qatorini ustiga chiziladi;
- **line-through** - Matnni har bir qatori chizib tashlangan holda aks etadi;
- **blink** — ekranda yonib,o'chib turadigan(мигающий текст) aks etadi.

Text-align

Bu hususiyat Matnning tekislash usulini o'rnatish imkonini beradi. mumkin bo'lgan qiymatlari:

- left** chap hoshiyaga tekislash;
- center** o'rta tekislash;
- right** o'ng hoshiyaga tekislash;
- justify** butun sahifa bo'ylab tekislash.

Text-indent

Bu hususiyat orqali abzatsning birinchi qatoridagi otstup beriladi. Matnli blokni ko'rsatishda ishlatiladi. birinchi galda abzats va sarlavhalar uchun qo'llaniladi.

text-transform

Bu hususiyat Matnni registrini almashtirishda qo'llaniladi.Quyidagi qiymatlardan birini qabul qiladi:

- capitalize** — har bir so'zning birinchi harfini registri o'zgaradi, yani bosh harfda yoziladi.
- uppercase** - hamma harflarning registri bosh harfga almashadi.
- lowercase** - hamma harflarning registri yzma harfga almashadi
- none** - harflarning registri almashmaydi.

Textning hususiyatlari

Hususiyat	Tushuntirish	Misol
text-decoration	none underline overline blink line-through	u {text-dekoration:underline ;} .old{text-decoration:line-through;}
text-align	center righr left justify	p{text-align:justify;} h4{text-align:center'}
text-indent	birinchi qatordagi otstup	p{text-indent: 5em;} h2{text-indent:25px;}
text-transform	capitalize uppercase lowercase none	h1{text-transform:uppercase;}
vertical-align	baseline sub super top text-top middle bottom text-bottom	img { vertical-align:top;}

Intervallar

Intervallar CSS vositasida beriladi..

word-spacing

Bu hususiyat so'zlar orasidagi masofani berish imkoniyatini yaratadi.Uzunlik o'lchov birliklaridan foydalaniladi.

letter-spacing

Bu hususiyat harflar orasidagi masofani berish imkoniyatini yaratadi. Uzunlik o'lchov birliklaridan foydalaniladi.

line-height

Bu hususiyat Matnning qatorlari orasidagi masofani berish imkoniyatini yaratadi. Uzunlik o'lchov birliklari va protsentsdan foydalaniladi.

Hususiyat	Tushuntirish	Misol
letter-spacing	harflar orasidagi interval	P{ letter-spacing:0,1em ;}
word-spacing	so'zlar orasidagi interval	p { word-spacing:0,5em;}
line-height	Matnqatorlari orasidagi interval	p{line-height: 18pt;} h2{line-height: 200%;}

2- Misol.

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
  <title>fon rasmini vertikal yo'nalishda takrorlash</title>
  <style type="text/css">
    BODY {
      padding-left: 100px; /* chap hoshiya */
      background:
        url(images/bg-green.png) repeat-y; /* faylga yo'l va fonni takrorlash */
    }
  </style>
</head>
<body>
  <p>Inson ba'zan o'z ustidan kula olishi kerak, aks holda u aqldan ozadi. Bu haqda ko'pchilik
  bilmaydi, shuning uchun ham dunyoda juda ko'p jinnilar mavjud.
  </p>
  <p>Rafael Sabatini</p>
</body>
</html>
```

Nazorat uchun savollar

1. Element ichidagi Matnning rangi qanday belgilanadi?
2. Element uchun fon rangi qanday belgilanadi?
3. Fon rasni va uning joylashtirish usullarini tushuntiring?
4. Shriftning hususiyatlarini berish usullarini tushuntiring?
5. Matnning hususiyatlarini berish usullarini tushuntiring?
6. Intervallarni berish usulini tushuntiring?

5- mavzu. Bloklarni formatlash

Maqsad: Blok va uni formatlash haqida tushuncha hosil qilish.

Kalit so'zlar: padding, border, margin, content, width, height, list-style-type

Asosiy savollar:

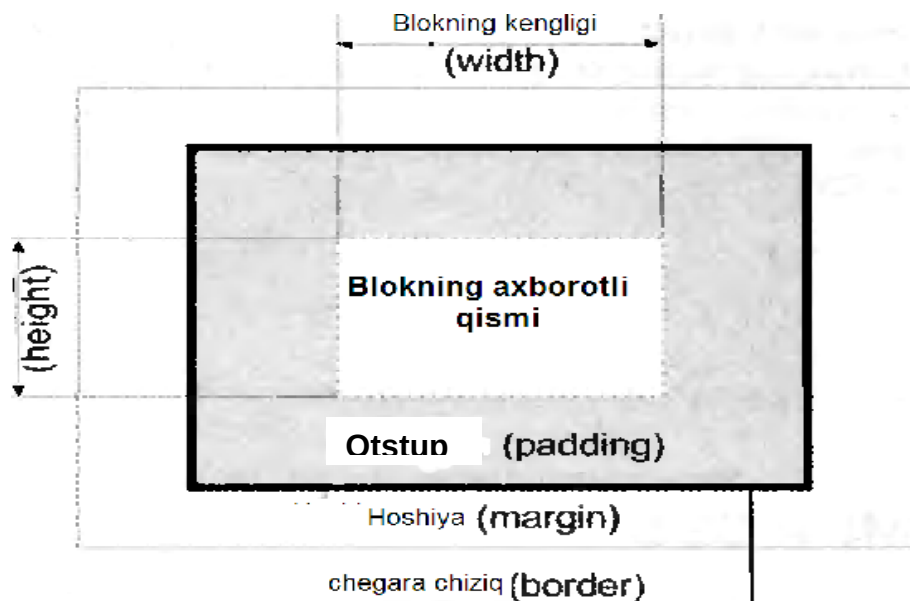
1. HTML va CSS da bloklar
2. Hoshiya (margin)
3. Chegara (border)
4. Otsup (padding)
5. Ro'yxatlar

HTML va CSS da bloklar

HTML ning hamma elementlarini ikkita tipga bo'lish mumkin: blok va qator tiplari. Blok tipidagi elementlar mustaqil ko'rgazmali strukturaga ega bo'ladi. Blok tipiga misol sifatida h1-h6, p, div kabi elementlarni olish mumkin. Qator tipidagi elementlar, chiziqli qator sifatida ko'rinadi. Masalan: I, B, U, S, EM va boshqalar. Ikkita blokli element bitta qatorda joylasha olmaydi.

CSS da hujjatni formatlashda ham blok tushunchasidan foydalaniladi. CSS da bloklar to'g'ri to'rtburchak ko'rinishiga ega bo'lgan mustaqil strukturaviy birlik hisoblanadi. CSSda bloklar struktura jixatidan olganda, bir-biridan ajratilgan. Qator ko'rinishidagi bloklar, strukturali blokni ichida joylashadi. Har bir qator ko'rinishidagi blok, axborotli qismga ega bo'ladi. bular tekst, rasm, yoki boshqa axborot bo'lishi mumkin. Blokning bu qismi content yoki hujjatning mazmuni deyiladi. Kontentning atrofidagi bo'sh soha hoshiya (**padding**) deyiladi. Dizayn nuqtaiy nazaridan hoshiya kontent uchun estetik ko'rinish beradi. Hoshiya esa chegara chizig'i (**border**) bilan qoplanadi. Bu chiziq turli stil va qalinlikga ega bo'ladi. Chegara chizig'i atrofidagi qo'shimcha bo'sh joy (**margin**) bo'lishi mumkin. CSS spetsifikatsiyasiga ko'ra, padding, border, margin, blokni kengligiga kirmaydi. Shunday qilib, blokni kengligini berganda, uni axborotli qismi nazarda tutiladi. Quyidagi rasmda blokni strukturasi keltirilgan,

Hamma hoshiya, chegaralar va otstuplar mos ravishda yuqori, quyi, chap va o'ng bo'laklarga bo'linadi. Har bir bo'lakga aloxida qiymat berish mumkin.



Hoshiya

Bu blokning chegarasi atrofidagi maxsus bo'sh soha. Hoshiyaning hususiyatlari:

- margin top yuqori hoshiyaning kengligi;
- margin right o'ng hoshiyaning kengligi;
- margin bottom quyi hoshiyaning kengligi;
- margin left chap hoshiyaning kengligi.

Qiymatlar ixtiyoriy uzunlik birliklarida yoki protsentlarda berilishi mumkin.

Agar sahifa kengligini 10% ni hoshiya uchun ishlatsak, yuqori va quyida hoshiya kerak bo'lmasa, quyidagicha stil yozamiz:

```
BODY {  
Margin top: 0px;  
Margin right: 10%;  
Margin bottom: 0px;  
Margin left: 10%; }
```

Qiymat sifatida xizmatchi so'z auto, dan ham foydalanish mumkin.

```
p {  
margin top: auto;  
margin right: auto;  
margin bottom: auto;  
margin left: auto; }
```

Chegara chiziqlari

Blokning chegara chiziqlari ham to'rtta segmentga bo'linadi. Har bir aloxida segment o'zining xarakteristikasiga ega: chiziqning rangi, qalinligi, tipi.

Border-top-width, border-right-width, border-bottom-width, border-left-width hususiyatlar chiziqning qalinligini aniqlaydi.

Qiymatlar quyidagi xizmatchi so'zlar orqali berilishi ham mumkin yoki uzunlik o'lchov birliklarida beriladi.

- thin ingichka chiziqli chegara;
- medium o'rtacha qalinlikdagi chegara;
- thick qalin chiziqli chegara.

Blok atrofidagi hamma chiziqlarni qalinligi bir xil bo'lsa:

```
P {  
border width: 1px }
```

Bu abzats atrofi 1 px qalinlikdagi ramka bilan ajratilgan.

```
DIV {  
border width: thin medium }
```

Bu misolda DIV konteynerning yuqori va quyi chegarasi ingichka(thin) va chap va o'ng chegarasi o'rtacha (medium) qalinlikdagi chiziq bilan qoplanadi.

Border-color

Bordercolor hususiyati chegara chiziqning rangini o'rnatadi:

- border-top-color yuqori segmentning rangi;
- border-right-color o'ng segmentning rangi;
- border-bottom-color quyi segmentning rangi;
- border-left-color chap segmentning rangi.

Bu hususiyatning qiymatlari quyida ko'rsatilgan uch xil usuldan biri orqali berilishi mumkin:

- xizmatchi so'zlar (red, yellow, silver);
- o'n oltilik kod (#00FFCC, #EEDDAA);
- RGB modelidagi o'nlik kod (color: rgb (25, 205, 172)).

Border-style

Bu hususiyat chiziqning tipini o'rnatadi. Agar to'rt tomondagi ramka uchun individual stil bermoqchi bo'lsak:

- border top style yuqori chegaradagi chiziqning stili;
- border right style o'ng chegaradagi chiziqning stili;
- border bottom style quyi chegaradagi chiziqning stili;
- border left style chap chegaradagi chiziqning stili.

Bu hususiyatlar quyidagi qiymatlarning birini qabul qiladi.

- none chiziqning qalinligi nolga teng.
- Hidden chiziq ekranda aks etmaydi.
- Dotted chegara chizig'i nuqtalardan iborat;
- Dashed chegara chizig'i punktir chiziq.
- Solid chegara chizig'i uzluksiz chiziq.
- Double chegara chizig'i ikkilangan uzluksiz chiziq.

Otstup (padding)

Otstuplar blokning mazmunini chegara chiziqlardan ajratish imkonini beradi.

Bu hususiyatni to'rt tomondan berish quyidagicha tarzda beriladi.

- padding top
- padding right .
- padding bottom .
- padding left

Bu hususiyatning qiymatlari ixtiyoriy uzunlik o'lchov birligi va protsentlarda berish mumkin.

Misol :

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
  <title>margin-bottom</title>
  <style type="text/css">
.layer1 {
  background: #008B66; /* fon rangi */
  color: white; /* Matnni rangi */
  padding: 10px; /* Matnatrofidagi hoshiya */
  margin-bottom: -7px; /* quyi hoshiya */
}
.layer2 {
  margin-left: 60px; /* chap hoshiya */
  background: #ccc; /* fon rangi */
  padding: 20px; /* Matnatrofidagi hoshiya */
  font: large; border-bottom-style: dashed; border-bottom-color: #FF0000;
border-bottom-width: thick; text-align: justify;
}
</style>
</head>
<body>
  <div class="layer1">
    <h2>Abu Rayhon Beruniy hikmatlari</h2>
  </div>
  <div class="layer2">
    Inson hayotidagi zarurat ,ehtiyotlar ularning ilmlarga
    bo'lgan talablarini keltirib chiqardi. Shu ehtiyojlarga ko'ra
    ilmlar har xil tarmoqlarga bo'lindi. Ilmning foydasi ochko'zlik
    bilan oltin-kumush to'plash uchun bo'lmay, balki u orqali inson
    uchun zarur narsalarga ega bo'lishdir.
  </div> </body>
</html>
```

natija quyidagi ko'rinishda bo'ladi:

Abu Rayhon Beruniy hikmatlari

Inson hayotidagi zarurat ,ehtiyotlar ularning ilmlarga bo'lgan talablarini keltirib chiqardi. Shu ehtiyojlarga ko'ra ilmlar har xil tarmoqlarga bo'lindi. Ilmning foydasi ochko'zlik bilan oltin-kumush to'plash uchun bo'lmay, balki u orqali inson uchun zarur narsalarga ega bo'lishdir.

Ro'yxatlar

HTML tilida avtomatik ravishda ro'yxat yaratishning ikki xil turi mavjud: nomerlangan va markerlangan. CSS ro'yxat yaratishning kengroq imkoniyati mavjud. Ro'yxatlar tiplarining hususiyatlarini ko'rib chiqamiz.

list-style-type

Bu hususiyat yordamida markerning tashqi ko'rinishi beriladi. Hususiyatning qiymati quyidagi xizmatchi so'zlar orqali beriladi:

- disk- marker doira shaklida;
- circle - marker aylana shaklida;
- square - marker kvadrat shaklida.

Nomerlangan ro'yxat yaratish uchun hususiyatning qiymatlari quyidagi xizmatchi so'zlar orqali beriladi.

- **decimal** 1 dan boshlanuvchi o'nli sonlar.
- **decimal leading zero** 0 dan boshlanuvchi o'nli sonlar (01,02,03, ...,98,99).
- **lower roman** lotin alifbosining yozma harflaridagi rim raqamlari (i, ii, iii, iv, v).
- **upper roman** lotin alifbosining bosma harflaridagi rim raqamlari (I,II, III IV,).
- **hebrew** yevrey raqamlari
- **georgian** gruzin raqamlari.
- **armenian** arman raqamlari.
- **lower greek** grek alifbosining harflari,

list style image

Bu hususiyat yordamida marker sifatida ixtiyoriy rasmni berish mumkin. Rasm grafik redaktorda tayyorlanadi va url adresi ko'rsatiladi.:

```
UL {  
list style image: url(marker.gif);  
}
```

misol keltiramiz

```
<html>  
<head>  
  <title>list-style-image</title>  
  <style type="text/css">  
UL {list-style-image: url('images/book1.gif');}  
LI {padding-left: 20px;}  
  </style>  
</head>  
<body>  
  <h2>Hotiraning turlari</h2>  
  <ul>  
    <li>Qattiq disk HDD</li>  
    <li>Tashqi hotira CD,DVD</li>  
    <li>flash</li>  
    <li>operativ hotira RAM</li>  
    <li>Doimiy hotira ROM</li>  
    <li>Kesh hotira</li>  
    <li>CMOS hotira</li>  
  </ul>  
</body>  
</html>
```

Hotiraning turlari

-  Qattiq disk HDD
-  Tashqi hotira CD,DVD
-  flash
-  operativ hotira RAM
-  Doimiy hotira ROM
-  Kesh hotira
-  CMOS hotira

list-style position

Bu hususiyat markerning joylashgan o'rini ko'rsatadi. Uning qiymati quyidagi xizmatchi so'zlar orqali beriladi:

- outside marker asosiy blokning tashqarisida joylashadi;
- inside marker asosiy blokning ichida joylashadi.

Misol:

```
<html>
```

```

<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
<title>list-style-position</title>
<style type="text/css">
  UL {
    list-style-type:square; list-style-position: inside;
  }
div.bir{background-color:#99FFCC }
div.ikki{background-color:#CC99FF}
ol{list-style-type: lower-latin; list-style-
position:outside;}
</style>
</head> <body>
<div class="bir"> Web -sahifa yaratish uchun
zaruriy tillar
<ul>
<li>HTML</li>
<li>CSS</li>
<li>XML</li>
<li>PHP</li>
</ul> </div>
<p>
<div class="ikki"> Web -sahifa yaratish uchun zaruriy tillar
<ol>
<li>HTML</li>
<li>CSS</li>
<li>XML</li>
<li>PHP</li>
</ol> </div> </p>
</body>
</html>

```

Web -sahifa yaratish uchun zaruriy tillar

- HTML
- CSS
- XML
- PHP

Web -sahifa yaratish uchun zaruriy tillar

1. HTML
2. CSS
3. XML
4. PHP

Ro'yxatlar , unung hususiyatlari va qabul qilishi mumkin bo'lgan qiymatlarni quyidagi jadvalda keltiramiz:

Hususiyat	Tushuntirish	Misol
List-style-type	Markerning tashqi ko'rinishi Disc circle square	UL{list-style-type:square;}
	Decimal decimal-leading-zero Lower-roman upper-roman Lower-greek lower-alpha Lower-latin upper-alpha Upper-latin	OL{list-style-type: lower-roman}
List-style-image	Rasmli markerning url adresini aniqlaydi.	UL{list-style-image:url(bullet.gif);}
List-style-position	Blokning ichida yoki tashqarisida joylashish o'rni Inside outside	LI {list-style-position:outside }

Nazorat uchun savollar

1. Blok nima va uning qanday hususiyatlari bor?
2. Blok uchun hoshiya qanday o'rnatiladi?
3. Blok uchun chegara chiziqlari qanday o'rnatiladi?

4. Padding nima va qanday oʻrnatiladi.
5. CSS yordamida roʻyxat yaratishni qanday usullari mavjud?

Uyga vazifa.

Shunday stil yaratingki, hujjat brayzer oynasida quyidagi tarzda aks etsin:

Html tilida formaning atributlari

1. ACTION
2. METHOD
3. ENCTYPE

Input elementi qoʻllaniladigan tiplar

- I. TYPE=text
- II. TYPE=password
- III. TYPE=radio
- IV. TYPE=checkbox
- V. TYPE=hidden
- VI. TYPE=submit
- VII. TYPE=reset

Adabiyotlar:

1. Мальчук, Е.В. HTML и CSS. Самоучитель. М. Издательский дом "Вильямс", 2008.
2. Михаил Дубаков ВЕБ-МАСТЕРИНГ Средствами CSS .Санкт-Петербург, 2002
3. Eric A. Meyer, Cascading Style Sheets. The Definitive Guide. O'Reilly 2008

6 – mavzu. Bloklarni joylashtirish usullari

Maqsad: Bloklarni web sahifada joylashtirish haqida tushuncha hosil qilish.

Kalit soʻzlar: position, static, relative, absolute, fixed, float, clear, level.

Asosiy savollar

1. Bloklarni joylashtirish
2. Normal oqim
3. Pozitsiyaning xususiyatlari
4. Fiksirlangan bloklar
5. Suzuvchi bloklar
6. CSS da qatlamlar

Bloklarni joylashtirish

Blok yaratilib, uning xususiyati berilgach, brauzer oynasida joylashish oʻrnini aniqlash kerak. Umumiy holda blokning oʻrni bir nechta parametrlarga bogʻliq. Asosiy rolni esa blokni joylashtirish sxemasi oʻynaydi. Blokni joylashtirish sxemasining toʻrt turi mavjud.

1. Normal oqim
2. Nisbiy pozitsiya.
3. Absolyut pozitsiya.
4. Suzuvchi bloklar modeli

Bundan tashqari hujjatning kodida blokning joylashgan oʻrni ham muhim roʻl oʻynaydi. Monitor ekranining oʻlchami va koʻrsatish qobiliyati ham ahamiyatli. Koʻpchilik web-dizaynerlar minimal koʻrsatish qobiliyatni 800x600 deb olib, ishlarini shunga moslaydilar.

Normal oqim

Normal oqim boʻyicha blokni joylashtirish sxemasi oddiy. Hamma struktura yoki qator tipidagi bloklar vertical yoʻnalish boʻyicha yuqoridan pastga qarab ketma-ket joylashaveradi va shu tarzda monitor ekranida aks etadi. Bunda ikkita strukturali blok hech qachon bir gorizontal

chiziqda joylasha olmaydi. Qator tipli bloklarni brauzer gorizontal yo‘nalish bo‘yicha ketma-ket joylashtiradi. Bunday qator chiziqli blok deyiladi, va bir nechta qator tipli bloklar bitta chiziqli blokda joylashishi mumkin. Agar joy yetmasa bertikal bo‘yicha keyingi qatorga o‘tadi. Normal oqim o‘z hususiyatiga ko‘ra HTML kod qanday tartibda yozilsa, shi tartibda ish bajaradi. Agar jadvaldan foydalanilmasa, normal oqim yordamida, murakkab sahifalar tayyorlash mumkin emas.

Kengligi 80%, qalinligi 1 px va yashil ramka chiziq bilan o‘ralgan, hamda sahifaning o‘rtasida joylashadigan asosiy blok yaratamiz.

```
#main {  
border:1px green solid;  
margin:0 auto;  
width:80%; }
```

HTML kodida esa quyidagicha konstruktsiya yoziladi:

```
<body>  
<div id= "main">  
</div>  
</body>
```

Hamma qolgan bloklar asosiy blok “main” ning ishiga joylashadi. Birinchi shunday blok sarlavha ya’ni saytning nomi bo‘ladi. Bu blokni head, deb nomlaymiz. Bu yerdagi Matnkattaroq razmerli shriftda yoziladi. Bundan tashqari sarlavha uchun toq kulrang fon, Matnning rangini oq qilib belgilaymiz. Sarlavhani o‘rtaga joylashtiramiz.

```
#head {  
background:#A5A5A5;  
color: #FFF;  
font size: 2,5em;  
text-align: center; }
```

Endi sahifaning HTML kodi quyidagi ko‘rinishni oladi:

```
<body>  
<div id="main">  
<div id="head">  
Ivan Kuprinning On-line kundaligi.  
</div>  
</div>  
</body>
```

Keyingi blok links ochroq fonda beriladi. Bu yerda o‘rtada saytning boshqa bo‘limlariga ssilkalar joylashadi.

```
#links {  
background:#EDED;ED;  
padding: 2px 0;  
text-align:center }
```

Ssilkalarni bir biridan ajratish uchun uzluksiz probel va vertical chiziqlar qo‘yilgan. Bundan tashqari ssilkalar teksti qalin shriftda va tagiga chizilmagan holda aks etsin. Buning uchun konMatnselektorga quyidagicha qoida yozamiz.

```
#links A {  
color:#000;  
font weight: bold;  
text decoration: none; }
```

Blokning HTML kodi quyidagi ko‘rinishni oladi:

```
<body>  
<div id="main">  
<div id="head">
```

Ivan Kuprinning On-line kundaligi

```

</div>
<div id="links">
<a href="#">kundalik</A>&nbsp;&nbsp;&nbsp;&||&nbsp;&nbsp;&nbsp;&
<a href="#">fotolar</A>&nbsp;&nbsp;&nbsp;&||&nbsp;&nbsp;&nbsp;&
<a href="#">dil izhorlari kitobi</a>
</div>
</div>
</body>

```

Ivan Kuprinning On-line kundaligi

kundalik || fotolar || dil izhorlari kitobi

Sahifaning davomiga yangiliklar uchun blok joylasgadi. Chap va o'ng tomondan 0.5 em hoshiya qolsin.

```
#news { padding:0px 0.5em; }
```

Pozition

Position hususiyati quyidagi to'rtta qiymatlardan birini qabul qilishi mumkin.

1. static bu oddiy blok bo'lib, normal oqim qoidalari bo'yicha joylashadi.
2. relative normal oqimga nisbatan nisbiy joylashish.
3. absolute absolyut joylashish.
4. fixed fiksirlangan joylashish. Blok avval absolyut joylashadi, yani normal oqimdan ajralib chiqib, alohida joylashadi, keyin uning shu joylashish holati fiksirlanadi. Fiksirlangan blok sahifa ko'rib chiqish jarayonida surilganida, o'z holatini o'zgartirmaydi.

Nisbiy pozitsiyalash(relative)

Bu usuldan foydalanganda, avval blokning holati normal oqimga nisbatan aniqlanadi, so'ngra blok o'z joyiga nisbatan suriladi. Surilish quyidagi hususiyatlar yordamida beriladi:

- left blokni chap chegarasi, konteynerni chap chegarasiga nisbatan suriladi;
- right blokni o'ng chegarasi, konteynerni o'ng chegarasiga nisbatan suriladi;
- top blokni yuqori chegarasi, konteynerni yuqori chegarasiga nisbatan suriladi;
- bottom blokni quyi chegarasi, konteynerni quyi chegarasiga nisbatan suriladi.

Qiymatlar uzunlik o'lchov birligida yoki protsentlarda beriladi.

Oddiy misol ko'ramiz. Sahifaga uchta blokni joylashtiramiz.

```

div{ border: 1px dotted #000; }
<div>Birinci blok</div><br>
<div>Ikkinchi blok</div> <br>
<div>Uchinchi blok</div>

```

Normal oqim qoidasiga ko'ra bloklar quyidagi rasmdagidek akas etadi.

```

-----
Birinci blok
-----
-----
Ikkinchi blok
-----
-----
Uchinchi blok
-----

```

Bloklardan birini chapga 20 px, yuqoridan pastga 10 px suramiz. Buning uchun "move" nomli klas yaratamiz:

```
.move{ position: relative; left: 20px; top: 10px; }
```

```

<div> Birinci blok</div><br />
<div class="move"> Ikkinchi blok</div><br />
<div> Uchinchi blok</div>

```

Klasni ikkinchi blok uchun qo'llasak, quyidagi natija chiqadi:

Birinchi blok

Ikkinchi blok

Uchinchi blok

Amalda nisbiy pozitsiyalash kam isjlatiladi.

Absolyut pozitsiyalash

Absolyut pozitsiyalashda blok normal oqimdan butunlay ajralib chiqadi. Bunda ham bloklarni surish quyidagi qiymatlardan birini oladi:

- left
- right ;
- top;
- bot tom.

Bizning misolimizda ikkinchi blok uchu konteyner vazifasini brauzer oynasi o'taydi. Shuning uchun surishni oynaga nisbatan beramiz. Suriladigan blok uchun "abs" deb identifikator belgilaymiz.

```
#abs{ position: absolute; left: 60px; top: 120px; }
```

```
<div> Birinchi blok</div><br />
```

```
<div id="abs"> Ikkinchi blok</div><br />
```

```
<div> Uchinchi blok</div>
```

Natija quyidagi rasmda tasvirlangan:

Birinchi blok

Uchinchi blok



Ikkinchi blok

Fiksirlangan bloklar

Fiksirlangan blok hech qayerga surilmaydi va har doim o'rnatilgan sohada o'zgarishsiz turadi.

Aniq misol ko'ramiz, ikkita blok yaratamiz. Birinchi blokni fiksirlab, brauzerning chap yuqori oynasiga joylashtiramiz. Bu blokga menu punktlarini joylashtiramiz. Blokning kengligini 100 piksel bilan cheklaymiz.

```
#fix{ position: fixed; left: 0px; top: 0px; width: 100px; }
```

Ikkinchi blok absolyut joylashgan, u birinchi blok bilan ustma-ust tushib qolmasligi uchun, o'ng tomonga 150 piksel surilgan.

```
#move{ position: absolute; left: 150px; top: 0px; }
```

Endi bu bloklardan sahifa yaratamiz.

```
<div id="fix">
```

```
1 punkt<br>
```

```
2 punkt<br>
```

```
3 punkt<br>
```

```
4 punkt </div>
```

```
<div id="move">
```

```
Sahifaning asosiy teksti<br>
```

```
Sahifaning asosiy teksti<br>
```

```
Sahifaning asosiy teksti<br>
```

```
</div>
```

1 punkt
2 punkt
3 punkt
4 punkt

Sahifaning asosiy teksti
Sahifaning asosiy teksti
Sahifaning asosiy teksti

Suzuvchi blok

Float hususiyati elementni sahifaning chap yoki o'ng chetiga siqib olib keladi. Matnesa uning atrofidan aylanib o'tadi. Matnning bunday harakati toshni yonidan oqib o'tuvchi suvni eslatadi, shuning uchun ham bunday hususiyatli bloklar, suzuvchi blok deyiladi. Absolyut pozitsiyadan farqli o'laroq, suzuvchi bloklar aniq koordinataga ega bo'lmaydi. Ularni aniq berilgan nuqtaga joylashtirish mumkin emas, lekin ular o'ziga xos muhim xarakterga ega.

Float qo'llaniladigan ishlarni sanab o'tamiz:

- Fotografiyalar galereyasi;
- Ikki va uch ustunli(kolonka)maketlar;
- Gorizontal menyu;
- Matnni mazmunini yorituvchi rasmlar;
- Ko'p ustunli tekst.

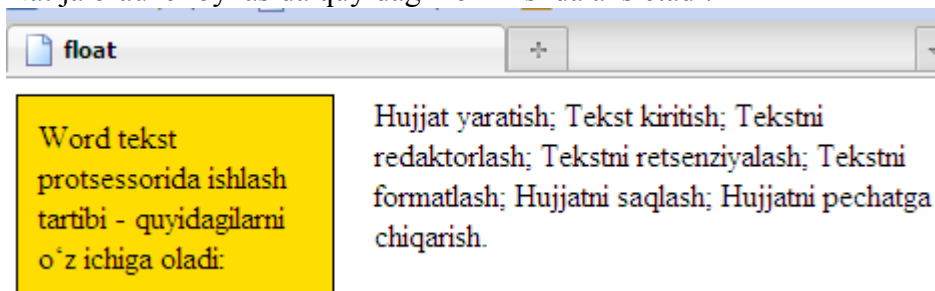
Float quyidagi qiymatlardan birini qabul qilishi mumkin.

- **left** strukturali blok chapga suriladi, hujjatning qolgan qismi esa blokning o'ng tomoniga o'tadi.
- **right** strukturali blok o'ng tomonga suriladi, hujjatning qolgan qismi esa blokning chap tomoniga o'tadi.
- **none** blok surilmaydi, yani positionda ko'rsatilgan algoritm bo'yicha joylashadi. Agar position hususiyati ko'rsatilmasa, normal oqim holiday bo'ladi.
- **Inherit** o'zi joylashgan konteynerni qiymatini oladi.

Misol keltiramiz:

```
<html>
<head>
  <title>float</title>
  <style type="text/css">
    .comment { float: left; background: #fd0; border: 1px solid black; padding: 10px;
    margin-right: 20px; /* ong hoshiya */
    width: 30%; /* blokni kengligi */
  }
</style>
</head>
<body>
  <div class="comment">
    Word Matn protsessorida ishlash tartibi - quyidagilarni o'z ichiga oladi:
  </div>
  <div>
    Hujjat yaratish; Matnkiritish; Matnni redaktorlash; Matnni retsenziyalash;
    Matnni formatlash; Hujjatni saqlash; Hujjatni pechatga chiqarish.
  </div>
</body>
</html>
```

Natija brauzer oynasida quyidagi ko'rinishda aks etadi:



Clear

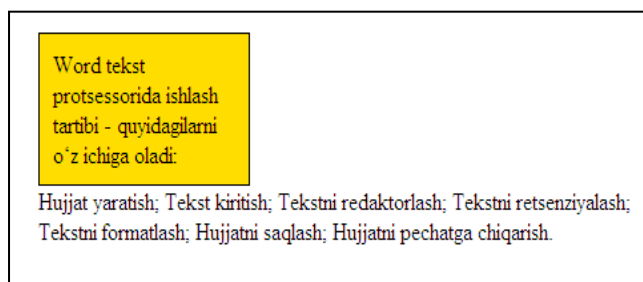
Bu xususiyat suriluvchi bloklarni qaysi tomoni , boshqa suriluvchi blok bilan yonma – yon joylasha olmasligini aniqlatdi.U faqat blok darajasidagi elementlar uchun qo‘llaniladi va quyidagi qiymatlardan birini qabul qiladi.

- **left** blok hamma chap tomondagi suzuvchi bloklardan keyin joylashishi kerak.
- **right** blok hamma o‘ng tomondagi suzuvchi bloklardan keyin joylashishi kerak .
- **both** blok hamma suzuvchi bloklardan keyin joylashishi kerak .
- **none** blokni joylashuviga cheklanish qo‘yilmaydi.
- **inherit** o‘zi joylashgan konteynerni qiymatini oladi.

Bu xususiyatni ixtiyoriy strukturali bloklarga qolash mumkin.

Misol keltiramiz:

```
<head>
<title>float</title>
<style type="text/css">
.comment {
float: left;
background: #fd0;
border: 1px solid black;
padding: 10px;
margin-right: 20px; /* ong hoshiya */
width: 30%; /* blokni kengligi */
}
```



```
</style>
```

```
</head>
```

```
<body>
```

```
<div class="comment">
```

Word Matn protsessorida ishlash tartibi - quyidagilarni o‘z ichiga oladi:

```
</div>
```

```
<div style="clear:left">
```

Hujjat yaratish; Matnkiritish; Matnni redaktorlash; Matnni retsenziyalash;

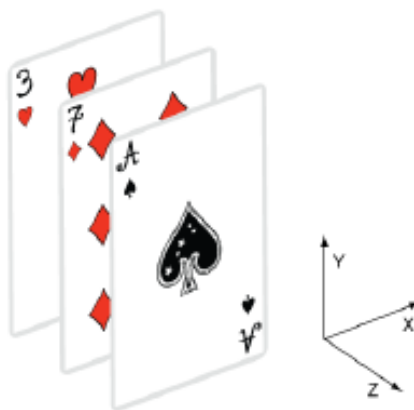
Matnni formatlash;Hujjatni saqlash; Hujjatni pechatga chiqarish.

```
</div>
```

```
</body>
```

CSS da qatlamlar

Agar web sahifani uch o‘lchovli fazo deb faraz qilsak, bu fazoda ustma –ust joylashgan sahifalar Z o‘qi bo‘yicha tartib bilan joylashadi. Hujjatning kodida esa tartib avtomatik tarzda aniqlanadi. Normal oqim bo‘yicha elementning kodi qancha quyida bo‘lsa, u Z o‘qi bo‘yicha shuncha yuqori qiymatni qabul qiladi.



CSS da Z- o‘qi bo‘yicha joylashuvga z-index xususiyati javob berradi,ya’ni element bizga yaqin yoki yzoqligini aniqlaydi. Qiymat sifatida butun sonlar beriladi, qiymat qancha katta

son bo'lsa shu element boshqalarga nisbatan yuqorida joylashadi. Web –sahifaning o'zi uchun z-index 0 ga teng.

Z-index hususiyatni, faqatgina positionga absolute, fixed yoki relative qiymatlar berilgan elementlar uchun qo'llash mumkin.

Nazorat uchun savollar

1. Bloklarni joylashtirish haqida ma'lumot bering.
2. Suzuvchi bloklar?
3. Css da qatlamlar?

Adabiyotlar

1. Мальчук, Е.В. HTML и CSS. Самоучитель. М. Издательский дом "Вильямс", 2008.
2. Михаил Дубаков ВЕБ-МАСТЕРИНГ Средствами CSS .Санкт-Петербург, 2002
3. Eric A. Meyer, Cascading Style Sheets. The Definitive Guide.O' Reilli 2008

7- mavzu.Jadval uchun display hususiyatning qiymatlari.

Maqsad: Jadval uchun display hususiyatining qiymatlari haqida tushuncha hosil qilish.

Kalit so'zlar: display, inline, table, table-row, yacheyka.

Asosiy savollar

1. Jadvalni formatlash
2. Jadvalni visual tashkil qilish
3. Jadval yaratish qoidalari
4. Jadval uchun display hususiyatning qiymatlari

Jadvallarni formatlash

Jadvalni o'lchami va uni yacheykalarini ramkasini chizishdan oldin, uni fundamental taxrirlash va jadvalni elementlarini bir-biri bilan bog'lanishini diqqat bilan o'rganish kerak. Bu jarayon jadvalni formatlash (table formatting) deb ataladi va jadvalni tahrirlash jarayonidan farq qiladi, yani jadvalni tahrirlash faqat jadvalni formatlashdan keyin amalga oshiriladi.

Jadvalni visual tashkil qilish.

Birinchi navbatda, CSS jadvalni qanday tashkil qilishini aniqlashini tushunish kerak. Bu tushunchalar sodda bo'lishiga qaramay, ular jadvalni uslubiy bezatishda hal qiluvchi o'ringa ega. CSS jadvalni elementlari va uning ichki elementlarini farqlaydi. CSS da ichki elementlar bo'shliq va ramkani o'z ichiga o'luvchi ammo chegaraga ega bo'lmagan to'g'ri burchak bloklarni generatsiya qiladi. Shundan kelib chiqqan holda maydon chegaralarini o'rnatish jarayonida yacheykalar orasidagi masofani aniqlab bo'lmaydi. CSS ga mos keladigan brauzerlar yacheykalarga, qatorlarga yoki jadvalning boshqa elementiga maydon qo'llash harakatlarini barchasini inkor etadi.

Jadval yaratish qoidalari.

Har bir qator blogi yacheyka to'ringining bir qatorini o'z ichiga oladi. Jadvalning har bir qatorining bloklari uni dastlabki hujjatda joylashishiga qarab yuqoridan pastga bo'lgan tartibda to'ldiriladi. Shunday qilib jadval o'z ichida qancha qator elementlari bo'lsa, shuncha qatorga ega bo'ladi.

1. Qator guruhining bloki o'zini shakllantiruvchi qator bloklari qancha to'r yacheyklarini egallasa, shuncha to'r yacheyklarini egallaydi.
3. Ustun bloki bir yoki bir nechta to'r yacheykasining ustunlarini o'z ichiga oladi. Ustunlarning barcha bloklari ko'rsatilgan tartibda ketma ket joylashadi. Ustunning birinchi bloki yozuvi chap tomondan o'nga qarab boshlanadigan tillar uchun chapda va aksincha bo'lgan tillar uchun o'ngda joylashadi.
4. Ustun guruhining bloki o'zini shakllantiruvchi ustun bloklari qancha to'r yacheyklarini egallasa, shuncha to'r yacheyklarini egallaydi.

5. Yacheykalar bir nechta qator yoki ustunlarni birlashtirishi mumkun, ammo CSS da ushbu jarayon qanday kechishi nomalum. Bu jarayon uchun hujjatning tili javob beradi. Har bir birlashtirilgan yacheyka- bu balandligi va eni bir yoki birnechta to`r yacheykasiga to`g`ri keladigan to`g`ri burchak blo`kdir. Ushbu to`g`riburchakni tepa qatori yacheykaning asosiy qatorida joylashgan. Yacheykaning to`g`riburchagi chap tomondan o`nga qarab yoziladigan tillarda imkon qadar chaproq joylashmog`i lozim, ammo yacheykalarning boshqa blo`klarini berkitma olmaydi. Hamda chap tomondan o`nga qarab yoziladigan tillarda u barcha, shu qatorda va oldin dastlabki hujjatda uchratilgan yacheykalardan o`ngroq joylashmog`i lozim. O`ngdan chap tomonga yoziladigan tillarda birlashtirilgan yacheyka imkon qadar o`ngroq, boshqa yacheykalarni berkitmagan holda joylashmog`i lozim va shu qatorda va undan oldin dastlabki hujjatda uchratilgan yacheykalardan chaproq joylashmog`i lozim.
6. Yacheykaning bloki jadval yoki qatorlar guruhining oxirgi qatorining blo`ki chegarasidan chiqma olmaydi. Agar jadvalning tuzilmasi buni shartlasa, yacheyka shunday qisqartirilishi lozimki, u o`zi joylashgan jadval yoki qatorlar guruhiga sig`ishi kerak.

Jadval uchun display hususiyatning qiymatlari

display	
Qiymat:	none inline block inline-block list-item run-in table inline-table table-row-group table-header-group table-footer-group table-row table-column-group table-column table-cell table-caption inherit
Boshlang`ich qiymat:	inline
Ishlatish sohasi:	Hamma elementlar

HTML da qaysi elementlar jadvalga tegishli ekanligini tushinich oson, chunki browserlarga tr va td elementlarini aniqlash funksiyasi o`rnatilgan. XML da esa aksincha qaysi elementlar jadval elementlari bo`lishi mumkinligini aniqlash mumkin emas. Bu yerda display xossasining qiymatlari asosiy o`ringa chiqadi.

Bu mavzuda biz jadvallarga tegishli qiymatlar bilan tanishamiz. Jadvalga tegishli qiymatlarni quyidagicha xarakterlash mumkin:

table

By qiymat element blok darajasidagi jadvalni aniqlashini ko`rsatadi. Shunday qilib u blok elementi kabi aniqlanuvchi to`g`rito`rtburchak blokni aniqlaydi. Unga mos HTML element bu – table.

Inline-table

By qiymat element satr darajasidagi jadvalni aniqlashini ko`rsatadi. Demak, element satr kabi aniqlanuvchi to`g`rito`rtburchak blokni aniqlaydi Unga yaqin element – inline – block qiymati.yaqin HTML element – table, lekin HTML- jadval satr elementi bo`lmaydi.

Table-row

Bu qiymat element yacheykalar satri bo`lishini aniqlaydi. Unga mos HTML – element – tr

Table-row-group

Bu qiymat element bir yoki bir nechta satrni birlashtirishini aniqlaydi. Unga mos HTML – element – tbody.

Table-header-group

Bu qiymat table-row-group qiymatiga o`xshaydi, faqat jadvalni vizual formatlaganda jadval nomlari guruhi, har doym hamma asosiy nomlardan keyin, qolgan hamma satrlardan oldin aks ettadi. Nom gruppasi bir nechta satrni tashkil qilishi mumkin. Unga mos HTML- - element thead.

Table-footer-group

Бу қиймат table-header-group қийматига мос, фақат пастги сатр қиймати қолган сатр ва сатрлар гуруҳидан кейин ва бошқа ҳар қандай пастги асосий сарлавҳадан кейин акс этади. Агар жадвални чоп қилиш учун бир нечта вароқ керак бўлса, чоп этишда фойдаланувчи агенти куйи сатр сарлавҳасини ҳар бир бет охирида такрорланши мумкин. Таснифлаш table-footer-group қийматни бир нечта элемент учун қўллаганда қандай ҳол рўй беришини аниқламаган. Unga mos HTML--element tfoot.

Nazorat uchun savollar

1. Jadvallarni formatlash.
2. Jadval yaratish qoidalari?
3. Display xususiyatining qiymatlari?

Adabiyotlar

1. Мальчук, Е.В. HTML и CSS. Самоучитель. М. Издательский дом "Вильямс", 2008.
2. Михаил Дубаков ВЕБ-МАСТЕРИНГ Средствами CSS .Санкт-Петербург, 2002
3. Eric A. Meyer, Cascading Style Sheets. The Definitive Guide. O'Reilly 2008

8 – mavzu. XML asoslari

Maqsad: XML ning maqsad va imkoniyatlarini tushuntirish

Kalit so'zlar: XML, SGML, Prolog, XHTML

Asosiy savollar:

- SGML standarti;
- XML hujjatning strukturasi;
- XML tilining kontseptual maqsadi;
- XML-hujjatni aks ettrish

SGML standarti

1986 yili SGML (Standard Generalized Markup Language), ishlab chiqildi. SGMLning muhim xususiyati shundaki, u yangi belgilashlar tilini yaratish imkonini beradi. SGML qoidalari bo'yicha ishlab chiqilgan har bir teglar guruhi DTD (Document Type Definition)-hujjatning tipini ta'rifida, teglar orasida o'zaro bog'lanish va ularni qo'llash qoidalari tushuntiriladi. SGML juda murakkab til bo'lib, uning bitta hususiyatnini o'zi beshyuz sahifadan ko'p.

XML tili

SGML va HTML orasidagi oltin qoidani ifodalovchi til, XML (extensible Markup Language) — kengaytirilgan belgilashlar tilidir. Bu til SGML dan ajratib olingan to'plam bo'lib, web- programmistga o'z teglarini yaratish imkonini beradi va ortiqcha murakkablikdan holi.

XML ni ishlab chiqish 1996 boshlandi. Bu ish bilan jamoat tashkiloti W3C(World Wide Web Consortium), shug'ullanadi. Uning asosiy sayti : <http://www.w3c.org/>. XML haqidagi ma'lumotlar <http://www.w3c.org/XML/> sahifaga joylashtirilgan 1998 bu tashkilot XML 1.0. versiyasini chiqardi. XML tobora takomillashib bormoqda. Uning oxirgi versiyasi quyidagi adresda bo'ladi: <http://www.w3c.org/TR/rec-xml>.

XML taraqqiyotining kelguvsidagi yo'nalishlari:

XFragment

XML hujjatning alohida qismlariga ma'no berishga urinish.

XML Schemas

XML alifbosi bo'yicha, XML hujjatning mazmunini ifodalash urinish.

XHTML

HTML hujjatni , XML qoidalariga asoslanib yaratishga urinish.

XML Query Language (XML zaproslar tili)

Hujjatda biror shartni qanoatlantiruvchi elementlarni izlashga urinish

Canonical XML

XML hujjatdagi elementning argumentlari oladigan qiymatlarni ko'rsatishni standart algoritmi.

XML Signatures

XML hujjatda raqamli podpisni ifodalashni standart vositalari

XML o'zini turli texnologiyalar uchun ishonchli, qat'iy asos ekanini ko'rsatdi.

XML hujjatning strukturasi

XML da tayyor teglar yo'q, balki o'z teglarini yozish qoidalar mavjud.

Aniq teglar haqida gapirganda XML ni amalda qo'llash nazarda tutiladi

- XML qoidalarini amalda qo'llab, ma'lum teglar guruhi tuzilgan, ular Aniq sohada ishlatiladi;
- XHTML — XML qoidalariga mos ravishda tuzilgan giperMatnni belgilash tili;
- Math ML — matematik formulalarni yozish tili;
- CML — ximik formulalarni yozish tili;
- Vox ML — ovozlarni yozish tili;
- WML —simsiz texnologiyada ishlatiladigan til.

XML hujjatni bezatishni va teglar yaratishni faqatgina umumiy qoidalarini beradi. Bu qoidalar XMLning hususiyatida ko'rsatiladi. Hujjat prolog keltirishdan boshlanadi.

XML hujjatning prologi

Prolog ikki qismdan iborat bo'ladi. Prologning birinchi qismida XML e'lon yoziladi. (XML Declaration). U `<?...?>` simvollar orasiga yoziladi, bu e'londa xml belgi va uning versiyasi keltiriladi. U quyidagi ko'rinishga ega bo'ladi:

`<?xml version="1.0"?>`

Bundan tashqari e'londa hujjat yozilgan tilning kodi ham keltirilishi mumkin

`<?xml version="1.0" encoding="utf-8"?>`

Kiril alfaviti ishlatilgan hujjatlar uchun, e'lonning ko'rinishi quyidagicha bo'ladi:

`<?xml version="1.0" encoding="windows-1251"?>`

E'londa yana standalone parametrini ham keltirilishi mumkin. Bu parametr ikki qiymatdan birini oladi: "yes" yoki "no". "no" qiymati shuni ko'rsatadiki, bu hujjarda ishlatilgan elementlarni ta'riflash uchun tashqi hujjatdan foydalanilgan. Odatda standalone "yes" qiymatini oladi. Demal XML hujjatni e'lon qilish quyidagi ko'rinishda bo'ladi:

`<?xml version="1.0" encoding="windows-1251" standalone="yes"?>`

Prologni ikkinchi qismida — hujjatning tipi e'lon qilinadi, DTD (Document Type Declaiatbn). Bu qismda hujjatda foydalanilgan teglar yoki alohida faylga ssilka bo'ladi.Hujjatni tipini ko'rsatish `<! DDCTYPE`, so'zi bilan boshlanadi va uchburchakli qavs belgisi `>` bilan tugaydi.

XML tilining kontseptual maqsadi

Quyidagi web-saytda xml tilininng vazifasi va amalda qo'llash maqsadi tushuntiriladi.
<http://www.w3.org/TR/REC-xml>.

1. XML Internetda bevosita foydalaniladigan til bo'lishi kerak.

XML asosan web- muhitida axborotni saqlash va tarqatis uchun ishlab chiqilgan til.

2. XML ko'p miqdordagi programmalar aro axborot almashishda qo'llaniladi

XML ni asosiy vazifasi serverlar va brauzerlar orqali axborot tarqatish bolsa ham, undan boshqa programmalarda ham foydalaniladi. Masalan finans programmalar aro axborot almashinuvi, programma maxsulotlarini yangilash va tarqatish.

3. XML, SGML bilan mos keladi.

XML aslida SGMLning maxsus shoxchasi hisoblanadi.

4. **XML-hujjatlarni qayta ishlovchi programmalarni yozish oson.**
5. **XMLning qo'shimcha funktsiyalarini miqdori minimal, idealda nolga teng bo'lsin.**
Qo'shimcha funktsiyalarni minimal bo'lishi xml –hujjatni qayta ishlovchi programmani yozishni soddalashtiradi
6. **XML-hujjatlar foydalanuvchi uchun tushunarli va ochiq- ravshan bo'lishi kerak.**
XML butun dunyodagi programmalar va foydalanuvchilar orasida axborot almashinuvi uchun universal til (lingua franca) boladi. Foydalanuvchilar XML- hujjatni osonlik bilan o'qiydilar, chunki u oddiy Matnsifatida ifodalanadi va daraxt ko'rinishidagi iyerarxik strukturaga ega.
7. **XML ni ishlab chiqish yetarli darajada tez tugallanishi kerak.**
8. **XML tili formal va qisqa bo'lishi kerak.**
9. **XML-hujjatlarni yaratish oson ish bo'lsin.**
10. **XML-hujjatlar siqilgan formatda bo'lsa , hajmi minimal qiymatni oladi.**

XML-hujjatni aks ettrish

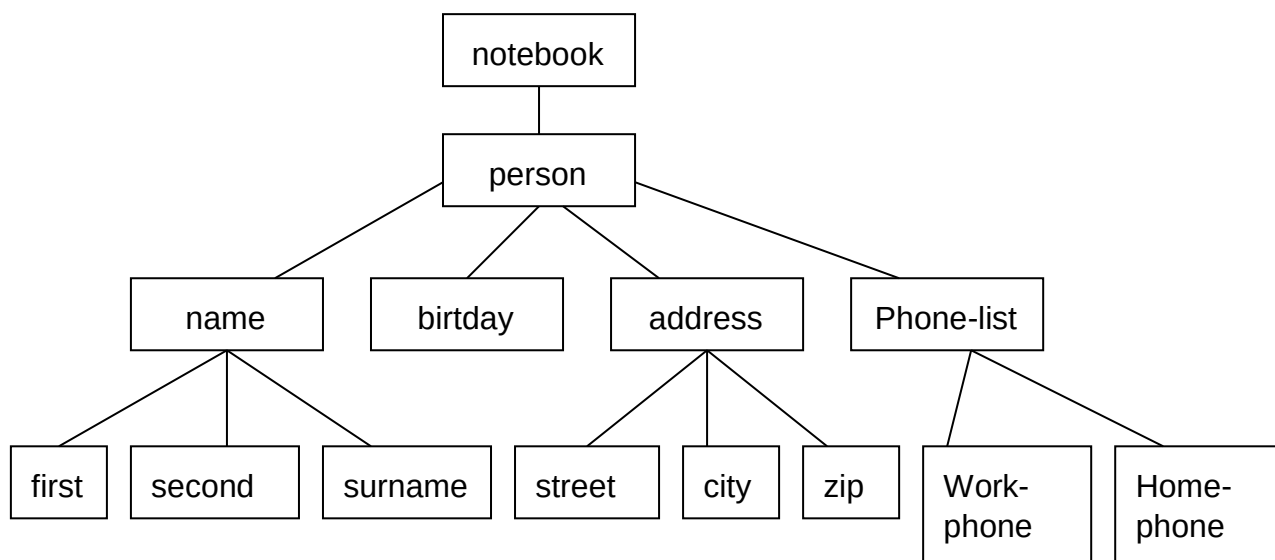
Brauzerda (xususan, Microsoft Internet Explorer 5) XML- hujjatni aks ettirishni, uch xil usuli mavjud:

- **Kaskad jadvallar stili.** Bu usuldan foydalanganda stil yozilgan fayl XML- hujjat bilan bog'lanadi. Kaskad jadvallar stili alohida fayl bo'lib, bu faylda XML –elementlarning har birini formatlash uchun ko'rsatma bo'ladi. Formatlash stili (Cascading Style Sheet – CSS), yordamida tayyorlanadi yoki (Extensible Stylesheet Language – XSL) dan, foydalaniladi. XSLT(Extensible Stylesheet Language Transform) aynan XML-hujjatni qayta ishlash uchun, maxsus programmalash tili.
- **Berilgan ma'lumotlarni bog'lash.** Bu usuldan foydalanganda, HTML-hujjat tayyorlanadi va XML –hujjat bilan bog'lanadi.
- **Senariy yozish.** Bu usuldan foydalanganda, HTML-hujjat tayyorlanadi , XML –hujjat bilan bog'lanadi va XML –elementlar uchun maxsus senariy yoziladi(JavaScript yoki Microsoft Visual Basic Scripting Edition [VBScript]). Brauzer XML-hujjatni ob'yekt sifatida idrok etadi (Document Object Model – DOM). DOM , ob'yektlar, hususiyatlar va buyruqlar to'plamidan iborat.

Adreslar kitobchasini belgilash misolini ko'ramiz:

XML tilida avvalo yaratilayotgan hujjatning strukturasi tuzishni o'ylash kerak. Biz o'z tanish bilishlarimiz haqida ma'lumot yozilgan adreslar kitobchasini tartibga solmoqchimiz. Bu kitobda tanishlarni familiyasi, ismi, yashash joyi, telefoni, tug'ilgan kuni yozilgan bo'lsin. Biz har bir elementni ko'rsatish uchun teglat tuzamiz va shu teglar orasidagi bog'lanishni aniqlaymiz.

Adreslar kitobchasini strukturasi quyidagi sxemada keltiramiz:



Yuqoridagi strukrura bo'yicha tuzilgan XML hujjat quyidagi ko'rinishni oladi:

Listing 1.1 XML hujjat

```
<?xml version="1.0" encoding="Windows-1251"?>
<?xml-stylesheet type="text/css" href="t1_1.css"?>
<notebook>
  <person>
    <name>
      <first-name>Aziz</first-name>
      <second-name>Alievich</ second-name>
      <surname>Akbarov</ surname>
    </name>
    <birthday>25.03.1977</birthday>
    <address>
      <street>Yunusobod, 23-15</street>
      <city>Toshkent </city>
      <zip>100055<zip>
    </address>
    <phone-list>
      <work-phone>1235431</work-phone>
      <home-phone>4567843</home-phone>
    </phone-list>
  </person>
  <person>
    <name>
      <first-name>Aziza</first-name>
      <second-name>Alievna</ second-name>
      <surname>Akbarova</ surname>
    </name>
    <birthday>29.04.1978</birthday>
    <address>
      <street>Yunusobod, 24-16</street>
      <city>Toshkent </city>
      <zip>100155<zip>
    </address>
    <phone-list>
      <work-phone>8973543</work-phone>
      <home-phone>4667944</home-phone>
    </phone-list>
  </person>
</notebook>
```

Bu hujjatni brauzerda aks ettirish uchun quyidagicha stil yozilgan fayl tayyorlaymiz va bu faylni nomi prologdan keyingi qatorda ko'rsatiladi. Bizni misolimizda bu faylni nomi t1_1.css.

```
/* File Name: t1_1.css */  
  
person,name {display:block; margin-top:12pt; font-size:14pt}  
  
first-name,second-name,surname {display:block; font-size:12pt; font-weight:bold;  
  
font-style:italic}  
  
birthday {display:block; margin-left:15pt; font-weight:bold}  
  
address ,phone-list {display:block; margin-left:15pt}  
  
steet,sity,zip {display:block; font-size:12pt; font-weight:bold; font-style:italic}  
  
work-phone,home-phone {display:block; margin-left:15pt}
```

Nazorat uchun savollar:

- XML tili nima maqsadda yaratildi?
- XML qanday vazifalarni bajarishga qodir?
- XML hujjarning strukturasi nimalardan tashkil topadi?

Uy ishi:

Shunday XML hujjar yaratingki. Unda sizning ismingiz, uy adresingiz, elektron pochtangiz, tug'ilgan kuningiz ko'rsatilsin. Bu hujjatni brauzerda aks ettirish uchun tashqi stil, yani .css kengaytmali fayl ham yozing.

Adabiyotlar

1. Хабибуллин И. Самоучитель XML. Санкт-Петербург. 2003.
2. Шапошников И.В. Справочник Web –мастера XML СПб: БХВ-Петербург 2001.-304 с
3. Гарольд Э., Минс С. XML. Справочник. – Пер. с англ. – СПб: Символ-Плюс, 2002. – 576 с.

9- mavzu. XML nomlar

Maqsad: XML hujjatning element va atributlarini nomlashni tushuntirish

Kalit so'zlar: Doctype, Element, DTD

Asosiy savollar:

1. Element va atributlarni nomlash
2. XML hujjat yaratish uchun asosiy qoidalar

Element va atributlarni nomlash

XML elementning nomlari alifboning simvollari va raqamlardan iborat bo'lishi mumkin. Bu simvollar lotin alifbosining A dan Z gacha va a dan z gacha, harflaridan, shuningdek 0 dan 9 gacha raqamlardan tuziladi. Ularda uchta punktuatsiya simvollari uchrashi mumkin:

_ tagiga chizish;
- defies;
. niqta.

Boshqa punktuatsiya simvollar , shuningdek probel ishlatilmaydi. Ikki nuqta nomlar fazosini ko'rsatishda ishlatiladi.

XML nomlar faqat harfdan boshlanadi. Nomni uzunligiga cheklash qo'yilmagan. Quyidagi elementlarni hammasi to'g'ri yozilgan:

- <Drivers_License_Number>98 NY 32</Drivers_License_Number>
- <month-day-year>7/23/2001</month-day-year>
- <first_name>Алан</first_name>
- <telephone>011 33 91 55 27 55 27</telephone>

Quyidagi elementlarda xatolik bor:

- <Driver's_License_Number>98 NY 32</Driver's_License_Number>
- <month/day/year>7/23/2001</month/day/year>
- <first name>Алан</first name>

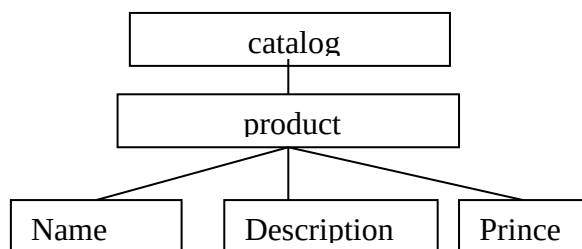
XML hujjat yaratish uchun asosiy qoidalar

Formatlangan XML hujjat yaratish uchun quyidagi asosiy qoidalarga rioya qilish kerak:

- Hujjatda faqat bitta yuqori qatlamdagi element (o'zak element) bolishi kerak. Qolgan hamma elementlar shu elementni ichida joylashadi.
- Agar element bosjqa elementning ichida boshlansa, shu elementni ichida yopilishi kerak..
- Har bir element ochuvchi va yopuvchi tegga ega bo'lishi kerak.
- Element tipining nomi ochuvchi va yopuvchi teglar uchun bir xil bo'lishi kerak.
- Elementni tipini yozganda registrga e'tibor berish kerak. Masalan quyidagi misolda boshlangich va oxirgi teglar bir –biriga mos kelmaydi:

<TITLE>Leaves of grass</Title> <!--noto'g'ri element -->

Tovarlar katalogi haqida xml hujjat va uni brauzerda aks ettirish stilini keltiramiz:



?xml version="1.0" encoding="utf-8" standalone="yes"?>

<?xml-stylesheet type="text/css" href="t2_2.css"?>

<!DOCTYPE catalog [<!ELEMENT catalog (product+)>

```

<!ELEMENT product (name, description?, price+)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT description (#PCDATA)>
<!ELEMENT price (#PCDATA)>
]>
<product>
<name>Fonar</name>
<description>Yorug'lik doimo siz bilan birga!
</description>
<price>$9.95</price>

<name>Neon lampa</name>
<description>Kunduzi xonani yoritish vositasi!
</description>
<price>$14.75</price>

<name>Elektr lampa</name>
<description>O'zidan yorug'lik va issiqlik chiqaradi!
</description>
<price>$4.25</price>
</product>
</catalog>

```

```

/* File Name: t2_2.css */
product
{display:block;
margin-top:12pt;
font-size:14pt}

name
{display:block;
font-weight:bold;
font-style:italic;
font:xx-large;
color:#0033CC;
}

description

```

O'zak element

Prologdan keyin o'zak element (root element) yoziladi. Bizning misolimizda o'zak element <catalog> bilan boshlanadi va </catalog> tegi bilan tugaydi. O'zak elementning nomi butun hujjatning nomi hisoblanadi va prologning ikkinchi qismida ko'rsatiladi. Hujjatning tipini e'lon qilish DTD (Document Type Declaration)-DOCTYPE so'zidan keyin hujjatning nomi va kvadrat qavsni ichida hujjatning tipini aniqlash DTD(Document Type Definition) yozaladi:

<! DOCTYPE catalog [hujjatning tipini aniqlash DTD]>

Ko'pincha hujjatni tipini aniqlovchi yozuvlar alohida fayl sifatida yoziladi. Bu holda prologni ikkinchi qismida kvadrat qavsni o'rniga SYSTEM yoki PUBLIC so'zlaridan biri yoziladi, SYSTEM so'zidan keyin DTD faylni URL adresi ko'rsatiladi. PUBLIC, so'zini ishlatganda esa faylni adresidan tashqari qo'shimcha axborot ham yoziladi.

DTD(Document Type Declaration) – hujjatning tipini e'lon qilish

DTD (Document Type Definition)- hujjatning tipini ta'riflash

To'g'ri yaratilgan hujjat

Biz tovarlar katalogini yozishda <catalog>, <product>, <name>, <description>, <price> teglardan foydalandik. Endi ularni e'lon qilish kerak. E'londa logic bog'lanishlarni umumiy alomatlari va ularni tipi ko'rsatiladi. Bunday o'zaro bog'lanishi ko'rsatilgan hujjat, to'g'ri hujjat bo'ladi. Hujjatni analiz qiluvchi programma, teglarni e'lon qilish bo'limidagi teglarni, hujjatdagi o'zaro bog'liq teglar bilan taqqoslaydi.

Yuqoridagi misolda teglarni e'lon qilish quyidagicha bo'lishi mumkin:

- <catalog> elementining ichida, nol yoki ko'plab <product> elementlar bo'lishi mumkin, boshqa hech narsa bo'lmaydi;
- <product> elementining ichida, bitta <name> elementi, nol yoki bir nechta <description> elementi, shuningdek bir nechta <price> elementi mavjud;
- <name> va <description> elementlari bittadan Matnli qatordan tashkil topadi;

} alohida
tushunchalar
ekaniga e'tibor
bering

- <price> elementi bitta Matnli qatordan tashkil topadi;

Bu soʻz bilan ifodalash, XML hujjatning sxemasi deyiladi. Sxema tuzishning ikki xil usuli mavjud:

1. DTD(Document Type Definition)
2. XSD(XML Schema Definition Language)

XML hujjatning elementlari

XML hujjat elementlardan(elements) tashkil topadi. Element uchburchakli qavsni ichidagi ochuvchi teg(start-tag) bilan boshlanadi, keyin elementning mazmuni (content) keladi, va undan keyin uchburchakli qavsni ichida yopuvchi teg (end-tag) keladi. Masalan:

<surname>Alieva</surname>

Agar elementdan keyin mazmun (content) keltirilmasa bu boʻsh element deyiladi:

</br>

Elementlar bir-birining ichiga joylashadi, lekin keshishmaydi. Toʻgʻri tuzilgan hujjatda, barcha tashkil qiluvchi elementlar, oʻzak elementning ichiga joylashadi. Hamma elementlar odatda quyi registrda yoziladi.

Atributlar

XML elementning atributlari boʻlishi mumkin. Atributning nomi qiymatidan tehglik belgisi bilan ajratiladi, qiymatlar qoʻshtirnoq ichga yoziladi. Atributlar elementning mazmunini kengroq yoritishga xizmat qiladi. Masalan:

1- misol

<person born="1912/06/23" died="1954/06/07">

Alan Turing

</person>

2-misol

<?xml version="1.0" encoding="utf-8"?>

<!DOCTYPE person [

<!ELEMENT first_name (#PCDATA)>

<!ELEMENT last_name (#PCDATA)>

<!ELEMENT profession (#PCDATA)>

<!ELEMENT name (first_name, last_name)>

<!ELEMENT person (name, profession*)>]>

<person>

<name>

<first_name>Alan</first_name>

<last_name>Turing</last_name>

</name>

<profession>Informatika mutahasisi</profession>

<profession>matematik</profession>

<profession>kriptogراف</profession>

</person>

Nazorat uchun savollar:

Uyga vazifa

Haftalik dars jadvalini ifodalovchi XML hujjat va unga mos stil yarating.

Adabiyotlar

1. Хабибуллин И. Самоучитель XML. Санкт-Петербург. 2003.
2. Шапошников И.В. Справочник Web –мастера XML СПб: БХВ-Петербург 2001.- 304 с

3. Гарольд Э., Минс С. XML. Справочник. – Пер. с англ. – СПб: Символ-Плюс, 2002. – 576 с.

10 - mavzu. Elementlarni e'lon qilish. Hujjatning tipini aniqlash.

Maqsad: XML hujjatning tipini aniqlashni tushuntirish

Kalit so'zlar: <! ELEMENT>, #PCDATA, ANY

Asosiy savollar:

1. Hujjatning tipini aniqlash
2. Elementlarni e'lon qilish

(Document Type Definition, DTD). Formal sintaksisga mos ravishda tuziladi. DTD hujjatda qanday elementlar va mohiyatlar mavjudligini, element va uning atributlarini mazmuni aniqlaydi. XML hujjatni qayta ishlovchi programma, DTD ko'rsatmalaridan foydalanib, hujjatni haqiqiy ekanini tekshiradi. Hujjatni haqiqiy ekanini tekshirish- XML hujjatni qayta ishlashda zaruriy bosqich emas.

Haqiqiy ekanini tekshirish

Agar haqiqiy (valid) hujjat bo'lsa, unda hujjatning tipi e'lon qilinadi. Bu e'londa qanday DTD berilgan hujjatni qano'atlantirishi ko'rsatiladi. DTD da hujjatda uchraydigan hamma elementlar, atributlar va mohiyatlar sanab o'tiladi va ular foydalanadigan konMatnham ko'rsatiladi. DTD mavjud va u qano'atlantiriladigan hujjat, haqiqiy hujjat hisoblanadi. Agar hujjat DTD ni qano'atlantirmasa u haqiqiy emas hisoblanadi.

DTD da ko'rsatilmaydigan muxim xususiyatlar ham mavjud:

- Hujjatning o'zak elementi qanday;
- Hujjatda har bir tipli element necha marta uchraydi;
- Element ichidagi simvolli ma'lumotlar qanday shaklda ko'rinadi;
- Elementning semantic qiymati qanday (sana yoki insonning nomidan iboratmi);

Hamma XML-hujjatlar uchun to'g'ri ekanligi zaruriy, lekin haqiqiy ekanini zaruriy emas.

Hujjatning tipini e'lon qilish

Agar DTD tashqi fayl sifatida olinsa, haqiqiy hujjatda shu faylga ssilka bo'ladi. Bu ssilka hujjatni tipini e'lon qilish bo'limida joylashadi. Bu e'lon taxminan quyidagi ko'rinishda bo'ladi:

```
<!DOCTYPE person SYSTEM "http://biblio.org/xml/dtds/person.dtd">
```

Bu hujjatning o'zak elementi person va DTD ning URL adresi,

<http://biblio.org/xml/dtds/person.dtd>.

"person" nomli haqiqiy hujjatga misol.

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE person SYSTEM "http://biblio.org/xml/dtds/person.dtd">
<person>
  <name>
    <first_name>Alan</first_name>
    <last_name>Turing</last_name>
  </name>
  <profession>informatika bo'yicha mutaxassis</profession>
  <profession>matematik</profession>
  <profession>kriptograf</profession>
</person>
```

Agar hujjat va DTD bitta saytda joylashgan bo'lsa, absolyut adresni o'rniga nisbiy adres ishlatish mumkin:

```
<!DOCTYPE person SYSTEM "/dtds/person.dtd">
```

Agar hujjat va DTD bitta katalogda joylashgan bo'lsa :

```
<!DOCTYPE person SYSTEM "person.dtd">
```

DTD ni ichki to'plam sifatida yozish

Boshlang'ich bosqichda DTD va hujjatni o'zini bitta faylda saqlash qulay, chunki ularni bir vaqtda baravar tekshirish va o'zgartirish mumkin. Hujjatni tipini e'lon qilganda o'zak elementdan keyin kvadrat qavsni ishiga DTD yoziladi.

“person” nomli haqiqiy hujjatga 2 misol.

```
<?xml version="1.0"?>
<!DOCTYPE person [
  <!ELEMENT first_name (#PCDATA)>
  <!ELEMENT last_name (#PCDATA)>
  <!ELEMENT profession (#PCDATA)>
  <!ELEMENT name (first_name, last_name)>
  <!ELEMENT person (name, profession*)>]>
<person>
  <name>
    <first_name>Alan</first_name>
    <last_name>Turing</last_name>
  </name>
  <profession>Informatika mutahasisi</profession>
  <profession>matematik</profession>
  <profession>kriptogrraf</profession>
</person>
```

Agar hujjatda tashqi DTD dan foydalanilsa, standalone atributi “no” qiymatni oladi:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
```

Elementlarni e'lon qilish

Elementni e'lon qilish quyidagi tartibda amalga oshadi:

```
<! ELEMENT elementning nomi (element mazmunining_modeli)>
```

#PCDATA

Bu element mazmunining_modelini eng sodda ko'rinishi. Agar biror tipdagi ichki elementlar mavjud bo'lmasa va elementning mazmuni simvolli ma'lumotlardan iborat bo'lsa, hizmatchi so'z, #PCDATA dan foydalaniladi.

Ichki elementlar

Agar elementning ishida bitta ichki element mavjud bo'lsa, elementning mazmuni quyidagicha yoziladi: <! ELEMENT fax (phone_number)>

Ketma-ketlik

Amalda esa bir nechta ichki elementlar mavjud bo'ladi, ular ketma –ket tarzda, vergul bilan ajratib yoziladi.

```
<! ELEMENT name (first_name, last_name)>
<name>
  <first_name>Мадонна</first_name>
  <last_name>Чикконе</last_name>
</name>
```

Ichki elementlar miqdori.

Ichki elementlar miqdorini ko'rsatishda quyidagi uch xil suffikslardan foydalanish mumkin:

? nol yoki bitta element mavjud

*** nol yoki bit nechta element mavjud**

+ bir yoki bit nechta element mavjud

Variantlar.

Variantlar – bu vertical chiziq bilan ajratilgan elementlar ro'yxatining nomi. Masalan quyidagi elonda ko'rsatilgan element method uchun ichki element faqat bitta bo'ladi, yoki params, yoki fault.

```
<! ELEMENT method(params | fault)>
```

Digit element uchun 10 tadan faqat bitta ichki element mavjud:

```
<! ELEMENT digit
```

(zero | one | two | three | four | five | six | seven | eight | nine)

>

Point elementini ko'rsatishda yoki dekart koordinatalari yoki qutb koordinatasidan foydalaniladi:

<! ELEMENT point ((x, y) | (r, α))>

Aralash elementni ko'rsatishga misol:

<! ELEMENT paragraph

(#PCDATA | name | profession | footnote | emphasize | date)* >

Bo'sh elementlarni ko'rsatish:

<image source="bus.jpg" width="152" height="345"/>

Bo'sh elementni ko'rsatishda xizmatchi so'z EMPTY dan foydalaniladi:

<! ELEMENT image EMPTY>

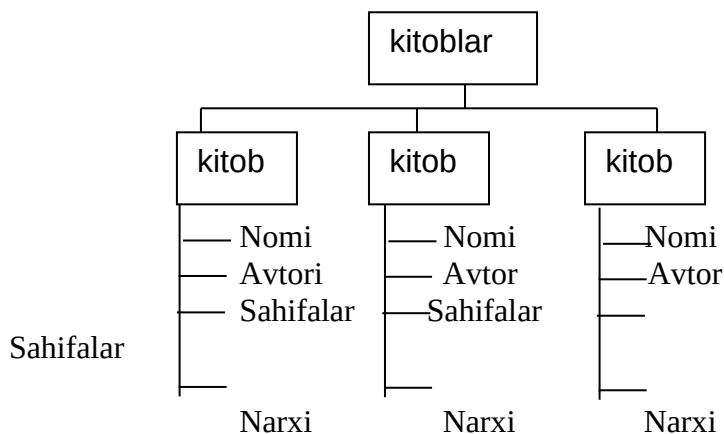
ANY

Agar elementda ixtiyoriy tipdagi ichki elementlar mavjud bo'lsa, uni ifodalashda ANY xizmatchi so'zidan foydalaniladi:

<!ELEMENT page ANY>

Kitoblar katalogini tuzishga doir misol ko'ramiz:

Kitoblar katalogining strukturasi



Kitoblar katalogini ifodalovchi XML hujjat

```
<?xml version="1.0" encoding="utf-8"
standalone="yes"?>
<?xml-stylesheet type="text/css" href="t3_3.css"?>
<!DOCTYPE books[
<!ELEMENT books (book)*>
<!ELEMENT book (name,authors,pages?,price+)>
<!ELEMENT authors (author+)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT author (#PCDATA)>
<!ELEMENT pages (#PCDATA)>
<!ELEMENT price (#PCDATA)>
<!ATTLIST price currency CDATA #REQUIRED>]>
<books>
<book>
<name>Informatika</name>
<authors>
<author>S.V.Simonovich</author>
</authors>
```

```
/* File Name: t3_3.css
*/

book{display:block;
margin-top: 12pt;
margin-left:15pt;}

name{display:block;
margin-left:15pt;
margin-top: 12pt;
font-style:italic;
font-size:18pt;}

author{display:block;
margin-left:15pt;
font-family:Arial,
Helvetica, sans-serif;
color:#0066CC;}

pages{ display:block;
margin-left:15pt;
font-variant:small-caps}

price{display:block;
font-weight:800}
```



```

<pages>650 varaq</pages>
<price currency="rub">40.25 rubl</price>
</book>
<book>
<name>Oliy matematika</name>
<authors>
<author>S.A.Alimov</author>
<author>M.T.Sobirov</author>
<author>B.A.Ayupov</author>
</authors>
<pages>750 varaq</pages>
<price currency="sum">40000.00 so'm</price>
</book>
</books>

```

Bu hujjat uchun stil:

Nazorat uchun savollar:

- Hujjatning tipi qanday tartibda e'lon qilinadi?
- Ichki elementlarning miqdori qanday ko'rsatiladi?

Uyga vazifa:

“Oziq-ovqat do'koni” mavzusi bo'yicha struktura tuzing va shu strukturani ifodalovchi XML hujjat yarating.

11- mavzu.XML atributlarni e'lon qilish

Maqsad: XML hujjatning atributlarini e'lon qilishni tushuntirish.

Kalit so'zlar: ATTLIST, CDATA, NMTOKEN, ENTITY, ID, IDREF, IDREFS, NOTATION.

Asosiy savollar:

- 1 Atributlarni e'lon qilish
- 2 Atributlarning tiplari

Atributlarni e'lon qilish

Atributlarni e'lon qilishda ATTLIST xizmatchi so'zidan foydalaniladi. Atributlar elementning mazmunini to'liqroq ochib berishga yordam beradi.

Bir xil tipdagi elementlar uchun bitta ATTLIST bilan bir nechta atributlarni e'lon qilish mumkin. Agar bir nechta elementlar faqat bir xil atributga ega bo'lsa, u har bir element uchun aloxida e'lon qilinadi.

Bu e'londa image elementi uchun, source atributini qiymati ko'rsatilgan:

```
<! ATTLIST image source CDATA #REQUIRED>
```

Bitta ATTLIST e'londa, bitta elementning, bir nechta atributlarini qiymati ko'rsatilishi mumkin:

```

<! ATTLIST image source CDATA #REQUIRED
width CDATA #REQUIRED
height CDATA #REQUIRED
alt CDATA #IMPLIED

```

>

Bu elonda source, width va height atributlar zaruriy, alt atributi zaruriy emasligi e'lon qilinadi.

Atributlarning tiplari

XML da 10 xil tipdagi atributlar mavjud:

- CDATA
- NMTOKEN
- NMTOKENS
- Sanab o'tish
- ENTITY

- ENTITIES
- ID
- IDREF
- IDREFS
- NOTATION

CDATA. CDATA atributning qiymati ixtiyoriy Matnli qatordan iborat bo'ladi. Bu atributning eng ko'p uchraydigan umumiy tipi

NMTOKEN. Bu qiymat elementar nomi (name token) degan mazmunni anglatadi, bir yoki bir nechta harf, raqam, nuqta, tire kabi simvollaridan tashkil topadi. Jurnal chiqarilgan yillar butun sonlarda beriladi, uni e'lon qilishda NMTOKEN tipi bilan berish mumkin.

<! ATTLIST journal year NMTOKEN #REQUIRED>

NMTOKENS

Bu atributni qiymati bir nechta elementar nomlardan tashkil topadi, qo'shtirnoq ichidagi nomlar probel bilan ajratiladi. Masalan

<! ELEMENT SHIRT (#PCDATA)>

<! ATTUST SHIRT Codes NMTOKENS #REQUIRED>

<SHIRT Codes="38 21 97">long sleeve Henley</SHIRT>

<performances dates="08-21-2001 08-23-2001 08-27-2001">

Kat and The Kings </performances>

Sanab o'tish

Bu tipdan foydalanganda xizmatchi so'z ishlatilmaydi. Atributni qabul qilishi mumkin bo'lgan qiymatlari ketma- ket sanab o'tiladi, ular vertical chiziq bilan bir-biridan ajratiladi.

Misol:

<! ATTLIST date month (yanvar | fevral | mart | aprel | may | iyun | iyul | avgust | sentyabr | oktyabr | noyabr | dekabr) #REQUIRED >

<! ATTLIST date day (1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31) #REQUIRED>

<! ATTLIST date year (1970 | 1971 | 1972 | 1973 | 1974 | 1975 | 1976

| 1977 | 1978 | 1979 | 1980 | 1981 | 1982 | 1983 | 1984 | 1985 | 1986

| 1987 | 1988 | 1989 | 1990 | 1991 | 1992 | 1993 | 1994 | 1995 | 1996

| 1997 | 1998 | 1999 | 2000 | 2001 | 2002 | 2003 | 2004 | 2005 | 2006

| 2007 | 2008 | 2009) #REQUIRED>

<! ELEMENT date EMPTY>

<date month="yanvar" day="22" year="2001"/>

ID

ID tipli atribut XML hujjat doirasida unikal nomdan iborat bo'lishi kerak. Elementda ID tipli atribut bittadan ortiq bo'lmaydi.

<! ATTLIST employee social_security_number ID #REQUIRED>

Sonlar XML nom bo'lmaydi, shuning uchun ID ga sonli qiymat berganda, shu sonning oldiga qabul qilingan simvollaridan biri qo'yiladi. Masalan:

<employee social_security_number="_123-45-6789"/>

IDREF

IDREF tipli atribut, elementning ID- atributiga ssilka sifatida ishlatiladi

IDREFS

IDREFS tipli atribut probel bilan ajratilgan xml nomlar ro'yhati bo'lib, ularni har biri ID identifikator bo'lishi kerak.

ENTITY

ENTITIES tipli atribut bitta analiz qilinmaydigan mohiyatlardan tashkil topadi

<! ATTLIST movie source ENTITY #REQUIRED>

ENTITIES

ENTITIES tipli atribut bir nechta analiz qilinmaydigan mohiyatlardan tashkil topadi, bu mohiyatlar bir-biridan probel bilan ajratiladi.

```
<! ATTLIST slide_show slides ENTITIES #REQUIRED>  
<slide_show slides="slide1 slide2 slide3 slide4 slide5 slide6 slide7 slide8 slide9 slide10"/>
```

NOTATION

NOTATION tipli atribut notatsiya nomlardan tashkil topadi. Bu atribut amalda eng kam qo'llaniladi. Quyidagi e'londa to'rtta notatsiya turli tipdagi tasvirlar uchun ko'rsatilgan.

```
<! NOTATION gif SYSTEM "image/gif">  
<! NOTATION tiff SYSTEM "image/tiff">  
<! NOTATION jpeg SYSTEM "image/jpeg">  
<! NOTATION png SYSTEM "image/png">
```

```
<! ATTLIST image type NOTATION (gif | tiff | jpeg | png) #REQUIRED>
```

Har bir image elementdagi, type atributi to'rtta qiymatgan birini olishi mumkin – gif, tiff, jpeg yoki png, va boshqa qiymatni olmaydi

Atributlarning qiymati

Har bir e'londa atributning tipi va qabul qilishi mumkin bo'lgan qiymati ham ko'rsatiladi. Qiymatni ko'rsatishda to'rtta imkoniyat mavjud:

#IMPLIED

Atribut zaruriy emasligini bildiradi. Bu atributga qiymat berish ham mumkin, bermaslik ham mumkin.

#REQUIRED

Atribut zaruriy ekanini bildiradi. Bu holda har bir element uchun atributni qiymatini berish kerak.

#AttValue

AttValue – atributni odatdagi qiymati.

#FIXED

Atributning qiymadi doimiy va o'zgarmaydi.

Nazorat uchun savollar:

1. Atributlarni e'lon qilish qanday tartibda amalgam oshadi?
2. Atributlarnig qanday tiplari mavjud?

Uyga vazifa: Atributlarni ishlatishga misol yozib keling.

Adabiyotlar

1. Хабибуллин И. Самоучитель XML. Санкт-Петербург. 2003.
2. Шапошников И.В. Справочник Web –мастера XML СПб: БХВ-Петербург 2001.-304 с
3. Гарольд Э., Минс С. XML. Справочник. – Пер. с англ. – СПб: Символ-Плюс, 2002. – 576 с.

12 mavzu. XSLT shakl almashtirish tili

Maqsad: XSLT tilining maqsad va imkoniyatlarini tushuntirish

Kalit so'zlar: XSLT, xsl: stylesheet, xsl: template, match, select

Asosiy savollar:

1. XSLT nima?
2. Shablonlar

XSLT nima?

XSLT (eXtensible Stylesheet Language Transformations) – kengaytiriladigan stillar varaqlarini shakl almashtirish tili.

XSLT tili berilgan Matnni erkin ravishda shakl almashtirishga yordam beruvchi translyatoridir. Ma'lumotlarni saqlash va masofaga uzatish uchun mo'ljallangan universal til XML ni tasdiqlashda, XSLT hal qiluvchi ahamiyatga ega. Amalda XSLT ni qo'llash doirasi juda keng -electron kommertsiyadan boshlab, simsiz Web gacha.

Berilgan XML hujjat va unga mos yozilgan XSLT stil, sintetik XSLT analizatorga (ya'ni XSLT-protessor) yuborilgach, natijaviy hujjat to'plami yuzaga keladi.

Web muhitida XSLT dan foydalanganda, sintetik analiz jarayoni yoki foydalanuvchining brauzerida yoki server tomonida yuz beradi. Kliyent tomonidagi analiz, HTML uchun kaskad jadvallar stilini qo'llash ishiga o'xshaydi. Berilgan hujjatga *xsl fayl ilova qilinadi:

```
<? xml-stylesheet type="text/xsl" href="transform.xsl" ?>
```

Bu yerda transform.xsl – XSLT stil yozilgan faylni nomi.

Shablonlar

Har bir XSLT element **xsl:** so'zi bilan boshlanadi. **xsl: stylesheet** element esa XSLT stil uchun konteyner vazifasini o'taydi.

XSLT shakl almashtirish shablonlarga asoslanib bajariladi. Shablon esa xsl: template

ko'rsatmalar majmuasidan iborat.

XSLT-protessor berilgan hujjatni analiz qiladi va unga mos shablonni qidirishga intiladi. Agar shunday shablon topilsa uni ichidagi ko'rsatmalar bajariladi.

Qayta ishlash ishi har doim match="/" shablonidan boshlanadi. Bu adreslash yo'lining qiymati o'zak tugunga mos (1-misol o'zak tugun - kitob) keladi. Masalan.

XML hujjat

```
<?xml version="1.0" encoding="WINDOWS-1251" ?>
```

```
<?xml-stylesheet type="text/xsl" href="t01.xsl" ?>
```

```
<kitob>
```

```
<nomi>Farxod va Shirin</nomi>
```

```
<muallif>Alisher Navoiy</muallif>
```

```
</kitob>
```

Shakl almashtiruvch XSLT (faylni nomi- t01.xsl)

```
<?xml version="1.0" encoding="WINDOWS-1251" ?>
```

```
<xsl:stylesheet version = "1.0"
```

```
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
```

```
<xsl:template match="/">
```

```
<h1 align="center">
```

```
<xsl:value-of select="//nomi"/>
```

```
</h1>
```

```
<h2 align="right">
```

```
<xsl:value-of select="//muallif"/>
```

```
</h2>
```

```
</xsl:template>
```

```
</xsl:stylesheet>
```

1 - misolni natijasi:

Farxod va Shirin.

Alisher Navoiy

Berilgan hujjatdan elementning mazmunini ikki xil usul bilan chiqarib olish mumkin:

1. **xsl:value-of** ko'rsatma yordamida. Bu holda elementning mazmunidan qayta ishlanmagan holda foydalaniladi (2 misol);
2. **xsl:apply-templates** ko'rsatma yordamida. Bu holda XSLT-protessor tanlangan elementlar uchun shablonda ko'rsatilgan ishlarni bajaradi (3 misol).

2-misol. Berilgan XML hujjat

```
<?xml version="1.0" encoding="WINDOWS-1251" ?>
```

```
<?xml-stylesheet type="text/xsl" href="t02.xsl" ?>
```

```
<book>
```

```
<name>Saddi Iskandariy.</name>
```

```
<author>Alisher Navoiy</author>
```

```
</book>
```

XSLT shakl almashtirish (t02.xsl)

```
<?xml version="1.0" encoding="WINDOWS-1251" ?>
<xsl:stylesheet version = "1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
  <h2>
    <xsl:value-of select="."/>
  </h2>
</xsl:template>
<xsl:template match="author">
  <i>
    <xsl:value-of select="."/>
  </i>
</xsl:template>
</xsl:stylesheet>
```

2 - misolni natijasi:

Saddi Iskandariy. Alisher Navoiy

3 – misol. Berilgan XML hujjat

```
<?xml version="1.0" encoding="WINDOWS-1251" ?>
<?xml-stylesheet type="text/xsl" href="t02.xsl" ?>
<book>
  <name>Layli va Majnun.</name>
  <author>Alisher Navoiy</author>
</book>
```

XSLT shakl almashtirish (t03.xsl)

```
<?xml version="1.0" encoding="WINDOWS-1251" ?>
<xsl:stylesheet version = "1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
  <h2>
    <xsl:apply-templates/>
  </h2>
</xsl:template>
<xsl:template match="author">
  <i>
    &#160;<xsl:value-of select="."/>
  </i>
</xsl:template>
</xsl:stylesheet>
```

3 - misolni natijasi:

Layli va Majnun. Alisher Navoiy

match va select atributlarnirning qiymati sifatida ishlatiladigan ifodalarni sintaksisi, fayl sistemasidagi marshrutga o'xshaydi:

Ifodalar	Tushuntirish
/	O'zak tugun
.	Joriy tugun
..	Joriy tugundan yuqoridagi tugun
fruit	fruit tugun

Fruit / lime	Fruit tugunni ostidagi lime tugun
fruit/*	fruit tugunning hamma davomchilari
/fruit	fruit tugun, o‘zak tugunning bevosita davomchisi
@taste	Joriy tugunning atributi taste
@*	Joriy tugunning hamma atributi
fruit@taste	fruit tugunning atributi taste
fruit/lime@taste	fruit tugunning, ostidagi lime tugunning atributi taste
..@taste	Yuqori pog‘onadagi tugunning atributi taste
//	Ixtiyoriy miqdordagi oraliq tugunlar
fruit//lime	Fruit ni ostidagi hamma lime tugunlar
	Aniq tugunlarni ajratish belgisi
lime grape	lime tuguni va grape tuguni
[]	Predikat ifoda
fruit[lime]	Fruit tuguning, davomchisi lime tuguni
fruit[lime="fine"]	Fruit tuguning, davomchisi lime tuguni, uni qiymati fine ga teng
fruit[@taste]	Fruit tugunining atributi taste mavjud
fruit[@taste="5"]	Fruit tugunining atributi taste ning qiymati 5 ga teng
count(fruit/*)	fruit tugunining davomchilari soni
name()	Joriy tugunning nomi

Adreslash yo‘lini ko‘rsatishda “//” belgi ko‘p uchraydi. Adreslash yo‘lining boshida bu belgi hujjatdagi ma’lum tipli hamma tugunlarni tanlashni bildiradi. Adreslash yo‘lining ichida esa, u yo‘lni birinchi qismidagi tugunning hamma davomchilarini tanlash kerakligini bildiradi.

ATRIBUTLAR. Elementga qanday murojaat qilinsa, atributga ham shunday murojaat qilinadi. Faqatgina atributning oldiga “@” belgisi qo‘yiladi.

4- Misol. Berilgan XML hujjat

```

<?xml version="1.0" encoding="WINDOWS-1251" ?>
<?xml-stylesheet type="text/xsl" href="t15_5.xsl" ?>
<polka1>
  <kitob isbn="5-7890-0248X">
    <berilgan nomi="Web-texnologiyalar"
      avtor="Kolesnikov.D.G."/>
  </kitob>
  <kitob isbn="5-78-02X">
    <berilgan nomi="HTML, XHTML I CSS na 100 Procentov"
      avtor=" Kvint I."/>
  </kitob>
</polka1>

```

XSLT shakl almashtirish (t15_5.xsl)

```
<?xml version="1.0" encoding="WINDOWS-1251" ?>
<xsl: stylesheet version = "1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl: template match="kitob">
  <p> <b>
    <xsl: text>ISBN: </xsl: text>
  </b>
    <xsl: value-of select="@isbn"/>
  </p>
  <p> <b>
    <xsl: text>Avtor: </xsl:text>
  </b>
    <xsl:value-of select="berilgan/@avtor"/>
  </p>
  <p> <b>
    <xsl: text>Kitobning nomi: </xsl:text>
  </b>
    <xsl :value-of select="berilgan/@nomi"/>
  </p>
</xsl: template>
</xsl: stylesheet>
```

Natija quyidagi ko‘rinishda bo‘ladi:

ISBN: 5-7890-0248X

Avtor: Kolesnikov.D.G.

Kitobning nomi: Web-texnologiyalar

ISBN: 5-78-02X

Avtor: Kvint I.

Kitobning nomi: HTML, XHTML I CSS na 100 Procentov

Nazorat uchun savollar:

1. XSLT tilining vazifasi nimadan iborat?
2. XSLT shakl almashtirish uchun hujjat qanday tartibda yaratiladi?

Uyga vazifa:

“Kutubxona” mavzusi bo‘yicha struktura tuzing va shu strukturani ifodalovchi XML hujjat yarating, va unga mos *xsl fayl tuzing.

Adabiyotlar

1. Хабибуллин И. Самоучитель XML. Санкт-Петербург. 2003.
2. Шапошников И.В. Справочник Web-мастера XML СПб: БХВ-Петербург 2001.- 304 с
3. Гарольд Э., Минс С. XML. Справочник. – Пер. с англ. – СПб: Символ-Плюс, 2002. – 576 с.
4. Валиков А.Н. Технология XSLT-СПб: БХВ-Петербург. 2002 – 544 с.

13 - mavzu. Mohiyatlarni aniqlash

Maqsad: Mohiyatlarni aniqlash ni tushuntirish.

Kalit so‘zlar: Mohiyat, tashqi va ichki mohiyat

Asosiy savollar:

1. Ichki mohiyatlar
2. Tashqi mohiyatlar

3. Parametrlashgan mohiyatlar

Ichki mohiyatlar

XML hujjatning bor quvvatini yetarli darajada ko'rsatish uchun, XML hujjat alohida nomlangan ob'yektlarga bo'lib o'rganiladi. Shu alohida ob'yektlar mohiyatlar deyiladi.

Mohiyatlar ampersand belgisi bilan boshlanadi va nuqtali vergul bilan tugaydi. Mohiyatlar (entity), hujjatning DTD bo'limida e'lon qilinadi.

Hamma mohiyatlarni uchta guruhga bo'lish mumkin:

- Ichki mohiyatlar- mohiyatni e'lon qilish jarayonida ko'rsatiladi
- Tashqi mohiyatlar – alohida faylga joylashadi;
- Parametrlashgan mohiyatlardan- faqatgina DTD tushuntirish bo'limoni ichidagina foydalaniladi.

Ichki mohiyatni ta'riflash quyidagi ko'rinishda bo'ladi:

```
<!ENTITY nom "qiymat">
```

Masalan:

```
<!ENTITY author "A.S.Pushkin">
```

Bu usuldan nomlangan Matnli ob'yektni ko'rsatishda foydalaniladi.

Mohiyatga havola esa &nom; (ampersant, mohiyatning nomi, nuqtali vergul yoziladi).

Misol.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<!DOCTYPE advert [
```

```
<!ENTITY mumtoz "Alisher Navoiy">
```

```
]>
```

```
<advert>
```

```
<book>&mumtoz; o&apos;zbeq adabiyotining shakllanishida
```

```
o&apos;chmas iz qoldirgan shaxs!
```

```
&mumtoz;-15-16 asrlarda o'rta osiyoda yashagan hamda
```

```
yuksak ma'naviyat va nafosatni o'zida jamlagan
```

```
&quot;Hamsa &quot; dostonini yozib qoldirgan.
```

```
</book>
```

```
</advert>
```

XML spetsifikatsiyasida bir nechta ichki simvolli mohiyatlar mavjud, ularni quyidagi jadvalda keltiramiz:

Mohiyatning nomi	Qiymati	Tushuntirish
lt	<	"kichik" belgisini ifodalaydi
gt	>	"katta" belgisini ifodalaydi
Amp	&	Ampersand belgisini ifodalaydi
apos	'	Apostrof yoki bittali qoshtirnoq belgisini ifodalaydi
Quot	"	Ikkitalik qo;shtirnoq belgisini ifodalaydi

Simvolli mohiyatlardan XML hullatda uchraydigan simvollarni belgi sifatida aks ettirish uchun foydalaniladi.

5-Misol.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<advert>
```

```
<product title="gul">
```

```
Narxi &lt;3000 bolgan atir gullar sotiladi!
```

```
Firma &quot;kaktus;&quot;;
```


</product>

</advert>

Tashqi mohiyatlar tashqi faylda saqlanadi. Agar hujjatda tashqi faylga havola ko'r satilsa tashqi faylning mazmuni mohiyatning o'rniga ko'chadi.

[Tashqi mohiyatlar](#)

Tashqi mohiyatning sintaksisi quyidagi ko'rinishda bo'ladi:

<!ENTITY nom SYSTEM "URL">

Bu yerda nom huddi ichki mohiyatdagi nom kabi aniqlanadi, URL esa tashqi faylning absolyut yoki nisbiy adresini aniqlaydi.

6-Misol

<!ENTITY animal SYSTEM "ent/animal.ent"> - nisbiy adres korsatilayapti.

<!ENTITY animal SYSTEM "http://www.animalhost.com/animal.ent"> - absolyut adres ko'rsatilayapti.

Tashqi mohiyatga murojaat ham huddi ichki mohiyat kabi yoziladi:

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE advert [

<!ENTITY animal SYSTEM "ent/animal.ent">

]>

<advert>

<product title="fil">

Ogirligi &qt; 5 bo'lgan &animal; sotiladi!

Qiziquvchilar 3455678 telefon orqali murojaat qiling!

</product>

</advert>

Tashqi mohiyat oldindan e'lon qilingan serverda saqlanayotgan bo'lsa, unga murojaat quyidagi ko'rinishda bo'ladi:

<!ENTITY nom PUBLIC "identifikator" "URL">

Masalan.

<!ENTITY animal PUBLIC "-//Z00//Elephant//Description"

"http://www.animalhost.com/animal.ent">

Qachon mohiyatlardan foydalanish mumkinligini xulosa qilamiz:

1. hujjatda uchraydigan ko'p takrorlanadigan fragmentlarni almashtirishda;
2. bitta XML- hujjatni alohida modullarga ajratishda;
3. Belgi sifatida ishlatiladigan simbollarni hujjatning ishida aks ettirishga zarurat yuzaga kelsa;
4. Unicode simvollarini mos keluvchi kodi bilan yozishga zarurat yuzaga kelsa.

[Parametrlashgan mohiyatlar](#). Parametrlashgan mohiyatlardan hujjatning DTD sxemasida foydalaniladi. Shuningdek parametrlashgan mohiyatlardan elementlarning umumiy atributlari guruxini korsatishda ham foydalaniladi. Parametrlashgan mohiyatlar ham ichki va tashqi bo'lishi mumkin. Parametrlashgan mohiyatlar DTD sxemada (%) belgisi bilan. Uning sintaksisi quyidagicha ko'rinishda:

<!ENTITY % havolaning_nomi havolaning_ifodalanishi>

Protsent belgisi havolaning nomidan probel bilan ajralib turishi kerak.

Faraz qilaykik siz schet-faktura uchun DTD sxema yaratishingiz kerakki, unda to'lov adresi, yetkazib beriladigan adres, sotuvchining adresi, tovar saqlanadigan joyni adresi ko'rsatilishi zarur bo'lsin.

Agar hamma adreslar umumiy strukturaga ega bo'sa siz parametrlashgan mohiyatdan yaratib, undan DTD sxemada birnecha bor foydalanish mumkin.

<!ENTITY % address

"street1 CDATA #REQUIRED

```

street2 CDATA #IMPLIED
city CDATA #REQUIRED
State CDATA #REQUIRED
zip CDATA #REQUIRED">
Schet- faktura uchun DTD ning to'liqroq sxemasi quyidagi ko'rinishda bo'ladi:
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE invoice [
<!ELEMENT invoice bill,ship,vendor,warehouse)>
<!ELEMENT bill (#PCDATA)>
<!ATTLIST bill %address;>
<!ELEMENT ship (#PCDATA)>
<!ATTLIST ship %address;>
<!ELEMENT vendor (#PCDATA)>
<!ATTLIST vendor %address;>
<!ELEMENT warehouse (#PCDATA)>
<!ATTLIST warehouse %address;>
<!ENTITY % address
"street1 CDATA #REQUIRED street2 CDATA #IMPLIED
city CDATA #REQUIRED
State CDATA #REQUIRED
Zip CDATA #REQUIRED">
]>

```

Bu misolda bill, ship, vendor, warehouse elementlar uchun address nomli parametrdan foydalanilgan.

Tashqi parametrli mohiyat aloxida hujjatda joylashadi. Aslida DTD sxemani ham tashqi parametrli havola deb qarash mumkin.

Mohiyatdan foydalanishga 2 misol.

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE contract [
<!ELEMENT contract (para1,para2)>
<!ELEMENT para1 (#PCDATA)>
<!ELEMENT para2 (#PCDATA)>
<!ATTLIST para2
section CDATA #REQUIRED>
<!ENTITY buyer "Devan Shepherd">
]>
<contract>
<para1>&buyer; agrees to buy the property defined below
on the terms and conditions set forth in this
contract.</para1>
<para2 section="1">&buyer; will take title of the real
property
described below, for which consideration Sbuyer; agrees
10: to pay... </para2>
</contract>

```

Mavzu uchun topshiriq.

Element va atributlardan foydalangan holda “Kutubxona” mavzusi bo'yicha XML hujjat yarating.

14 – mavzu. XHTML(extensible hyper text markup language)

Maqsad: XHTML ning imkoniyatlari va vazifalarini tushuntirish

Kalit so'zlar: DOCTYPE, Strict, Transitional, Frameset

Asosiy savollar:

1. XHTML nima ?
2. XHTML- hujjatning strukturasi.
3. XHTML tilining sintaksisi.

1. XHTML nima?

HTML tilining rivojlanishi ma'lum ma'noda chorasiz holatga kelib qoldi — yangi teglar kerak emas, chunki mavjud teglar ehtiyoj uchun yetarli, web- sahifani tayyorlashda e'tibor stillarga ko'chdi, chunki aynan ular sahifani bezash uchun keng imkoniyat beradi. Tabiiyki, CSS hech qachon HTMLni o'rnini butunlay egallay olmaydi, lekin cheklangan teglar guruhidan foydalanishga imkon beradi, elementning turi, joylashish o'rnini, turli parametrlari stil yordamida beriladi.

HTML formal tillar guruhiga kiradi, chunki uni teglari va iyerarxik struktura qat'iy belgilab qo'yilgan. Shuning uchun XML ommalasha boshladi, chunki uni yordamida foydalanuvchi o'z tegini yaratadi va mos strukturani shakllantiradi. HTML va XML o'rtasidagi farq faqatgina teglar emas, balki ularni yozish qoidalarida ham mavjud.

Brauzer HTML hujjat bilan ishlaganda kichik xatolarni ko'rmay o'tib ketadi. XML bilan ishlaganda esa brauzer har qanday xatolikni ko'rsati va xatolik tuzatilmaguncha hech qanday natijani chiqarmaydi.

Foydalanuvchilar to'g'ri fikr yuritishlari uchun, ularning kod yozish stilini o'zgartirish maqsadida va shuningdek HTML va XML orasidagi uzilishni to'ldirish maqsadida, XHTML tili ishlab chiqildi. Bu til HTML va XML orasidagi etapda turadi.

XHTML (Extensible HyperText Markup Language)- kengaytirilgan giperMatnni belgilashlar tili, HTMLning qat'iy qoidalar kiritilgan versiyasi hisoblanadi. Umuman olganda, W3C konsortsiyumi XHTML ni HTMLning oxirgi versiyasi deb ta'riflaydi, chunki u vaqti kelib, HTML ni butunlay siqib chiqarishi mumkin. Shunday qilib, hozirgi kunda XHTML 1.0 ishlab chiqilgan bo'lib, u HTML ga o'xshaydi, lekin qat'iy sintaksisga ega.

2. XHTML-hujjatning strukturasi.

Har qanday XHTML- fayl to'rtta bo'limdan tashkil topadi — <!DOCTYPE>, (<html>) konteyner, sarlavha (<head>) va hujjatning tanasi (<body>). Quyidagi misol xhtml hujjatning strukturasi ifodalaydi.

2.1_kod. Oddiy XHTML-hujjat.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
```

```
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
```

```
<!--Bu bo'lim sahifaning sarlavhasini va texnik axborotlarni  
yozishga mo'ljallangan. -->
```

```
</head>
```

```
<body>
```

```
<!-- bu bo'limda brauzer oynasida ko'rinishi kerak bolgan,  
sahifaning mazmuni beriladi. -->
```

```
</body>
```

```
</html>
```

<!DOCTYPE> elementi joriy hujjatning tipi va uni qanday interpretitsiyalash haqida brauzerga ma'lumot beradi. Различают три типа документа по спецификации XHTML 1.0 spetsifikatsiyasiga ko'ra hujjatning uch xil tipi mavjud:

1. Strict (Qat'iy);
2. Transitional (O'tuvchi);

3. Frameset (Freymli sahifa).

XHTML 1.0 Strict

Agar hujjatning mazmuni va bezatish stili aniq ajratilgan bo'lsa, XHTML 1.0 strict dan foydalaniladi. Bu holda web- sahifaning kodi faqat belgilashlar tegi bilan beriladi,elementning tashqi ko'rinishi stil orqali beriladi. (Listing 2.2).

Listing 2.2.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Qat'iy hujjatga misol</title>
<meta http-equiv="Content-Type" content="text/html;
charset=windows-1251" />
</head>
<body>
<p>...</p>
</body>
</html>
```

XHTML 1.0 Transitional

Agar hujjatning mazmuni va bezatish stili aniq ajratilmagan bo'lsa, XHTML 1.0 transitional dan foydalaniladi. Bu holda web- sahifaning kodi belgilashlar tegi bilan birgalikda bezash stili ham qisman beriladi,elementning tashqi ko'rinishi stil orqali beriladi. (Listing 2.3).

Listing 2.3. «O'tuvchi» hujjat

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>O'tuvchi</title>
<meta http-equiv="Content-Type" content="text/html;
charset=windows-1251" />
</head>
<body>
<p>...</p>
</body>
</html>
```

XHTML 1.0 Frameset

Agar brauzer oynasi ikki yoki undan ko'p freymlarga bo'linsa , XHTML 1.0 Framesetdan foydalaniladi. (Listing 2.4).

Listing 2.4. Freymli hujjat.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml-frameset.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Freymlar</title>
<meta http-equiv="Content-Type" content="text/html;
charset=windows-1251" />
</head>
<frameset cols="100,*">
<frame src="menu.html" name="menu" />
<frame src="content.html" name="content" />
</frameset>
</html>
```

<!DOCTYPE> zaruriy element husoblansa ham, XHTML-hujjatning bevosita qismi emas, shuning uchun unga yopuvchi teg kerak emas.

HTML konteyner

<html> tegi hamma elementlarni o'zida saqlovchi asosiy konteyner bo'lib chiqadi va u <!DOCTYPE> dan keyin yoziladi. <html>da zaruriy parametr – xmlns ni albatta ko'rsatish kerak, uni qiymati "http://www.w3.org/1999/xhtml". U nomlar fazosini aniqlaydi. Nomlar fazosi – bu hamma element va atributlarning belgilab olingan nomi XHTML-hujjat doirasida unikal ekanini bildiradi.

Hujjatni kodlash

Hujjatning HTML nkodi <meta> tegi orqali beriladi va <head> konteyneri ichiga joylashadi. XHTML kodlash ikki yo'l bilan berilishi mumkin. U buyruq;ar yordamida <?xml version="1.0" encoding="UTF-8"?>, bunda <!DOCTYPE>dan oldin birinchi qatorga yoziladi (2.5 listing) .

Listing 2.5. Hujjatni kodlash

```
<?xml version="1.0" encoding="windows-1251"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>Kodlash</title>
  </head>
  <body>
    <p>Kiril alifbosining harflari, Windows-1251 kodlash
sistemasida beriladi.</p>
  </body>
</html>
```

3. XHTML tilining sintaksisi

XHTML kod yozganda HTML uchun xarakterli sintaksisdan foydalaniladi. Faqatgina HTML va XHTML orasida farqni ko'rsatuvchi ba'zi qoidalar mavjud.

XHTML qoidalari quyidagilar.

1. Hamma teglar va ularning paranetrlari quyi registrda yozilishi kerak.
 2. Har qanday parametrning qiymati qo'shtirnoq ichida berilishi kerak.
 3. Hamma teglar yopilishi kerak, hatto yopuvchisi bo'lmagan teglar ham.
 4. Teglarini ichma-ich yozish qoidalariga amal qilinishi kerak, yani ular bir-biri bilan kesishmasligi kerak.
 5. teglani qisqartirilgan atributlari ishlatilmaydi.
 6. name parametrining o'rniga id parametri ko'rsatiladi.
 7. <!DOCTYPE> elementi yordamida DTD (document type definition, hujjatning tipini tushuntirish) ko'rsatilishi kerak. U birinchi qatorda yoziladi.
- XHTML da hamma parametrlarning qiymatlari qo'shtirnoq ichida berilishi zarur.

(Listing 3.3).

Listing 3.3. Qo'shtirnoqdan foydalanish

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="text/html;
charset=windows-1251" />
    <title>XHTML</title>
  </head>
  <body>
    <table width="200" border="1" cellpadding="5" cellspacing="0">
```

```

<tr>
  <th scope="col">Abdulla Oripov</th>
  <th scope="col">Muhammad Yusuf</th>
</tr>
<tr>
  <td>1</td>
  <td>5</td>
</tr>
<tr>
  <td>4</td>
  <td>13</td>
</tr>
</table>
</body>
</html>

```

Toq teglarni yopishda `</>` konsrtruktsiyadan oldin probel qo'yiladi.

HTMLva XHTML dagi toq teglarni taqqoslash	
HTML-teg	XHTML-teg
<code>
</code>	<code>
</code>
<code><hr></code>	<code><hr /></code>
<code><input></code>	<code><input /></code>
<code></code>	<code></code>
<code><meta></code>	<code><meta /></code>

Teglarning qisqartirilgan atributi deganda – qiymat berilmagan parametr nazarda tutiladi. Parametrdagi «Bo'sh qiymatni», yo'qotish uchun atributni nomi bilan mos keluvchi qiymat beriladi.

HTML va XHTML parametrlarini taqqoslash	
HTML	XHTML
<code>checked</code>	<code>checked="checked"</code>
<code>compact</code>	<code>compact="compact"</code>
<code>disabled</code>	<code>disabled="disabled"</code>
<code>ismap</code>	<code>ismap="ismap"</code>
<code>multiple</code>	<code>multiple="multiple"</code>
<code>nohref</code>	<code>nohref="nohref"</code>
<code>noresize</code>	<code>noresize="noresize"</code>
<code>noshade</code>	<code>noshade="noshade"</code>
<code>nowrap</code>	<code>nowrap="nowrap"</code>
<code>readonly</code>	<code>readonly="readonly"</code>
<code>selected</code>	<code>selected="selected"</code>

Listing 3.4. Rasmning identifikatori

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="text/html;
charset=windows-1251" />
    <title>XHTML</title>
  </head>
  <body>
    <p></p>
  </body> </html>

```

Mustaqil bajarish uchun topshiriq.

«Supermarket» mavzusi bo'yicha XHTML hujjat yarating.