

9 лабораториялық жұмыс. CSS құру негіздері

1. Стильдің мұра ретінде берілуі

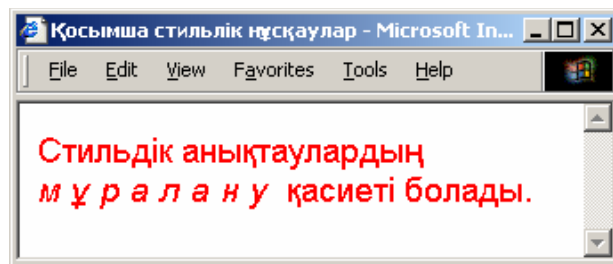
HTML-кодтың құрылымы сатылы түрде болады. Бар құжат <body>. . .</body> тәгтері ішінде орналасады. Ал <P> абзацы ішінде тәгімен белгіленген элементтер болуы мүмкін.

Стильдің мұра ретінде берілуі <body> тәгі үшін анықталған стильдің құжаттағы барлық тәгтерге әсер ететіндігін білдіреді. Сәйкесінше егер бір тәг үшін стиль құрылса, онда сол тәг ішіндегі басқа тәгтерге де осы стиль міндетті түрде беріледі.

Мысалы, <P> тәгі үшін келесі стиль берілген болсын:

```
P
{
  color: red; font-size:14pt;
  font-family: Arial, sans-serif; }
```

Онда осы абзац ішіне орналасқан тәгінің элементтері де экранда қызыл түспен, 14 пункт көлемде жұмыр қаріппен шығарылады.



Мысалы:

<P>

Стильдік анықтаулардың мұралану қасиеті болады.

Ал егер тәгі үшін мәтін символдары арасын ашып жазу қажет болса, онда мұндай қасиет қосымша беріледі:

<HTML>

<HEAD>

<TITLE> Қосымша стильдік нұсқаулар </TITLE>

<STYLE type="text/css">

<!--

P {color:red;font-size:14pt;
font-family:Arial,sans-serif}

-->

</STYLE>

</HEAD>

<BODY bgcolor=white text=black>

<P>

Стильдік анықтаулардың <EM style="letter-spacing: 6pt;"> мұралану

қасиеті болады.

</BODY>

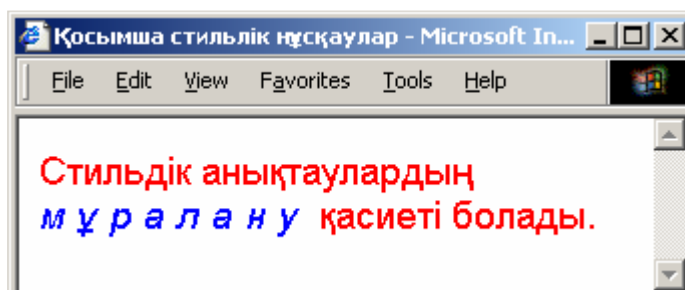
</HTML>

 тәгінің ішкі мәтіні <P> тәгінің мәтіндері сияқты шығады (14 пункт, қызыл түсті, жұмыр қаріп), ал style="letter-spacing: 6pt" қосымша анықтаудың әсерінен сөздегі әріптердің арасы 6 пункт болып ашылып орналасады.

Сонымен ішкі қосымша тәгте жаңа стильдік анықтаулар кіргізіп қана қоймай, оның ата тегі болып саналатын сыртқы тәгтің де қасиеттерін өзгертуге болады.

Мысалы, келесі мәтіндегі “мұралау” (наследование) сөзі көк түске боялып тұрады.

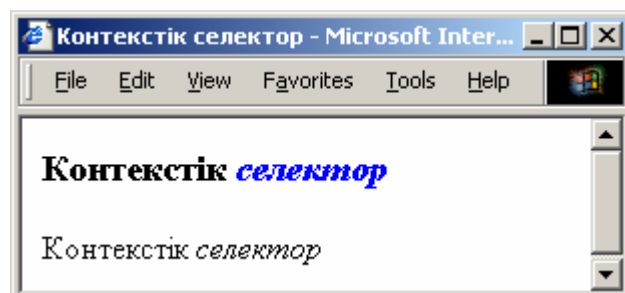
```
<HTML>
<HEAD>
  <TITLE> Стыльлік нұсқауларды қайта анықтау
</TITLE>
  <STYLE type="text/css">
    <!--
      P {color: red;font-size:14pt; font-family:Arial,sans-serif}
    -->
  </STYLE>
</HEAD>
<BODY bgcolor=white text=black>
  <P> Стыльдік анықтаулардың <EM style="letter-spacing: 6pt; color:blue
">
    мұралану </EM> қасиеті болады.
</BODY>
</HTML>
```



2. Контекстік селекторлар

Стыльдік анықтауларды, тәгтердің кірістіріліп орналасу реттілігіне сәйкес орындалатындай етіп жазуға болады. Мысалы, тәгіндегі мәтін түсі ол <H3> тәгінің ішінде орналасқан жағдайда ғана көк болатындай етіп орналастырайық:

```
<HTML>
<HEAD>
  <TITLE>Контекстік селектор
</TITLE>
  <STYLE type="text/css">
    <!-- H3 EM {color: blue}
    -->
  </STYLE>
</HEAD>
<BODY bgcolor=white text=black>
  <H3> Контекстік <EM>селектор </EM></H3>
  <P> Контекстік <EM>селектор </EM></P>
</BODY>
</HTML>
```



Экранға «селектор» сөзі бірінші қатарда көк түспен (<H3> тәгінің ішінде), ал екінші қатарда қара түспен (<P> тәгінің ішінде) шығады. **Стыльдік анықтауларда үтір қойылмағандығына назар аударыңыз.** Бұл контекстік анықтаулардың белгісі. Егер H3, EM { color: blue } түрінде жазсақ, онда <H3> тәгімен қатар тәгі де көк түске боялады, яғни үтір бірнеше тәгтер тобы үшін бірдей болатын ортақ стильді анықтайды.

3. Кластар

Стильдік анықтауларды тәгтерді көрсетпей-ақ жазуға болады. Мұндай жағдайда, берілген стилді нақты тәгке сәйкестендіру үшін, әрбір анықтауға арнайы атау меншіктелуі тиіс. Мұндай стильдік анықтаулар кластар деп аталады. Класс келесі түрде жазылады:

```
.аты
{ сипаттама: мәні;
...
сипаттама: мәні;
}
```

Басқаша айтқанда, анықтаулар әдеттегідей жазылады, бірақ тәгтердің аттары орнына **.аты** конструкциясы қолданылады.

Мысалы, **def** атты стильдік класс жазып шығайық:

```
.def
{color: red; font-size: 16pt;
font-family: Geneva, Helvetica, sans-serif;
border: solid 0.2cm blue }
```

Мұнда **border: solid 0.2cm blue** қатары бір стильдік нұсқауда бірнеше параметрлердің қызметін жазуды көрсетеді (мұнда: стиль, жақтау қалыңдығы, және оның түсі). Мұндай біріктіруді **border** сияқты жалпыландырылған арнайы стильдік қасиеттерге қолдануға болады.

Стильдік класты тәгпен сәйкестендіру үшін **class** атрибуты қолданылады:

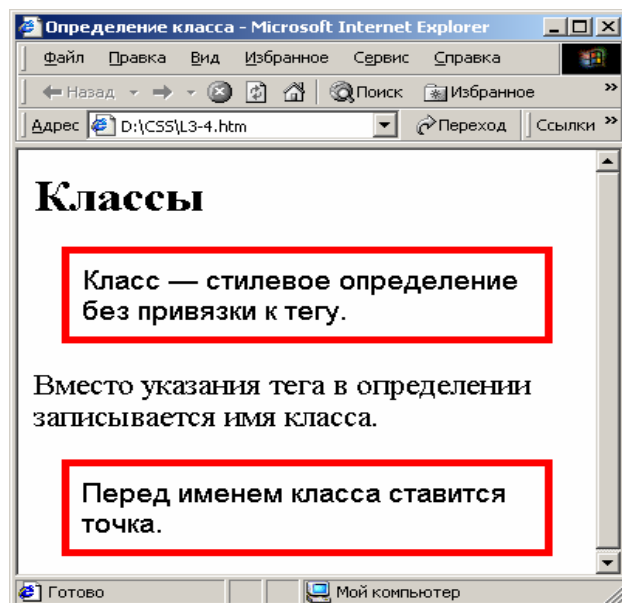
```
<P class=def > текст </P>
```

Бұл кодтың қалай жұмыс істейтінін қарастырайық.

```
<HTML>
<HEAD>
<TITLE>Определение
класса</TITLE>
<STYLE type="text/css">
.def
{font-family:Helvetica;font-size:14pt;
border: solid 4pt red;padding: 6pt;
margin-left:5%;margin-right:5% }
</STYLE>
</HEAD>
<BODY bgcolor=white text=black>
<H2>Классы</H2>
<P class=def>
Класс&nbsp;&#151; стиливое
определение без привязки к тегу.
<P> Вместо указания тега в
определении записывается имя класса.
<P class=def>
Перед именем класса ставится точка.
</BODY>
</HTML>
```

Мысалы, **def** атты стильдік класс жазып шығайық:

```
.def
{color: red; font-size: 16pt;
font-family: Geneva, Helvetica, sans-serif;
border: solid 0.2cm blue }
```



Мұнда border: solid 0.2cm blue қатары бір стиль-дік нұсқауда бірнеше параметрлердің қызметін жазуды көрсетеді (мұнда: стиль, жақтау қалыңдығы, және оның түсі). Мұндай біріктіруді border сияқты жалпыланды-рылған арнайы стильдік қасиеттерге қолдануға болады.

Стильдік класты тәгпен сәйкестендіру үшін **class** атрибуты қолданылады:

<P class=def > текст </P>

Бұл кодтың қалай жұмыс істейтінін қарастырайық.

<HTML>

<HEAD>

<TITLE>Определение класса</TITLE>

<STYLE type="text/css">

.def

{font-family:Helvetica;font-size:14pt;

border: solid 4pt red;padding: 6pt;

margin-left:5%;margin-right:5% }

</STYLE>

</HEAD>

<BODY bgcolor=white text=black>

<H2>Классы</H2>

<P class=def>

Класс — стилевое определение без привязки к тегу.

<P>Вместо указания тега в определении записывается имя класса.

<P class=def>

Перед именем класса ставится точка.

</BODY>

</HTML>

Құрылған стильдік анықтаулар негізінде кластар құруға болады. Келесі мысалда def класының негізі ретінде **<P>** тәгі үшін анықталған стиль алынып, оған қосымша жаңа қасиеттер қосылып жазылған:

<HTML>

<HEAD>

<TITLE>Определение класса</TITLE>

<STYLE type="text/css">

P {font-family:Helvetica}

P.def

{text-align: justify;

background:#CFB597;

font-size:14pt;

border:solid 4pt red;

padding:6pt;

margin-left:5%;margin-right:5%}

</STYLE>

</HEAD>

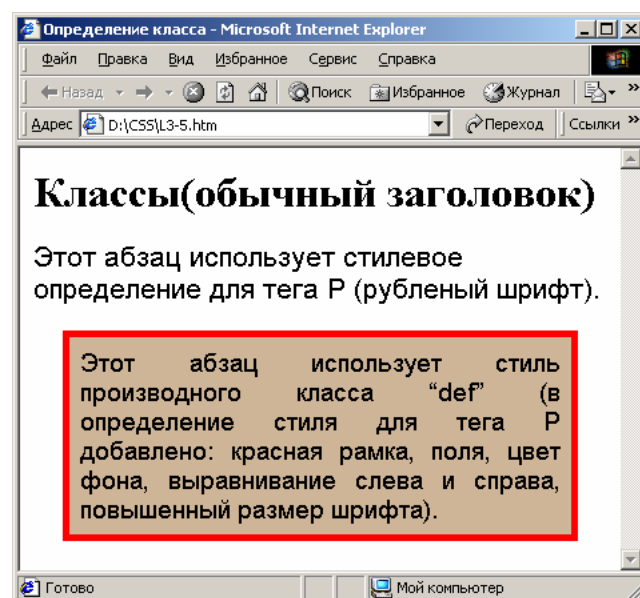
<BODY bgcolor=white text=black>

<H2> Классы(обычный заголовок)</H2>

<P> Этот абзац использует стилевое определение для тега P (рубленный шрифт).

<P class=def>

Этот абзац использует стиль производного класса “def”



(в определение стиля для тега P добавлено: красная рамка, поля, цвет фона, выравнивание слева и справа, повышенный размер шрифта).

</BODY>

</HTML>

4. <DIV> және тәгтері

Бұл тәгтер CSS үшін маңызды болып табылады. Олар құжаттағы жеке бір бөліктерді ерекшелеп алып, оларға арнайы қасиеттер беру ісін атқарады. Ол үшін керекті элементтерді <DIV>. . .</DIV> немесе . . . тәгтері ішіне орналастыру керек.

Бұлардың айырмашылығы мынада: <DIV> блогынан соң браузер жаңа жолға көшіреді, ал блогынан кейін — бұрынғы жолда қала береміз. Сонымен, тәгін пайдалану - бір жолдағы сөздерге не символдарға жеке стильдік қасиеттер тағайындай алады. Осыларға мысал келтірейік.

<DIV> тәгін пайдалану:

<HTML>

<HEAD>

<TITLE> **Использование тега**
DIV</TITLE>

<STYLE type="text/css">

.area1

{ color:red; font-weight:bolder;
font-size:40pt; background:aqua}

.area2

{ color:black; background:#CFB597}

.area3

{ color:blue;background:#C0C0C0}

</STYLE>

</HEAD>

<BODY bgcolor=white text=black>

<DIV class=area1> Где</DIV>

<DIV class=area2>начало того конца,</DIV>

<DIV class=area3>которым оканчивается начало?

</DIV>

</BODY>

</HTML>

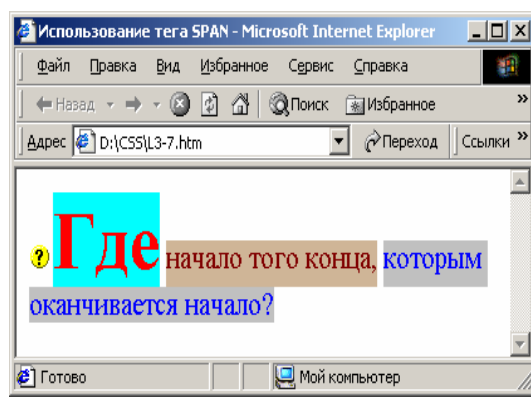
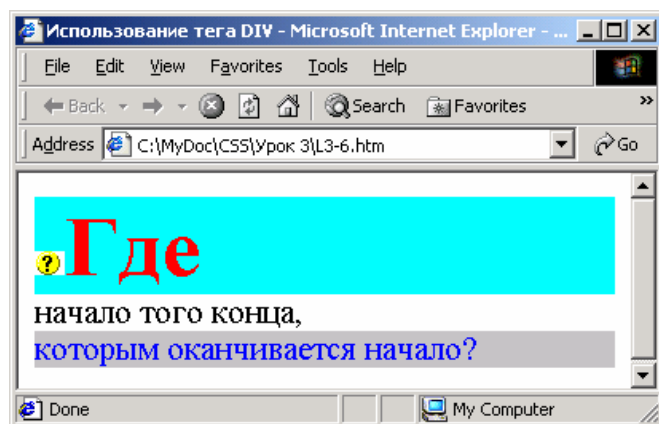
 тәгін пайдалану мысалы:

<HTML>

<HEAD>

<TITLE> **Использование тега**
SPAN</TITLE>

<STYLE type="text/css">



```

.area1
{ color:red; font-weight:bolder;
  font-size:40pt; background:aqua}
.area2
{ color:maroon; background:#CFB597;
  padding:6pt}
.area3
{ color:blue;background:#C0C0C0;
  padding:6pt}
</STYLE>
</HEAD>
<BODY bgcolor=white text=black>
<SPAN class=area1><IMG src=vopros.gif>Где</SPAN>
<SPAN class=area2>начало того конца,</SPAN>
<SPAN class=area3>которым оканчивается начало?
</SPAN>
</BODY>
</HTML>

```

5. Абсолюттік түрде орналастыру

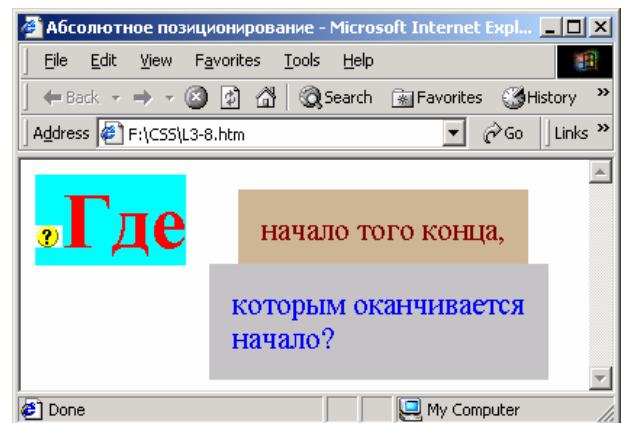
CSS арқылы экрандағы элементтерді браузер терезе-сінің сол жақ жоғары бұрышынан бастап нақты координаталар бойынша орналастыра аламыз. Мұндай мүмкіндікке position стильдік қасиетінің absolute мәні арқылы қол жеткізуге болады. Координаталар пиксель бірлігімен left (горизонталь координата) және top (вертикаль координата) сөздері арқылы тағайындалады.

Енді мысалдар қарастырамыз.

```

<HTML>
<HEAD>
<TITLE> Абсолютное позиционирование</TITLE> <STYLE type="text/css">
.area1
{ position:absolute; top:10; left:10;
  color:red; font-weight:bolder;
  font-size:40pt; background:aqua}
.area2
{ position:absolute; top:20; left:150;
  color:maroon; background:#CFB597;
  padding:12pt}
.area3
{ position:absolute; top:70; left:130;
  color:blue; background:#C0C0C0;
  padding:12pt}
</STYLE>
</HEAD>
<BODY bgcolor = white text = black>
<DIV class =area1> <IMG src=vopros.gif> Где </DIV>
<DIV class =area2> начало того конца, </DIV>
<DIV class =area3> которым оканчивается начало?
</DIV>
</BODY>
</HTML>

```



Мұнда браузер үш <DIV> тәгімен берілген эле-менттерді көрсетілген координаталар бойынша орналастырып шықты. Элементтер бірінің үстіне бірі шығып тұрғанын байқаған шығарсыңдар.

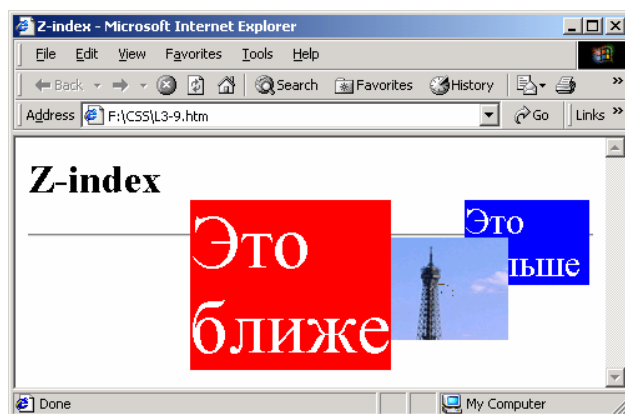
Ең үстіне, яғни бізге жақын түрде HTML-кодта ең соңғы көрсетілген элемент шығады. Егер <DIV> тәгтерінің орнын ауыстырсақ, онда элементтердің қабаттаса орналасу реттілігі де өзгеріске ұшырай-ды. Дегенмен, CSS элементтерді қабаттастыра орналасу реттілігін жылдам өзгертетін тағы бір мүмкіндікті - Z-index тәсілін де ұсынады.

5. Z-index

Бұл стильдік қасиет экрандағы элементтің қай деңгейде (қай қабатта) орналасуы керек екендігін анықтайды. Мұндағы негізгі деңгейдің нөмірі (элементтер стильдік анықтаусыз-ақ шығатын деңгей) нөлге тең болып саналады. Сәйкесінше, экран қабатын анықтайтын теріс Z-index мәні бар элементтер төменде (алыста), ал оң мәнді элементтер — жоғарыда (жақын) орналасады. Егер элементтердің Z-index мәндері бірдей болса, олар бір қабатта орналасады. Сонымен Z-index мәні үлкен элемент жоғары деңгейде орналасады екен.

Енді мысал қарастырайық.

```
<HTML>
<HEAD>
  <TITLE> Z-index </TITLE>
</HEAD>
<BODY bgcolor=white text=black>
  <H2>Z-index</H2> <HR>
  <DIV style="position:absolute; top:50;
    left:140; height:130; width:100;
    background:red; font-size:60;
    color:white; z-index:3">
    Это ближе </DIV>
  <DIV style="position:absolute; top:50; left:360; height:30; width:100;
    background:blue; font-size:30; color:white; z-index:1">
    Это </DIV>
  <DIV style="position:absolute; top:80; left:270; width:125; z-index:2">
    <IMG src=img555.jpg width=125 height=82 alt ="">
  </DIV>
</BODY>
</HTML>
```



6. Сатылай орналастыру (каскадирование)

CSS көмегімен тұтынушылар экрандағы ақпаратты ыңғайлы түрде бейнелейтін мүмкіндік алады.

Тек жасалған стильдердің бір-бірімен өзара байланысын және тәгтер атрибуттарының жазылуын (олар да стильді анықтайды) браузердің өз стилімен (үнсіз келісім бойынша анықталатын стиль) шатастырып алмау керек.

Жоғарыда айтылғандай, программалаушы HTML-құжатта үш түрлі стиль пайдалана алады:

- Құрамдас (встроенный - inline). Style атрибуты арқылы тәг ішінде жазылған стиль, ол жеке тәг жұмысын бақылайды.

- Енгізілген (внедренный - embedded). HTML-файл тақырыбында <style>...</style> тәгтері арқылы жа-зылған стиль, ол жеке HTML-құжаттағы стильді бақылайды.

· Байланысқан (связанный - linked). Жеке CSS-файлда жазылған стиль. Ол көптеген HTML-құжаттар стилін бақылай алады. Мұнда стильдік файлға сілтеме жасау үшін HTML-құжаттың бас жағына <link> тәгі жазылады.

Бір құжатта осы айтылған үш стильдер механизмі де қолданыла алады. Бұған тәгтердің кәдімгі атрибуттары арқылы берілетін стильді және браузердің “келісім бойынша” тағайындалған өз стилін қосайық. Бұлар бір-бірімен қалай әрекеттесер екен, компьютер экранына құжатты шығарарда браузер қай стильді пайдаланады?

Құжатты шығарарда кәдімгі браузер ұстанатын ережелер *каскадты*, яғни *сатылы* деп аталады. Бұл браузер үшін ең негізгі стиль – құрамдас стиль деген сөз, одан кейін төмен қарай реті бойынша енгізілген және байланысқан стильдер қолданылады.

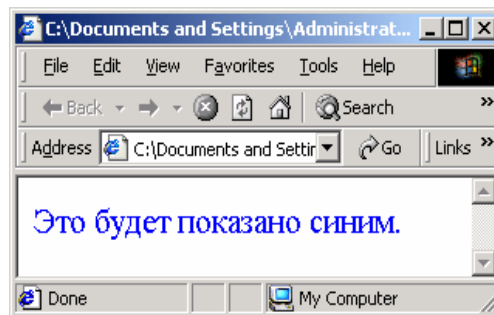
Сонымен браузер үшін енгізілген және байланысқан стильдер бірдей болып саналады, ол тек соңғы қолданылуға байланысты өзгереді.

Стильдік реттілік бойынша ең төменгі сатыда «үнсіз келісім бойынша» стиль тұрады. Оны браузер ешқан-дай стильдік нұсқау болмаған жағдайда қолданады. «Каскадтау - сатылау» ұғымына арғы тегінен берілу – мұралау (наследования) механизмі де жатады, ол бойынша өз стилі жоқ бала-тәг ата-тәгтің стилін қабылдайды (к потомку без собственных стилей применяется стиль родителя).

Кәдімгі атрибуттар арқылы берілген стильдер осы сатылау ережесі бойынша жұмыс істейді.

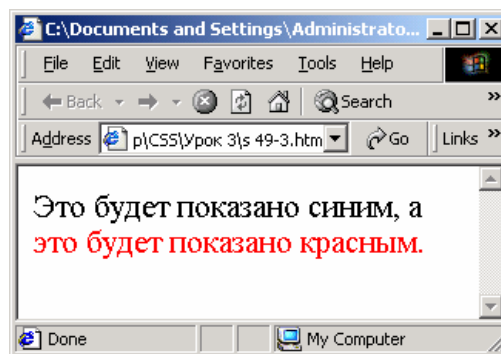
1 мысал.

```
<FONT color=red>  
<P style="color:blue">  
Это будет показано  
синим.  
</P>  
</FONT>
```



2 мысал.

```
<P style="color:blue">  
Это будет показано  
синим, а  
<FONT color=red>  
это будет показано  
красным.  
</FONT>  
</P>
```



СҰРАҚТАР

1. «Иерархия», яғни “сатылы” деген ұғым нені білдіреді?
2. Для каждой из приведенных ниже схем (рис. 3.10, 3.11) запишите HTML-коды, которые им соответствуют.

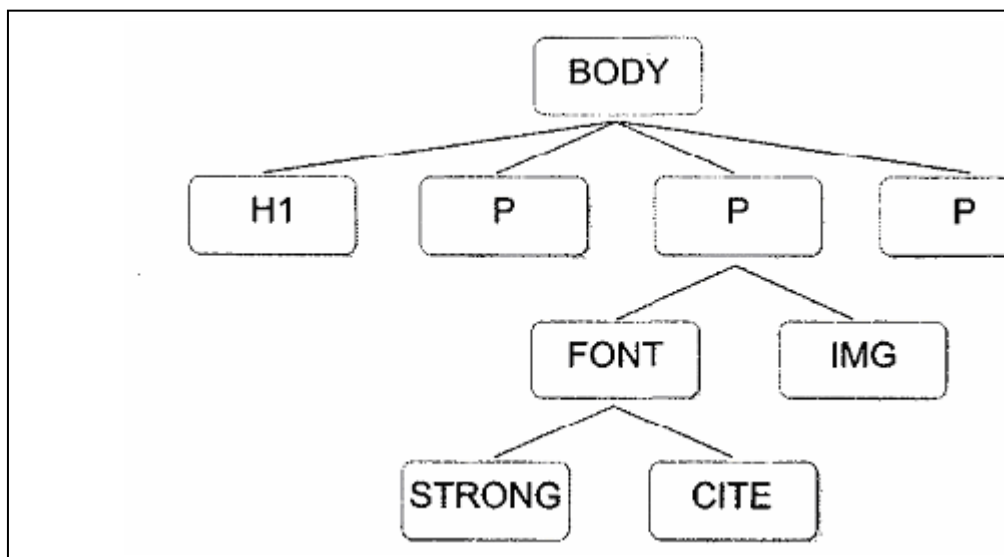


Рис.3.10. Схема 1

3. Опишите механизм наследования стилей.
4. Можно ли переопределить наследуемый стиль? Если да, то как?
5. Для тега `<P>` задан стиль `P {color:red}`. Каким цветом будет выведено слово «сложный» в следующих кодах:
 - а) `<H1>Это сложный пример</H1>`;
 - б) `<H1>Это <SPAN style="Color:Blue">сложный</SPAN> пример.</H1>`;
 - в) `<P>Это сложный пример.</P>`;
 - г) `<P>Это<SPAN style="color:blue">сложный</SPAN> пример.</P>`;
 - д) `<P>Это<FONT color= blue>сложный</SPAN> пример.</P>`;
 - е) `<P>Это<SPAN style="font-size:smaller">сложный</SPAN> пример.</P>`.
6. Что такое «контекстное определение»?
7. Задан стиль `P EM {color:red}`. Каким цветом будет выведено слово «сложный» в следующих кодах:
 - а) `<H1>Это сложный пример</H1>`;
 - б) `<H1>Это <EM>сложный</EM> пример</H1>`;
 - в) `<P>Это сложный пример.</P>`;
 - г) `<P>Это <EM>сложный</EM> пример</P>`?

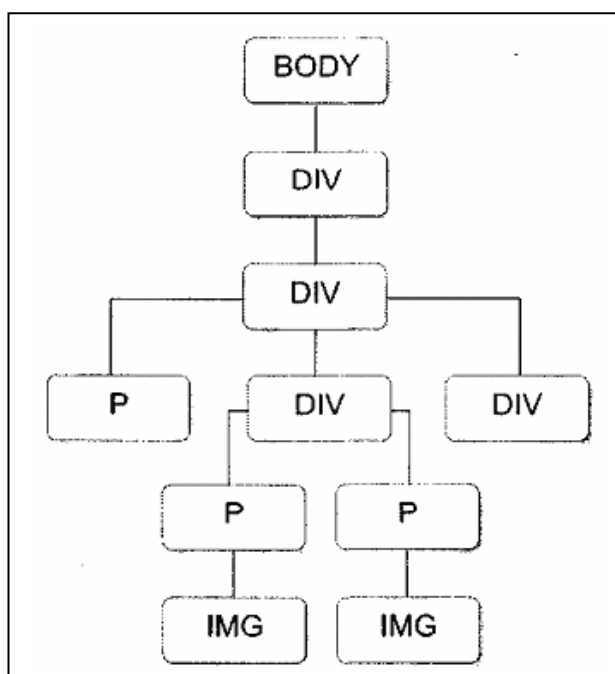


Рис. 3.11. Схема 2

8. Постройте иерархическое дерево для каждой приведенной ниже программы.

Программа 1	Программа 2
<BODY>	<BODY>
<DIV>	<DIV>
...	<DIV>
</DIV>	...
<DIV>	</DIV>
<DIV>	<DIV>
<DIV>	...
<DIV>	<DIV>
...	</DIV>
</DIV>	<DIV>
</DIV>	...
<DIV>	</DIV>
<DIV>	</DIV>
...	<DIV>
</DIV>	<DIV>
</DIV>	...
<DIV>	</DIV>
...	</DIV>
</DIV>	<DIV>
</BODY>	</BODY>

9. Задан стиль P, EM {color:red}. Каким цветом будет выведено слово «сложный» в следующих кодах:

- а) <H1>Это сложный пример</H1>;
б) <H1>Это сложный пример</H1>;
в) <P>Это сложный пример.</P>;
г) <P>Это сложный пример</P>.

10. Опишите назначение тегов `<div>` и `<span>`. В чем сходство работы этих тегов, а в чем различие?

11. Что такое «абсолютное позиционирование», каким стилевым указанием оно задается?

12. Что принимается за начало координат при абсолютном позиционировании?

13. По какому правилу абсолютно позиционированные области накладываются друг на друга?

14. Какие значения может принимать стилевое свойство `z-index` и как оно влияет на экранный образ объекта?
15. Что понимается под каскадированием стилевых указаний?

Задания

При подготовке заданий, представленных ниже, рекомендуется использовать справочник свойств CSS, в Приложениях к учебникам.

1. Сделайте страницу, в которой все абзацы выровнены по левому и правому краям, и имеют красную строку в один сантиметр.

2. Создайте новый стиль `P.def`. Абзацы этого стиля должны выравниваться по ширине, иметь отступ слева 2 см и справа 1 см. Расположите на странице обычные абзацы вместе с абзацами `P.def`.

3. Определите стили для написания старой и новой цены товара. Старая цена — серого цвета, перечеркнутая. Новая цена — красного цвета, на 50% более крупного кегля, чем остальной текст. Напишите список товаров со старыми и новыми ценами.

4. Определите два стиля. В первом стиле:

- буквы должны печататься коричневым по светло-серому фону;
- расстояние между содержимым и рамкой элемента должно составлять 0,5 см;
- текст выравнивается по левому и правому краям.

Во втором стиле:

- фон бирюзовый;
- расстояние между содержимым и рамкой элемента должно составлять 0,5 см;
- поля слева и справа от элемента по 1 см;
- рубленый шрифт.

Подготовьте документ с двумя разделами. Первый раздел определяется первым стилем, второй — вторым. Второй раздел должен быть вложен в первый, чтобы было видно наследование. Какие стилевые указания наследуются во втором разделе, а какие нет?

5. Сделайте стиль `.nb` такой, чтобы элемент заключался в рамку (бorders), занимал (по ширине) половину окна браузера (независимо от его размера), был расположен у левого края, а остальные элементы страницы «обтекали» бы этот элемент справа.

6. Подготовьте стиль для абзаца, у которого слева и справа проводится вертикальная черта (на всю высоту абзаца).

7. Подготовьте стиль абзаца «подложка» (`.ground`) и стиль «надпись» (`.poster`).

Используя созданные стили, создайте страницу с надписью поверх подложки.

Подложка пишется очень крупными буквами мягкого светло-серого цвета. Надпись пишется коричневыми буквами обычного размера.

8. Используя стили, сделайте страницу, в которой текст выводится в две колонки (как в газете). Таблицы при этом не используются.

9. Постройте на экране две области с линейками прокрутки и поместите в них информационные элементы.

10. Используя свойство `z-index`, постройте на экране несколько перекрывающихся друг друга областей.