

CSS

Лекція 3

Лекція 3

- Введение
- Селекторы
- Специфичность
- Важность
- Наследование
- Каскад
- Значения и единицы измерения
- Добавление CSS на страницу
- Типы устройств

Вступ

CSS - мова стилів, що визначає відображення HTML-документів.

CSS працює зі шрифтами, кольором, полями, рядками, висотою, шириною, фоновими зображеннями, позиціонуванням елементів і багатьма іншими речами.

Основні етапи розвитку

- CSS рівень 1 (1996, 1999) - параметри шрифтів, кольору, ...
- CSS рівень 2 (12 травня 1998), CSS 2.1 (7 червня 2011) - блокова верстка, селектори, ...
- CSS рівень 2.1 (7 червня 2011)
- CSS рівнів 3 знаходиться в стадії розробки - трансформації, анімація, ...
- CSS рівнів 4 розробляється з 29 вересня 2011 року.

Вступ

HTML – це трафарет

`<body>`

`<h1>Основний заголовок</h1>`

`<p>Текст параграфа 1</p>`

`<h2>Підзаголовок 1</h2>`

`<p>Текст параграфа 2</p>`

`<h2>Підзаголовок 1</h2>`

`<p>Текст параграфа 3</p>`

`</body>`

Основний заголовок

Текст параграфа 1

Підзаголовок 1

Текст параграфа 2

Підзаголовок 1

Текст параграфа 3

Основний заголовок

Текст параграфа 1

Підзаголовок 1

Текст параграфа 2

Підзаголовок 2

Текст параграфа 3

Вступ

Розмітка, але вже з доданими тегами оформлення

```
<body bgcolor="silver">
```

```
  <h1 align="center"><font size="16px" color="red"  
face="Tahoma">Основний заголовок</font></h1>
```

```
  <p><font size="8px" color="gray" face="Arial">Текст  
параграфа 1</font></p>
```

```
  <h2><font size="10px" face="Tahoma">Підзаголовок  
1</font></h2>
```

```
  <p><font size="8px" color="gray" face="Arial">Текст  
параграфа 2</font></p>
```

```
  <h2><font size="10px" face="Tahoma">Підзаголовок  
2</font></h2>
```

```
  <p><font size="8px" color="gray" face="Arial">Текст  
параграфа 3</font></p>
```

```
</body>
```

Вступ

CSS: причини появи

- Надлишковість HTML
- Більше можливостей для оформлення
- Спрощення оформлення сторінки
- Поділ контенту і візуальної складової сторінки

```
<body>
  <h1>Основний заголовок</h1>
  <p>Текст параграфа 1</p>
  <h2>Підзаголовок 1</h2>
  <p>Текст параграфа 2</p>
  <h2>Підзаголовок 1</h2>
  <p>Текст параграфа 3</p>
</body>
```

```
<style>
  body {
    background: #f1f1f1;
  }
  h1 {
    font-size: 34px;
    color: red;
    font-face: Tahoma;
    text-align: center;
  }
  h2 {
    font-size: 24px;
    font-face: Tahoma;
  }
  p {
    font-size: 18px;
    color: gray;
    font-face: Arial;
  }
</style>
```

Селектори



селектор ,

селектор ,

селектор { /* блок объявления стилей */

свойство : значение ;

свойство : значение ;

свойство : значение ;

свойство : значение ;

свойство : значение

}

Селектори

```
p {  
    font-size: 12px;      /* неизвестное значение (правильно  
font-size) */  
    color: ultraviolet;  /* неизвестное значение (нет  
такого цвета) */  
}  
  
h1 {  
    font: italic 40px Tahoma; /* три значения для одного  
свойства */  
}
```


Селектори

Групування селекторів

```
h1, p {  
    color: blue;  
}
```

Універсальний селектор

```
* {  
    color: green;  
}
```

Основний заголовок

Текст параграфа 1

Підзаголовок 1

Текст параграфа 2

Підзаголовок 1

Текст параграфа 3

Основний заголовок

Текст параграфа 1

Підзаголовок 1

Текст параграфа 2

Підзаголовок 1

Текст параграфа 3

Селектори

Селектори класів

```
.caution {  
    color: red;  
}
```

```
<h2 class = "caution">Важливий заголовок</h2>
```

```
<p>Параграф, в якому є <span class = "caution"> важливий  
текст </span> </p>
```

Важливий заголовок

Параграф, в якому є **важливий текст**

Селектори

Множинні класи

```
<div class="heading red">Червоний заголовок</div>
```

```
<div class="red">Червоний заголовок</div>
```

```
div.heading {  
    font-size: 50px;  
}
```

```
div.red {  
    color: red;  
}
```




Селектори

Селектори ідентифікаторів

Атрибут `id` – задає унікальний ідентифікатор елемента.

```
<div id="red"> Червоний заголовок </div>
```

```
div#heading {  
    font-size: 50px;  
}  
  
div#red {  
    color: red;  
}
```



Червоний заголовок

Селектори

```
<div id="heading red"> Червоний заголовок </div>
```

```
div#heading {  
    font-size: 50px;  
}  
div#red {  
    color: red;  
}
```



Селектори

Універсальний селектор

```
* {  
    color: red;  
}
```

```
<h1>Основний заголовок</h1>  
<p>Текст параграфа 1</p>  
<h2>Підзаголовок 1</h2>  
<p>Текст параграфа 2</p>  
<h3>Підзаголовок 2</h3>  
<p>Текст параграфа 3</p>
```

Основний заголовок

Текст параграфа 1

Підзаголовок 1

Текст параграфа 2

Підзаголовок 2

Текст параграфа 3

Селектори

Комбінування селекторів

```
div.myclass { color: red; }  
div.class1.class2 { color: red; }  
div#id1 { color: red; }  
div#id1.class1 { color: red; }  
div#id1#id2 { color: red; }  
*.class1 { color: red; }  
.class1 { color: red; }
```

Селектори

Селектори атрибутів

```
.italic {  
    font-style: italic;  
}
```

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

```
<p class="italic">Цей параграф буде виділений курсивним шрифтом.</p>
```

```
<p class="italic">Цей параграф буде виділений курсивним шрифтом.</p>
```

Цей параграф буде виділений курсивним шрифтом.

Цей параграф буде виділений курсивним шрифтом.

Селектори

Простий вибір атрибутів

```
[title] { color: red; }
```

Вибір на підставі конкретного значення атрибуту

```
a[target="_blank"] { color: red; }
```

Вибір по частковому значенню атрибуту

* Будь-яка частина рядка

\$ кінець рядка

^ початок рядка

```
[href*="google"] { color: red; }
```

```
[href^="https://"] { color: red; }
```

```
[href$="google.com"] { color: red; }
```

Селектори псевдокласів

Динамічні псевдокласи

```
a:link { color: blue }  
a:active { color: red }  
a:hover { color: green }  
a:visited { color: purple }  
a:focus { color: yellow }
```

Червоний заголовок

Псевдокласи стану

```
<input type="text" placeholder="У фокусі зміниться колір"  
<button>Якщо клікнути зміниться колір</button>  
<a href="/">Під час наведення зміниться колір</a>
```

```
input: focus {  
  background: blue;  
}  
button: active {  
  background: green;  
}  
a: hover {  
  color: red;  
}
```



У фокусі зміниться колір

Якщо клікнути зміниться колір

Під час наведення зміниться колір

Псевдокласи стану

- `:enabled`
- `:disabled`
- `:checked`
- `:indeterminate`
- `:read-only`
- `:valid`

Структурные псевдоклассы

```
li:first-child  
{  
  color: green;  
}
```

<p>Способы подключения
CSS на страницу</p>

```
<ul>  
  <li>Связанные стили</li>  
  <li>Глобальные стили</li>  
  <li>Внутренние стили</li>  
</ul>
```

Способы подключения CSS на страницу

- Связанные стили
- Глобальные стили
- Внутренние стили

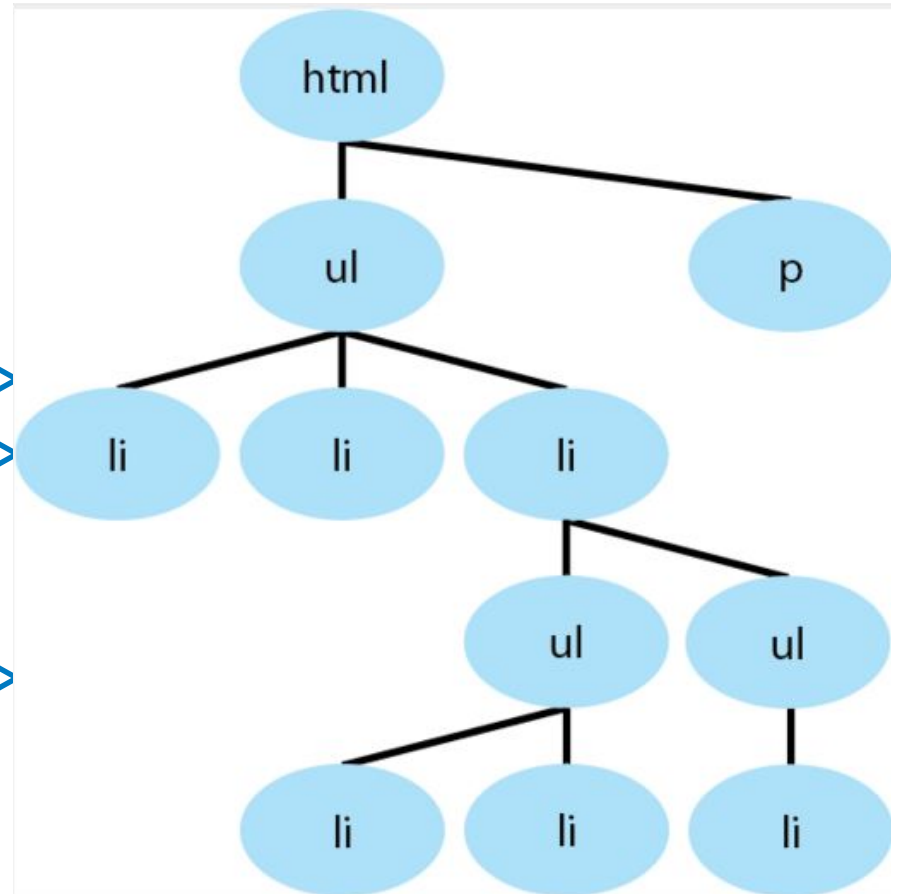
Структурные псевдоклассы

Структурные псевдоэлементы

- `:root`
- `:first-child/last-child`
- `:nth-child() /nth-last-child()`
- `:first-of-type/ :last-of-type`
- `:nth-of-type() /nth-last-of-type()`
- `:only-child/only-of-type`
- `:empty`

Структурные псевдоклассы

```
<ul> ul 1.1
  <li> li 2.1 </li>
  <li> li 2.2 </li>
  <li> li 3.2
    <ul> ul 3.1
      <li> li 4.1 </li>
      <li> li 4.2 </li>
    </ul>
    <ul> ul 3.2
      <li> li 4.3 </li>
    </ul>
  </li>
</ul>
<p>p1.2</p>
```



Структурные псевдоклассы

```
<ul> ul 1.1
  <li> li 2.1 </li>
  <li> li 2.2 </li>
  <li> li 3.2
    <ul> ul 3.1
      <li> li 4.1 </li>
      <li> li 4.2 </li>
    </ul>
    <ul> ul 3.2
      <li> li 4.3 </li>
    </ul>
  </li>
</ul>
<p>p1.2</p>
```

ul 1.1

li 2.1

li 2.2

li 3.2

ul 3.1

li 4.1

li 4.2

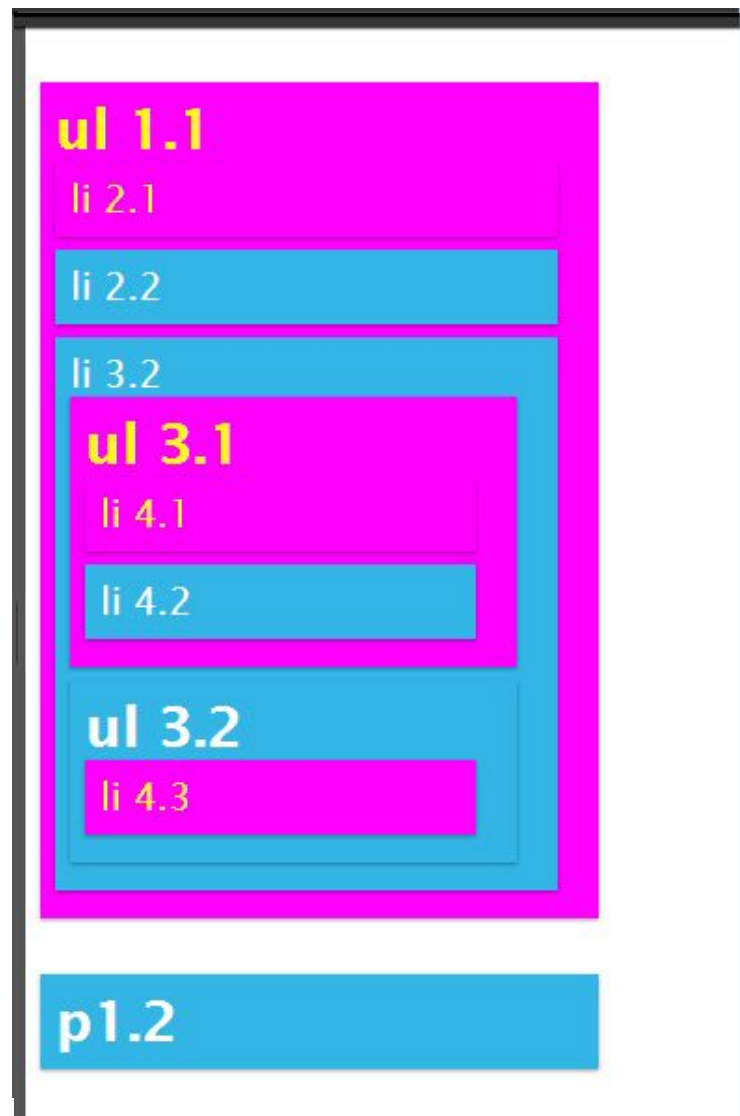
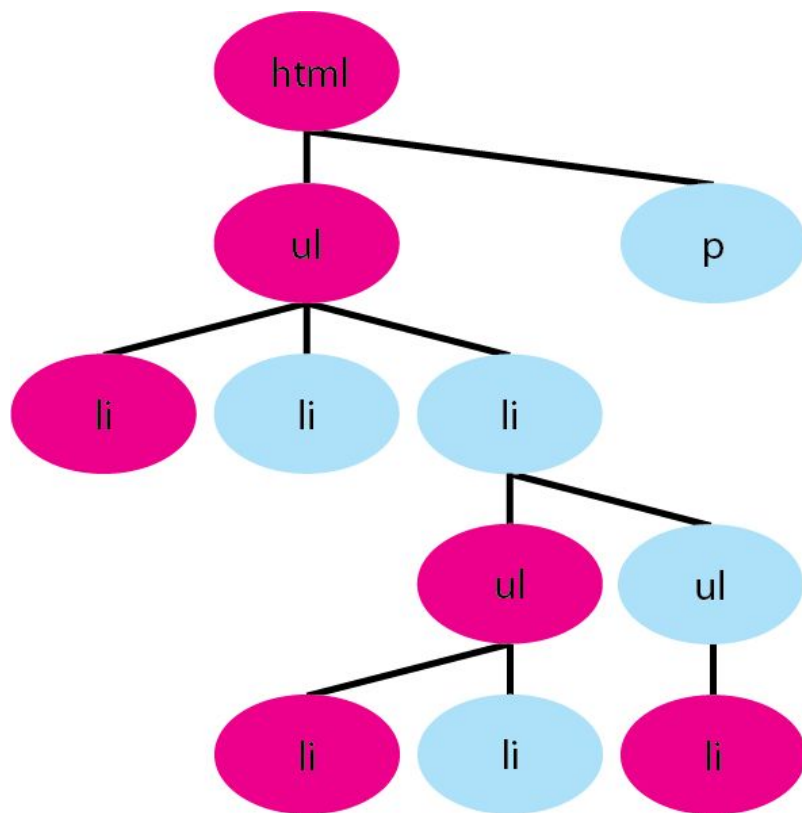
ul 3.2

li 4.3

p1.2

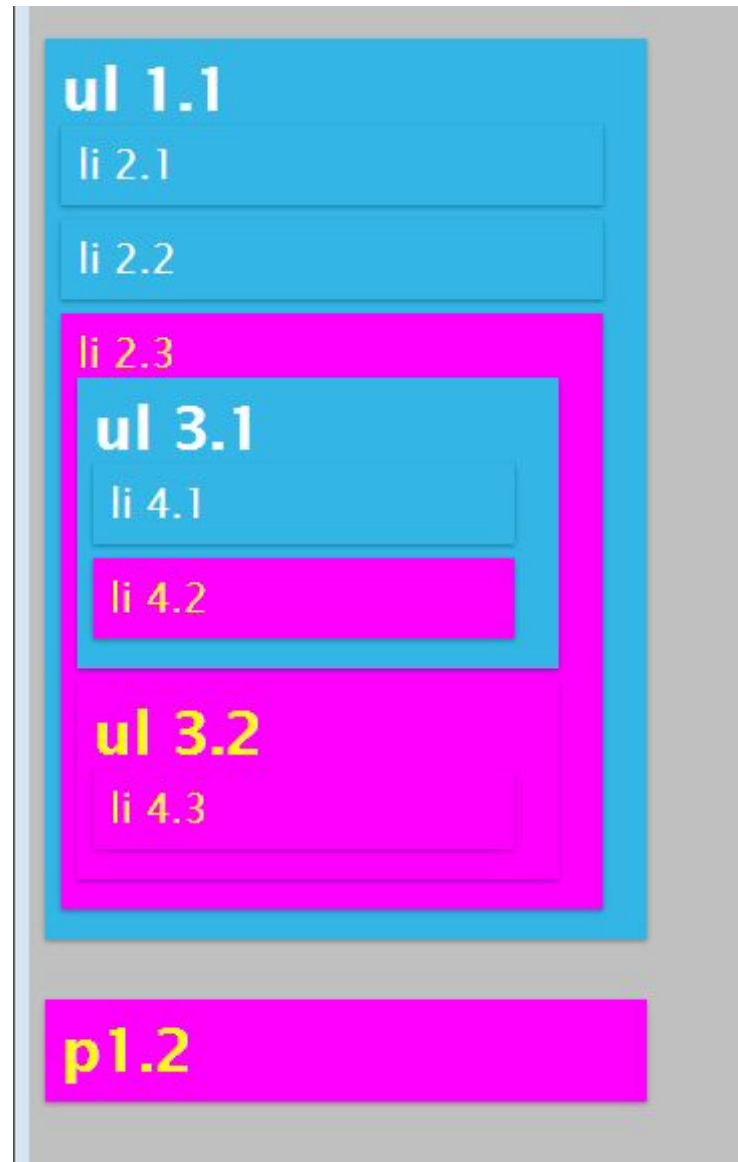
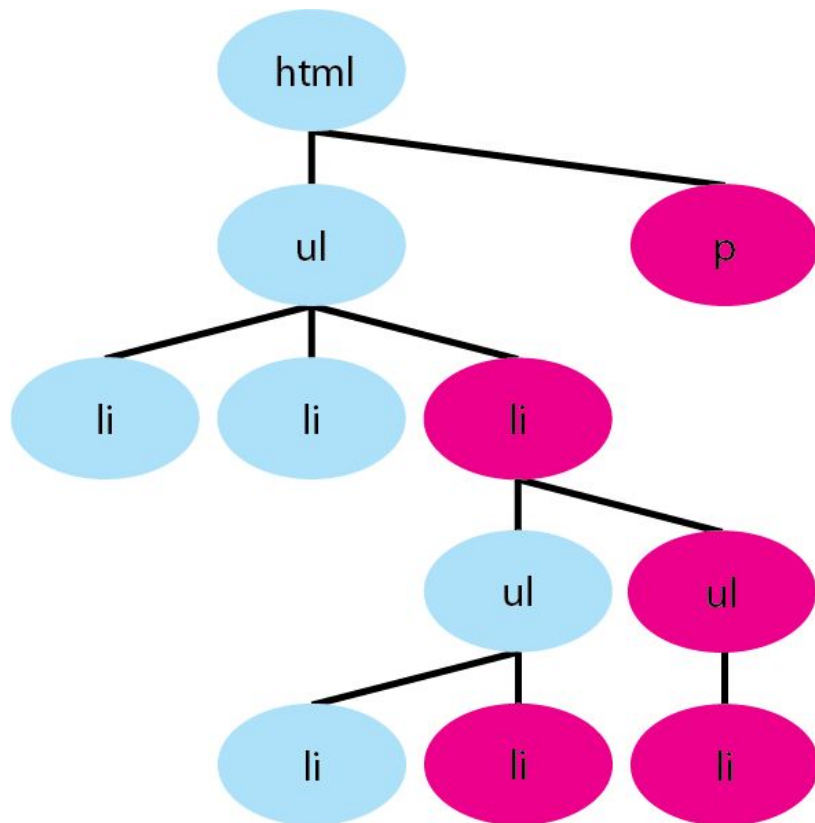
Структурные псевдоклассы

```
:first-child {  
  background-color :#FF00FF;  
  color: yellow;  
}
```



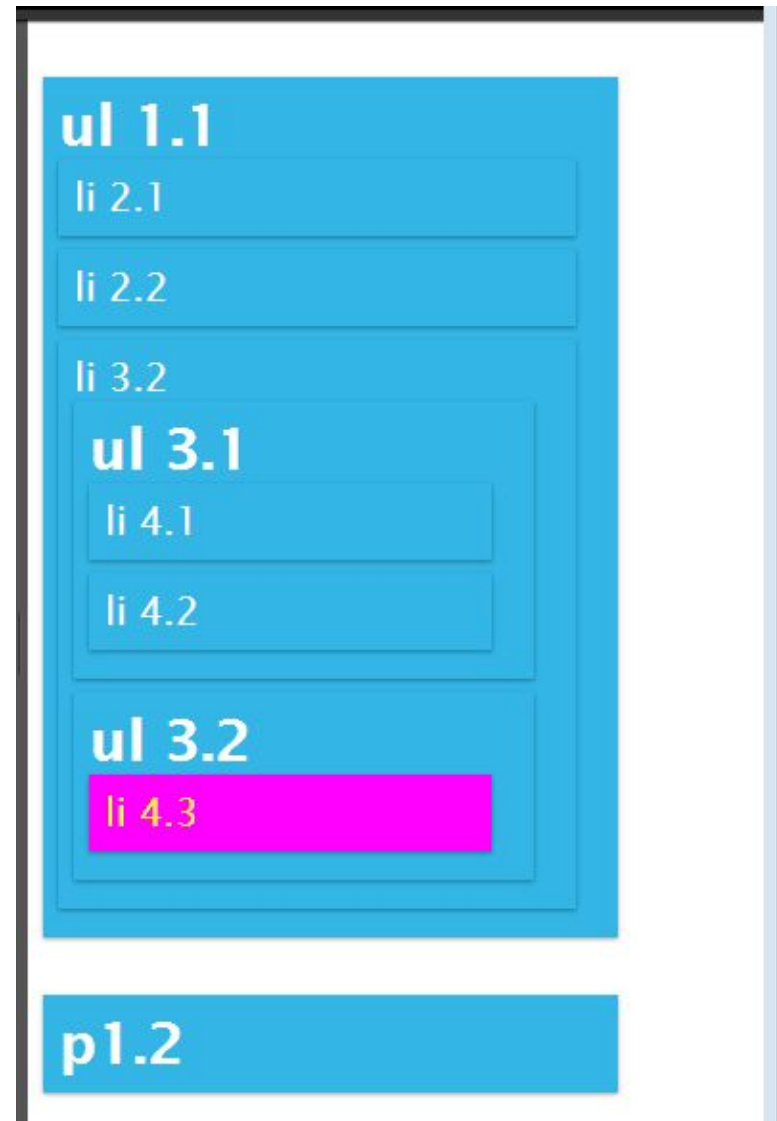
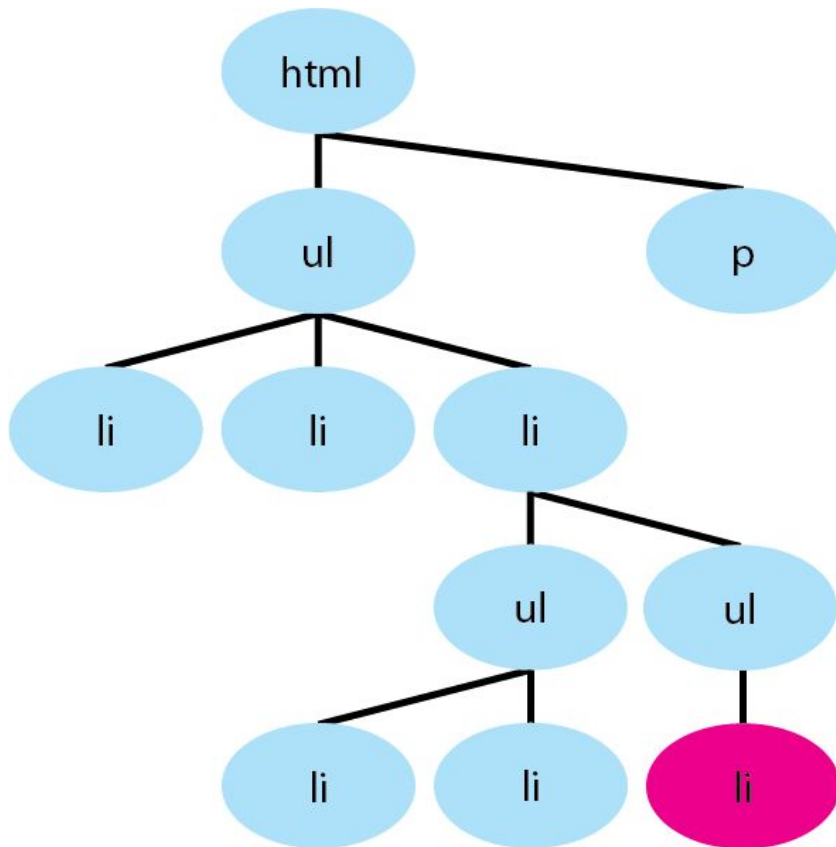
Структурные псевдоклассы

```
:last-child {  
  background-color :#FF00FF;  
  color: yellow;  
}
```



Структурные псевдоклассы

```
:only-child {  
  background-color :#FF00FF;  
  color: yellow;  
}
```



Структурные псевдоклассы

```
:nth-child(2n) {  
  background-color: #FF00FF;  
  color: yellow;  
}
```



Структурные псевдоклассы

```
li:nth-of-type(2n) {  
  background-color :#FF00FF;  
  color: yellow;  
}
```

ul 1.1

li 2.1

li 2.2

li 2.3

ul 3.1

li 4.1

li 4.2

ul 3.2

li 4.3

p1.2

Структурные псевдоклассы

```
:nth-of-type(2n) {  
  background-color: #FF00FF;  
  color: yellow;  
}
```



Селекторы псевдоэлементов

```
::first-letter { color: red }
```

```
::first-line { color: blue }
```

Перший рядок

Другий рядок

Перший рядок

Другий рядок

Селекторы псевдоэлементов

```
body:before {  
  content: '=початок=';  
  color: green;  
}
```

```
body:after {  
  content: '=кінець=';  
  color: red;  
}
```

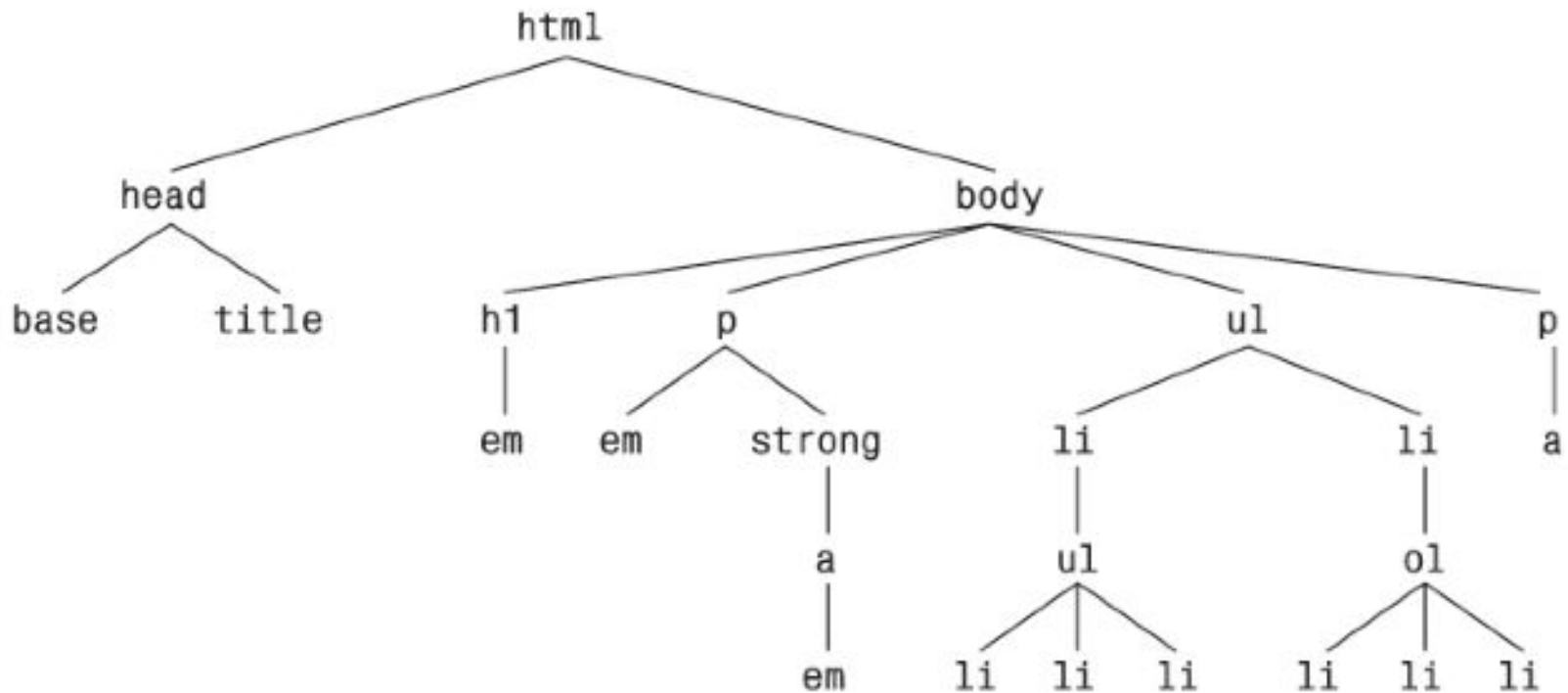
=початок=

CSS

Каскадные таблицы стилей (CSS – Cascading Style Sheets) – мощный механизм управления представлением отдельных документов или их наборов.

=кінець=

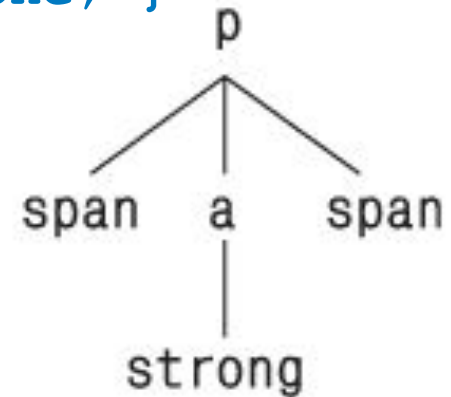
Отношения родитель-потомок



Отношения родитель-потомок

Селекторы потомков

```
nav a { color: red; text-decoration: none; }
```



Выбор дочерних элементов

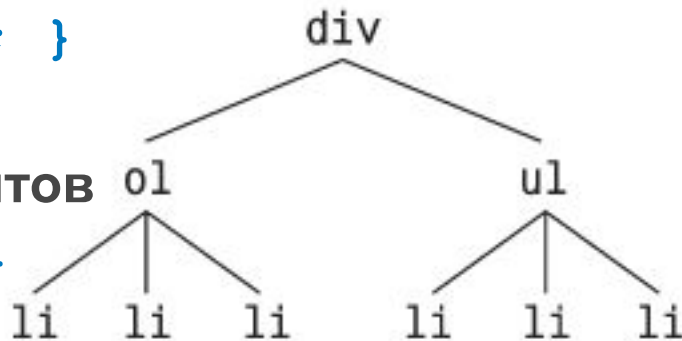
```
h1 > strong { color: red; }
```

Выбор соседних элементов

```
h1 + p { margin-top: 0; }
```

Выбор сестринских элементов

```
li ~ li { color: blue; }
```



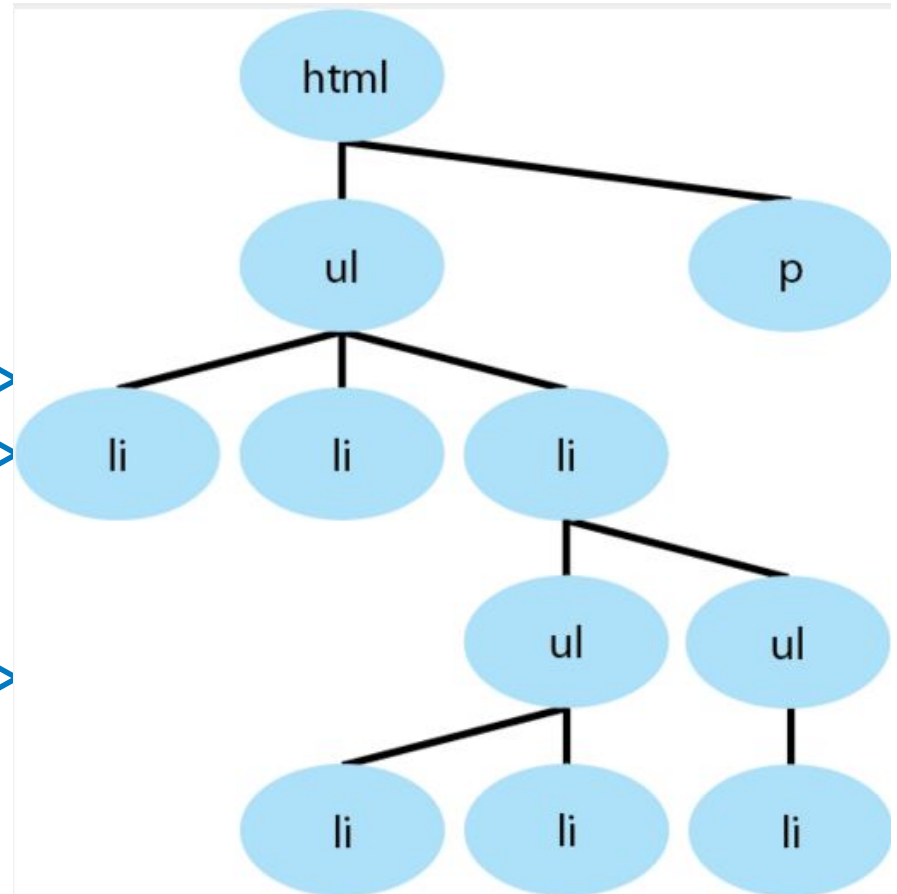
Селектор отрицания

```
td:empty { background: red; }
```

```
td:not(:empty) { background: green; }
```

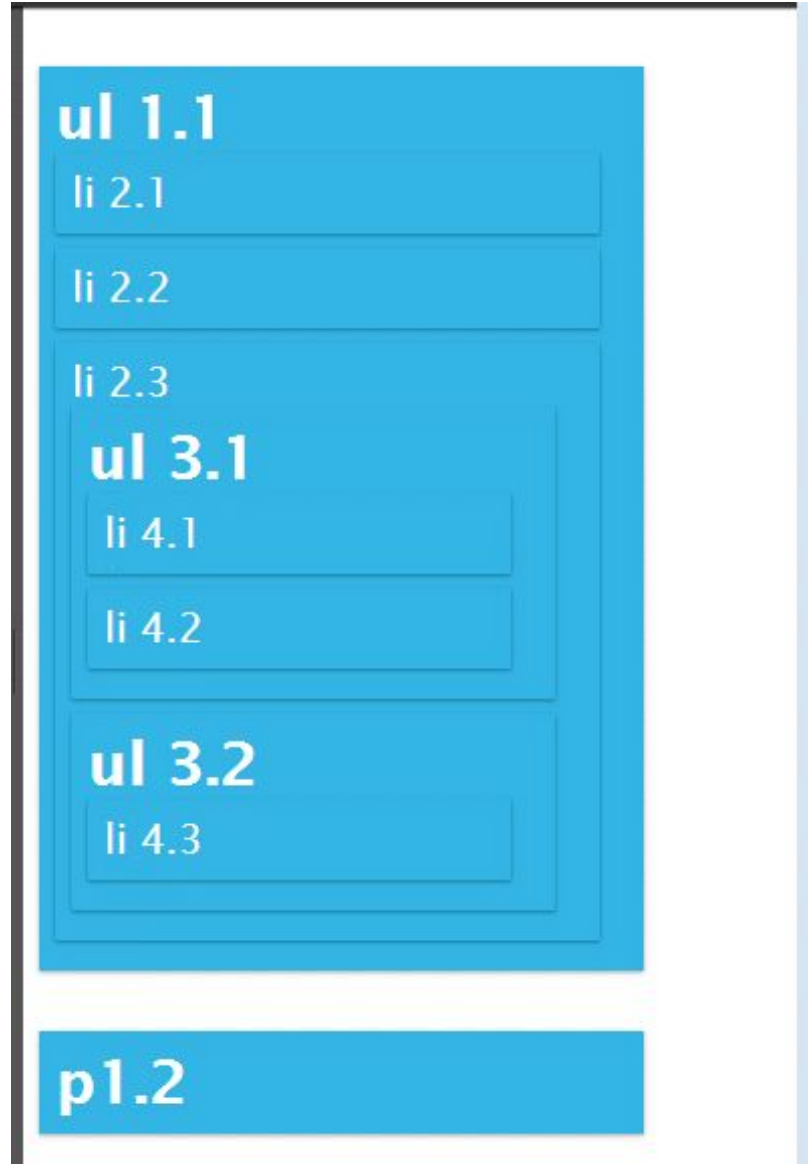
Отношения родитель-потомок

```
<ul> ul 1.1
  <li> li 2.1 </li>
  <li> li 2.2 </li>
  <li> li 2.3
    <ul> ul 3.1
      <li> li 4.1 </li>
      <li> li 4.2 </li>
    </ul>
    <ul> ul 3.2
      <li> li 4.3 </li>
    </ul>
  </li>
</ul>
<p>p1.2</p>
```



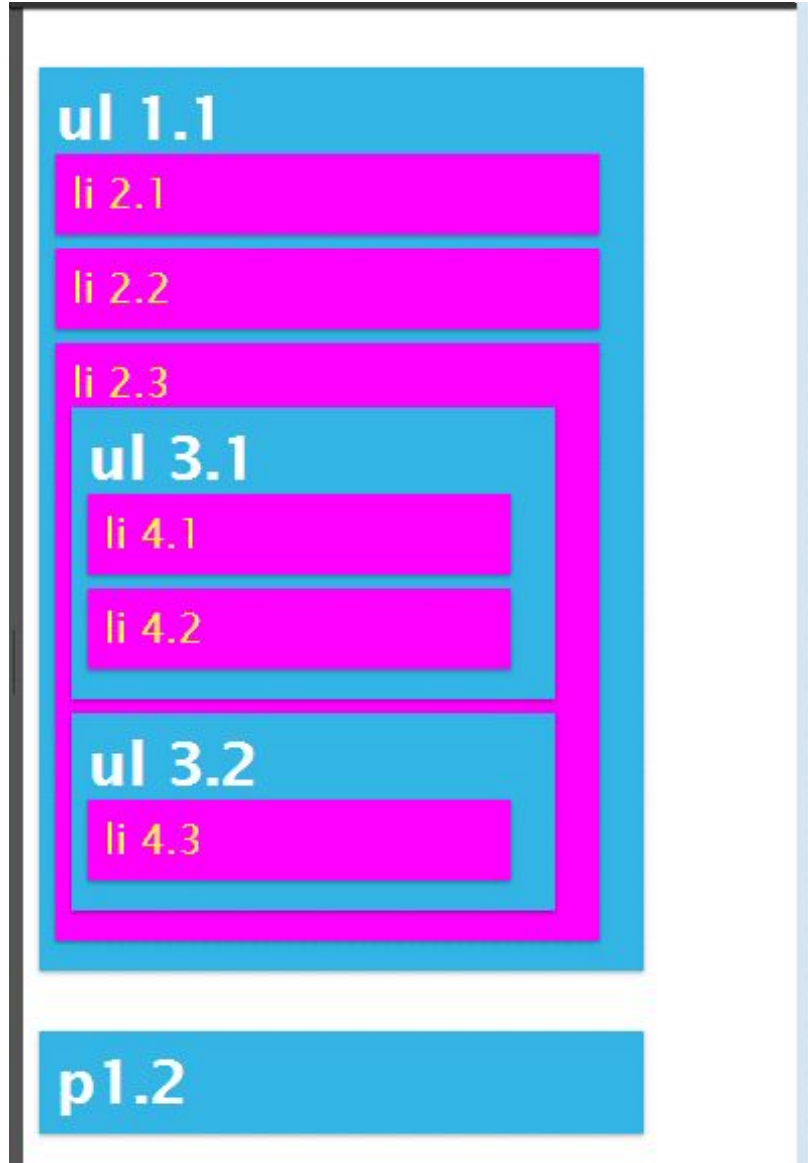
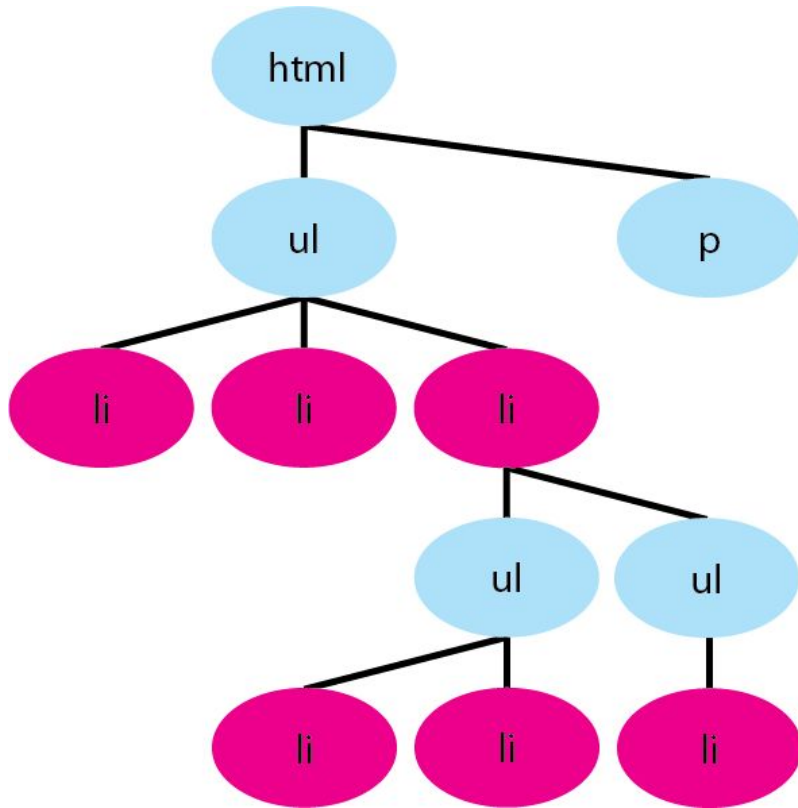
Отношения родитель-потомок

```
<ul> ul 1.1
  <li> li 2.1 </li>
  <li> li 2.2 </li>
  <li> li 2.3
    <ul> ul 3.1
      <li> li 4.1 </li>
      <li> li 4.2 </li>
    </ul>
  <ul> ul 3.2
    <li> li 4.3 </li>
  </ul>
</li>
</ul>
<p>p1.2</p>
```



Отношения родитель-потомок

```
ul li {  
  background-color :#FF00FF;  
  color: yellow;  
}
```



Отношения родитель-потомок

```
li li{  
  background-color :#FF00FF;  
  color: yellow;  
}
```

ul 1.1

li 2.1

li 2.2

li 2.3

ul 3.1

li 4.1

li 4.2

ul 3.2

li 4.3

p1.2

Отношения родитель-потомок

```
body > *{  
  background-color :#FF00FF;  
  color: yellow;  
}
```

ul 1.1

li 2.1

li 2.2

li 2.3

ul 3.1

li 4.1

li 4.2

ul 3.2

li 4.3

p1.2

Отношения родитель-потомок

```
li:first-child + li{  
  background-color :#FF00FF;  
  color: yellow;  
}
```

ul 1.1

li 2.1

li 2.2

li 2.3

ul 3.1

li 4.1

li 4.2

ul 3.2

li 4.3

p1.2

Отношения родитель-потомок

```
li:first-child ~ li{  
  background-color :#FF00FF;  
  color: yellow;  
}
```

ul 1.1

li 2.1

li 2.2

li 2.3

ul 3.1

li 4.1

li 4.2

ul 3.2

li 4.3

p1.2

Отношения родитель-потомок

```
li:not(:first-child) {  
  background-color :#FF00FF;  
  color: yellow;  
}
```

ul 1.1

li 2.1

li 2.2

li 2.3

ul 3.1

li 4.1

li 4.2

ul 3.2

li 4.3

p1.2

Специфичность

```
h1 {  
  color: red;  
}
```

```
body h1 {  
  color: green;  
}
```

CSS

Каскадные таблицы стилей (CSS – Cascading Style Sheets) – мощный механизм управления представлением отдельных документов или их наборов.

Специфичность

Значение специфичности : **0, 0, 0, 0**.

- Для каждого указанного в селекторе значения идентификатора к специфичности добавляется **0, 1, 0, 0**.
- Для каждого указанного в селекторе имени класса, псевдокласса или атрибута к специфичности добавляется **0, 0, 1, 0**.
- Для каждого заданного в селекторе элемента и псевдоэлемента к специфичности добавляется **0, 0, 0, 1**.
- Универсальный селектор не учитывается

Специфичность

Примеры:

```
h1 { color: red; } /* 0, 0, 0, 1 */
p em { color: purple; } /* 0, 0, 0, 2 */
.grape { color: purple;} /* 0, 0, 1, 0 */
*.bright { color: yellow;} /* 0, 0, 1, 0 */
p.bright em.dark{ color: maroon;} /* 0, 0, 2, 2 */
#id216 { color: blue; } /* 0, 1, 0, 0 */
div#sidebar [href] { color: silver; } /* = 0, 1, 1, 1 */
* { color: yellow; } /* 0, 0, 0, 0 */
```

```
h1 {color: red;} /* 0, 0, 0, 1 */
body h1 {color: green;} /* 0, 0, 0, 2 (победитель) */
```

Специфичность

Специфичность с группированными селекторами

```
h1, h2.section { color: silver; background: black; }
```

```
h1 { color: silver; background: black; }
```

```
/* 0, 0, 0, 1 */
```

```
h2.section { color: silver; background: black; }
```

```
/* 0, 0, 1, 1 */
```

Специфичность селекторов идентификаторов и атрибутов

```
#uniq { color: green; }
```

```
/* 0, 1, 0, 0 */
```

```
[id='uniq'] { color: red; }
```

```
/* 0, 0, 1, 0 */
```

Специфичность

Инлайн и !important

```
<div style="color: blue"></div> /* 1, 0, 0, 0 */
```

```
div {  
    color: green !important;  
}
```

- Инлайн стили по-умолчанию приоритетнее стилей в CSS ()
- Стили в CSS с !important приоритетнее инлайн стилей
- Инлайн стили с !important приоритетнее всего

Наследование

1. Некоторые стили применяются не только к целевому элементу, но и к его потомкам
2. Унаследованные стили не имеют специфичности, т.е их всегда перебивает любой селектор
3. Именно поэтому не рекомендуют пользоваться селектором *

```
div {  
    color: green; /* 0 0 1 */  
}  
<div>  
    <abbr title="Cascading Style Sheets">  
        CSS  
    </abbr>  
    - Каскадные таблицы стилей  
</div>
```

CSS - Каскадные таблицы стилей

Наследование

```
* {  
    color: red; /* 0 0 0 */  
}  
div {  
    color: green; /* 0 0 1 */  
}
```

CSS - Каскадные таблицы стилей

```
<div>  
    <abbr title="Cascading Style Sheets">  
        CSS  
    </abbr>  
    - Каскадные таблицы стилей  
</div>
```

Наследование

Стили которые наследуются

- color
- cursor
- direction
- empty-cells
- font-family
- font-size
- font-weight
- font-style
- font-variant
- font
- letter-spacing
- list-style-type
- list-style-position
- list-style-image
- list-style
- line-height

Каскад

```
h1 { color: red; }  
h1 { color: blue; }
```

1. Найти все правила, содержащие селектор, сопоставляемый с данным элементом.
2. Провести сортировку согласно явной приоритетности всех применяемых к элементу объявлений
3. Провести сортировку всех объявлений, применяемых к элементу, согласно их специфичности.
4. Провести сортировку всех объявлений, применяемых к элементу, в соответствии с очередностью расположения.

Каскад

Сортировка по приоритетности и источнику

Порядок применения:

1. Стили браузера
2. Стили пользователя и/или плагинов браузера
3. Стили страницы
4. Стили страницы с !important
5. Стили пользователя и/или плагинов браузера с !important

Сортировка по порядку расположения:

```
h1 { color: red; }  
h1 { color: blue; } /* имеет более высокий приоритет*/
```

Значения и единицы измерения

Числовые значения:

- Целые числа (1, 2, 3...)
- Дробные числа (1.5)
- Процентные значения (50%)

Значения и единицы измерения

Абсолютные единицы измерения длины

- Миллиметры (mm)
- Сантиметры (cm)
- Дюймы (in)
- Пункты – используется в типографиях, 1 дюйм = 72 пункта (pt)
- Пики – 12 пунктов (pc)

Абсолютные единицы удобны при определении CSS для печати. В вебе же практически не используются

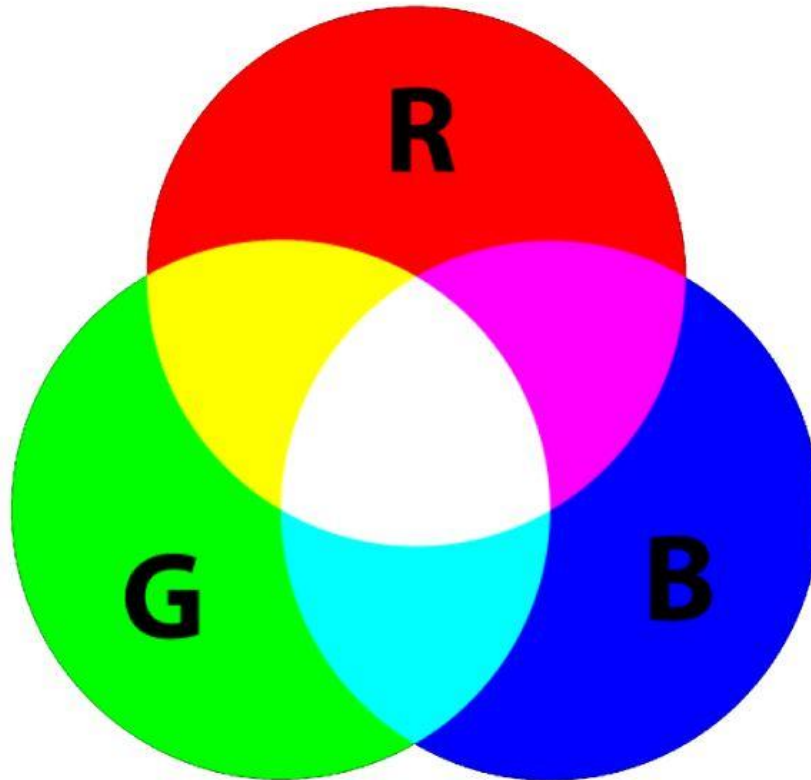
Относительные единицы измерения длины

- пикселы (px)
- em – зависит от размера шрифта (em)
- rem – зависит от размера шрифта корневого элемента (rem)
- ex – зависит от высоты символа x в данном шрифте
- ch – зависит от ширины символа 0 в данном шрифте
- vh/vw – 1/100 высоты и ширины viewport'a соответственно
- vmax – 1/100 от максимума между высотой и шириной viewport'a
- vmin – 1/100 от минимума между высотой и шириной viewport'a

Цвета

Способы задания красного цвета

- Именованные цвета – `red`
- Функциональный – `rgb(255, 0, 0)` или `rgba(255, 0, 0, 1)`
- Шестнадцатиричный – `#ff0000` или `#f00`



Цвета

```
h1 { color: fuchsia; }  
p { background: yellow; }
```

```
.red { color: #ff0000; }  
.orange { color: #eea837; }
```

```
.red { color: #f00; }           /* #f00 = #ff0000 */  
.gray { color: #888; }         /* #888 = #888888 */
```

```
p { color: rgb( 40%, 40%, 40% ); }  
p { color: rgb( 102, 102, 102 ); }
```

```
p { color: rgba( 40%, 40%, 40%, 0.5 ); }  
p { color: rgba( 102, 102, 102, 0.8 ); }
```


Добавление CSS на страницу

```
<div style="color: red"></div>
```

```
<style>@import (<url to css>)</style>
```

```
<style>div { color: red }</style>
```

```
<link rel="stylesheet" href="<url to css>">
```

Типы устройств

Обозначение	Типы устройств
all	Таблица стилей используется для всех устройств
aural	Для речевых синтезаторов
braille	Устройства для слепых людей
embossed	Страничные принтеры для слепых людей
handheld	Для устройств с небольшими экранами (мобильные телефоны, карманные компьютеры и т.д.)
print	Используется при выводе документа на печать
projection	Для проектора
screen	Экран компьютерного монитора
tty	Устройства, использующие символьную сетку экрана фиксированного шага, например телетайп
tv	Для экранов подобно телевизионным (низкое разрешение, ограниченная цветопередача, отсутствует прокрутка и т.д.)

Дякую!