

Internet- технології

Каскадні таблиці стилів.
CSS-препроцесори

Зміст

- призначення каскадних таблиць стилів;
- основні поняття CSS;
- способи підключення каскадних таблиць стилів;
- пріоритет стилів;
- CSS-препроцесори.

Рекомендовані джерела

- Специфікація <https://www.w3.org/TR/css3-values/>
- Про каскадність у CSS <https://css-live.ru/css/nikto-ne-znaet-css-specificnost-ne-kaskad.html> и <https://css-live.ru/css/pravilnaya-shpargalka-po-css-kaskadu.html>
- Рекомендації щодо іменування CSS-сущностей <https://habr.com/company/ruvds/blog/347194/>
- Про обробку CSS браузером <http://habrahabr.ru/post/224187/>
- Одиниці виміру у CSS <http://shpargalkablog.ru/2013/03/px-em-calc.html>,
<https://html5book.ru/edinicy-izmereniya-vh-vw-vmin-vmax/>
- Препроцесори <http://frontender.info/why-sass/>, <http://sass-scss.ru/documentation/sintaksis/> и <https://habrahabr.ru/post/140612/>

Призначення каскадних таблиць стилів

CSS (Cascading Style Sheets, каскадні таблиці стилів) – це засіб, що визначає зовнішній вигляд документа, написаного за допомогою мови розмітки (зазвичай, HTML). CSS використовується при створенні веб-сторінок для завдання кольору, шрифтів, розташування окремих блоків та інших аспектів представлення зовнішнього вигляду веб-документів.

Переваги CSS:

- дозволяє **розділити** структурований контент сторінки та її оформлення;
- можливість керування відображенням безлічі документів за допомогою **однієї** єдиної таблиці стилів;
- більш точний **контроль** за зовнішнім виглядом сторінок, порівняно з засобами оформлення у мові розмітки;
- можливість описати різні **представлення** для різних засобів відображення (екран, друк, тощо).

Синтаксис CSS

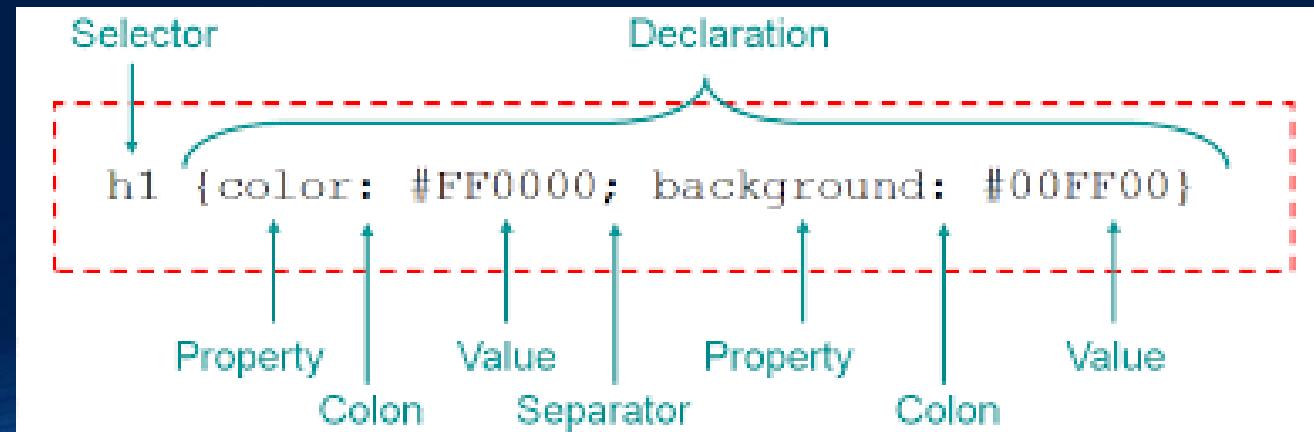
Таблиця стилів містить **стильові правила**.

Стильове правило складається із **селектора** та набора **стильових властивостей** із вказаними певними значеннями.

Селектор – визначає те, до чого пристосовується набір стильових властивостей (тобто, якась множина елементів сторінки).



```
body {  
    background-color: #FFoooo;  
}
```



Синтаксис CSS. Опис стилевого правила

селектор { властивість: значення;}

- Якщо **значення** складається з кількох слів, воно поміщається в одинарні чи подвійні лапки. Усередині рядка можна комбінувати типи лапок або додати перед лапкою слеш.
- Якщо потрібно перерахувати **кілька значень** стилової властивості, всі вони поділяються **пропуском**.
- Якщо необхідно перерахувати **альтернативні значення**, вони поділяються **комою**.
- Якщо необхідно визначити **більше однієї властивості**, то вони поділяються **крапкою з комою**.
- Селектори, котрим визначають однакові властивості, допускається **групувати**. Селектори у групі поділяються комами.

```
h1, h2, h3 {  
    font-family: 'Times New Roman', serif;  
    font-size: 14pt; /* розмір тексту */  
}
```

Синтаксис CSS. Приклад

Каскадність таблиць полягає у можливості успадкування вкладеними елементами стилів елементів-контейнерів. Крім того, стилі, підключенню різними способами, можуть бути перевизначені.

Опис стилів:

```
body {  
    font-family: Arial, Verdana, sans-serif;  
    font-size: 11pt;  
    background-color: #fofofo;  
    color: #333;  
}  
  
h1 {  
    color: #a52a2a;  
    font-size: 24pt;  
    font-family: Georgia, Times, serif;  
    font-weight: normal;  
}
```

Тіло документа:

```
<body>  
    <h1>Introduction</h1>  
    <p>A single CSS file can describe a common  
    "style" to be applied to many HTML, XHTML, and  
    XML documents.</p>  
</body>
```

Introduction

A single CSS file can describe a common "style" to be applied to many HTML, XHTML, and XML documents.

Одиниці виміру в CSS

Міри довжини, що використовуються при вказівці значень властивостей для вказівки розмірів:

- **відносний** розмір у відсотках (%)
- відносний розмір за допомогою словесного опису (larger, smaller, xx-small, x-small, small, medium, large, x-large, xx-large)
- відносно розміру шрифту: em – значення font-size батька, rem – font-size кореневого елемента документа
- відносно розміру вікна: vw – 1% від ширини, vh – 1% від висоти, vmin або vmax – менше/більше значення з vw та vh
- **абсолютний** розмір у друкарських одиницях – розмір може задаватися у пунктах (pt – points, точка ($1\text{pt} = 1/72\text{in}$)), піках (pc – picas ($1\text{pc} = 12\text{pt}$)), пікселях (px)
- абсолютний розмір у стандартних одиницях довжини – розмір може задаватися у сантиметрах (cm), міліметрах (mm), дюймах (in)

Варіанти вказівки кольору в CSS

Колір для тих властивостей, де це потрібно, може бути визначений одним із таких способів:

- за допомогою **назви** кольору: yellow, red, green, grey,..
- шістнадцятковим завданням кольору у форматі **#rrggbb**: #ffoooo, #883490, #ffffff,..
- шістнадцятковим завданням кольору у форматі **#rgb** (наприклад, #осо)
- десятковим завданням складових кольору у форматі **rgb(x,x,x)**, де "x" число від 0 до 255 (rgb(red, green, blue) – наприклад, rgb(255,0,0), rgb(100,23,78),..)
- **rgb(x%,x%,x%)** – де "x%" число від 0.0 до 100.0 (наприклад, 0%,80%,0%)

Підключення таблиць стилів

Існують такі способи використання стилів:

- **inline-опис** – використовуючи атрибут `style`, у якому можна задати значення властивостей елемента відповідно до синтаксису каскадних таблиць стилів;
- **опис у секції заголовка в стильовому блоці**, обмеженому тегами `<style>` та `</style>`;
- **описати у зовнішньому файлі**, що підключається до веб-документу за допомогою тега `<link>`, що задається у розділі `<head>`;

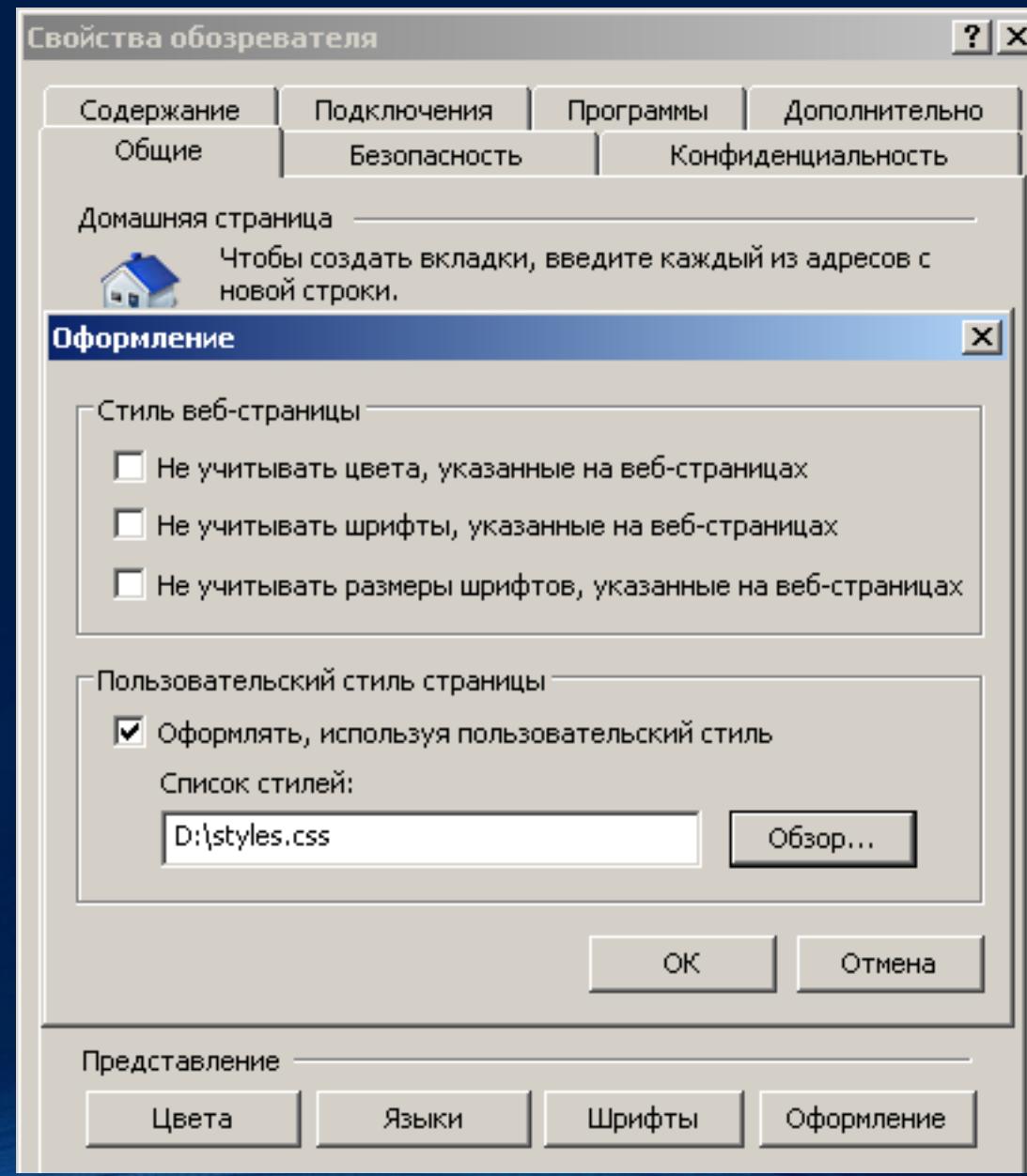
Один із проявів **каскадності** таблиць стилів у тому, що стилі, підключені різними способами, можуть перевизначати одне одного. Наведений вище перелік методів застосування стилів відповідає порядку перевизначення. Вищерозташований спосіб підключення може перевизначати нижчезаведений.

Пріоритет стилів

Пріоритети використання стилів, за зростанням:

- Найнижчим пріоритетом має стиль **браузера**;
- Наступним за значимістю є стиль, заданий користувачем браузера у його налаштуваннях;
- Найвищий пріоритет має стиль, заданий автором сторінки:
 - Найнижчий пріоритет має стилі, успадковані в документі елементом від своїх предків;
 - Вищий пріоритет мають стилі, задані у зовнішніх таблицях стилів, підключених до документа;
 - Ще більш високим пріоритетом мають стилі, що містяться в контейнерах **style** даного документа.
 - Ще більш високим пріоритетом мають стилі, оголошені безпосередньо в тезі даного елемента за допомогою **атрибуту style**;
 - І нарешті найвищий пріоритет мають стилі, оголошені автором сторінки за допомогою супровідного слова **!important**.

Приклад визначення стилю браузера



Підключення таблиці стилів. *inline*-опис

inline-опис – різновид внутрішніх таблиць стилів.

Стильові властивості можуть розміщуватися у веб-документі в тілі окремого тега (за допомогою його атрибуту **style**). Описане оформлення діятиме лише на вміст цього тега.

```
<body>
  <p style="font-size: 20px; color: green; font-family: arial, helvetica,
  sans-serif">
    Параграф із крупним текстом зеленого кольору.
  </p>
</body>
```

Параграф із крупним текстом зеленого кольору.

Підключення таблиць стилів у секції заголовка

Таблиця стилів може бути описана у самому документі, розміщуючись у ньому між тегами `<style>` і `</style>` (які, у свою чергу, розташовуються в цьому документі в секції заголовку, усередині тегу `<head>`).

Усі правила підключеної таким чином таблиці діють протягом усього документа.

```
<head>
  <title>Example</title>
  <style type="text/css">
    body {
      color: red;
    }
  </style>
</head>
<body>
  <p>This is a red page</p>
</body>
```

This is a red page

Підключення таблиці стилів. Зовнішній файл

Таблиці стилів можуть бути розташовані в окремих файлах, що мають формат CSS (це звичайний текстовий файл, в якому немає нічого, крім правил CSS і коментарів до них). Усі правила такої таблиці є дійсними для усіх документів, до яких була підключена.

Підключення виконується за допомогою тега `<link>` секції заголовка, який має атрибут `href` із значенням адреси файлу таблиці стилів, що підключається. Таким чином, можна підключати також таблицю стилів, яка знаходиться на іншому сайті.

```
<head>
  <title>Example</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <h1>Заголовок</h1>
  <p>Текст</p>
</body>
```

```
h1 {
  color: #000080;
  font-size: 200%;
  text-align: center;
}
p {
  padding-left: 20px;
}
```

Заголовок

Текст

Підключення таблиці стилів. Зовнішній файл

Підключення до веб-документу таблиці стилів, що розміщена у окремому css-файлі, можливе також за допомогою директиви @import, що міститься в цьому документі між тегами <style> та </style>, яка також вказує на адресу таблиці стилів.

Правило @import дозволяє імпортувати вміст файлу CSS у поточну стильову таблицю.

Підключення до веб-документу:

```
<style media="screen">  
    @import url(styles.css);  
</style>
```



Імпорт до складу стильового файлу:

```
@import "style/print.css";  
@import "style/palm.css";  
  
h1 {  
    color: #000080;  
    font-size: 200%;  
    text-align: center;  
}
```

CSS-препроцесори. Sass

Sass (Syntactically Awesome Stylesheets) – препроцесор CSS, що дозволяє створювати інструкції CSS із використанням програмних конструкцій замість статичних правил. Розширення файлів SASS можуть бути .sass і .scss – це залежить від вибраного синтаксису.

SCSS:

```
$blue: #3bbfce; $margin: 16px;  
$fontSize: 14px;  
.content {  
    border: 1px solid $blue;  
}  
.border {  
    padding: $margin / 2;  
    margin: $margin / 2;  
    border-color: $blue;  
}
```

Результат в CSS:

```
.content {  
    border: 1px solid #3bbfce;  
    color: #217882;  
}  
.border {  
    padding: 8px;  
    margin: 8px;  
    border-color: #3bbfce;  
}
```

Sass. Приклади

Фрагменти коду, що повторюються, можна зберегти в окремій змінній і вставляти в потрібних місцях за допомогою @mixin.

SCSS:

```
@mixin border-radius($radius) {  
    -webkit-border-radius: $radius;  
    -moz-border-radius: $radius;  
    border-radius: $radius;  
}  
.box-1 {  
    @include border-radius(10px);  
}  
.box-2 {  
    @include border-radius(5px);  
}
```

Результат в CSS:

```
.box-1 {  
    -webkit-border-radius: 10px;  
    -moz-border-radius: 10px;  
    border-radius: 10px;  
}  
.box-2 {  
    -webkit-border-radius: 5px;  
    -moz-border-radius: 5px;  
    border-radius: 5px;  
}
```

Sass. Компілятори

Sass можна використовувати як інструмент командного рядка (необхідно, щоб був встановлений Ruby). Тоді для запуску Sass із командного рядка використовується команда:

```
sass input.scss output.css
```

Також можливо вказати Sass стежити за файлом і автоматично компілювати його в CSS при будь-якій зміні:

```
sass --watch input.scss:output.css
```

Сторонні компілятори:

- CodeKit (платний, Mac)
- Compass.app (платний, Win/Mac)
- Hammer (платний, Mac)
- Koala (**безкоштовний**, Win/Mac)
- LiveReload (платний, Win/Mac)
- Mixture (**безкоштовний**, Win/Mac)
- Scout (**безкоштовний**, Win/Mac)

```
$globalVar: 10px; // global variable  
.enclosing {  
    $enclosingVar: 20px; // local variable  
    .closure {  
        $closureVar: 30px; // local variable  
        font-size: $closureVar + $enclosingVar + $globalVar; // 60px  
    }  
}
```

Контрольні запитання

- Для чого використовуються каскадні таблиці стилів?
- Яким чином здійснюється підключення таблиць стилів?
- У чому різниця при використанні між підключенням зовнішнього CSS-файлу за допомогою тегу `link` та його підключенням за допомогою директиви `@import`?
- У чому полягає каскадність таблиць стилів?
- Наведіть приклади стильових властивостей.

- Для чого використовуються CSS-препроцесори?