

QA Manual

Лекція 15. CSS

Що таке CSS

CSS(Cascading Style Sheets) — це мова таблиці стилів. Вона дозволяє вибірково застосовувати стилі відображення до елементів HTML(тегів). По суті ці таблиці описують зовнішній вигляд HTML документу.

CSS — описують правила форматування елементів за допомогою **властивостей** та допустимих **значень** цих властивостей. Для кожного елементу можна використовувати обмежений набір властивостей, решта властивостей не будуть мати на нього ніякого впливу.



Що таке CSS

Селектор — ім'я HTML елемента на початку набору правил. Він вибирає елемент(и) для застосування стилю.

Значення властивості — значення для обраного виду стиля HTML елемента. В прикладі далі це код кольору у шістнадцятирічній системі числення — **#ffffff;** (білий колір, після виду стилю обов'язково ставиться точка з комою).

Властивість — Вид стилю який буде застосовуватися до HTML елемент. В прикладі вище це — **color:** (після виду стилю обов'язково ставиться двокрапка).

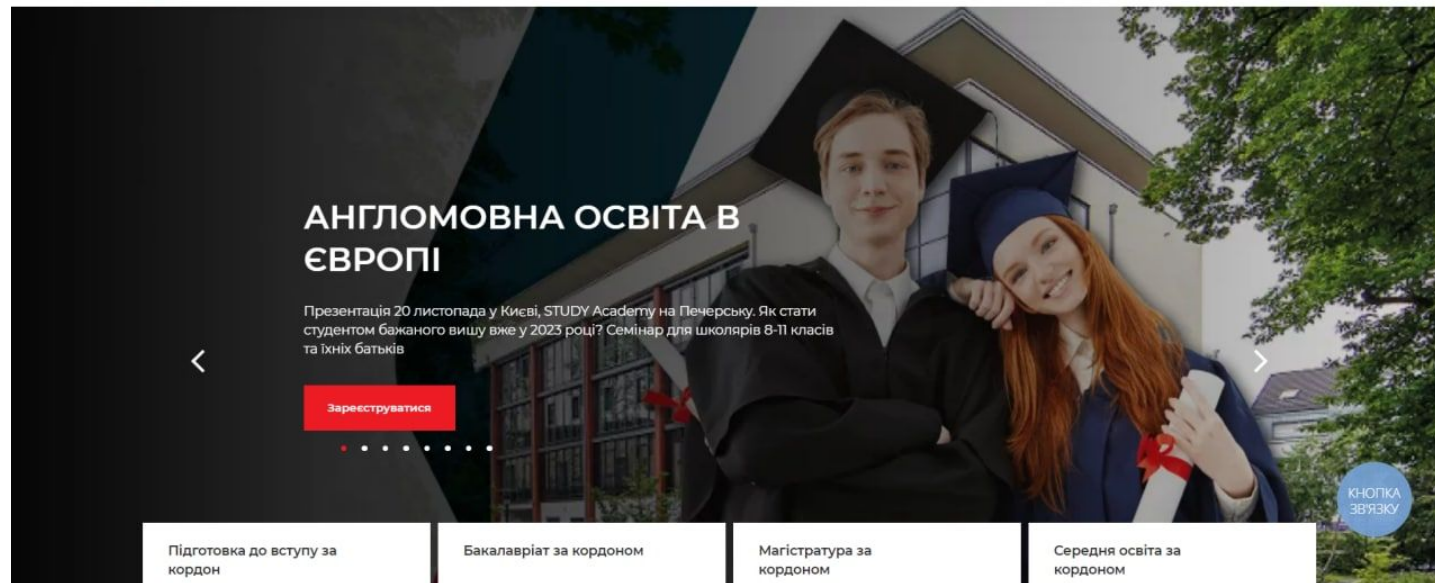
Блок оголошень — починається з символу **{** та закінчується символом **}**



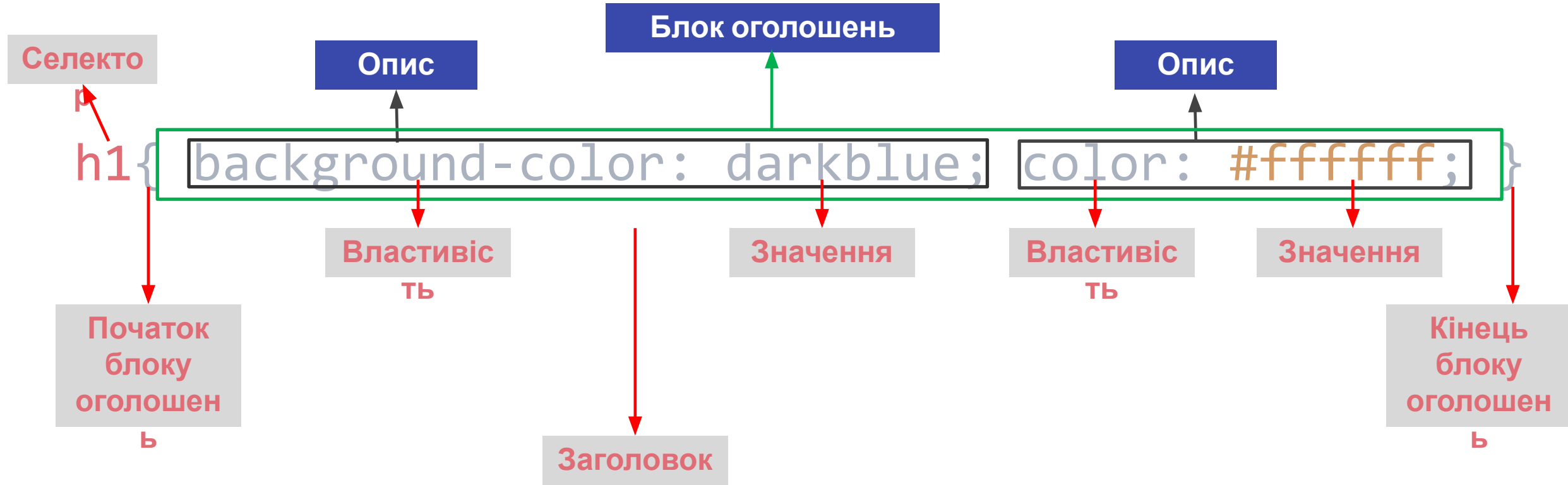
HTML



HTML+CSS



Що таке CSS



Підключення стилів — зовнішні таблиці стилів

Зовнішня таблиця стилів — це текстовий файл з розширенням **.css**, в якому знаходиться набір css правил. Файл створюється в редакторі коду, як і HTML-сторінка. Усередині файлу можуть бути лише стилі, без HTML-розмітки.

Тег `link` — Записується у тегу **`head`** та підключає зовнішні таблиці стилів.

Атрибут `href` — в цей атрибут записується посилання на таблицю стилів.

Атрибут `rel="stylesheet"` — вказує тип посилання (посилання на таблицю стилів).

```
<head>
<link rel="stylesheet" href="https://unpkg.com/swiper@8/swiper-bundle.min.css"/>
<link rel="stylesheet" href="/css/style.css">
</head>
```

Можливо підключати одразу декілька таблиць, тип посилання на таблицю може бути відносний або абсолютний. Ви також можете підключати стилі зі сторонніх сайтів та ресурсів.

Підключення стилів — внутрішні стилі

Внутрішні стилі — записуються всередині тегу **head** у HTML сторінці.

Тег **style** — в середині цього тегу записуються стилі CSS

```
<head>
  <style>
    h1 {
      background-color: darkblue;
      color: #ffffff;
    }
  </style>
</head>
```

УВАГА!

Стилі записані у внутрішніх стилях мають перевагу над стилями, які записані у зовнішніх стилях. Тобто при однакових записах до елемента будуть примінятися стилі із внутрішньої таблиці.



Підключення стилів — вбудовані стилі

Вбудовані стилі — це стилі які можуть записуватися безпосередньо у сам тег. Такі стилі діють лише той елемент у якому вони задані.

Атрибут **style** — використовується для запису стилів у тег.

```
<h1 style="background-color: darkblue; color: #ffffff;">Заголовок</h1>
```

УВАГА!

Стилi записанi у вбудованих стилях мають перевагу над стилями, якi записанi у зовнiшнiх та внутрiшнiх стилях. Тобто при однакових записах до елемента будуть примiнитися стилi



Підключення стилів — імпорт стилів

Директива **@import** — дозволяє завантажувати зовнішні таблиці стилів. Щоб директива **@import** працювала, вона повинна розташовуватися в таблиці стилів (зовнішньої або внутрішньої) перед іншими правилами.

```
<head>
  <style>
    @import url(mobile.css);
    h1 {
      background-color: darkblue;
      color: #ffffff;
    }
  </style>
</head>
```

Директива **@import** також використовується для підключення веб-шрифтів



Селектори

Селектор — визначає, до якого елемента застосовувати те чи інше CSS-правило. Селекторами можуть бути елементи, їх класи та ідентифікатори, а також псевдокласи та псевдоелементи.

Клас `class` — задає стилі елементу відповідно до назви атрибута `class`. У тег необхідно додати атрибут `class=""` та в лапках вказати ім'я класу. Для того щоб вказати що ми будемо шукати саме клас у css використовується символ крапка — `.` після якого вказується назва класу: `.contacts{/*стилі класу*/}`

Ідентифікатор `id` — задає стилі елементу відповідно до назви атрибута `id`. У тег необхідно додати атрибут `id=""` та в лапках вказати ім'я ідентифікатору. Ім'я ідентифікатору не повинно повторюватися в жодному іншому тегу. Для того щоб вказати що ми будемо шукати саме ідентифікатор у css використовується символ — `#` після якого вказується назва ідентифікатору: `#contacts{/*стилі*/}`

Ім'я класу та ідентифікатору може містити цифри та латинські букви, символи підкреслення та дефіс і повинно починатися з букви, підкреслення або одного символу дефісу.

Селектори, приклад

```
<div class="title"><h1 id="title-text">NVIDIA at VMware Explore 2022</h1></div>
```

```
.title{  
  background-color: #000;  
}  
#title-text{  
  padding: 20px;  
  color: #fff;  
  font-size: 20px;  
}
```

Такий запис **padding** означає
встановити відступ 20px одразу з усіх

NVIDIA at VMware Explore 2022

Селектори

Універсальний селектор `*` — відповідає будь-якому HTML-елементу. Наприклад вираз `{margin: 0;}`

обнуляє зовнішні відступи для всіх елементів сайту. Такий підхід часто використовується при створенні так званих **стилів обнулення** які видаляють усі стилі використовувані браузером за умовчанням.

Селектор елемента `тег` — відповідає вказаному тегу. Усі відповідні теги на сторінці будуть мати стилі які вказані у селекторі. Наприклад: `p{font-size: 20px;}` **усі теги p** на сторінці матимуть розмір шрифту 20px.



Приклад

```
<div class="title">
<h1>GeForce RTX 30 <a href="https://nvidia.com/">NVIDIA</a></h1>
  <p>The Ultimate Play</p>
</div>
<div class="contacts">
  <p>NVIDIA® GeForce RTX™ 30 Series Laptop GPUs</p>
  <span>Options </span><a href="https://nvidia.com/">GeForce RTX</a>
</div>
```

GeForce RTX 30 NVIDIA

The Ultimate Play

NVIDIA® GeForce RTX™ 30 Series Laptop GPUs

Options GeForce RTX

```
*{margin: 0;}
p{font-weight: 600;}
a{text-decoration: none;}
```

GeForce RTX 30 NVIDIA

The Ultimate Play

NVIDIA® GeForce RTX™ 30 Series Laptop GPUs

Options GeForce RTX



Селектори

Селектор нащадка — застосовують стилі до елементів, розташованих всередині елемента-контейнера. Наприклад: `ul li {text-transform: uppercase;}` вибере всі елементи `li`, що є нащадками всіх елементів `ul`.

Дочірній селектор `>` — це прямий нащадок батьківського елемента. Дочірній селектор дозволяє застосувати стилі лише якщо дочірній елемент йде відразу за батьківським елементом і між ними немає інших елементів. Наприклад: `div > p{/*стилі*/}`

Дочірній селектор `+` — `div + p{/*стилі*/}` обере перші абзаци, що йдуть безпосередньо після тегу `div`

Дочірній селектор `~` — `div ~ p{/*стилі*/}` обере всі абзаци що йдуть безпосередньо після тегу `div`



Селектори, приклад

GeForce RTX 30 NVIDIA

The Ultimate Play

Optimized Power and Performance

Call Now

We will call you in 30 seconds!

Free consultation

GeForce RTX 30 NVIDIA

THE ULTIMATE PLAY

Optimized Power and Performance

Call Now

We will call you in 30 seconds!

Free consultation

HTML :

```
<div class="title">
  <h1>GeForce RTX 30 NVIDIA</h1>
  <p>The Ultimate Play</p>
  <p>Optimized Power and Performance</p>
  <div class="call">
    <button>Call Now</button>
    <p>We will call you in 30 seconds!</p>
  </div>
  <p>Free consultation</p>
</div>
```

CSS:

```
.title > p {font-weight: 600;}
h1 + p{text-transform: uppercase;}
h1 ~ p{text-decoration: underline double;}
```

Селектор атрибуту

Селектор атрибуту — обирає елементи на основі імені атрибута або значення атрибута.

[атрибут] — всі елементи, що містять вказаний атрибут.

[value] — всі елементи, для яких заданий атрибут **value**

Селектор [атрибут] — елементи даного типу, що містять вказаний атрибут.

option[value] всі теги **option** для яких заданий атрибут **[value]**

селектор[атрибут="значення"] — елементи даного типу, що містять вказаний атрибут з конкретним значенням **option[value="0"]** або **img[alt="frontend"]**

селектор[атрибут ~="значення"] — стиль застосовується в тому випадку, якщо у атрибута є вказане значення або воно входить в список значень, що розділяються пробілом

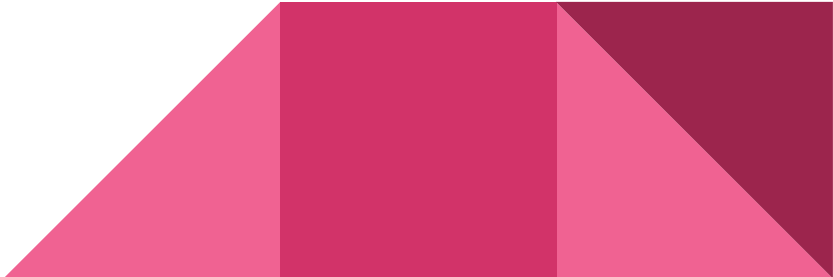


Селектори, успадкування та каскад

Спадкування — полягає в тому, що елементи успадковують властивості свого батька (елемента, що їх містить). Тобто це механізм, за допомогою якого певні властивості передаються від предка до його нащадків

Каскад — проявляється у тому, як різні види таблиць стилів застосовуються до документа, і як правила що конфліктують перевизначають одне одного. Тобто це механізм, який керує кінцевим результатом у ситуації, коли до одного елемента застосовуються різні CSS-правила.

Каскадування — це механізм, який керує кінцевим результатом у ситуації, коли до одного елемента застосовуються різні CSS-правила. Стили, розташовані в таблиці стилів нижче, скасовують ти ж самі стилі, розташовані в таблиці вище. **Каскадування** — допомагає уникнути конфліктів.




Селектори, успадкування та каскад

Властивості які відносяться до текстового вмісту сторінки — **успадковуються**:

- color
- font
- letter-spacing
- line-height
- list-style
- text-align
- text-indent
- text-transform
- Visibility
- white-space
- word-spacing

Властивості які відносяться до форматування блоків — **не успадковуються**:

- background
 - border
 - display
 - float
 - list-style
 - height
 - width / min-max-height
 - min-max-height
 - margin
 - outline
 - Overflow
 - Padding
 - Position
 - text-decoration
 - vertical-align
 - z-index
- 

Шрифти

font-family — визначає пріоритетний список із однієї або декількох назв сімейства шрифтів та/або загальне ім'я шрифту для вибраного елемента. Значення розділені комами, щоб вказати, що вони є альтернативами. Браузер вибере зі списку перший шрифт, який встановлено або може бути завантажено

```
h1{  
  font-family: Courier, 'Courier New', monospace;  
  font-size: 60px;  
  font-weight: 400;  
  font-style: italic;  
}
```

Якщо назви шрифтів у **font-family** складаються з одного слова, їх можна писати без лапків.

Відступи

margin — `margin: 10px 20px 15px 15px;` вказує значення **margin-top**, **margin-right**, **margin-bottom** та **margin-left**.

Також можна записати так `margin: 10px 15px;` — цей запис встановлює значення **margin-top** та **margin-bottom** у 10px, а **margin-right** та **margin-left** у 15px.

padding — `padding: 10px 20px 15px 15px;` вказує значення **padding-top**, **padding-right**, **padding-bottom** та **padding-left**.

Також можна записати так `padding: 10px 15px;` — цей запис встановлює значення **padding-top** та **padding-bottom** у 10px, а **padding-right** та **padding-left** у 15px.



Границі блоків

border-width — `border-width: 1px 2px 3px 4px;` визначає ширину рамки елемента (з верху 1px, з права 2px, з низу 3px та з ліва 4px). Може використовувати такі одиниці виміру: **px**, **rem**, **em**.

border-style — встановлює стиль лінії рамки для всіх чотирьох сторін елемента.

Значення: **solid** — одна пряма суцільна лінія;

dotted — ряд округлих точок;

dashed — короткі квадратні тире

double — дві прямі лінії

border-color — визначає колір лінії рамки. Може приймати значення кольору у css функціях `rgb(a)` / `hsl(a)` або у шістнадцятирічному форматі (повному та скороченому).

border-radius — визначає як округлюються кути блоку. Може використовувати такі одиниці виміру: **px**, **rem**, **em**.

border — скорочений запис який об'єднує властивості **border-width**, **border-style** та **border-color** `border: solid 3px #000;`



Ширина блоків

width — встановлює ширину блоку. Може використовувати такі одиниці виміру: **px**, **%**, **rem**, **em**.

min-width — ширина блоку не буде меншою за вказане значення, може збільшуватися. Може використовувати такі одиниці виміру: **px**, **%**, **rem**, **em**.

max-width — ширина блоку не буде більшою за вказане значення, може зменшуватися. Може використовувати такі одиниці виміру: **px**, **%**, **rem**, **em**.

```
<div style="border: 3px solid #000;"><h1 style="background-color: #00ffff;">Дуже довгий заголовок</h1></div>
```

```
h1{  
  width: 400px;  
}
```

Дуже довгий заголовок

```
h1{  
  width: 50%;  
}
```

Дуже довгий
заголовок

Дуже довгий заголовок

Приклади

```
h1{
  min-width: 400px;
}
```

Дуже довгий заголовок

```
h1{
  max-width: 400px;
}
```

Дуже довгий заголовок

```
h1{
  width: 100px;
}
```

Дуже
довгий
заголовок

```
div{
  width: 300px;
}
h1{
  min-width: 400px;
}
```

Дуже довгий заголовок

```
div{
  width: 300px;
}
h1{
  max-width: 400px;
}
```

Дуже довгий
заголовок



Висота блоків

height — встановлює висоту блоку. Може використовувати такі одиниці виміру: **px**, **%**, **rem**, **em**.

min-height — висота блоку не буде меншою за вказане значення, може збільшуватися. Може використовувати такі одиниці виміру: **px**, **%**, **rem**, **em**.

max-height — висота блоку не буде більшою за вказане значення, може зменшуватися. Може використовувати такі одиниці виміру: **px**, **%**, **rem**, **em**.

Поведінка цих властивостей така ж сама, як і поведінка властивостей ширини блоку, тільки тут змінюється значення висоти

