

Лекція №2

Мова розмітки HTML5

Викл. Коваль В.В.

ОСІТ

2021р.



Основи web-розробки



Мова розмітки HTML5

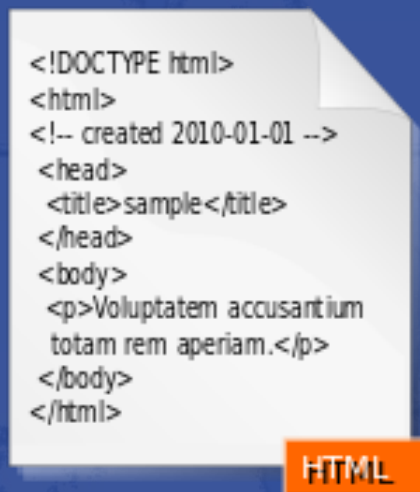
Історія виникнення мови HTML

- 1969 р. – Чарльз Гольдфарб запропонував прототип мови для технічної документації GML
- 1986 р. – представлений стандарт SGML (Standard Generalized Markup Language)
- 1989 р. – Тім Бернерс-Лі розробив систему розмітки гіпетекстових документів HTML
- 1994 р. – створений консорціум W3
- 1995 р. – HTML 2.0
- 1997 р. – HTML 4.0
- 2002 р. - XHTML 1.0
- 2014 р. – HTML 5.0

Історія виникнення мови HTML



В 1989 році Тім Бернерс Лі запропонував керівництву міжнародного центра високих енергій (CERN) проект розподіленої гіпертекстової системи World Wide Web (WWW).

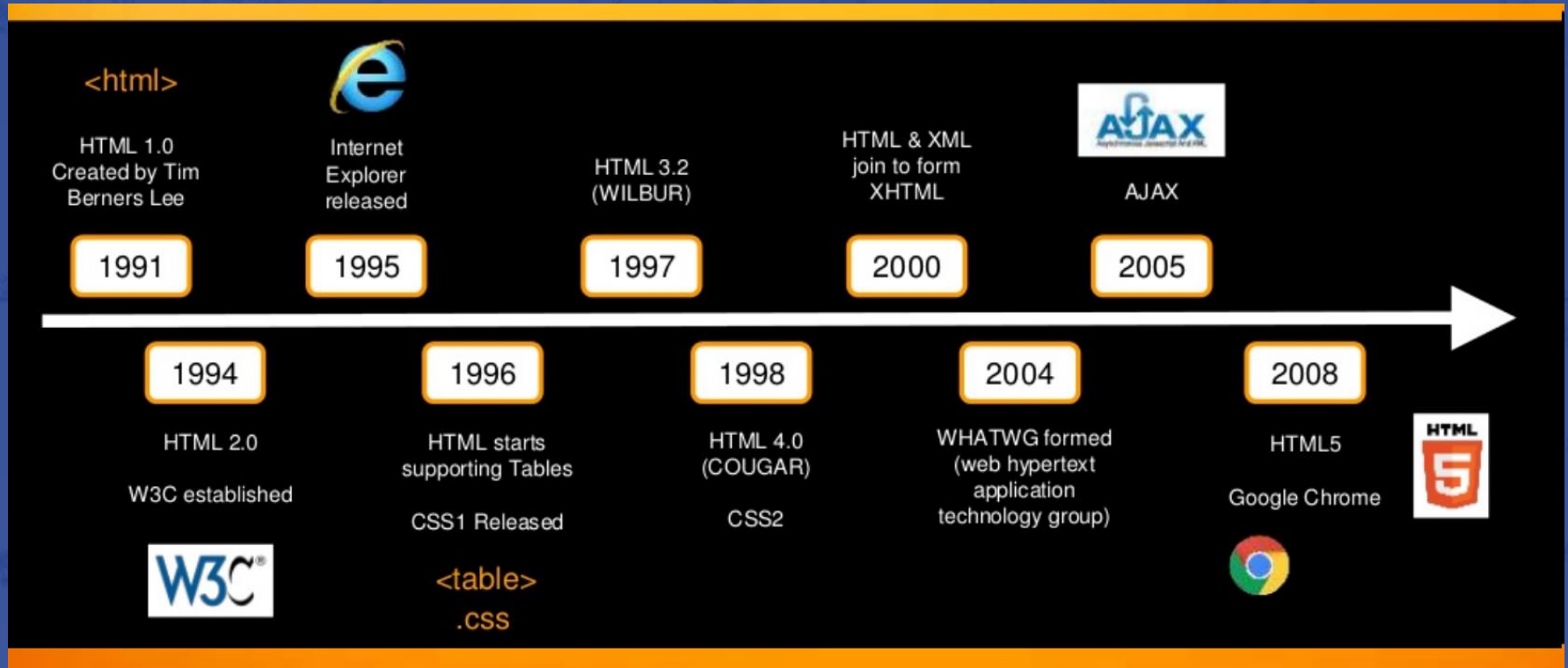


- Спочатку ідея системи полягала в тому, щоб за допомогою гіпертекстової навігаційної системи об'єднати всю безліч інформаційних ресурсів CERN у єдину інформаційну систему. Технологія виявилася настільки вдалою, що дала поштовх до розвитку однієї із самих популярних у світі глобальних інформаційних систем.



HTML (*HyperText Markup Language, 1989*) – стандартна мова розмітки веб-документів.

Консорціум Всесвітнього павутиння (*World Wide Web Consortium, W3C*) — головна міжнародна організація, що розробляє і впроваджує технологічні стандарти для Всесвітнього павутиння. Заснована 1994 року Тімом Бернерсом-Лі.



Особливості W3C HTML 5.2

1. Структура (семантика) документа

В HTML є цілий ряд нових семантичних тегів, які служать для створення більш змістовної організації веб-сторінок. Це такі теги, як: `<header>`, `<article>`, `<footer>`, `<nav>`, `<aside>`, `<section>`.

2. Графічні можливості

Елемент `<canvas>` створений для роботи з 2D графікою, що створює безліч нових можливостей для впровадження на сторінки. `<canvas>` являє собою динамічну поверхню для програмного малювання та різних операцій над графікою.

3. Мультимедіа

Додана підтримка мультимедійного контенту за допомогою тегів `<video>` та `<audio>`.

4. Геолокація

Браузер буде запитувати дозволу перед проведенням операції, оскільки в специфікації вказано, що без дозволу браузери не мають права відправляти геодані.

5. JavaScript API

API для роботи з графікою і мультимедією в новому HTML5, перетягування об'єктів (Drag & Drop методи) та інші можливості.

6. Нові елементи веб-форм

З'явилися також нові елементи для веб-форм, які знову ж таки розширюють функціонал сторінок.

7. Локальне сховище

Кешування даних- локальне сховище на стороні клієнта. Інформацію можна зберігати на пристрої клієнта в набагато більшому розмірі і з ширшими можливостями, ніж у куки.

Декларація HTML 4

Суворий (*Strict*): не містить nereкомендованих елементів (*deprecated*)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/strict.dtd">
```

Перехідний (*Transitional*): містить застарілі теги з метою сумісності зі старими версіями HTML.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01  
Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
```

Набір фреймів (Frameset): містить також теги для створення наборів фреймів.

- ```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">
```



# Синтаксис HTML

## Елемент

`<tag attr1="value" attr2="value2">` *Вміст елементу* `</tag>`

## Коментар

`<!--...-->`

## Декларація типу документа

`<!DOCTYPE HTML>`

```
<!DOCTYPE HTML>
<html lang="en">
 <head>
 <meta charset="utf-8">
 <title>Title of the document</title>
 </head>
 <body>
 The content of the document.....
 </body>
</html>
```





# Елементи



Кожен HTML-документ складається з дерева HTML-елементів і тексту.

Кожен HTML-елемент позначається початковим і кінцевим тегом.

Відкриваючий і закриваючий теги містять ім'я теги.

## Типи елементів:

- порожні елементи - `<area>`, `<base>`, `<br>`, `<col>`, `<embed>`, `<hr>`, `<img>`, `<input>`, `<keygen>`, `<link>`,  
`<menuitem>`, `<meta>`, `<param>`, `<source>`, `<track>`, `<wbr>`;
- елементи з неформатованим текстом - `<script>`, `<style>`;
- елементи, що виводять неформатований текст - `<textarea>`, `<title>`;
- елементи з іншого простору імен - MathML та SVG;
- звичайні елементи

# Атрибути



- Надають додаткову інформацію про елементи HTML
- Всі елементи HTML можуть мати **атрибути**
- Атрибути завжди вказуються в **початковому тезі**
- Атрибути зазвичай входять в пари ім'я/значення, наприклад: **name = "value"**

# Структура HTML-документа

```
<html>
```

```
<head>
```

```
<title>Page title</title>
```

```
</head>
```

```
<body>
```

```
<h1>This is a heading</h1>
```

```
<p>This is a paragraph.</p>
```

```
<p>This is another paragraph.</p>
```

```
</body>
```

```
</html>
```

## Заголовкова частина

<title>, <style>, <meta>, <link>, <script>, та  
<base>

```
<meta charset="UTF-8">
```

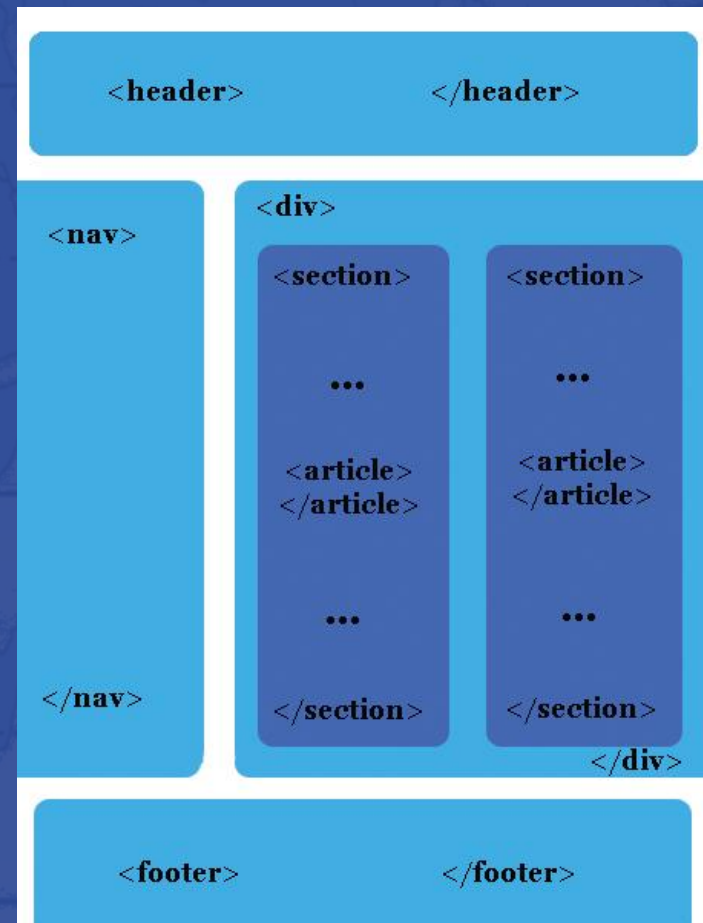
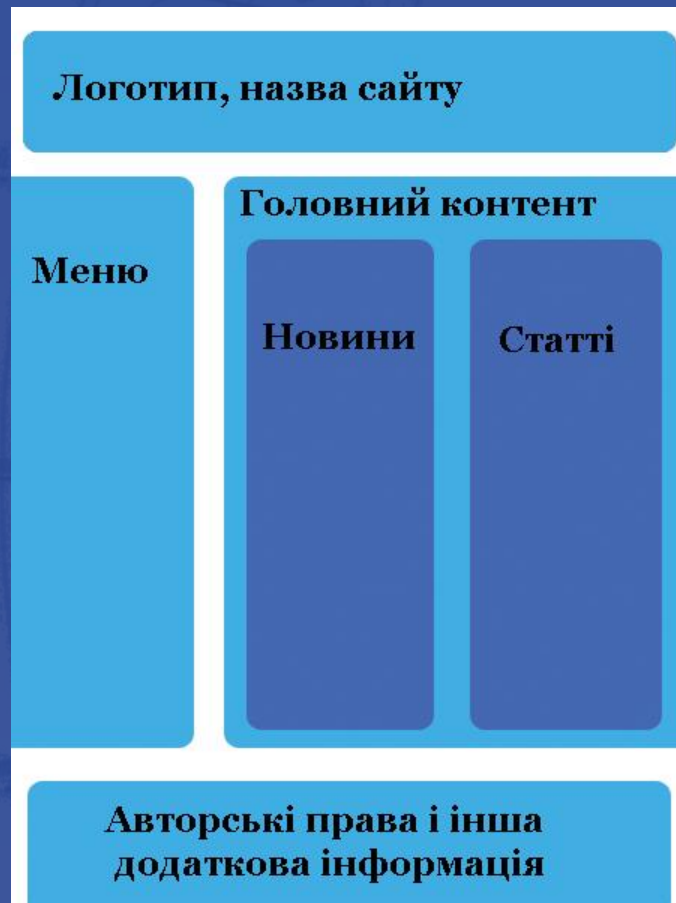
```
<meta name="description" content="Lecture">
```

```
<meta name="keywords" content="HTML, CSS, XML, JavaScript">
```

```
<meta name="author" content="Alex Lang">
```



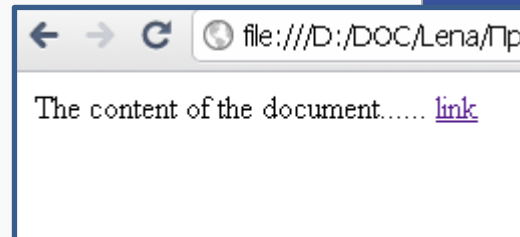
# Структура документа



# A

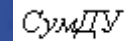
`<a href="http://site.ua" target="_blank">У новому вікні</a>`

href	URL	URL для переходу
hreflang	код мови	мова URL <code>&lt;a href="http://www.w3schools.com" hreflang="en"&gt;W3Schools&lt;/a&gt;</code>
rel	author bookmark help icon index next nofollow noreferrer pingback prefetch prev search sidebar stylesheet tag up	Зв'язок між поточним і URL в href <code>&lt;a rel="nofollow" href="http://www.functravel.com/"&gt;Cheap Flights&lt;/a&gt;</code>
target	_blank _self	_blank - у новому вікні _self – у поточному
type	mime_type	MIME тип файлу в URL



## <address>

```
<!DOCTYPE HTML>
<html>
<head><title>Title of the document</title></head>
<body>
<address>СyмΔY</address></body>
</html>
```

A small logo with the text "СyмΔY" in a stylized font, enclosed in a white rectangular box.

## <abbr>

The `<abbr title="World Health Organization">WHO</abbr>` was founded in 1948.



## <article>

```
<article>
<a href="http://blog.netscape.com/2010/12/8/
end-of-support-for-netscape-web-browsers">
Netscape

... has a long history on the internet, being one of
the first companies to really get people online.....
</article>
```

`<p>` - параграф

`<p>текст</p>`

`<br>`

перенесення рядку

`<aside>`

```
<p>My family and I visited Kyiv this summer.</p>
```

```
<aside>
```

```
<h4>Kyiv</h4>
```

```
My address in Kyiv is...
```

```
</aside>
```

My family and I visited Kyiv this summer.

**Kyiv**

My address in Kyiv is...

`<audio>`

```
<audio src="horse.ogg" controls="controls">
```

Your browser support the audio element.

```
</audio>
```



<a href="#">autoplay</a>	<a href="#">autoplay</a>	Якщо присутній, то звук буде транслюватися
controls	controls	кнопка
loop	loop	циклічно
preload	auto metadata none	Зазначає, чи потрібно завантажувати аудіо при завантаженні сторінки
SRC	URL	ресурс



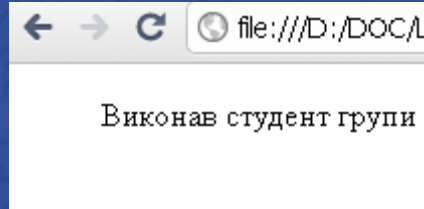
<b> ВИДІЛЕННЯ ЧАСТИНИ ТЕКСТУ

<blockquote>

<blockquote >

Виконав студент групи

</blockquote>



<em> <strong> <dfn> <code> <samp> <kbd>

<var> <cite>

<em>Emphasized text</em>

<strong>Strong text</strong>

<dfn>Definition term</dfn>

<code>Computer code text</code>

<samp>Sample computer code text</samp>

<kbd>Keyboard text</kbd>

<var>Variable</var>

<cite>Citation</cite>

<b> - Bold text

<strong> - Important text

<i> - Italic text

<em> - Emphasized text

<mark> - Marked text

<small> - Small text

<del> - Deleted text

<ins> - Inserted text

<sub> - Subscript text

<sup> - Superscript text

*Emphasized text* **Strong text** Definition term Computer code text Sample computer code text Keyboard text *Variable* *Citation*

<i>The lightning</i>

# <dl><dt><dd>

```
<dl>
<dt>Coffee</dt>
<dd>Black hot drink</dd>
<dt>Milk</dt>
<dd>White cold drink</dd>
</dl>
```

Coffee	Black hot drink
Milk	White cold drink

# <del><ins>

```
<p>My favorite color is blue <ins>red</ins>!</p>
```

My favorite color is ~~blue~~ red!

# <details>

```
<details open="open">
<summary>This document was written in 2018.</summary>
</details>
```

# <embed>

```
<embed src="helloworld.swf" />
```

World!

<h1> ... <h6>

<!DOCTYPE HTML>

<html>

<body>

<h1>This is header 1</h1>

<h2>This is header 2</h2>

<h3>This is header 3</h3>

<h4>This is header 4</h4>

<h5>This is header 5</h5>

<h6>This is header 6</h6>

</body>

</html>

**This is header 1**

**This is header 2**

**This is header 3**

**This is header 4**

**This is header 5**

**This is header 6**

<footer>

<footer>This document was written in 2018</footer>

<hr>

## HTML

HTML is a language for describing web pages.

## CSS

CSS defines how to display HTML elements.

<img>

alt	<i>text</i>	Короткий опис
src	<i>URL</i>	URL картинки
height	<i>pixels</i> %	висота
width	<i>pixels</i> %	ширина



# Елементи

`<mark>` - помічає фрагмент тексту

`<time>` - виведення часу, дати у текстовому блоці.

`<meter>` - для подання чисел в певному діапазоні(value, min, max, low, high і optimum).

`<progress>` - індикатор стану процесу, наприклад, завантаження файлу

## Типи полів введення

datetime

datetime-local

date

month

week

time

number

range

email

url

# Списки

`<ol>`

тег визначає початок упорядкованого списку.

<code>reversed</code>	<code>reversed</code>	При наявності списку порядок повинен бути зворотний
<code>start</code>	<i>number</i>	Вказує порядковий номер, з якого починається список

```
<!DOCTYPE HTML>
<html>
<body>

 Coffee
 Tea

 <ol start="10">
 Coffee
 Tea

</body>
</html>
```

```
1. Coffee
2. Tea

50. Coffee
51. Tea

1. Coffee
2. Tea
3. Геометрия
```

`<ul>`

Тег визначає неупорядкований список.

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<body>
```

```

```

```
 Coffee
```

```
 Tea
```

```
 Milk
```

```

```

```
</body>
```

```
</html>
```

- Coffee
- Tea
- Milk

# <table>

Тег визначає таблицю HTML.

Проста таблиця HTML складається з елементів таблиці і одного або більше **tr**, **th** і **td** елементів.

- **tr** елемент визначає рядок таблиці,
- **th** елемент визначає заголовок таблиці,
- **td** елемент визначає комірку в рядку таблиці.

Більш складні HTML-таблиці можуть також включати `caption`, `col`, `colgroup`, `thead`, `tfoot`, `tbody` елементи.

Атрибут

<b>summary</b>	<i>text</i>	Вказує резюме змісту таблиці
----------------	-------------	------------------------------



`<li>`

Тег визначає початок списку.

Використовується в упорядкованих списках (`<ol>`), маркованих списках (`<ul>`) і в списках меню (`<menu>`).

value	<i>number</i>	Визначає кількість елементів списку. У наступному списку пунктів буде збільшуватися від цього числа (лише в <code>&lt;ol&gt;</code> )
-------	---------------	---------------------------------------------------------------------------------------------------------------------------------------

`<ol>`

`<li>Coffee</li>`

`<li value=7>Tea</li>`

`<li>Milk</li>`

`</ol>`

1. Coffee

7. Tea

8. Milk

• Coffee

• Tea

• Milk

`<ul>`

`<li>Coffee</li>`

`<li>Tea</li>`

`<li>Milk</li>`

`</ul>`

```
<!DOCTYPE HTML>
<html>
<head><title>Title of the document</title></head>
<body>
<table summary="month">
 <caption>Monthly savings</caption>
 <col span="2" style="background-color:green" />
 <thead>
 <tr>
 <th>Month</th>
 <th>Savings</th>
 </tr>
 </thead>
 <tr>
 <td>January</td>
 <td>$100</td>
 </tr>
 <tr>
 <td>February</td>
 <td>$200</td>
 </tr>
 </table>
</body>
</html>
```

Monthly savings

Month	Savings
January	\$100
February	\$200

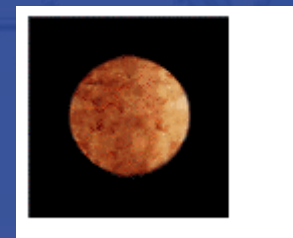
**<img>**

Тег визначає рисунок на сторінці.

alt	<i>text</i>	Визначає короткий опис зображення
src	<i>URL</i>	Адреса зображення
height	<i>pixels</i> %	Визначає висоту зображення
width	<i>pixels</i> %	Встановлює ширину зображення











```
<!DOCTYPE HTML>
<html>
<body>

</body>
</html>
```



# Мнемоніки

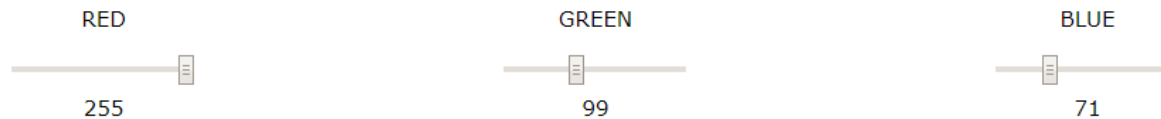
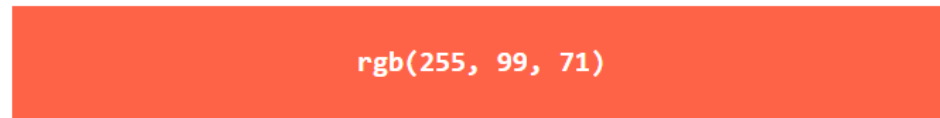
	&nbsp;
	&shy;
<	&lt;
>	&gt;
«	&laquo;
»	&raquo;
‹	&lsaquo;
›	&rsaquo;
“	&quot;
’	&prime;
”	&Prime;
‘	&lsquo;
’	&rsquo;

	&#9999;
	&#10000;
	&#10001;
	&#10002;
	&#9884;
	&#9937;
	&#9885;
	&#10052;
	&#10084;
	&#10053;

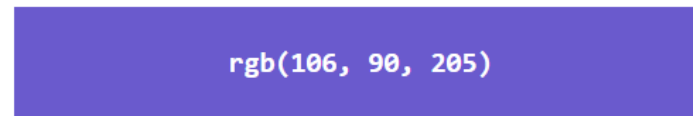
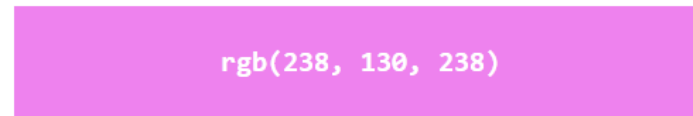
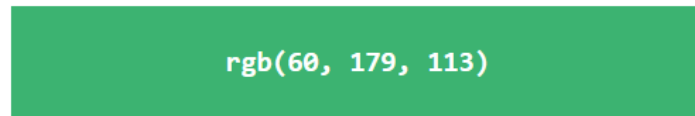
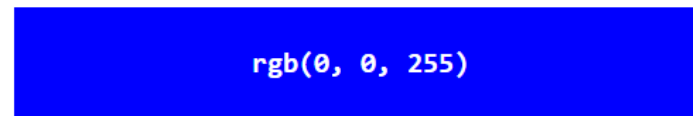
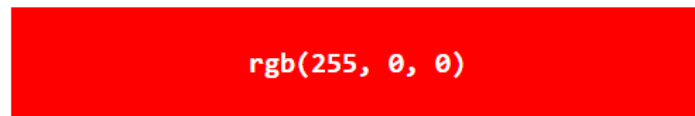


# Кольори

HTML-кольори зазначаються з використанням передвизначених імен кольорів або значень RGB, HEX, HSL, RGBA, HSLA.



## Example



# Приклади кольорів

`rgb(0, 0, 0)`

`rgb(60, 60, 60)`

`rgb(120, 120, 120)`

`rgb(180, 180, 180)`

`rgb(240, 240, 240)`

`rgb(255, 255, 255)`

`#ff0000`

`#0000ff`

`#3cb371`

`#ee82ee`

`#ffa500`

`#6a5acd`

# Система передвизначених імен

Color Name	HEX	Color
<u>AliceBlue</u>	<u>#F0F8FF</u>	
<u>AntiqueWhite</u>	<u>#FAEBD7</u>	
<u>Aqua</u>	<u>#00FFFF</u>	
<u>Aquamarine</u>	<u>#7FFFD4</u>	
<u>Azure</u>	<u>#F0FFFF</u>	
<u>Beige</u>	<u>#F5F5DC</u>	
<u>Bisque</u>	<u>#FFE4C4</u>	
<u>Black</u>	<u>#000000</u>	
<u>BlanchedAlmond</u>	<u>#FFEBCD</u>	
<u>Blue</u>	<u>#0000FF</u>	
<u>BlueViolet</u>	<u>#8A2BE2</u>	
<u>Brown</u>	<u>#A52A2A</u>	
<u>BurlyWood</u>	<u>#DEB887</u>	
<u>CadetBlue</u>	<u>#5F9EA0</u>	
<u>Chartreuse</u>	<u>#7FFF00</u>	
<u>Chocolate</u>	<u>#D2691E</u>	
<u>Coral</u>	<u>#FF7F50</u>	
<u>CornflowerBlue</u>	<u>#6495ED</u>	
<u>Cornsilk</u>	<u>#FFF8DC</u>	
<u>Crimson</u>	<u>#DC143C</u>	
<u>Cyan</u>	<u>#00FFFF</u>	
<u>DarkBlue</u>	<u>#00008B</u>	
<u>DarkCyan</u>	<u>#008B8B</u>	
<u>DarkGoldenRod</u>	<u>#B8860B</u>	
<u>DarkGray</u>	<u>#A9A9A9</u>	

# Приклад представлення кольору “tomato” у схемах RGB, HEX, HSL, RGBA та HSLA

`rgb(255, 99, 71)`

`#ff6347`

`hsl(9, 100%, 64%)`

Same as color name "Tomato", but 50% transparent:

`rgba(255, 99, 71, 0.5)`

`hsla(9, 100%, 64%, 0.5)`

## Example

```
<h1 style="background-color:rgb(255, 99, 71);">...</h1>
<h1 style="background-color:#ff6347;">...</h1>
<h1 style="background-color:hsl(9, 100%, 64%);">...</h1>

<h1 style="background-color:rgba(255, 99, 71, 0.5);">...</h1>
<h1 style="background-color:hsla(9, 100%, 64%, 0.5);">...</h1>
```



# Система HSL

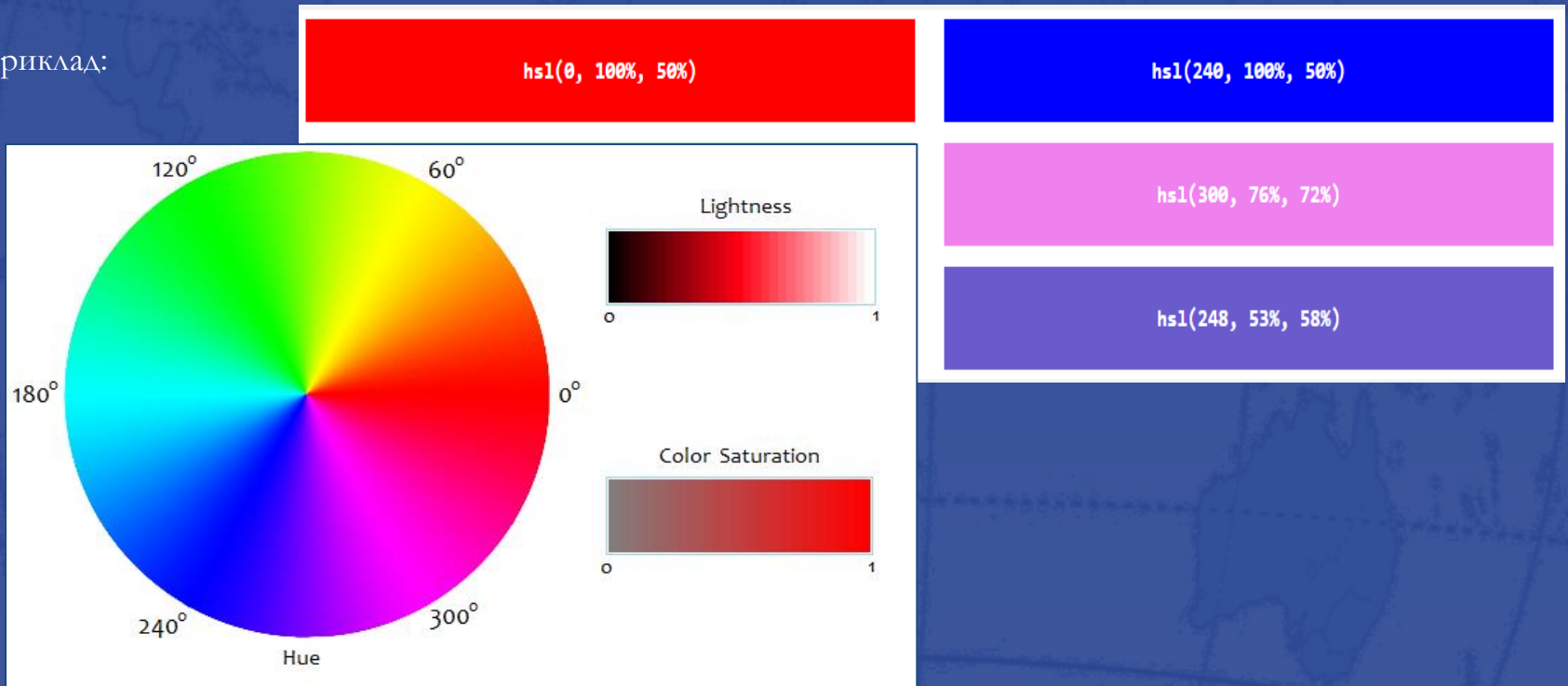
**HSL** ( відтінок, насиченість, легкість )

**Значення відтінку**- це градус на колі кольорів від 0 до 360. 0 червоний, 120 зелений, а 240 - синій.

**Насиченість** - це процентное значення, 0% - відтінок сірого, а 100% - повний колір.

**Легкість** - процент, 0% - чорний, 50% - не світлий не темний, 100% білий

Приклад:



# Насиченість та легкість кольору

`hsl(0, 100%, 50%)`

`hsl(0, 80%, 50%)`

`hsl(0, 60%, 50%)`

`hsl(0, 40%, 50%)`

`hsl(0, 20%, 50%)`

`hsl(0, 0%, 50%)`

`hsl(0, 100%, 0%)`

`hsl(0, 100%, 25%)`

`hsl(0, 100%, 50%)`

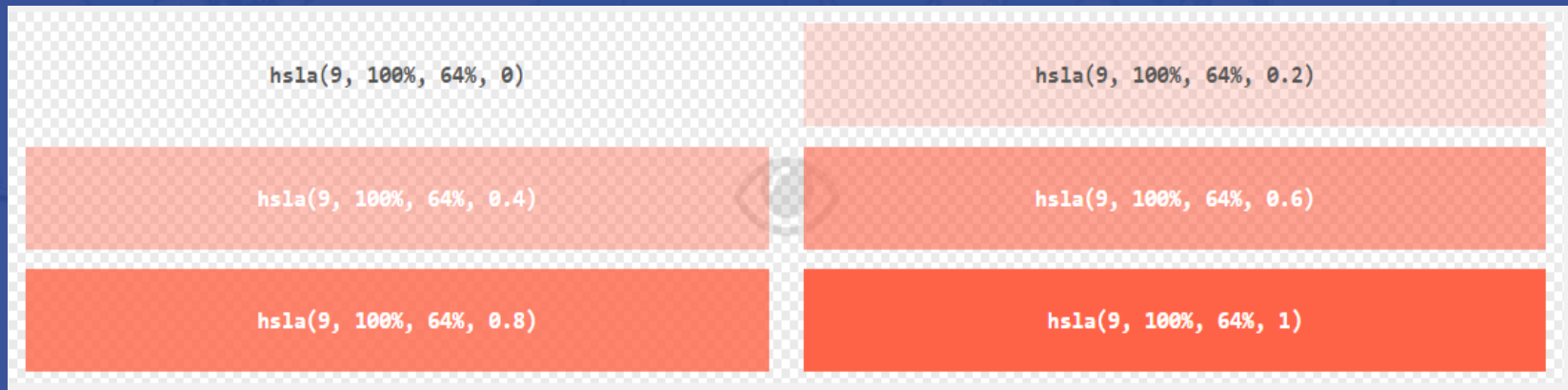
`hsl(0, 100%, 75%)`

`hsl(0, 100%, 90%)`

`hsl(0, 100%, 100%)`

# Прозорість у RGBA і HSLA

- Додається значення альфа-каналу  $[0,1]$  – степінь прозорості.





Дякую за увагу!



# *Подивитись*

---

<https://habr.com/ru/post/50147/>

<https://developer.mozilla.org/ru/docs/Web/HTTP/Methods>

<https://dev.opera.com/articles/http-lets-get-it-on/>

<https://lectureskpd.readthedocs.io/kpd/3.http.html#openssl>

[https://developer.mozilla.org/ru/docs/Learn/Getting\\_started\\_with\\_the\\_web/How the Web works](https://developer.mozilla.org/ru/docs/Learn/Getting_started_with_the_web/How_the_Web_works)

[https://developer.mozilla.org/ru/docs/Learn/Getting\\_started\\_with\\_the\\_web/HTML basics](https://developer.mozilla.org/ru/docs/Learn/Getting_started_with_the_web/HTML_basics)