

# Вставка зображень і редагування їх властивостей

## Формати графічних файлів, використовувані у веб-дизайні

Широке поширення для веб-графіки отримали два формати – **GIF** і **JPEG**. Їх багатофункціональність, універсальність, невеликий об'єм початкових файлів при достатній для сайту якості, фактично стали причиною того, що їх визначили як стандарт веб-зображень. Також при додаванні зображень використовується інший відомий формат **PNG**, який існує в двох різновидах - **PNG-8** і **PNG-24**. Однак популярність PNG сильно поступається визнанню форматів GIF і JPEG. Розглянемо ці формати трохи докладніше.

### Формат GIF

**GIF (Graphics Interchange Format)** – формат графічних файлів, широко застосовуваний при створенні сайтів. GIF використовує 8-бітовий колір і ефективно стискає ділянки суцільної кольорової області, при цьому зберігаючи деталі зображення.

#### Особливості

Кількість кольорів у зображенні може бути від 2 до 256, але це можуть бути будь-які кольори з 24-бітної палітри. Файл в форматі GIF може містити прозорі ділянки. Якщо використовується відмінний від білого кольору фон, він буде переглядати крізь «дірки» в зображенні.

Цей формат підтримує покадрову зміну зображень, що робить формат популярним для створення банерів і простої анімації. Використовує вільний від втрат метод стиснення.

#### Область застосування

Текст, логотипи, ілюстрації з чіткими краями, анімовані малюнки, зображення з прозорими ділянками, банери.

### Формат JPEG

**JPEG (Joint Photographic Experts Group)** – популярний формат графічних файлів, широко застосовуваний при створенні сайтів і для зберігання зображень. JPEG підтримує 24-бітову палітру кольорів і зберігає яскравість і відтінки кольорів у фотографіях незмінними. Даний формат використовує стиснення з втратами. Оскільки алгоритм JPEG вибірково відкидає дані, це може внести спотворення в малюнок, особливо якщо він містить текст, дрібні деталі або чіткі краї. Формат JPEG не підтримує прозорість. Коли фотографію зберігають у цьому форматі, прозорі пікселі заповнюються певним кольором.

#### Особливості

Кількість кольорів у зображенні – близько 16 мільйонів, що цілком достатньо для збереження фотографічної якості зображення.

Основна характеристика формату – якість, яка дозволяє управляти кінцевим розміром файлу. Підтримує технологію, так званий прогресивний JPEG, при якій версія малюнка з низькою роздільною здатністю з'являється у вікні перегляду до повного завантаження самого зображення.

#### Область застосування

Використовується переважно для фотографій. Не дуже підходить для малюнків, які містять прозорі ділянки, дрібні деталі або текст.

### Формат PNG-8

**PNG-8 (*Portable Network Graphics*)** – формат за своєю дією аналогічний GIF. ПО запевненням розробників, використовує покращений формат стиснення даних, але, як показує практика, це не завжди так.

#### *Особливості*

Використовує 8-бітову палітру (256 кольорів) в зображенні, за що і отримав у своїй назві цифру вісім. При цьому можна вибирати, скільки кольорів буде зберігатися в файлі – від 2 до 256. На відміну від формату GIF, даний формат не підтримує анімацію ні в якому вигляді.

#### *Область застосування*

Текст, логотипи, ілюстрації з чіткими краями.

#### **Формат PNG-24**

**PNG-24** – формат, аналогічний PNG-8, але використовує 24-бітову палітру кольорів подібно до формату JPEG, зберігає яскравість і відтінки кольорів у фотографіях. Подібно до форматів GIF і PNG-8, зберігає деталі зображення, як, наприклад, в лінійних малюнках, логотипах або ілюстраціях.

#### *Особливості*

Використовує приблизно 16,7 млн. колірів у файлі. Саме тому цей формат застосовується для повнокольорових зображень.

Підтримує багаторівневу прозорість. Це дозволяє створювати плавний перехід від прозорої області зображення до кольорового, так званий градієнт.

Через те, що використовуваний алгоритм стиснення зберігає всі кольори і пікселі в зображенні незмінними, якщо порівнювати з іншими форматами, то у PNG-24 кінцевий об'єм графічного файлу виходить найбільшим.

#### *Область застосування*

Фотографії, малюнки, які містять прозорі і напівпрозорі ділянки, малюнки з великою кількістю кольорів і чіткими краями зображень.

#### **Формат SVG**

**SVG (*Scalable Vector Graphics*)** з'явився в 1999 році як результат роботи W3C консорціуму. Задуманий і реалізований був для того, щоб графіку, яку розміщують на сайті, можна було описати мовою, тобто всі фігури, об'єкти, малюнки в **.svg** файлах описуються текстово (XML).

#### *Особливості*

На відміну від попередніх, растрових форматів, рисунки у графічному форматі SVG складаються із різноманітних геометричних фігур, що дозволяє збільшувати їх масштаб без погіршення якості зображення. Об'єкти **.svg** займають набагато менше місця, ніж їх близнюки, створені як об'єкт растрової графіки. Але зі збільшенням кількості геометричних фігур, з яких складається рисунок, значно зростає і розмір файлу.

За допомогою CSS можна швидко змінити параметр графіки на сайті, наприклад, колір фону або позицію логотипу на сторінці. Для цього можна відредагувати файл в будь-якому текстовому редакторі. А переглянути вміст файлу **.svg** можна практично будь-яким браузером (IE, Chrome, Opera, FireFox, Safari тощо.)

#### *Область застосування*

Найчастіше **.svg** використовують в дизайні іконок для веб-сайтів. Зображення без втрат розтягується і стискається, менше важить і не виглядає розмитим на пристроях з високою щільністю пікселів. Це досить актуально для смартфонів планшетів. Крім того відкриваються ці файли будь-яким браузером.

### Додавання малюнка

Додавання зображення на веб-сторінку відбувається в два етапи: спочатку готується графічний файл бажаного розміру, потім він додається на сторінку за допомогою тегу **<img>**. Сам HTML призначений тільки для того, щоб відобразити необхідну картинку, при цьому ніяк її не змінюючи. При підготовці зображень слід врахувати, що об'єм графічного файлу істотно впливає на швидкість завантаження веб-сторінки.

Як вже було сказано, щоб додати зображення на веб-сторінку використовується тег **<img>**, атрибут **src** якого визначає адресу графічного файлу. Загальний синтаксис додавання зображення буде наступний.

```
<img src = "URL" alt = "альтернативний текст" >
```

**URL (Universal Resource Locator, універсальний покажчик ресурсів)** – це шлях до графічного файлу. Для його вказівки можна використовувати як абсолютну, так і відносну адресу, використовуючи ті ж [правила](#), що і для створення гіперпосилань.

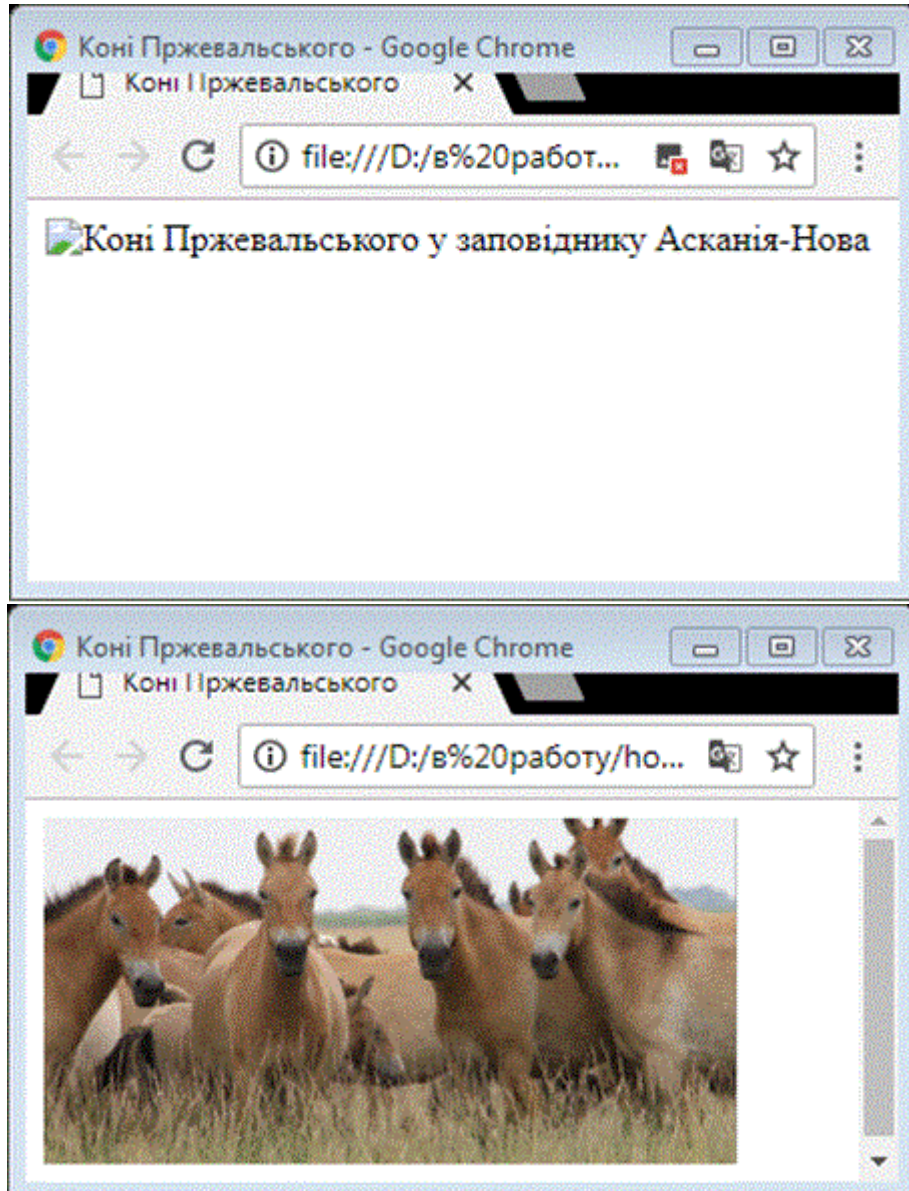
Слід зазначити, що у випадку використання абсолютного посилання зображення завжди буде завантажуватися із зазначеної адреси в Інтернеті, навіть при збереженні веб-сторінки на локальному комп'ютері.

#### *Приклад вставки зображення в документ*

```
<!DOCTYPE html>
<html>
  <head>
    <title> Вставка зображення </title>
  </head>
  <body>
    <p><!-- Адреса розміщення зображення щодо кореня сайту,
    працює тільки при розміщенні сайту на веб-сервері -->
    </p>
    <p><!--Адреса розміщення зображення щодо поточного HTML-документа
    -->
    <img src ="images/home.png"></p>
    <p><!--Абсолютне посилання -->
    <img src                                ="http://art-
coral.com/sites/default/files/newsimgs/134213.png ">
    </p>
  </body >
</html>
```

## Альтернативний текст та спливаюча підказка

Альтернативний текст дозволяє отримати текстову інформацію про малюнок при відключеному в браузері показі картинок, у випадку відсутності графічного файлу за вказаною адресою або під час його завантаження. Такий текст з'являється раніше самого зображення і дає уявлення про його зміст. Потім зарезервоване порожнє поле замінюється картинкою.



Вид підказки, а саме, її колір, фон, шрифт та інші параметри задаються за допомогою налаштувань операційної системи і не можуть бути змінені через HTML-файл.

Для створення альтернативного тексту використовується атрибут `alt` тега `<img>`, як показано в наступному прикладі.

```
<!DOCTYPE html>
<html>
  <head>
    <title> Альтернативний текст </title>
  </head>
  <body>
    <p> <img src = "images/horses.jpg"
```

```
alt= "Коні Пржевальського у заповіднику Асканія-Нова" >
</p>
</body>
</html>
```

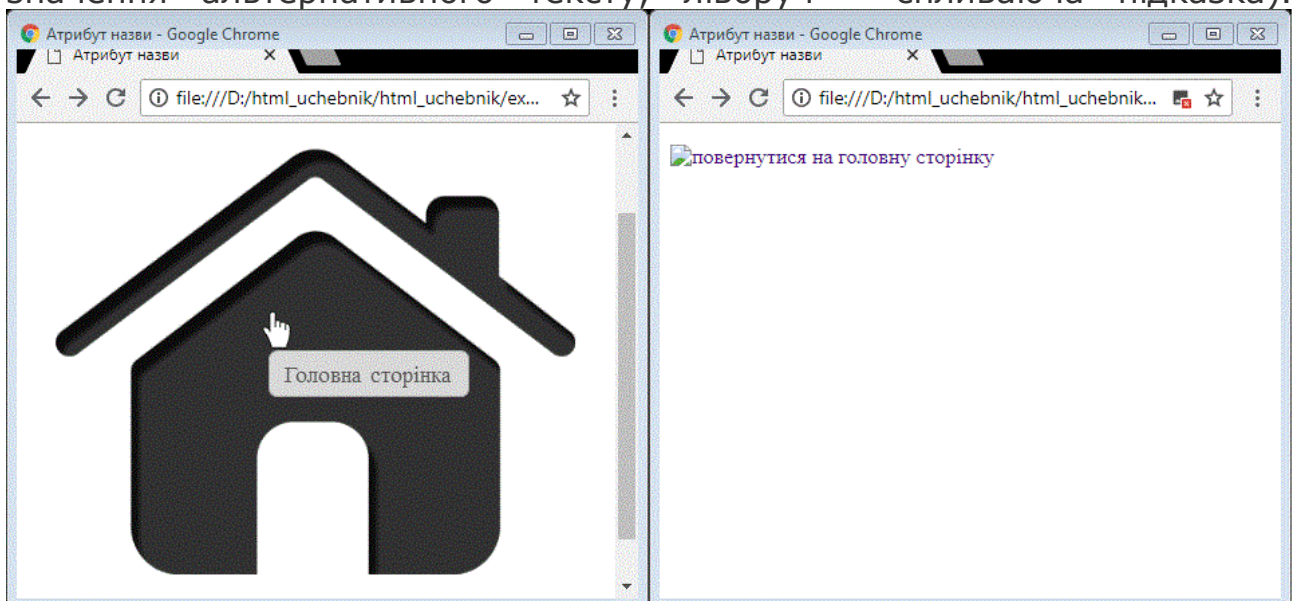
Врахуйте, що текст в атрибуті **alt** обов'язково повинен бути взятий в лапки, як у наведеному прикладі.

Але не всі браузери відображають альтернативний текст у вигляді підказки. Тому для її створення використовуйте атрибут **title**.

*Приклад: Спливаюча підказка*

```
<!DOCTYPE html>
<html>
  <head>
    <title> Атрибут назви </title>
  </head>
  <body>
    <p> < a href = "index.html" >
      <img src = "images/home.png"
        alt= "повернутися на головну сторінку "
        title = " Головна сторінка ">
    </p>
  </body>
</html>
```

Зверніть увагу, що у даному прикладі значення атрибутів **alt** і **title** відрізняються (див. наступний рисунок: праворуч значення альтернативного тексту, ліворуч - спливаюча підказка).



Відмінні значення даних атрибутів дозволяють встановити певний текст для різних випадків. Проте, необхідно враховувати, що довгий текст буде «обрізатися» і відображатися не весь. Але пошукові системи, для яких зокрема і застосовують атрибути **alt** і **title**, читають його повністю.



## Визначення розмірів малюнка

Якщо для елемента `<img>` не поставити атрибути `width` (ширина) і `height` (висота), то браузер самостійно визначить ширину і висоту зображення після завантаження файлу і покаже його в початковому розмірі. Малюнок у форматі [SVG](#) браузери, за винятком IE, виводять на всю доступну ширину.

Зміна масштабу і пропорцій зображення робиться як через атрибути `<img>`, так і через стилі. Оскільки стилі будуть розглядатися у іншому розділі посібника, зупинимось тільки на використанні атрибутів.

Будь-яку картинку можна як збільшувати, так і зменшувати в розмірах, задаючи значення висоти або ширини в пікселях або відсотках від розміру батьківського елемента. Якщо встановлена тільки ширина або висота, то друга величина обчислюється автоматично виходячи з пропорцій картинки. Два заданих значення можуть спотворити пропорції, особливо якщо вони задані невірно. У наступному прикладі показані різні варіанти завдання розмірів в пікселях.

*Приклад встановлення розмірів картинки*

```
<!DOCTYPE html>
<html>
  <head>
    <title>Розміри зображення</title>
  </head>
  <body>
    
    
    
  </body>
</html>
```

Як ви могли побачити, в прикладі використовувалася одна і та ж фотографія. Для першого `<img>` розміри явно не вказані, тому браузер додав зображення в початковому вигляді. Для другої фотографії вказана ширина 300 пікселів, що призвело до збільшення її розмірів при збереженні пропорцій. Третя фотографія спотворена, оскільки вихідне зображення не квадратне, а при цьому задані однакові ширина та висота.

Аналогічним чином задаються розміри в процентах від ширини батьківського елемента. Таким чином можна встановити картинку на всю ширину вікна браузера. У наступному прикладі показано додавання трьох фотографій в ряд, ширина яких прив'язана до ширини вікна. (Переглядаючи приклад, змінюйте розмір браузера та звертайте увагу, як змінюються розміри зображення)

*Приклад: Розміри зображень у відсотках до батьківського елемента*

```
<!DOCTYPE html>
<html>
  <head>
    <title>Розміри зображення</title>
  </head>
```

```
<body>
  
  
  
</body>
</html>
```

У даному прикладі кожне зображення займає 30% екрану. Разом це складає 90%, тому зправа залишається вільне місце. Та оскільки браузер додає відступи між картинками, вказувати ширину 33% не доцільно, зображення можуть не поміститися в один ряд. Пропонуємо вам поекспериментувати, натиснувши кнопку "Редагувати".

Нагадаємо, що використовувати атрибути у випадку, коли можна застосувати стилі CSS, зайва витрата часу. Тому повернемося до питання встановлення розмірів рисунків у відповідному розділі CSS.

**ЗАУВАЖЕННЯ:** Якщо розміри зображення вказані явно, то браузер використовує їх для того, щоб відображати відповідну зображенню порожню область в процесі завантаження документа. В іншому випадку браузер чекає, коли малюнок завантажиться повністю, після чого змінює ширину і висоту картинки. При цьому може відбутися переформатування тексту, оскільки спочатку розмір картинки не відомий і автоматично він встановлюється невеликим.

Ширину і висоту зображення можна змінювати як в меншу, так і більшу сторону. Однак на швидкість завантаження малюнка це ніяк не впливає, оскільки розмір файлу залишається незмінним. Тому з обережністю зменшуйте зображення за допомогою стилів, бо це може викликати здивування у відвідувача сайту, чому такий маленький малюнок так довго вантажиться. А ось збільшення розмірів призводить до зворотного ефекту – розмір зображення великий, але файл у порівнянні із іншим зображенням аналогічного розміру (розмір на екрані відповідає розмірам фактичним), завантажиться швидше.

Така зміна розмірів називається ресемпліруванням, при цьому алгоритм браузера за своїми можливостями поступається графічним редакторам. Тому збільшувати таким способом зображення потрібно тільки в особливих випадках, адже у цьому випадку якість картини суттєво погіршується. Краще підготувати зображення із використанням якої-небудь графічної програми. Винятком є малюнки, які містять прямокутні області.

### Розташування зображень у тексті

Відразу необхідно зауважити, що способи визначення положення рисунків по відношенню до тексту, про які далі піде мова, застаріли. Зараз вплив на розташування зображення реалізується за допомогою CSS. Але з іншими варіантами верстальник може стикнутися у випадку редизайну сайту, який було створено значно раніше.

Для вирівнювання зображення по центру колонки тексту тег `<img>` поміщається в контейнер `<p>`, для якого встановлюється атрибут `align = "center"`. Таке положення малюнка щодо тексту схематично показано на рис. 1.

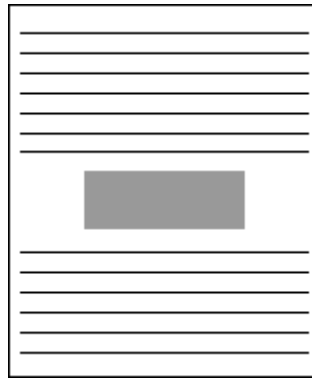


Рис. 1. Рисунок в центрі колонки тексту

Обтікання картинки текстом – це один із популярних прийомів верстки веб-сторінок, коли зображення розташовується по краю вікна браузера, а текст обтікає його з інших сторін (рис. 2). Для створення обтікання зображення текстом існує кілька способів, пов'язаних, як з можливістю тегів HTML, так і з застосуванням стилів. Застосування стилів ми будемо розглядати пізніше у відповідному розділі CSS.

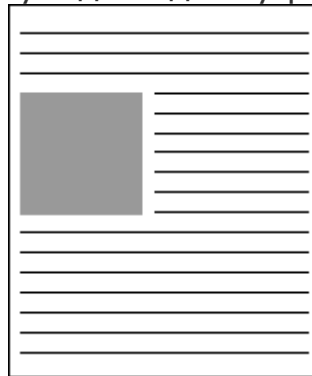


Рис. 2. Текст обтікає рисунок, вирівняний по лівому краю

Тег `<img>` має атрибут `align`, який визначає вирівнювання зображення. Цей атрибут задає, біля якого краю вікна буде розташовуватися малюнок, одночасно встановлюючи і спосіб обтікання тексту. Щоб вирівняти зображення по правому краю і задати обтікання зліва, використовують значення `right`, для вирівнювання по лівому краю застосовують `left`. Атрибут `align` часто використовують в зв'язці з іншими атрибутами тега `<img>` – `vspace` і `hspace`. Вони визначають відстань від обтічного тексту до зображення. Без цих атрибутів зображення і текст будуть занадто щільно прилягати одне до одного.

*Приклад розміщення зображень*

```
<!DOCTYPE html>
<html>
  <head>
    <title> Обтікання картинок </title>
    <style type="text/css">
      body {
        width: 600px;
        font-size: 16px;
      }
      img {
        width: 20%;
      }
    </style>
```



```

</head>
<body>
  
  Lorem ipsum dolor sit amet, consectetur
  adipisicing elit. Perspiciatis aliquid
  itaque dolor adipisci. Ab dicta qui odit
  ...
  ...
  <p align="center"></p>
  Laborum officia sapiente exercitationem, quas
  possimus debitis quam vitae perferendis eos
  facere cum, saepe, suscipit at! Iste quis est,
  architecto iusto voluptates hic porro magnam
  eos earum, dolorum eveniet. Iste.
  Nulla,
  facere suscipit sed dolore enim optio aperiam
  qui obcaecati soluta. Accusamus culpa ipsum
  sed suscipit id officiis qui, ea recusandae,
  nesciunt adipisci optio dicta alias tempore,
  obcaecati quam accusantium.
  ...
  ...
</body>
</html>

```

### Зображення у якості посилання

Інколи у якості посилання використовують зображення. Для цього треба об'єднати два відомі теги `<img>` і `<a>` наступним чином:

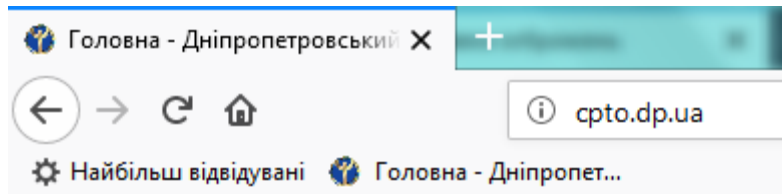
```

<!DOCTYPE html>
<html>
  <head>
    <title> Image-Link </title>
  </head>
  <body>
    <a                                href="https://www.google.com"></a>
  </body>
</html>

```

### Іконка сайту у браузері

Зазвичай на вкладці веб-сторінки, ліворуч від її назви, розташовано маленьке зображення-логотип, яке ще називають *favicon*. Це ж зображення відображається поряд з назвою при додаванні сайту в закладки.



Щоб додати таке зображення до своєї веб-сторінки, необхідно створити його за допомогою будь-якого графічного редактора та зберегти, наприклад, у папці сайту `images`, в одному з наступних форматів - `.ico`, `.png`, `.jpg`, `.gif`. Розмір іконки на вкладці складає `16×16` також може бути `32×32`, а у деяких випадках `48×48`. Тому рекомендується підготувати зображення у найбільшому розмірі, а браузер, при необхідності, його зменшить. Також можна підготувати зображення у трьох розмірах, та підключити їх із використанням атрибуту `sizes`, як показано далі.

Щоб додати іконку до сторінки, необхідно в `html`-документі в контейнер `<head>...</head>` додати наступний код:

- із використанням одного зображення:

- `<link rel="shortcut icon" type="image/png"`

`href="images/favicon.png">`

- із використанням зображень різних розмірів:

- `<link rel="icon" type="image/png" href="/favicon-16x16.png"`
- `sizes="16x16">`
- `<link rel="icon" type="image/png" href="/favicon-32x32.png"`
- `sizes="32x32">`
- `<link rel="icon" type="image/png" href="/favicon-96x96.png"`

`sizes="96x96">`