

Урок №70

HTML- макети. Адаптивна верстка

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>CSS Template</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
* {
  box-sizing: border-box;
}

body {
  font-family: Arial, Helvetica, sans-serif;
}

/* Style the header */
header {
  background-color: #666;
  padding: 30px;
  text-align: center;
  font-size: 35px;
  color: white;
}

/* Create two columns/boxes that floats next to each other */
nav {
  float: left;
  width: 30%;
  height: 300px; /* only for demonstration, should be removed */
  background: #ccc;
  padding: 20px;
}

/* Style the list inside the menu */
nav ul {
  list-style-type: none;
  padding: 0;
}

article {
  float: left;
  padding: 20px;
  width: 70%;
  background-color: #f1f1f1;
  height: 300px; /* only for demonstration, should be removed */
}
```

```

/* Clear floats after the columns */
section:after {
  content: "";
  display: table;
  clear: both;
}

/* Style the footer */
footer {
  background-color: #777;
  padding: 10px;
  text-align: center;
  color: white;
}

/* Responsive layout - makes the two columns/boxes stack on top of each other
instead of next to each other, on small screens */
@media (max-width: 600px) {
  nav, article {
    width: 100%;
    height: auto;
  }
}
</style>
</head>
<body>

```

<h2>CSS Layout Float</h2>

<p>In this example, we have created a header, two columns/boxes and a footer. On smaller screens, the columns will stack on top of each other.</p>

<p>Resize the browser window to see the responsive effect (you will learn more about this in our next chapter - HTML Responsive.)</p>

```

<header>
  <h2>Cities</h2>
</header>

<section>
  <nav>
    <ul>
      <li><a href="#">London</a></li>
      <li><a href="#">Paris</a></li>
      <li><a href="#">Tokyo</a></li>
    </ul>
  </nav>

  <article>
    <h1>London</h1>

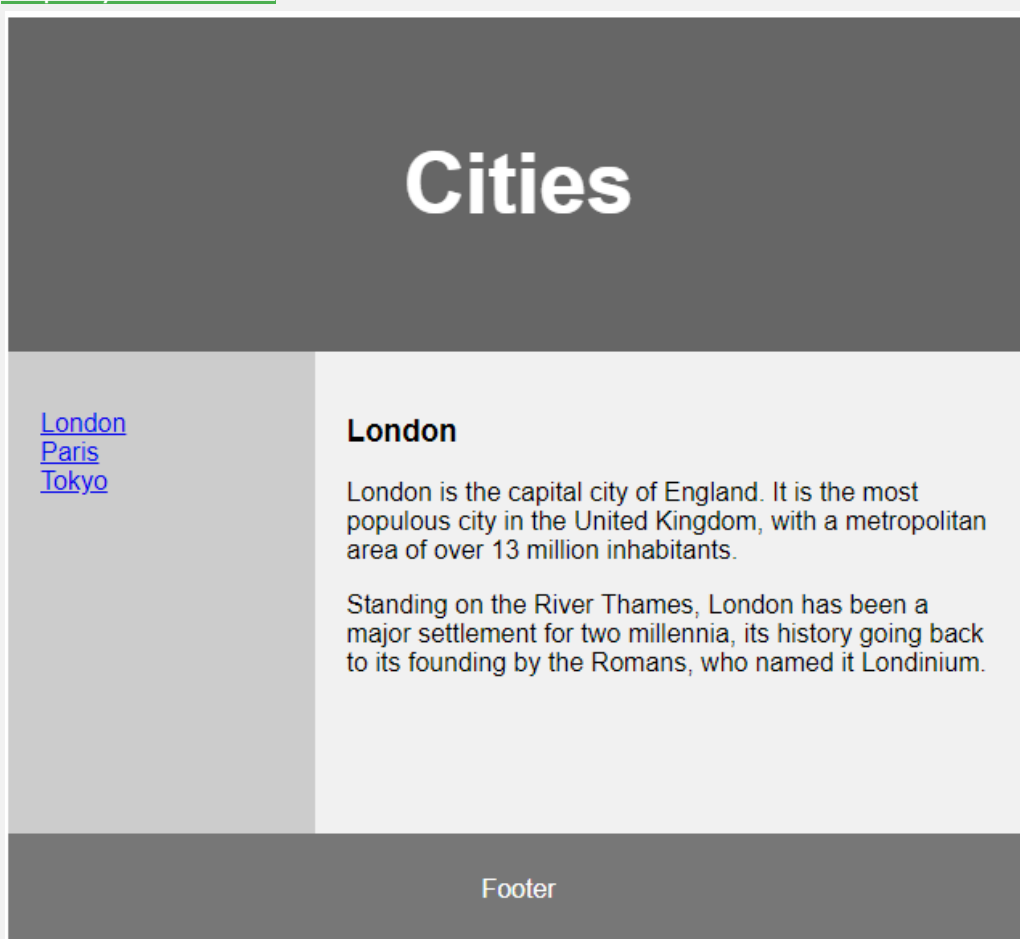
```

```
<p>London is the capital city of England. It is the most populous city in the United Kingdom, with a metropolitan area of over 13 million inhabitants.</p>
<p>Standing on the River Thames, London has been a major settlement for two millennia, its history going back to its founding by the Romans, who named it Londinium.</p>
</article>
</section>

<footer>
  <p>Footer</p>
</footer>

</body>
</html>
```

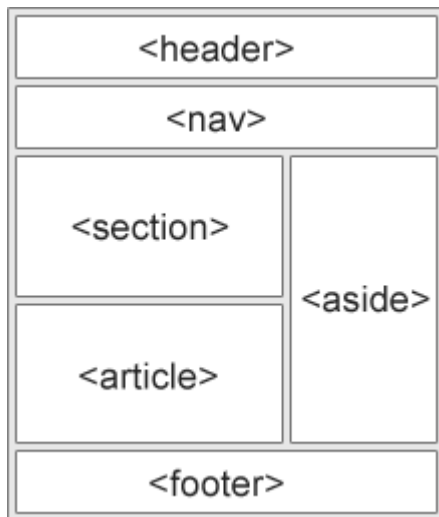
[Спробуйте самі »](#)



Елементи розкладки HTML

Веб-сайти часто відображають вміст у кількох стовпцях (наприклад, журналі чи газети).

HTML5 пропонує нові семантичні елементи, які визначають різні частини веб-сторінки:



- **<header>** - визначає заголовок документа або розділу
- **<nav>** - Визначає контейнер для навігаційних посилань
- **<section>** - Визначає розділ у документі
- **<article>** - визначає незалежну автономну статтю
- **<aside>** - визначає внутрішню статтю, окрім вмісту (наприклад, бічна панель)
- **<footer>** - визначає колонтитул для документа або розділу
- **<details>** - Визначає додаткові деталі
- **<summary>** - визначає заголовок для елемента **<details>**

Методика розмітки HTML

Існує п'ять різних способів створення багатостонових макетів. Кожен шлях має свої плюси і мінуси:

- Таблиці HTML (не рекомендується)
- Властивість CSS float
- CSS flexbox
- Структура CSS
- Сітка CSS

Який вибрати?

Таблиці HTML

Елемент `<table>` не був розроблений як інструмент компоновання! Метою елемента `<table>` є відображення табличних даних. Отже, не використовуйте таблиці для макета сторінки! Вони принесуть безлад у ваш код. І уявіть, як важко буде редизайн вашого сайту через пару місяців.

Порада: НЕ використовуйте таблиці для розташування сторінки!

CSS Frameworks

Якщо ви хочете швидко створити макет, ви можете використовувати фреймворк, наприклад [W3.CSS](#) або [Bootstrap](#).

CSS Floats

Це звичайне використання цілих веб-макетів за допомогою властивості CSS **float**. Його легко впізнати - потрібно просто запам'ятати. **Недоліки:** Плаваючі елементи прив'язані до документообігу, що може пошкодити гнучкість. Докладніше про float можна дізнатися з розділу [CSS Float i Clear](#).

CSS Flexbox

Flexbox - це новий режим компоновання в CSS 3.

Використання **flexbox** гарантує, що елементи ведуть себе передбачувано, коли макет сторінки має відповідати різним розмірам екрану та різним пристроям відображення. **Недоліки:** Не працює в IE10 і раніше.

Дізнайтеся більше про flexbox в главі [CSS Flexbox](#).

Приклад Flexbox

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>CSS Template</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
* {
  box-sizing: border-box;
}

body {
  font-family: Arial, Helvetica, sans-serif;
}

/* Style the header */
header {
  background-color: #666;
  padding: 30px;
  text-align: center;
  font-size: 35px;
  color: white;
}

/* Container for flexboxes */
section {
  display: -webkit-flex;
  display: flex;
}

/* Style the navigation menu */
nav {
  -webkit-flex: 1;
  -ms-flex: 1;
  flex: 1;
  background: #ccc;
  padding: 20px;
}

/* Style the list inside the menu */
nav ul {
  list-style-type: none;
  padding: 0;
}
```

```

/* Style the content */
article {
  -webkit-flex: 3;
  -ms-flex: 3;
  flex: 3;
  background-color: #f1f1f1;
  padding: 10px;
}

```

```

/* Style the footer */
footer {
  background-color: #777;
  padding: 10px;
  text-align: center;
  color: white;
}

```

/* Responsive layout - makes the menu and the content (inside the section) sit on top of each other instead of next to each other */

```

@media (max-width: 600px) {
  section {
    -webkit-flex-direction: column;
    flex-direction: column;
  }
}

```

```

</style>
</head>
<body>

```

```

<h2>CSS Layout Flexbox</h2>

```

<p>In this example, we have created a header, two columns/boxes and a footer. On smaller screens, the columns will stack on top of each other.</p>

<p>Resize the browser window to see the responsive effect.</p>

<p>Note: Flexbox is not supported in Internet Explorer 10 and earlier versions.</p>

```

<header>
  <h2>Cities</h2>
</header>

```

```

<section>
  <nav>
    <ul>
      <li><a href="#">London</a></li>
      <li><a href="#">Paris</a></li>
      <li><a href="#">Tokyo</a></li>
    </ul>
  </nav>

```

```

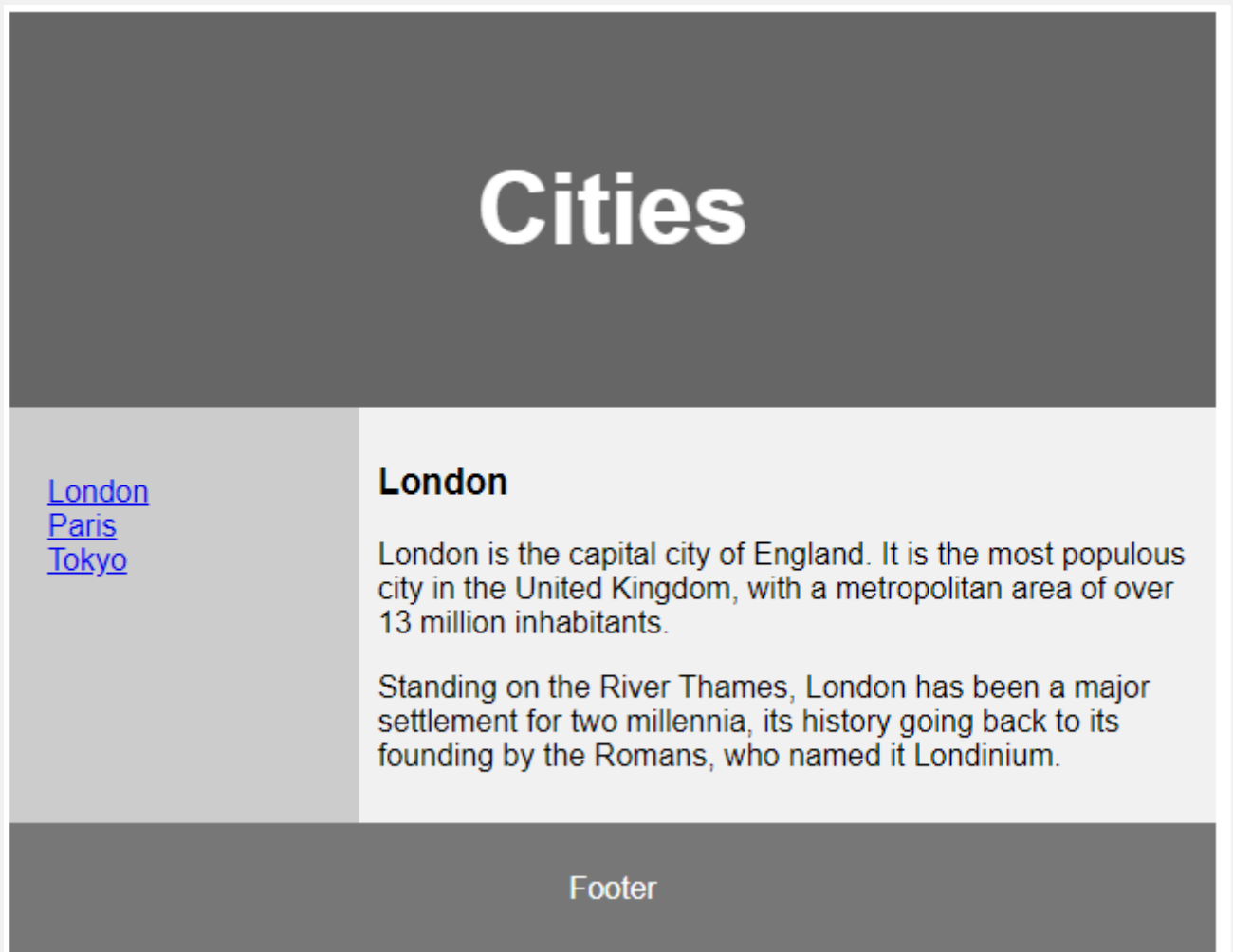
<article>

```

```
<h1>London</h1>
<p>London is the capital city of England. It is the most populous city in the United Kingdom, with a metropolitan area of over 13 million inhabitants.</p>
<p>Standing on the River Thames, London has been a major settlement for two millennia, its history going back to its founding by the Romans, who named it Londinium.</p>
</article>
</section>

<footer>
  <p>Footer</p>
</footer>

</body>
</html>
Спробуйте самі »
```



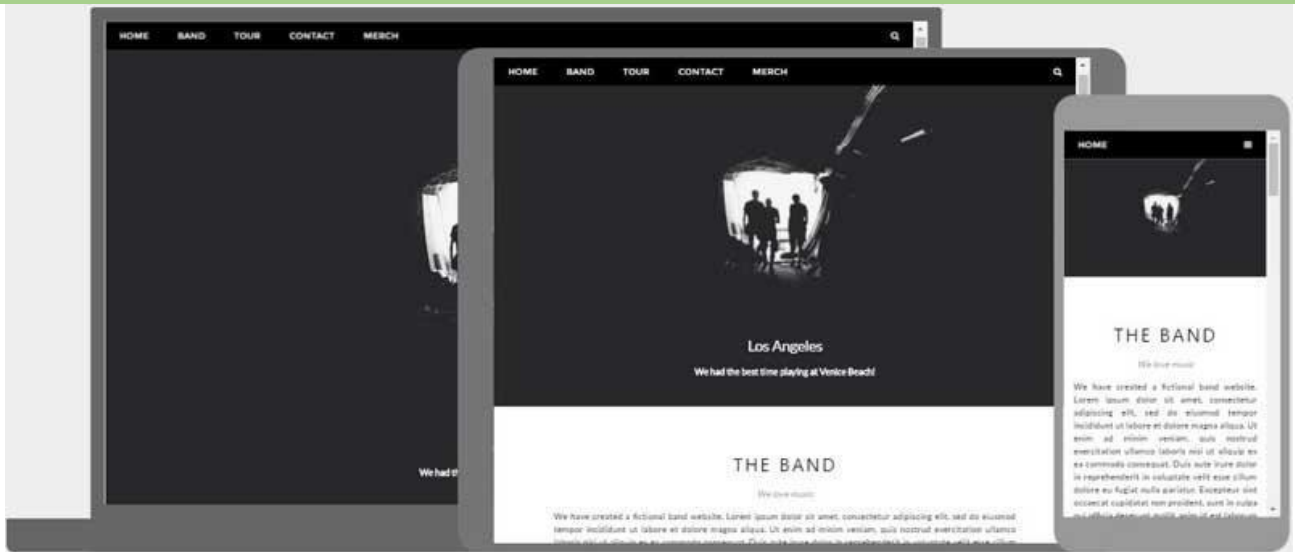
Перегляд сітки CSS

Модуль CSS Grid Layout пропонує систему компонування на основі сітки, з рядками і стовпцями, що полегшує розробку веб-сторінок без використання обтікання і позиціонування.

Недоліки: Не працює в IE, ні в Edge 15, ні раніше.

Докладніше про сітки [CSS див](#) . У розділі Розділ [CSS Grid](#) .

HTML- адаптивний веб-дизайн



Що таке чуйний веб-дизайн?

Чуйний веб-дизайн - це використання HTML і CSS для автоматичного зміни розміру, приховування, зменшення або збільшення веб-сайту, щоб він виглядав добре на всіх пристроях (настільних комп'ютерах, планшетах і телефонах):

[Спробуйте самі »](#)

Примітка: веб-сторінка повинна добре виглядати на **будь-якому пристрої** !

Встановлення вікна перегляду

При створенні веб-сторінок, які відповідають нашим потребам, додайте до **<meta>** всіх своїх веб-сторінок наступний елемент:

Приклад

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

[Спробуйте самі »](#)

Це дозволить встановити вікно перегляду сторінки, який надасть інструкціям браузера, як керувати розмірами та масштабуванням сторінки.

Нижче наведено приклад веб-сторінки без мета-тега вікна перегляду та тієї ж веб-сторінки з метатегом вікна перегляду:

Без мета-тега вікна перегляду:



3 мета-тегом вікна перегляду:



Порада. Якщо ви переглядаєте цю сторінку на телефоні або планшеті, можна натиснути два посилання вище, щоб побачити різницю.

Відповідні зображення

Чуйними зображеннями є зображення, які красиво розміщуються для будь-якого розміру веб-переглядача.

Використання властивості **width** (ширини)

Якщо **width**-властивість CSS встановлено на 100%, зображення буде реагувати і масштабуватиметься вгору та вниз:



Приклад

```

```

[Спробуйте самі »](#)

Зверніть увагу, що в наведеному вище прикладі зображення може бути розширено, щоб бути більшим, ніж його початковий розмір. Краще рішення, у багатьох випадках, буде використовувати **max-width** властивість.

Використання властивості **max-width**

Якщо `max-width` - властивість встановлено на 100%, зображення зменшуватиметься, але ніколи не збільшується до розміру, що перевищує його початковий розмір:



Приклад

```

```

[Спробуйте самі »](#)

Показати різні зображення залежно від ширини браузера

Елемент HTML `<picture>` дозволяє визначати різні зображення для різних розмірів вікна браузера.

Змініть розмір вікна веб-переглядача, щоб побачити, як змінюється зображення в залежності від його ширини:

Приклад

```
<picture>
  <source srcset="img_smallflower.jpg" media="(max-width: 600px)">
  <source srcset="img_flowers.jpg" media="(max-width: 1500px)">
  <source srcset="flowers.jpg">
```

```
  
</picture>
```

[Спробуйте самі »](#)

Show Different Images Depending on Browser Width

Resize the browser width and the image will change at 600px and 1500px.



Show Different Images Depending on Browser Width

Resize the browser width and the image will change at 600px and 1500px.



Show Different Images Depending on Browser Width

Resize the browser width and the image will change at 600px and 1500px.



Розмір тексту

Розмір тексту можна встановити за допомогою модуля "**vw**", що означає "ширину вікна перегляду".

Таким чином, розмір тексту буде відповідати розміру вікна веб-переглядача. Змініть розмір вікна браузера, щоб побачити розмір тексту.

Приклад

```
<h1 style="font-size:10vw">Hello World</h1>
```

[Спробуйте самі »](#)

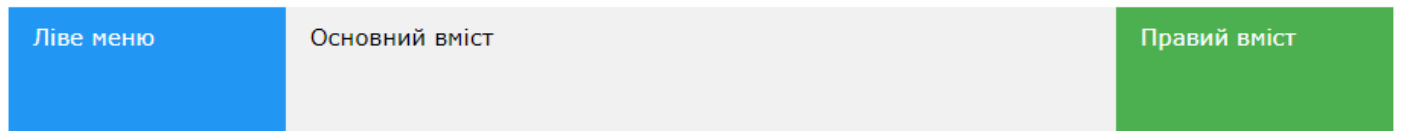
Viewport - це розмір вікна браузера. $1\text{vw} = 1\%$ від ширини вікна перегляду. Якщо область перегляду становить 50см, $1\text{vw} = 0,5\text{ см}$.

Запити медіа

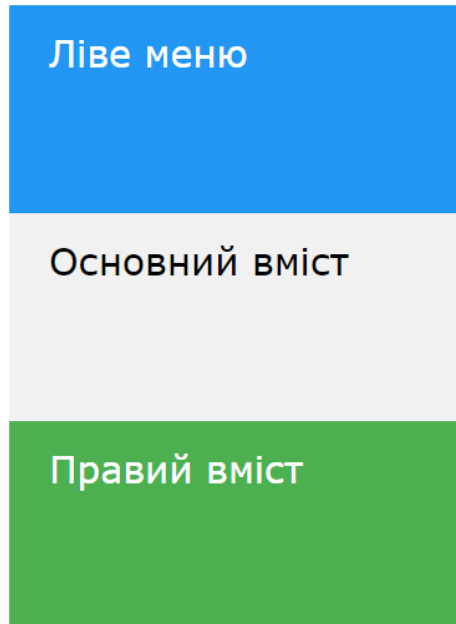
Окрім зміни розміру тексту та зображень, також часто використовуються медіазапити у веб-сторінках, які відповідають вимогам.

За допомогою медіа-запитів ви можете визначити різні стилі для різних розмірів браузера.

Приклад: змініть розмір вікна веб-переглядача, щоб побачити, що три елемента **div** нижче відображатимуться горизонтально на великих екранах і розташовуватися вертикально на невеликих екранах:



Або



Приклад

```
<style>
.left, .right {
  float: left;
  width: 20%; /* The width is 20%, by default */
}

.main {
  float: left;
  width: 60%; /* The width is 60%, by default */
}

/* Use a media query to add a breakpoint at 800px: */
@media screen and (max-width: 800px) {
  .left, .main, .right {
    width: 100%; /* The width is 100%, when the viewport is 800px or smaller */
  }
}
</style>
```

[Спробуйте самі »](#)

Порада. Щоб дізнатися більше про запити медіа та адаптивний веб-дизайн, прочитайте [підручник з RWD](#) .

Відповідна веб-сторінка – повний приклад

Відповідна веб-сторінка повинна добре виглядати на великих екранах робочого столу та невеликих мобільних телефонах.

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
* {
  box-sizing: border-box;
}
.menu {
  float:left;
  width:20%;
  text-align:center;
}
.menu a {
  background-color:#e5e5e5;
  padding:8px;
  margin-top:7px;
  display:block;
  width:100%;
  color:black;
}
.main {
  float:left;
  width:60%;
  padding:0 20px;
}
.right {
  background-color:#e5e5e5;
  float:left;
  width:20%;
  padding:15px;
  margin-top:7px;
  text-align:center;
}
```

```

@media only screen and (max-width:620px) {
  /* For mobile phones: */
  .menu, .main, .right {
    width:100%;
  }
}
</style>
</head>
<body style="font-family:Verdana;color:#aaaaaa;">

<div style="background-color:#e5e5e5;padding:15px;text-align:center;">
  <h1>Hello World</h1>
</div>

<div style="overflow:auto">
  <div class="menu">
    <a href="#">Link 1</a>
    <a href="#">Link 2</a>
    <a href="#">Link 3</a>
    <a href="#">Link 4</a>
  </div>

  <div class="main">
    <h2>Lorum Ipsum</h2>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.</p>
  </div>

  <div class="right">
    <h2>About</h2>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
  </div>
</div>

<div style="background-color:#e5e5e5;text-align:center;padding:10px;margin-top:7px;">© copyright
w3schools.com</div>

</body>
</html>

```

[Спробуйте самі »](#)

Hello World

[Link 1](#)

[Link 2](#)

[Link 3](#)

[Link 4](#)

Lorum Ipsum

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

About

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Hello World

[Link 1](#)

[Link 2](#)

[Link 3](#)

[Link 4](#)

Lorum Ipsum

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

About

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

© copyright w3schools.com

Чуйний веб-дизайн – фреймворки

Існує багато CSS-фреймворків, які пропонують чуйний дизайн. Вони безкоштовні і прості у використанні.

Використання W3.CSS

Відмінний спосіб створити адаптивний дизайн - це використання відповідного стилю, наприклад [W3.CSS](#)

W3.CSS дозволяє легко розробляти сайти, які добре виглядають у будь-якому розмірі; настільний, ноутбук, планшет або телефон:

Демонстрація W3.CSS

Змініть розмір сторінки, щоб побачити реакцію!

Приклад

```
<!DOCTYPE html>
<html>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
<body>
```

```

<div class="w3-container w3-green">
  <h1>W3Schools Demo</h1>
  <p>Resize this responsive page!</p>
</div>

<div class="w3-row-padding">
  <div class="w3-third">
    <h2>London</h2>
    <p>London is the capital city of England.</p>
    <p>It is the most populous city in the United Kingdom,
    with a metropolitan area of over 13 million inhabitants.</p>
  </div>

  <div class="w3-third">
    <h2>Paris</h2>
    <p>Paris is the capital of France.</p>
    <p>The Paris area is one of the largest population centers in Europe,
    with more than 12 million inhabitants.</p>
  </div>

  <div class="w3-third">
    <h2>Tokyo</h2>
    <p>Tokyo is the capital of Japan.</p>
    <p>It is the center of the Greater Tokyo Area,
    and the most populous metropolitan area in the world.</p>
  </div>
</div>

</body>
</html>

```

[Спробуйте самі »](#)

Демонстрація W3Schools

Змінити розмір цієї сторінки!

Лондон

Лондон є столицею Англії.

Це найбільш густонаселене місто у Сполученому Королівстві, де мегаполіс мешкає понад 13 мільйонів жителів.

Париж

Париж - столиця Франції.

Район Парижа є одним з найбільших населених пунктів у Європі з населенням більше 12 мільйонів чоловік.

Токіо

Токіо - столиця Японії.

Він є центром Великого Токійського району і найбільш густонаселеним районом у світі.

Демонстрація W3Schools

Змінити розмір цієї сторінки!

Лондон

Лондон є столицею Англії.

Це найбільш густонаселене місто у Сполученому Королівстві, де мегаполіс мешкає понад 13 мільйонів жителів.

Париж

Париж - столиця Франції.

Район Парижа є одним з найбільших населених пунктів у Європі з населенням більше 12 мільйонів чоловік.

Токіо

Токіо - столиця Японії.

Він є центром Великого Токійського району і найбільш густонаселеним районом у світі.

Щоб дізнатися більше про W3.CSS, прочитайте наш [підручник W3.CSS](#) .

Bootstrap

Інший популярний фреймворк - Bootstrap, він використовує HTML, CSS і jQuery для створення адаптивних веб-сторінок.

Приклад

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootst
rap.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.0/jquery.min.js"></scrip
t>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></sc
ript>
</head>
<body>
```

```
<div class="container">
  <div class="jumbotron">
    <h1>My First Bootstrap Page</h1>
  </div>
  <div class="row">
    <div class="col-sm-4">
      ...
    </div>
    <div class="col-sm-4">
      ...
    </div>
    <div class="col-sm-4">
      ...
    </div>
  </div>
</div>

</body>
</html>
```

[Спробуйте самі »](#)

My First Bootstrap Page

Resize this responsive page to see the effect!

Column 1

Lorem ipsum dolor sit amet, consectetur adipisicing elit...

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris...

Column 2

Lorem ipsum dolor sit amet, consectetur adipisicing elit...

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris...

Column 3

Lorem ipsum dolor sit amet, consectetur adipisicing elit...

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris...

My First Bootstrap Page

Resize this responsive page to see the effect!

Column 1

Lorem ipsum dolor sit amet,
consectetur adipisicing elit...

Ut enim ad minim veniam, quis
nostrud exercitation ullamco
laboris...

Column 2

Lorem ipsum dolor sit amet,
consectetur adipisicing elit...

Ut enim ad minim veniam, quis
nostrud exercitation ullamco
laboris...

Column 3

Lorem ipsum dolor sit amet,
consectetur adipisicing elit...

Ut enim ad minim veniam, quis
nostrud exercitation ullamco
laboris...

Щоб дізнатися більше про Bootstrap, перейдіть до [посібника з Bootstrap](#) .