



web дизайн

Урок 4

Класи й ідентифікатори в HTML і CSS. Тег <div>

Зміст

Класи й ідентифікатори в HTML і CSS. Тег <div>	3
Класи та ідентифікатори	3
Структура сайту.	
Створення блокових елементів	8
Блокова верстка.....	8

Матеріали уроку прикріплені до даного PDF-файлу. Для доступу до них, файл необхідно відкрити в програмі Adobe Acrobat Reader.

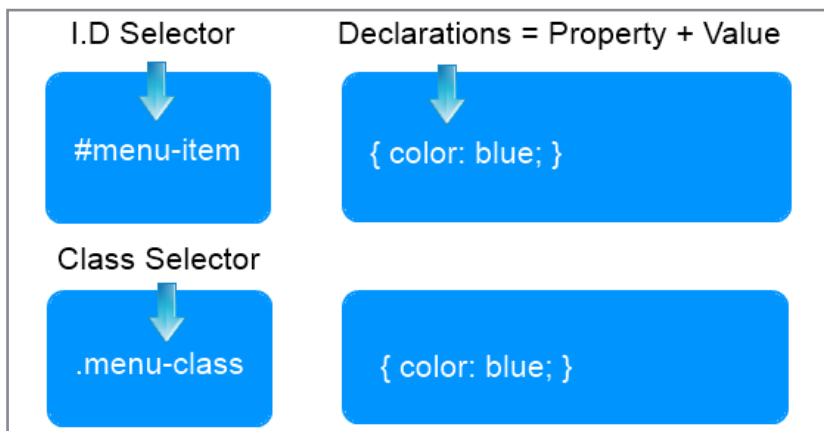
Класи й ідентифікатори в HTML і CSS. Тег <div>

Класи та ідентифікатори

- Ви можете переглянути проєкт у CodePen, для цього перейдіть за [посиланням](#).

Стилізація й оформлення сторінки — це важлива складова в розробці. Змінити вигляд вебсторінки можна за допомогою таблиць стилів CSS. При створенні різних проєктів використовують класи та ідентифікатори, їх також називають селекторами. **Селектор** — це елемент або група елементів, до яких застосовується стиль.

Селектори незамінні в тих випадках, коли розробник має по-особливому оформити сторінку або її елементи, наприклад, зображення, текст або блок тексту. Тобто задати особливий стиль поодиноким елементам.



Малюнок 1

Для задання класів у таблиці стилів перед ім'ям необхідно поставити крапку, а для ID — ґрати (#). Безпосередньо в коді HTML всередині тега, до якого будемо застосовувати селектор, додамо `class=""` або `id=""`, а всередині лапок треба вказати унікальне ім'я (мал. 1).

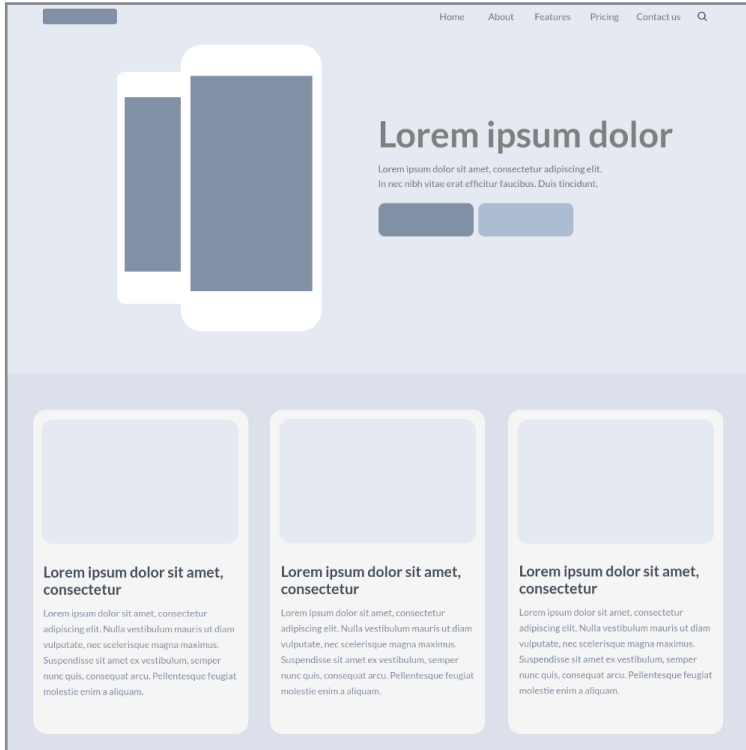
Кожен із селекторів має свої особливості. Загальним для них є те, що і класи, і ідентифікатори чутливі до регістру. Відповідно, потрібно стежити за тим, літери якого регістра вводимо.

Розглянемо відмінності. Ідентифікатори мусять бути унікальними і траплятися в коді тільки один раз. Також вони мають пріоритет перед класами. А класи можна застосовувати неодноразово і комбінувати між собою. При поєднанні різних класів, якщо задані однакові властивості (наприклад, двічі визначено колір фону або розмір шрифту), буде застосовано ту властивість, яку в таблиці стилів записано в останню чергу.

Тобто ID потрібно використовувати в тих випадках, якщо треба визначити стиль тільки для одного елемента. Якщо ж потрібно задати один стиль для кількох елементів, більш доцільно використовувати клас.

Розглянемо на прикладі використання селекторів для оформлення елементів. Додамо текст *Lorem Ipsum* або, як його ще називають, текст-рибу. Він використовується для тимчасового наповнення блоків, щоб побачити, як загалом виглядатиме сторінка в завершеному проєкті (мал. 2).

Відкриємо посилання: <https://loremipsum.io>, в осередок введемо число 3 (кількість абзаців), залишимо Paragraphs і натиснемо Generate (мал. 3). Сервіс згенерує три абзаци, які можна буде додати в код.



Малюнок 2



Малюнок 3

Кожен абзац згенерованого тексту має бути всередині тегів `<p>`. Додамо наступний код у HTML-документ, щоб наповнити сторінку інформацією:

```
<p class="custom-class">
  ...
  ...
</p>

<p id="custom-id">
  ...
  ...
</p>

<p class="custom-class">
  ...
  ...
</p>
```

Замість крапок потрібно додати текст-рибу. Для першого й останнього абзаців `<p>` додамо клас `class="custom-class"`, а для другого — ідентифікатор `id="custom-id"`. Так буде наочно показано, як працюють ці селектори.

Тепер кожному з елементів треба задати свій стиль. Встановимо відступи і поля для всіх `<p>`:

```
p {
  padding: 15px;
  margin: 15px;
}
```

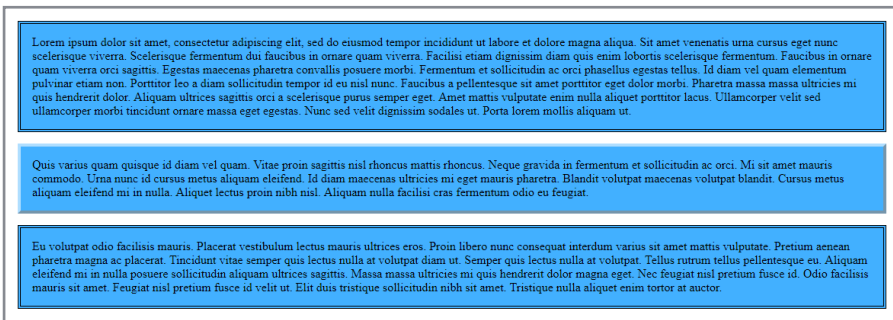
Спеціально для класу `custom-class` задамо колір фону та оформлення:

```
.custom-class {
  background-color: #42b0ff;
  border: 4px double black;
}
```

Аналогічно зробимо для ідентифікатора `custom-id`, але стиль задаємо інший:

```
#custom-id {
  background-color: #42b0ff;
  border: 4px outset #afdeff;
}
```

Результат буде виглядати так само, як показано на малюнку 4.

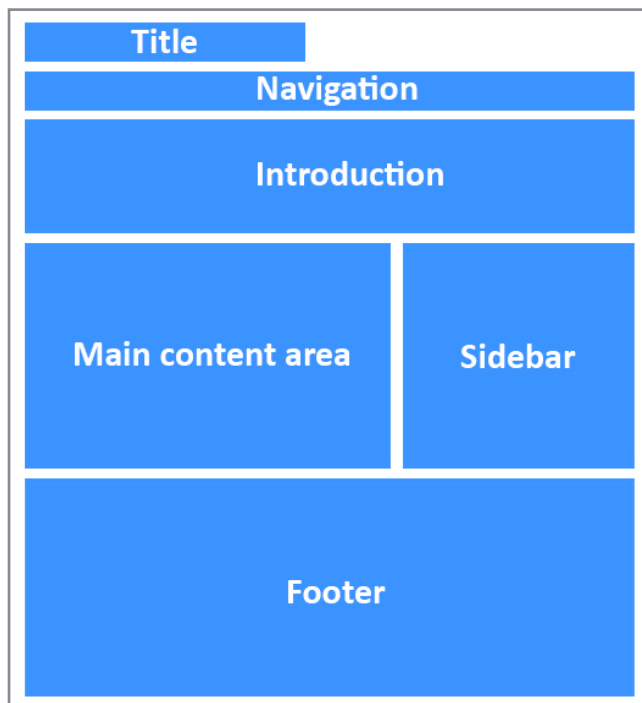


Малюнок 4

У результаті ми розглянули, як правильно працювати з селекторами, а також навчилися стилізувати елементи за допомогою класів і ідентифікаторів.

Структура сайту. Створення блокових елементів

Для створення будь-якої сторінки, розробник має певним чином розташовувати основні блоки сайту: хедер, меню, основну частину і футер. На малюнку 5 наведено елементи, які мають бути на сторінці.



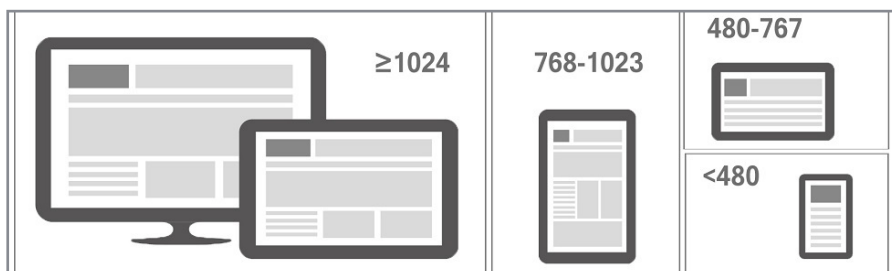
Малюнок 5

Блокова верстка

- Ви можете переглянути проєкт у CodePen, для цього перейдіть за [посиланням](#).

Використовуючи блокову верстку, можна задавати безліч параметрів. Тег `<div>` дає змогу створювати бло-

ки і їхній стиль. У такий спосіб в одному блоці можна буде об'єднати величезну кількість елементів, наприклад, зображення, текст, заголовки, а також налаштувати свої стилі. За допомогою цього тегу зручно створювати адаптивні сайти. Адаптивними називають такі вебсторінки, які, залежно від розміру екрана, змінюють розташування інформації (мал. 6).



Малюнок 6

Створити таку сторінку, як уже говорилося, можна за допомогою тегів <div>. У блоці HTML треба додати всі необхідні елементи, про які згадувалося раніше:

```
<div id="container">
  <div id="header"><h2>Header</h2></div>
  <div id="navigation"><h2>Nav</h2></div>
  <div id="content"><h2>Main content area</h2></div>
  <div id="sidebar"><h2>Sidebar</h2></div>
  <div id="clear"></div>
  <div id="footer"><h2>Footer</h2></div>
</div>
```

Тепер для кожного з елементів задамо стиль. Визначимо відступи для заголовків другого рівня:

```
h2 {  
  margin: auto;  
}
```

Для кожного елемента `div` встановимо відступи для кожної сторони, а також встановимо позиціонування (розташування) тексту:

```
div {  
  margin-top: 2px;  
  text-align: center;  
}
```

Основний елемент заданий за допомогою ID з ім'ям `container`. Для нього встановимо ширину `width`. Необхідно задати властивість так, щоб блоки були адаптивні, а значить, залежно від ширини екрана, зменшувалися або розширювалися. Зробимо це в такий спосіб:

```
#container {  
  width: auto;  
}
```

Для хедера і навігаційного меню встановимо колір фону і висоту:

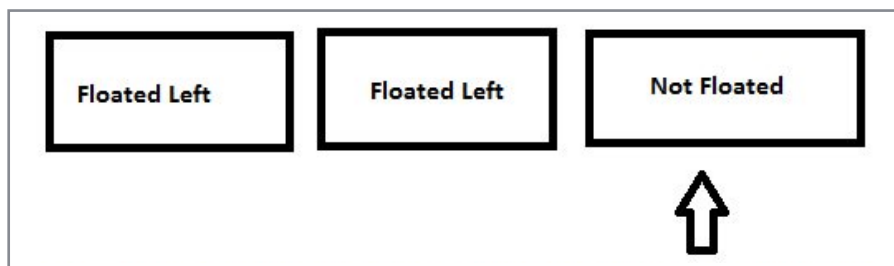
```
#header {  
  background: #85D6E5;  
  height: 150px;  
}
```

```
#navigation {  
  background: #afdeff;  
  height: 40px;  
}
```

В основній частині сайту буде розташований контент (content) і сайдбар (sidebar). Блок content розташуємо в лівій частині сайту, займати він буде більшу частину сторінки. Виділимо для нього 75% вільного місця. Вказуємо значення у відсотках, щоб сторінка залишалась адаптивною. Уся ширина екрана — це 100%, з них 75% займає блок content, а значить, для sidebar виділяємо решту 25%. У коді зі стилями це оформлено так:

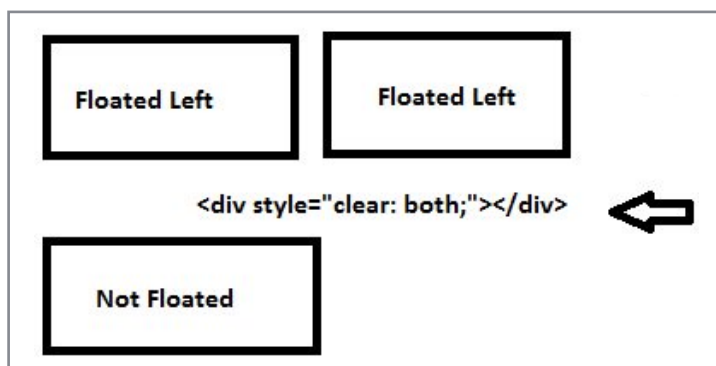
```
#content {  
  background: #85D6E5;  
  float: left;  
  width: 75%;  
  height: 250px;  
}  
  
#sidebar {  
  background: #9fd5fc;  
  float: right;  
  width: 25%;  
  height: 250px;  
}
```

Потім іде футер. Наше завдання зробити так, щоб цей елемент перебував під content і sidebar (мал. 7).



Малюнок 7

Це завдання можна виконати, задавши властивість `clear: both` перед футером (мал. 8).

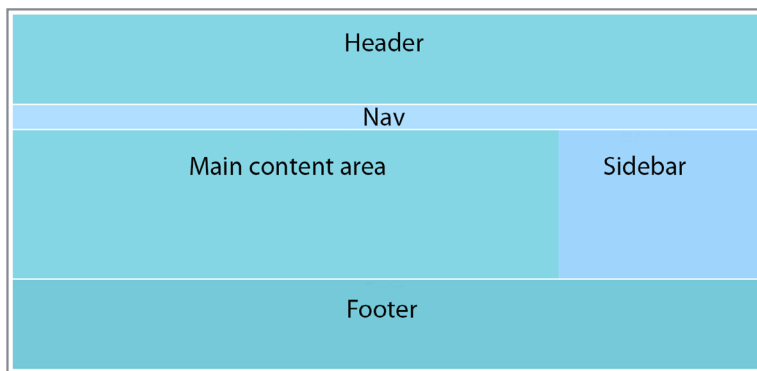


Малюнок 8

```
#clear {
  clear: both;
}

#footer {
  margin-top: 4px;
  background: #75c9d8;
  height: 150px;
}
```

У результаті блоки будуть розташовані саме так, як це і передбачалося. При зміні роздільності екрана змінюватиметься розмір усіх блоків (мал. 9).



Малюнок 9



Малюнок 10

На даний момент ми працювали з мовою HTML, але потрібно розуміти, що синтаксис різних версій мови відрізняється. Наприклад, якщо в першому випадку ми створювали блоки за допомогою тегу `<div>` і відповідних ID, то в HTML5 їх замінюють спеціальні слова, як у правій частині малюнка (мал. 10).

Дані теги зручніше використовувати, адже вони замінюють цілу конструкцію, а для їхнього запису вистачить одного слова.

STEP IT Academy, www.itstep.org

Усі права на захищені авторським правом фото, аудіо та відеотвори, фрагменти яких використані в матеріалі, належать їхнім законним власникам. Фрагменти творів використовуються з ілюстративною метою в обсязі, виправданому поставленим завданням, в межах навчального процесу і в навчальних цілях. Відповідно до ст. 21 і 23 Закону України «Про авторське право й суміжні права». Обсяг і спосіб цитованих творів відповідає прийнятим нормам, не завдає шкоди нормальному використанню об'єктів авторського права і не обмежує законні інтереси автора та правовласників. Цитовані фрагменти творів на момент використання не можуть бути замінені альтернативними, що незахищені авторським правом аналогами, і як такі відповідають критеріям сумлінного і чесного використання.

Усі права захищені. Повне або часткове використання матеріалів заборонено. Узгодження використання творів або їхніх фрагментів проводиться з авторами і правовласниками. Узгоджене використання матеріалів можливе лише за умов згадування джерела.

Відповідальність за несанкціоноване копіювання і комерційне використання матеріалів визначається чинним законодавством України.