19.10.21            11 клас                Вчитель: Балагуряк Є.Ю.

**Тема. Стильове оформлення сторінок. Проектування та верстка веб-сторінок**

**Ознайомтеся з інформацією**

Створивши сторінку, її інформаційне та змістове наповнення, можна приступати до дизайну – тобто графічного оформлення цієї сторінки. За стандартом HTML таке оформлення задається через стилі, а точніше **каскадні аркуші стилів**.

**Стиль** – це набір правил оформлення та форматування, який можна застосувати до різних елементів сторінки. **Каскадні аркуші стилів**, (так звані CSS, Cascading Style Sheets) містять опис формату частини або всього тексту. В одному документі можна описати кілька аркушів, браузер використовуватиме їх каскадом, відповідно до пріоритету цих описів.

Стиль можна задавати для **конкретного тега**, для **стандартних тегів** у документі чи кількох документах, а також для **ідентифікаторів** та **класів**, створених для проекту.

Таким чином виділяється зовнішній аркуш стилів, внутрішній, вбудований стиль тегу та стилі класів та ідентифікаторів. Вони застосовуються по порядку, і найменший пріоритет мають зовнішні аркуші стилів, а найвищий – стилі конкретних тегів. Стилі визначають **форматування**, а також **розташування** елементів у документі. Так, можна змінити вигляд усіх заголовків чи вирівнювання усіх зображень документа. Крім того можна задати певне оформлення для горища чи підвалу сайту, навігаційної панелі тощо.

Прикладами стильових ефектів є наступні:

background – колір тла;   
font-family – вид шрифта;   
font-size – розмір шрифта;   
color – колір шрифта;    
text-decoration – оздоблення тексту;   
font-weight – жирність шрифта;   
text-align - вирівнювання тексту;  
border-width - ширина границі;   
border-style - стиль оформлення границі;  
margin-top – відстань від верхнього рядка;   
line-height – висота рядка.

Довідник стилів: <https://css.in.ua/>

**Підключати аркуші стилів можна:**

* в окремому файлі:

<link rel="stylesheet" type="text/css" href="mysite.css">

* на початку веб-документа:

  <style type="text/css">

   ...

  </style>

* в конкретному тезі:

<h1 style="font-size: 120%; font-family: Verdana, Arial, Helvetica, sans-serif; color: #336">Заголовок </h1>

* для спеціально виділених блоків:

<div  style="font-size: 120%; font-family: Verdana, Arial, Helvetica, sans-serif; color: #336">Hello World!</div>

* для блоку з ідентифікатором:

**#**para1 {

    text-align: center;

    color: red;

}

<p **id**="para1">Hello World!</p>

* для блоків з селектором класу:

**.center**{

    text-align: center;

    color: red;

}

<h1 **class="center"**>Заголовок червоного кольору, вирівнювання по центру</h1>

<p **class="center"**>Текст червоного кольору, вирівнювання по центру</p>

**Перегляньте відео за посиланням**

<https://youtu.be/hZaEsJyCP3Q>

**Виконайте завдання за посиланням**

<https://dystosvita.org.ua/mod/assign/view.php?id=1200>

**Ознайомтеся з інформацією**

**Верстка сторінки**

Верстка сторінок може виконуватись різними способами, і найбільш поширеними є таблиці, фрейми та стилі. Тоді як фрейми на нинішній день використовуються дуже рідко, порівнюючи верстку таблицями та стилями, можна виділити такі переваги та недоліки кожного із цих способів:

|  | **Таблиця** | **Блоки та стилі** |
| --- | --- | --- |
| Колонки | Колонки формуються елементами [таблиці](https://dystosvita.org.ua/mod/page/view.php?id=1185), їх висота однакова і взаємопов'язана. | Колонки створюються різними блоками, їх висота різна і залежить від змісту. |
| Ширина | Якщо ширина [таблиці](https://dystosvita.org.ua/mod/page/view.php?id=1185) явно не вказана, вона обчислюється на основі вмісту [таблиці](https://dystosvita.org.ua/mod/page/view.php?id=1185). | За замовчуванням блок займає всю доступну йому ширину. |
| Розташування | Рядки [таблиці](https://dystosvita.org.ua/mod/page/view.php?id=1185) відображаються в тому порядку, як вони представлені в коді. Клітинки йдуть зліва направо і зверху вниз. | Порядок блоків у коді може не відповідати їх становищу в браузері. |
| Завантаження | Як правило, поки [таблиці](https://dystosvita.org.ua/mod/page/view.php?id=1185) не завантажиться повністю, вміст її не відображатиметься.  Якщо на веб-сторінці розміщена велика таблиця, завантаження сторінки істотно сповільнюється. | Блоки відображаються послідовно, у міру завантаження документа. |
| HTML-код | Код для створення таблиць може бути громіздкий, особливо якщо потрібно об'єднати багато клітинок або зробити кілька вкладених таблиць. | Код, як правило, компактний. |
| Вирівнювання по висоті | Вміст всередині клітинки легко вирівнювати по її верхньому, нижньому краю або середині. | Вирівнювання нетривіальне. |

**Основними принципами верстки стилями та блоками є:**

* розділення змісту та оформлення;
* використання тегу <div>;
* виділення елементів <header> <footer> <nav> <aside>;
* використання стилів CSS;
* використання таблиць лише для подання табличних даних.

Блоки створюються або спеціальними тегами <**header**> <**footer**> <**nav**> <**aside**> <**p**> <**h1**> <**table**>, або універсальним тегом  <**div**>. Такі блоки розміщуються по вертикалі один під одним і займають всю доступну площу за шириною. Якщо в атрибутах задана ширина блоку, то вона окрім власне ширини вмісту має розмістити границі, поля та відступи. Висота блоку визначається браузером автоматично за його вмістом.

При потребі визначити блок у межах рядка, використовується тег <**span**>, наприклад:

<span class="pose">Виділена частина</span> рядка

За рекомендаціями W3C, не можна вкладати блоковий елемент (визначає абзац) всередину рядкового (визначає сукупність символів, не обов'язково окремий абзац):

<a><h1>Заголовок</h1></a> - неправильно

<h1><a>Заголовок</a></h1> - правильно

**Показ елемента (властивість display)**

Ще однією важливою властивістю верстки є властивість показу display - вона вказує ЯК і ЧИ елемент показується на сторінці.

Так властивість **display: none;** приховує елемент, не залишаючи для нього місця (є ще одна схожа властивість: **visibility:hidden**; проте вона передбачає відведення простору під прихований елемент).

**display: inline;**перетворює елемент на рядковий. Якщо застосувати його, наприклад до тегу <div>, він стане вмістом, аналогічним <span> (наприклад, неможливо задати висоту).

**display: block;**- перетворює елемент на блок. Якщо застосувати його, наприклад до тегу <span>, він стане блоком, аналогічним <div>

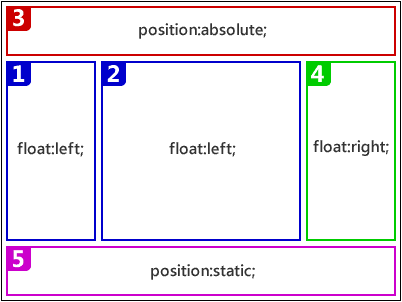
**display: inline-block;** - елемент, форматування вмісту якого відбувається блоком, а сам елемент розміщується на сторінці рядково.

Існують й інші стособи показу елементів.

**Позиціювання (властивість position)**

При розміщенні елементів на сторінці можна використовувати один із таких режимів:

* **static** – на вільне місце після попереднього об'єкта;
* **absolute** – точне місце розміщення;
* **fixed** - не рухається при прокручуванні;
* **relative** –розташування об'єкта відносно того місця, яке би він зайняв, якби для нього використовувалось статичне позиціонування.



На зображенні можна переглянути, як буде розміщено елемент у різних режимах: для блоку **div** можна задавати тло, наприклад **background: green;**  ширину **width: 70px;** та відступи **padding: 50px;** (відступи від всіх країв блоку до вмісту становлять 50px).

Для відносного та абсолютного позиціонування можна задавати відступи для розміщення об'єкта по горизонталі **left**(ліворуч) та **right**(праворуч), та по вертикалі **top**(згори) та **bottom**(знизу).

Наприклад,

**.first {position:fixed; top:450; left:500;}**

Елемент нерухомий при прокручуванні.

**.second {position:absolute; top:100; left:400}**

Елемент показаний поверх інших у заданому розміщенні.

**.third {position:relative; top:200;}**

На місці елемента – порожньо, а сам елемент показаний зі зміщенням.

Стиль

**.feedback {**

**position: fixed; /\* Фіксоване розміщення \*/**

**right: 0; top: 50%; /\* Координати \*/**

**background: red; /\* Колір тла\*/**

**width: 70px; /\* Ширина блоку\*/**

**height: 70px; /\* Висота блоку\*/**

**}**

визначає створення квадрату червоного кольору 70х70 пікселів посередині висоти сторінки і з нульовим відступом справа. Він може бути зафіксованим при прокручуванні, розміщений абсолютно **absolute** чи плаваюче **float**. Так можна додавати, так звані, врізки: цитати, відгуки, коментарі.

При позиціюванні варто враховувати, що існують стилі, визначені стандартно, наприклад відступи між заголовками чи на початку сторінки. Для того, щоб скинути ці значення варто задати стиль, який стосуватиметься усіх елементів (або вказувати конкретні теги, для яких видаляти стандартні відступи та поля):

**\* {margin: 0; padding: 0; }**

**Приклади та заготовки CSS-розміток:**<http://www.dynamicdrive.com/style/layouts/>

**Градієнти**

За допомогою стилів CSS можна створювати градієнти, це атрибут **linear-gradient,** у якому слід вказати напрямок градієнта та назви кольорів (два і більше).

Наприклад, **background: linear-gradient(red, yellow, green);** - лінійний градієнт з трьох кольорів.

Варіанти градієнта: **to right** (зліва направо), **to bottom right,** (кутовий - до правого нижнього кута), або за допомогою вказання кута повороту градієнта, наприклад **-90deg**.

**background: linear-gradient(to right, red,orange,yellow,green,blue,indigo,violet);** - горизонтальна веселка

**background: radial-gradient(red, yellow, green);** - радіальний градієнт

Наприклад, стиль

**.content {background: linear-gradient(orange, yellow);}**

створює градієнт на всю висоту сторінки (чи блоку) - вгорі буде оранжевий колір, внизу - жовтий.

Задавши додаткову властивість **background-attachment: fixed;** отримаємо зафіксований градієнт тла, яке не рухається при прокручуванні сторінки. У стилі можна вказати адресу зображення властивістю **background: url(header.jpg)**. У такому разі за умовчанням це зображення замостить простір по горизонталі та вертикалі. Якщо потрібно розмістити зображення лише по горизонталі, задають атрибут **repeat-x** (**repeat-y** - відповідно, по вертикалі). Зверніть увагу на скорочений запис того факту, що цей атрибут стосується властивості **background**, тому перед repeat-x **НЕМАЄ** крапки з комою.

**.header {background: url(header.jpg) repeat-x; }**

Можна, наприклад, вказати відсутність замощення і розташувати зображення по центру блоку **no-repeat center;**

Якщо ж додамо **background-attachment: fixed;** зображення тла буде зафіксоване при прокручуванні (щоправда, поки відвідувач не прокрутить вміст сторінки до наступного блоку).

Наприклад, **body { background: url(background.jpg) fixed no-repeat center;}** розміщує нерухоме зображення по центру сторінки.

**Поради**

Іноді елементи сторінки можуть накладатися один на одний. У нижній частині такої групи об'єктів буде розташований об'єкт, який описали першим, а зверху – той об'єкт, який описали останнім. Можна також задати властивість **z-index: 2**

Іноді потрібно задати обтікання об'єкту текстом: **float: right;**

Оскільки серед властивостей стилів немає можливості задати висоту блоку, то для розтягування блоку на всю висоту вікна можна скористатись комбінацією стилів **position: fixed; min-height: 100%;**

У разі, якщо вміст блоку перевищує його розмір (або може перевищити), варто задати властивість переповнення **overflow: auto;** (блок при цьому буде розтягуватись).

**Виконайте завдання**

Створити макет сайту на 2 колонки із горищем та підвалом. Ліворуч - блок меню, праворуч - вміст на градієнтному тлі. Прокручуватись має лише вміст, а горище, підвал та меню - залишатись на місці.

Завдання зручно виконувати в онлайновому редакторі <https://jsfiddle.net/> або <http://cssdeck.com> <http://jsbin.com/>

Результат роботи збережіть і надішліть учителю лінк (посилання) на виконану роботу