

Адаптивна верстка

Мета: Допомогти учням засвоїти факти та основні ідеї адаптивної верстки. Забезпечити якість вивчення нового матеріалу. Формувати вміння виділяти головне, актуалізувати, конспектувати, порівнювати, зіставляти. Забезпечити диференційований підхід. Встановити зв'язки між засвоєними та новими знаннями. Формувати групи компетентностей: соціально-трудову, інформаційну, загальнокультурну, соціально-трудову, уміння вчитися.

Тип уроку: засвоєння нових знань; формування вмінь і навичок;

Обладнання та наочність: комп'ютери, підручники, презентація, проектор.

Програмне забезпечення: Google Chrome, Notepad++

Хід уроку

I. Організаційний етап

- *Привітання з класом*
- *Повідомлення теми і мети уроку*

Сьогодні на уроці ми з вами навчимось будувати адаптивні структури сайтів.

II. Мотивація навчальної діяльності

Інформаційні технології постійно змінюються, створюються нові пристрої і веб сайти також повинні йти в ногу та підлаштовуватись під кожен. Саме в такому підлаштуванні нам допоможе адаптивна верстка.

Не забувайте про правила поведінки на уроці.

III. Актуалізація опорних знань

- *Повторення раніше вивченого матеріалу*
 1. Що таке верстка та яка вона буває?
 2. З яким типом верстки ми познайомились на уроці?
 3. Під час роботи за ПК які плюси та мінуси ви виділили в табличній верстці?

IV. Вивчення нового матеріалу

До 2000 року веб розробники користувались звичайним табличним дизайном. Чому? Тому, що не було потреби. Всі в основному монітори були розширенням 1024 на 768. Тому можна було бути впевненим, що сторінка аналогічним чином відображалася б і в користувача.

Наразі існують планшети, ноутбуки, різні монітори до комп'ютера, смартфони та мобільні телефони. І тоді постає питання: «А під яке розширення писати сайт?».

Щоб сайт виглядав коректно використовують адаптивну верстку (гумову, блокову). **Гумова (або блокова) верстка – спосіб опису блоків де розміри задаються не в статичному, а у відсотковому розмірі.**

Блоки можна створювати:

- `<div id="Block1"> div>` опис блоку, як ідентифікатор тегу.`</div>`
- `<div class="Block1"> </div>` опис як клас
- `<div> </div>` опис блоку окремим існуючим тегом
- `<Block1> </Block1>` опис власним тегом.

Додамо тег «header». Такий тег існує та спеціально орієнтований на опис блоку. Розглянемо основні властивості блоку в style.css:

- `border: 2px solid black;` - вкажемо контур блоку товщиною в 2 пункти та за допомогою `solid` вкажемо колір.
- `width: 60%;` - вказуючи ширину у відсотка ми вже робимо адаптивний блок
- `background-color: silver;` - вкажемо заливку блоку (сірий)
- `margin: 10px;` - дана команда означає зовнішній відступ блоку у пікселях
- `padding: 5px;` - відступ внутрішній у всередині блоку (те як текст буде відступати від рамки блоку).
- `border-radius: 15px;` - заокруглення кутів блоку.

Спробуємо додати решту блоків (лівий, правий, основний та підвал). Тег `footer` є зарезервований спеціально для підвалу, а от тих яких немає ми можемо переглядати в програмі Notepad++ забарвленими в чорний колір.

Якщо аналогічним чином описати блоки то ми можемо спостерігати, що ширина не встановлюється для них. Тому правильніше буде описувати блоки ідентифікаторами або класами.

Для скорочення програмного коду можна аналогічні налаштування описати для тегів чи ідентифікаторів вказавши їх через кому.

Окремі команди ми можемо описати окремо. Наприклад ширина блоків. Бокові будуть по 20% від ширини екрану та центральний 40%. Після опису ми бачимо, що місця вистачає і логічно, що блоки мають вибудуватись в рядок. Але кожен блок на початку має налаштування статичного розташування.

Для зміщення блоків потрібно вказати команду `float: left;`. Таке розташування потрібно, щоб було і в решти блоків (не тільки центральних). Якщо це не зробити то блоки «наїдуть» один на одного.

Наші блоки вирівнялись в лінію і вже починають утворювати адекватну структуру, та вони ще не рівні. За логікою опису ширини все вірно. Шапка та підвал по 80%, лівий та правий 20% в центральний 40%. Але ми встановлювали зовнішні відступи. Значення задавались в пікселях, і щоб

вирівняти блоки потрібно до 80% якимось чином додати пікселі. В CSS3 це стало можливим. Таку дію виконає команда `calc(80%+70px)`.

Тепер змінюючи масштаб блоки будуть рівні.

Щоб відцентрувати наші блоки потрібно встановити окремі відступи від лівого та правого краю. Цим нам допоможуть команди `margin-left: 5%;` та `margin-right: 5%;`.

V. Усвідомлення набутих знань й формування вмінь та навичок

● Практичне завдання

Відповідно до розробленої табличної структури сайту в дома розробіть аналогічну блочну (гнучку) структуру.

● Релаксація

Вправа для профілактики короткозорості та порушення зору

Вії опускаються...
Очі закриваються...
Ми спокійно відпочиваємо... (2 рази)
Сном чарівним засипаємо...
Наші руки відпочивають...
Важчають, засинають...(2 рази)
Шия не напружена,
А вона розслаблена...
Губи трохи відкриваються...
Так приємно розслабляються. (2 рази)
Дихається легко, рівно, глибоко.
Ми чудово відпочиваємо.
Сном чарівним засипаємо...

VI. Підведення підсумків уроку

● Бесіда за питаннями

1. В чому полягає суть гнучкого налаштування блоків?
2. Що означає код `margin-left: 5%;` ?
3. Що означає код `border-radius: 15px;` ?

VII. Домашнє завдання

Виконати практичне завдання та надіслати вчителю.