

Принципы построения разметки Grid.

Шаблоны сайтов

№ урока: 4 **Курс:** HTML5&CSS3 Advanced

Средства обучения: Visual Studio, Visual Studio Code, NotePad++

Обзор, цель и назначение урока

Цель урока – ознакомление с режимами разметки **страниц CSS3 Grid**. Выучить основные принципы работы с адаптивной разметкой. Использование сеток для контролируемого отображения элементов на странице.

Изучив материал данного занятия, учащийся сможет:

- Применять Grid модель на разметке страницы.
- Использовать шаблоны Grid модели для адаптивной верстки.

Содержание урока

1. Определение CSS3 Grid Layout.
2. Поддержка Grid Layout браузерами.
3. Определение сеток, колонок и строк.
4. Свойства Grid для отображения контейнеров.
5. Методы отдельных элементов для отображения.
6. Вложенные Grid.
7. Области Grid.

Резюме

- **CSS Grid Layout** – представляет собой двухмерную сетку на которой предполагается размещать элементы пользовательского интерфейса. Элементы можно расположить соответственно строкам и столбцам.
- `display: grid` — создание Grid сетки.
- `grid-template-columns: width width width` — определение количества и размера колонок.
- `grid-template-rows: width width;` — определение размера рядов. (Ряды которым не задана высота используют высоту по умолчанию).
- `grid-auto-rows` — определяет высоту по умолчанию.
- `grid-column-gap` — определяет отступы между колонками.
- `grid-row-gap` — определяет отступы между рядами.
- `grid-auto-flow` — определяют порядок отображения элемента.
- Единица длины `fr` представляет собой долю (fraction) доступного пространства в грид-контейнере и позволяет создавать эластичные (flexible) грид-треки.
- `repeat(n, width)` — позволяет повторять элементы разметки – `n` – количество повторений, `width` – ширина ячеек при повторении.
- `minmax(min, max)` — указывает на поведения блока `min` – минимальная ширина при смещении, `max` –максимальная.
- `Auto-fill` — позволяет элементам при нехватке места перемещается на следующую строку.
- `Auto-fit` — позволяет элементам растягиваться на всю ширину и при нехватке места перемещается на следующую строку.

- `grid-column-start: n` - позволяет позиционировать элемент относительно места от выбранного.
- `grid-column-end: n` - позволяет позиционировать элемент относительно места до выбранного.
- `justify-items` – позволяет элементам позиционироваться горизонтально (`start`, `end`, `center`, `stretch`, `baseline`).
- `align-items` – позволяет элементам позиционироваться вертикально (`start`, `end`, `center`, `stretch`, `baseline`).
- `align-self` – позволяет отдельному элементу позиционироваться вертикально (`start`, `end`, `center`, `stretch`, `baseline`).
- `justify-self` – позволяет отдельному элементу позиционироваться горизонтально (`start`, `end`, `center`, `stretch`, `baseline`).
- `grid-area` – позволяет создать область для управляемого элемента.
- `grid-template-areas` – позволяет элементам позиционироваться относительно значения.

Закрепление материала

- Чем характерна Grid модель в верстке?
- Что такое фракция?
- Что определяет свойство `auto-flow`?
- В чем разница `Autofill` и `autofit` свойств?
- Что позволяют сделать шаблоны в Grid верстке?

Дополнительное задание

Выполните макет страницы с `img02.jpeg` используя Grid модель. Видоизмените страницу под формат автомобилей.

Самостоятельная деятельность учащегося

Задание 1

Выучить основные понятия, рассмотренные на уроке.

Задание 2

Выполните макет страницы с `img01.jpeg` используя Grid модель.

Задание 3

Выполните макет страницы с `img02.jpeg` используя CSS3 Grid модель. Используйте медиа запросы и шаблоны для адаптивной верстки. (При уменьшении экрана – новости должны выстраиваться в колонку, текст прятаться – и оставаться только картинка с кнопкой на ссылку).

Рекомендуемые ресурсы

<http://htmlbook.ru/>

<https://html.com/>

<https://www.html5rocks.com/>