Сценарий

Фон: светлый

Одежда: черный

Доска для написания адреса сайта.

Я правша.

Текст

 **Слайд 1**

Привет, меня зовут Аида и сегодня мы поговорим о css-правилах.

 **Слайд 2**

На прошлом уроке мы познакомились с элементами html-разметки и у нас уже готова структура тестовой странички. \*\* Но пока она выглядит не очень привлекательно. Каким образом сделать так, чтобы она приобрела свой окончательный «красивый» вид? Узнаем в этом видео.

 **Слайд 3**

Допустим, вам нужно отправить посылку. Чтобы работники почты знали, куда ее доставить вы указываете имя человека и его адрес. \*\*

Здесь все работает примерно также - браузеру необходимо конкретно указать, *что именно* ***этот*** *элемент* из всего перечня нам нужен, для этого и используют селекторы.

\*\*

После выбора элемента, нужно указать, как именно он будет выглядеть, для этого описывают свойства селектора. Например, цвет, размер, ширину, прозрачность и т.д.

**Слайд 4**

Рассмотрим данный пример:

В первой части мы видим уже знакомую нам структуру html документа. \*\* То, что мы видим чуть ниже - называют css-правилами. CSS-правила отвечают за внешний вид элементов.

**Слайд 5**

Структурно css-правила выглядят следующим образом:

Это селектор, открывающая фигурная скобка, \*\* свойство, двоеточие, \*\* значение , точка с запятой, далее закрывающая скобка.

\*\*

То есть возвращаясь к примеру мы видим, что селекторами у нас являются \*\* класс .red-font и \*\* тэг р.

Свойств элемента может быть *сколько угодно*, все зависит от исходного дизайна.

 **Слайд 6**

Каким образом браузер интерпретирует css-правила? \*\*

Возьмем исходный html-документ, и запомним, как он выглядел без css-правил. А теперь посмотрим, что изменилось после их подключения. \*\*

Подключив правила мы видим, что текст во всех параграфах стал заглавным. Это произошло потому, что был использован селектор р, \*\* а параграфов в структуре несколько и явно не было указано, какой конкретно из них нам нужен.

 **Слайд 7**

Селекторы бывают нескольких типов: \*\*

* Простые селекторы \*\* – селекторы по тегам \*\* и по классам\*\*
* Составные или вложенные селекторы

Селекторы по тегам содержат имя тега \*\* без символов скобок и \*\* применяются ко всем подходящим тегам, \*\* это мы увидели ранее с тегом р.

Другой тип селекторов - селекторы по классам, \*\* они начинаются с точки, за которой идёт имя класса. \*\* Это наиболее популярный тип селекторов. Такие селекторы применяются ко всем тегам, содержащим данный атрибут.

 **Слайд 8**

Классы могут быть вложенными друг в друга. Можно комбинировать любые типы селекторов через пробел. Такие селекторы называются вложенными и читаются справа налево.

Рассмотрим пример: \*\*

Первое правило выберет все ссылки в списках и покрасит их в зеленый цвет. При этом учитывается вложенность элементов. \*\* Если вы заметили на слайде не указано, что ссылка лежит в элементе списка li, там сразу указан список ul.

Обратите внимание, что если списков несколько, то **все ссылки** в списках на страничке будут зелеными, а не только тот, что находится в блоке с классом box.

Теперь добавим 2 правило: \*\*

Так как мы увеличили уровень вложенности (элементов теперь 3), правило стало более приоритетным, из-за чего ссылка в списке из блока .box поменяла свой окрас. Но если второе правило стояло первым, сразу получился бы 2 вариант, т.к. это правило все еще более приоритетно, чем первое.

 **Слайд 9**

И на последок познакомимся с понятием «множественные классы».

Для этого рассмотрим пример:

Есть 3 одинаковых блока, у которых все стили кроме фона одинаковые. Фон 1го и 3го блоков серый, фон второго блока синий.

\*\*

Класс box с общими стилями есть у *всех* блоков. \*\* Класс box-blue с частными стилями есть только у второго. Стили класса .box-blue, \*\* а именно цвет заднего фона, заменили собой стили общего класса .box

Использование однотипных селекторов позволяет использовать самый простой механизм разрешения конфликтов — по порядку в коде. Размещаем частные CSS-правила после общего, и всё что нужно переопределится. Это будет работать, т.к. CSS – это каскадные таблицы стилей, то есть свойства, что находятся ниже и обладают одним приоритетом, переопределят свойства выше.

**Слайд 10**

На этом все! Надеюсь, это видео помогло вам лучше разобраться, что такое css-правила, для чего они нужны и как устроены. Если вы хотите ознакомиться с темой более подробно и попробовать свои знания на практике заходите на наш сайт **\*пиши текст\*** edu.kazgik.ru