

Конспект «Знакомство с HTML и CSS»

HTML

HTML расшифровывается как «Hypertext Markup Language», то есть «язык гипертекстовой разметки».

Язык HTML отвечает за структуру и содержание страницы. HTML состоит из тегов, а тег состоит из имени, заключённого между знаками «меньше» и «больше». Примеры тегов:

`<h1>`, `<p>`, ``.

Парные теги

Теги бывают парными и одиночными. Парные теги могут содержать текст и другие теги. У парных тегов, в отличие от одиночных, есть вторая половинка — закрывающий тег:

`<h1>Текст заголовка </h1>`

В закрывающей части парных тегов перед именем ставится символ `/` («слэш»).

В парные теги можно вкладывать другие теги. Например, как в списках:

```
<ul>
  <li>Элемент списка</li>
</ul>
```

У вложенных тегов всегда нужно следить за правильным порядком закрытия. Вложенный тег не может закрываться позже родительского:

```
<ul><li>Элемент списка</ul></li> <!-- Плохо -->
<ul><li>Элемент списка</li></ul> <!-- Хорошо -->
```

Не все теги можно вкладывать в другие теги, например, тег `<h1>` нельзя вкладывать в `<p>`.

Одиночные теги

Существуют не только парные, но и одиночные теги. Например, тег `` позволяет добавить картинку в разметку.

Сам по себе `` не имеет смысла. Чтобы этот тег был действительно полезен, необходимо написать внутри него адрес, ведущий к картинке. Делается это с помощью атрибута `src`:

```

```

У тега может быть несколько атрибутов. В этом случае они пишутся через пробел:

```
<тег атрибут1="значение1" атрибут2="значение2">
```

Например, картинке при желании можно задать размеры:

```

```

Комментарии

Код, заключённый между символами `<!--` и `-->`, работать не будет. Если эти символы удалить, то код заработает, это называется «раскомментировать». С помощью комментариев обычно временно отключают какой-то код или оставляют подсказки и разъяснения.

CSS

CSS расшифровывается как «Cascading Style Sheets», то есть «каскадные таблицы стилей».

Язык CSS отвечает за внешний вид страницы.

С помощью CSS можно задавать параметры для любого тега: ширину и высоту, отступы, цвет и размер шрифта, фон и так далее. Все эти параметры задаются с помощью свойств в следующем формате:

свойство: значение;

Например:

```
color: red;  
padding: 10px;
```

Стили к тегам добавляются чаще всего при помощи атрибута `class`.

Например, если мы хотим, чтобы определённые стили, описанные, допустим, в классе `feature-kitten`, применились к тегу `<p>`, то в разметке напишем так:

```
<p class="feature-kitten">...</p>
```

CSS-правила

Можно сказать, что CSS-правило — это группа свойств и их значений, которая целиком применяется к тем тегам, на которые указывает селектор.

И выглядит это так:

```
селектор {  
    свойство1: значение;  
    свойство2: значение;  
}
```

Задавать стили можно не только с помощью атрибута `class`, но и по тегам. Селектор указывает, к каким тегам применятся свойства из CSS-правила. Селекторы по тегам работают проще всего: они выбирают все теги с подходящим именем.

```
p { color: red; }
```

В примере селектором является `p`, и он выбирает все теги с именем `p` (то есть теги `<p>`), а теги с другим именем, например `h1`, не выбирает.

Когда же стилизация задаётся по классам, то стили применяются только к тегам с такими классами.

```
.название_класса {  
    свойство: значение;  
}
```

Миксование классов

У HTML-элемента может быть сколько угодно классов, в этом случае они перечисляются в атрибуте `class` через пробел, например:

```
<li class="product">Товар</li>  
<li class="product hit">Товар, а ещё хит продаж</li>  
<li class="product hit sale">Товар, хит продаж и со ски-и-идкой!</li>
```

Обычно миксование используют так: в один класс выносят общее оформление, а в дополнительных классах описывают его модификации.

Комментарии

В CSS тоже существуют комментарии, их отличие от HTML-комментариев в том, что код или подсказки пишутся между символами `/*` и `*/`.