**HTML теги. Классификация атрибутов**

Атрибуты или свойства тегов позволяют расширить функционал HTML-элементов либо предоставить дополнительные сведения о них. Когда идет речь об атрибутах, важно понимать следующее:

1. У всех тегов есть атрибуты (универсальные, уникальные, событийные или специфические);

2. Они необходимы для дополнения элемента новыми особенностями;

3. Они приводятся в **открывающем теге** через пробел;

4. Типичный вид атрибута – пара **ключ="значение"**.

**1. Универсальные атрибуты**

Исходя из названия такого рода свойства присущи каждому тегу в HTML. Никто не запрещает «навешивать» их налево и направо, но это не всегда разумно. В любом случае, если универсальный атрибут применен «не к месту», он не возымеет никакого действия, и, что хорошо, не приведет к ошибке.

Перечень универсальных свойств

|  |  |
| --- | --- |
| **Атрибут** | **Описание** |
| class | Указывает на имя класса элемента, по которому можно обращаться через каскадные таблицы стилей (или через js-скрипт). Классовое свойство с тем же именем может присутствовать у нескольких тегов |
| id | Указывает на идентификатор имени элемента, через который можно получить доступ к тегу. На странице может присутствовать только одно уникальное имя идентификатора (позволяет уникализировать элемент и получить возможность воздействовать на него скриптами, не затрагивая другие теги) |
| hidden | Если указать такое свойство, то элемент не будет отображаться на странице (в противном случае он видим) |
| accesskey | В качестве значения указывается клавиша на клавиатуре. Позволяет получить доступ к web-элементу – взять его в фокус или активировать – по определённой клавише. Работает пока что не всегда хорошо и имеет особенности в разных браузерах и операционных системах. Тем не менее, активно используется в поисковых системах |
| contenteditable | При выставленном значении «true» позволяет редактировать содержимое элемента |
| style | Применяется для задания CSS-свойств прямо в теге без использования CSS-файла |
| title | Дополнительные сведения об элементе. Если навести на него мышкой, то выводится всплывающая подсказка с содержимым атрибута title |
| dir | Актуален для языков с другим направлением письма (например, арабского). По умолчанию все теги имеют значение «ltr», т.е. слева-направо |
| draggable (принимает значения true или false) | Позволяет перемещать элемент на странице при помощи мыши. Дополнительно необходимо задание скрипта, обрабатывающего перемещение. Ссылки и картинки имеют это свойство по умолчанию |
| lang | Задает язык содержимого элемента (для русского, например, выглядит так: lang=”ru”) |
| spellcheck | (принимает значения true или false) Для проверки орфографии вводимого текста. Актуально для форм и тегов со свойством contenteditable. Важно, чтобы у браузера была включена эта опция также |
| tabindex | Для навигации по элементам страницы можно пользоваться клавишей «TAB». Изменить порядок фокусирования объектов можно через свойство tabindex. Оно принимает числовые значения. Если у первого тега стоит «2», а у второго «1», то при нажатии табуляции сначала фокус переместится ко второму тегу |
| translate (принимает значения yes или no) | Сообщает браузеру, переводить или нет содержимое элемента. Современные программы позволяют переводить содержимое страницы на нужный вам язык. Это можно принудительно запретить для части контента, если имеется такая потребность |
| data-\* | Особое свойство, позволяющее создавать свои собственные атрибуты. Их можно использовать в качестве объектов js-скриптов. Например, никто не запрещает создать такие атрибуты: data-lang=”python”, data-lang=”c++”, data-lang=”javascript” |

**2. Уникальные атрибуты**

Характерны только для одного элемента. Приведем несколько примеров:

1. **ismap**

Имеется только у тега <img> (позволяет получать координаты клика по картинке);

2. **high, low**

Уникальны для тега <meter> – для задания верхней и нижней границы предела определенного диапазона в рамках элемента (например, мы показываем пользователю процент прочтения статьи. Он может варьироваться от 0 до 100 %);

3. **list**

Принадлежит элементу <input> – указывает на id списка, из элементов которого можно выбирать значения для выбранного поля формы.

**3. Специфические атрибуты**

Существенная часть атрибутов относится не к одному тегу, а к нескольким. До HTML5 их было еще больше, но сейчас их запретили, так как предпочтительнее задавать их через таблицы стилей. Озвучим ряд из них, которые актуальны в HTML5:

1. **src**

Ссылка на источник медиа-файла (применим к тегам: <audio>, <embed>, <iframe>, <img>, <input>, <script>, <source>, <track>, <video>).

2. **target**

Задает источник целевого открытия ссылки или валидации формы. Актуально для элементов: <a>, <area>, <base>, <form>. Может принимать следующие значения: \_blank (откроется новая вкладка в браузере), \_self (останемся на текущей вкладке), \_parent (загрузка в родительский фрейм, если страница на них разделена), \_top (несмотря на наличие фреймов они все закроются, а ссылка откроется на всё окно браузера)

3. **height, width**

Определяют высоту и ширину элемента. Применимы для тегов <canvas>, <embed>, <iframe>, <img>, <input>, <object>, <video>. Значения – положительные числа (трактуются как пиксели). Можно использовать и проценты - считаются относительно родительского элемента.

4. **name**

Для задания имени элемента. Применяется для следующих тегов: <button>, <fieldset>, <form>, <iframe>, <input>, <map>, <meta>, <object>, <output>, <param>, <select>, <textarea>. По этому названию к объекту можно обращаться через JavaScript, например.

**4. Событийные атрибуты**

Связаны с JavaScript-событиями. При взаимодействии пользователя с элементами эти события могут срабатывать и изменять поведение объектов. Их практическое использование будет рассмотрено в теме по JavaScript. Сейчас лишь озвучим некоторые из них для понимания:

1. **onclick**

Скрипт срабатывает при клике мышкой по элементу (можно вызвать всплывающее окно, изменить размер или цвет, перенаправить на другую страницу и т.п.).

2. **oncopy**

Вызывает скрипт при попытке скопировать содержимое элемента (таким способом часто запрещают копировать содержимое либо добавляют ссылку на первоисточник, когда вы вставляете скопированное в текстовый редактор).

3. **onmousemove**

Эффект при движении мыши (позволяет создавать интересные эффекты на странице: смещение, увеличение, тряску и т.п.).

4. **onkeydown**

Обработка нажатия определенной клавиши.

5. **onfocus**

Изменение поведения объекта при фокусе на нем (например, когда поставили курсор в поле для ввода логина, само поле может изменить цвет, стать больше, дать какую-то подсказку).

6. **onoffline**

Поведение страницы в момент отсутствия активного подключения к сети (удобно, когда под рукой нет Интернета, а сайт продолжает функционировать, хоть и в ограниченном режиме).

Подытожим

Как видно, атрибуты тегов существенно повышают возможности HTML. Они позволяют не просто добавлять новые характеристики и менять свойства элементов, но и дают возможность использовать фактически неограниченный потенциал языка JavaScript.