

УДК 004.021:421

https://doi.org/10.33619/2414-2948/97/08

СОЗДАНИЕ ИНТЕРАКТИВНОГО САЙТА С ИСПОЛЬЗОВАНИЕМ ОБЪЕКТНОЙ МОДЕЛИ ДОКУМЕНТА

©*Кадыркулова Н. К., Ошский технологический университет им. акад. М.М. Адышева, г. Ош, Кыргызстан, kadyrkulova74@mail.ru*

©*Кыргызбаев У. Б., Ошский технологический университет им. акад. М.М. Адышева, г. Ош, Кыргызстан, kulukman@gmail.com*

CREATING AN INTERACTIVE SITE USING THE DOCUMENT OBJECT MODEL

©*Kadyrkulova N., Osh Technological University named by M.M. Adyshev, Osh, Kyrgyzstan, kadyrkulova74@mail.ru*

©*Kyrgyzbaev U., Osh Technological University named by M.M. Adyshev, Osh, Kyrgyzstan, kulukman@gmail.com*

Аннотация. Представлен ход создания интерактивного сайта с использованием объектной модели документа. В процессе исследования показаны возможности объектной модели для компонентной реализации интерактивного сайта с применением пользовательского интерфейса JavaScript. В результате работы был спроектирован и разработан простой интерфейс веб-страницы, формирующей список компонентов для магазина продаж компьютеров.

Abstract. We look at creating an interactive website using the document object model. The research process shows the capabilities of the object model for the component implementation of an interactive site using the JavaScript user interface. As a result of the work, a simple interface for a web page was designed and developed that forms a list of components for a computer sales store.

Ключевые слова: объектная модель документа, узлы, веб-браузер, CSS-файлы, HTML-документ, интерфейс JavaScript.

Keywords: document object model, nodes, web browser, CSS files, HTML document, JavaScript interface.

Веб-сайт должен состоять из HTML-документа index.html. Используя браузер, просматриваем веб-сайт, который отображает файлы HTML и любые файлы CSS, которые добавляют правила стиля и макета. Браузер также создает представление этого документа, известное как объектная модель документа. Используя DOM, JavaScript может получать доступ и изменять контент и элементы веб-сайта. DOM (*от англ. Document Object Model*) — является прикладным программным интерфейсом (API), который определяет порядок передачи информации к объектам, из которых является документ [1–6].

DOM-модель документа используется при работе с элементами сайта на стороне заказчика. Вы можете выполнить обработку содержания веб-страниц через JavaScript. Важность DOM-модели заключается в следующем: после получения HTML-документа, браузер создает соответствующий объект на ОЗУ для каждого HTML-элемента, с его свойствами, методами и событиями. Для получения ссылки на какой-либо объект можно



использовать JavaScript. Основная задача DOM заключается в возможности изменения HTML-страницы и того, как они отображаются на экране с помощью JavaScript. Связи между объектами различных уровней, которые показаны на Рисунке 1, означает, что объект верхнего уровня содержит ссылки на объект нижнего уровня. Например, существует связь между объектом окна с документом. Это значит, что у объекта Window есть свойство, которое называется Document, где содержится ссылка на объект типа Documents.

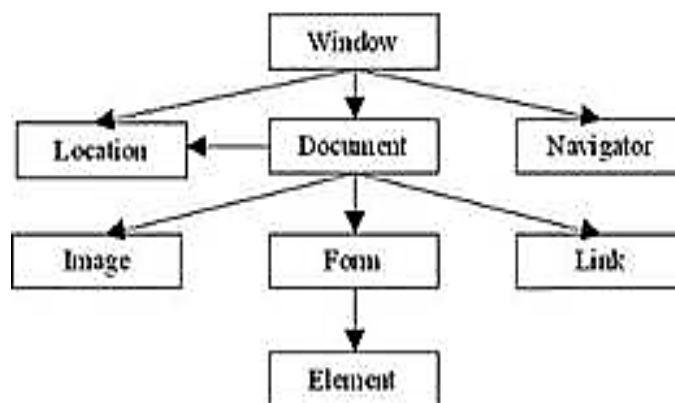


Рисунок 1. Иерархия объектов веб-страницы

Основным объектом для работы с DOM является глобальная переменная document, которая относится к объекту Document, описывающему документ HTML, отображаемый в окне браузера. Массивы используются для хранения произвольного числа ссылок по объектам, расположенным на веб-странице. Таким образом, свойства объекта Document, который предназначен для совместимости с компонентами веб-страницы выглядят следующим образом:

- links[] — массив ссылок на гипертекстовые ссылки, сформированные в HTML-документе с помощью элементов <a>;
- images[] — массив ссылок на изображения, включенные в веб-страницу с помощью элемента ;< li="">;< >
- forms[] — массив ссылок на формы, созданные с использованием элементов<form>.< li=""></form>.< >

В этих таблицах содержатся ссылки объектов Link, Image и Form так же, как и появляются в тексте HTML-документа. Так, images [0] ссылается на первое изображение в составе веб-страницы, images [1] — на второе изображение и т. д. Дерево представляет собой объекты, которые называются узлы. Существует множество видов узлов, однако большинство времени вы работаете с узлом элементов HTML, текстовыми узлом любого текстового содержимого и узлом комментариев, комментируемым кодом. Объектом документа является собственный раздел, который находится в корне каталога.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Nodes</title>
  </head>
  <body>
    <h1>This is an element node</h1>
    <!-- This is a comment node -->
    This is a text node.
```

```
</body>  
</html>
```

Узел от которого дуга идет к этому узлу, будем называть родительским узлом или родителем этого узла [7].

Работая с узлами DOM, их также называются родителями, потомками и братьями и сестрами, исходя из того, насколько они относятся к другим узлам. Объект документа — встроенное устройство, в котором содержится множество характеристик и способов. Мы получаем доступ к объекту и манипулируем его с помощью JavaScript. И манипулировать DOM и делать веб-страницу интерактивной! Поскольку уже не ограничивается простой созданием статического сайта с содержанием HTML (Рисунок 2).

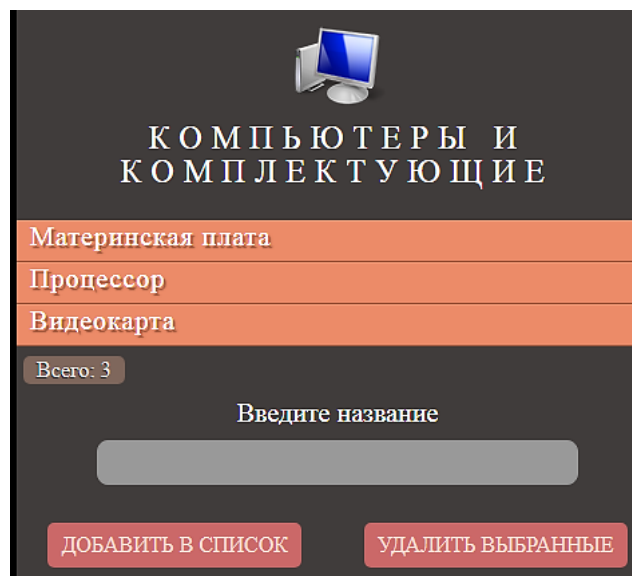


Рисунок 2. Фрагмент интерактивных сайтов

Для того, чтобы создать механизм управления страницами на стороне заказчика, предлагалось использовать модель объекта документа. Суть модели заключается в следующем: каждый HTML контейнер является объектом, идентифицированным тройкой: свойства; методы; события.

Объектная модель может рассматриваться как способ соединения страниц с браузерами. Модель объекта представляет объекты, методы, свойства и события, существующие и происходящие в браузере программное обеспечение, поэтому подойдет для использования HTML и скрипта исходного кода страницы. Можно его использовать для передачи своих пожеланий браузеру и потом посетителю сайта. Браузер выполнит наши инструкции и корректно изменит на экране страницу.

Объекты, имеющие одинаковый набор свойства, метода и события, объединены в классы единого типа объектов. Классы представляют собой описания возможного объекта. Сам объект отображается только при загрузке документа в браузере или при выполнении программы. Используются методы и особенности этой модели, чтобы выполнить каждое действие на странице. DOM- модель представляет собой веб-документ, в виде дерева тегов. Тег в этом дереве известен как узел элемента. Содержимое тега (текст) является текстовым узлом. Также атрибуты маркеры рассматриваются как узлы дерева. Узлы элемента и узлы текста — это типы элементов. Каждый тег дерева может быть доступен для получения или изменения содержимого дерева или выполнить какую-либо другой операции. Все это

осуществляется с помощью свойства и методов DOM. Мы все узнали о DOM-моделях и работе с деревом и узлами.

Используя JavaScript, вы получите доступ к DOM, чтобы управлять сайтом и делать его интерактивными. Для работы HTML-кода и сценария исходной страницы используются объекты, методы, свойства, события, присутствующие и происходящие в браузере.

Разработан интерактивный сайт с использованием языка Javascript с применением модели объектов DOM. Создана Web-страница, которая сформировала список компонентов для магазина продаж компьютеров, а также получение доступа к элементам при помощи программы JavaScript.

Список литературы:

1. Флэнаган Д. JavaScript. Подробное руководство. СПб: Символ Плюс, 2012. 1080 с.
2. Дарнелл Р. JavaScript: Справочник. СПб: Питер, 1998. 192 с.
3. Гончаров А. Самоучитель HTML. СПб.: Питер, 2001. 214 с.
4. Мейер Э. А. CSS. Каскадные таблицы стилей. Подробное руководство. СПб: Символ-Плюс 2008. 575 с.
5. Седерхольм Д. CSS ручной работы. СПб.: Питер, 2011. 240 с.
6. Мэтью М. Создание web-сайтов. Основное руководство. СПб.: Эксмо, 2010. 768 с.
7. Кадыркулова Н. К., Сатыбаев А. Д. Моделирование дистанционного обучения на основе байесовской сети // Проблемы автоматизации и управления. 2019. №1. С. 84-89.

References:

1. Flenagan, D. (2012). JavaScript. Podrobnoe rukovodstvo. St. Petersburg. (in Russian).
2. Darnell, R. (1998). JavaScript: Spravochnik. St. Petersburg. (in Russian).
3. Goncharov, A. (2001). Samouchitel' HTML. St. Petersburg. (in Russian).
4. Meier, E. A. (2008). CSS. Kaskadnye tablitsy stilei. Podrobnoe rukovodstvo. St. Petersburg. (in Russian).
5. Sederholm, D. (2011). CSS ruchnoi raboty. St. Petersburg. (in Russian).
6. Metyu, M. (2010). Sozdanie web-saitov. Osnovnoe rukovodstvo. St. Petersburg. (in Russian).
7. Kadyrkulova, N. K., & Satybaev, A. D. (2019). Modelirovanie distancionnogo obucheniya na osnove bajesovskoj seti. *Problemy avtomatiki i upravleniya*, (1), 84-89. (in Russian).

Работа поступила
в редакцию 25.10.2023 г.

Принята к публикации
04.11.2023 г.

Ссылка для цитирования:

Кадыркулова Н. К., Кыргызбаев У. Ы. Создание интерактивного сайта с использованием объектной модели документа // Бюллетень науки и практики. 2023. Т. 9. №8. С. 65-68. <https://doi.org/10.33619/2414-2948/97/08>

Cite as (APA):

Kadyrkulova, N., & Kyrgyzbaev, U. (2023). Creating an Interactive Site Using the Document Object Model. *Bulletin of Science and Practice*, 9(8), 65-68. (in Russian). <https://doi.org/10.33619/2414-2948/97/08>