

Применение JavaScript в системе Dreamweaver

Методические указания к лабораторной работе

Цель работы: научиться использовать стандартные и собственные решения на языке JavaScript при проектировании Web – сайтов.

Теоретическая справка

Web-страницы могут быть статическими и динамическими.

Статическая Web-страница имеет постоянную структуру и предназначена для представления гипертекстовых данных. Статические Web-страницы *макетируются* при помощи языка разметки HTML.

Динамическая Web-страница может выполнять, по-крайней мере, две функции:

- служить для ввода – вывода данных;
- изменять свое содержание в зависимости от запросов пользователя.

Динамические Web-страницы *программируются* при помощи языков программирования. Среди языков программирования Web-страниц популярен язык JavaScript – простой и мощный язык, позволяющий создавать динамические Web-страницы как на стороне клиента, так и на стороне сервера.

Язык JavaScript является системно-независимым и совместим между всеми типами компьютеров, работающими в Интернет. Программы на языке JavaScript включаются в состав HTML документа и, фактически, этот язык может считаться расширением состава команд HTML. Поэтому для создания программ на JavaScript не требуется никаких дополнительных средств, необходим лишь браузер, поддерживающий JavaScript (браузеры Netscape Navigator 2.0 и выше или Microsoft Internet Explorer).

1. Объектная модель JavaScript

JavaScript - это объектно-ориентированный язык, несмотря на то, что он реализует парадигму объектно-ориентированного программирования не в полном объеме.

Все операции, которые можно исполнять в программе на JavaScript, описывают действия над известными объектами.

Объектами являются *элементы рабочей области программы Netscape Navigator* и *контейнеры языка HTML*. Собственно объектная ориентированность JavaScript на этом и кончается. Есть только объекты с набором свойств и набор функций над объектами.

Все встроенные объекты JavaScript берут свое начало от рабочей области Netscape, и их можно представить в виде следующей иерархии, изображенной на рис.1.

Как видно из рисунка, классы объектов *Окно (window)* и *Документ (document)* являются основными.

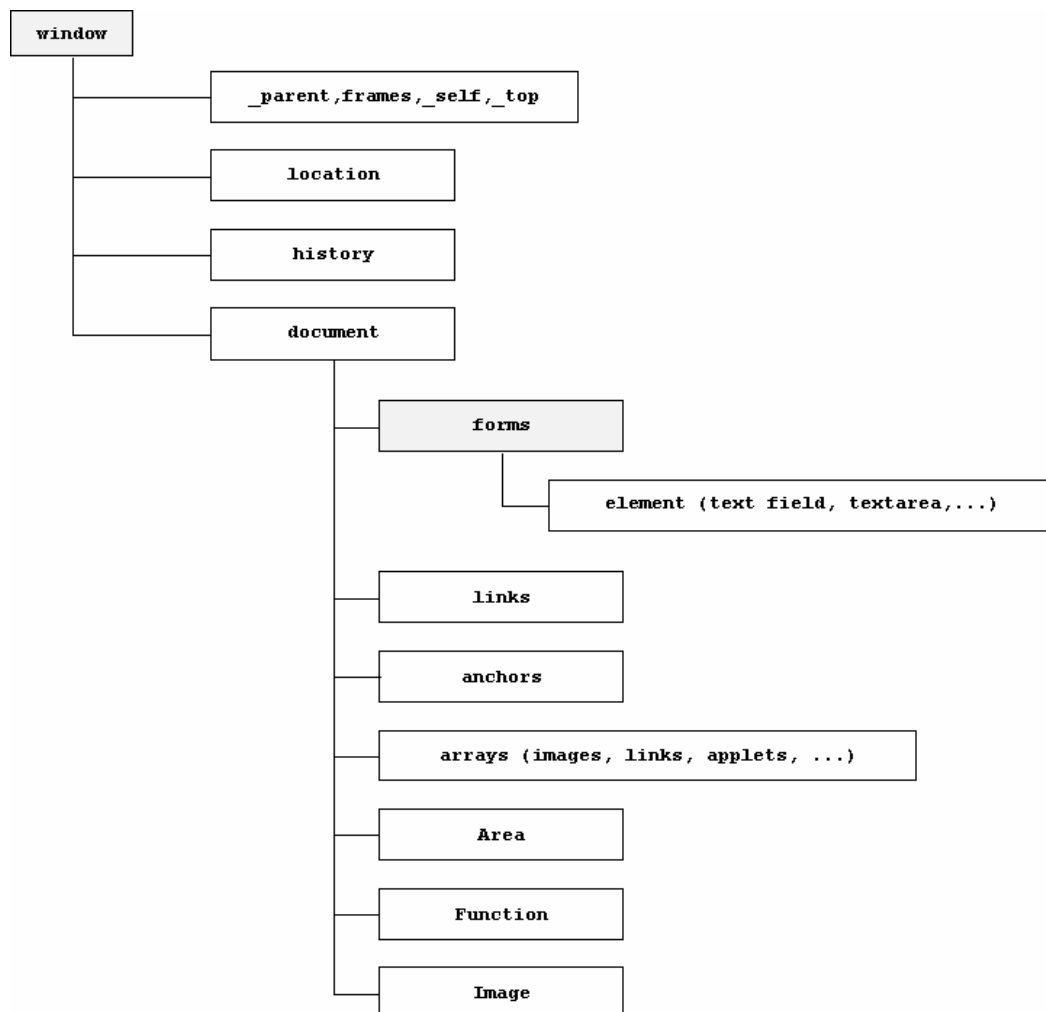


Рис.1. Объектная модель JavaScript

Объекты и свойства

Объект - это конструкция со свойствами, которые являются переменными JavaScript. Свойства могут быть другими объектами. Функции, связанные с объектом известны как *методы* объекта.

Объект JavaScript имеет свойства ассоциированные с ним. Обращение к свойствам объекта выполняется при помощи точечной нотации:

`objectName.propertyName`

Функции и методы

Функции - один из фундаментальных встроенных блоков в *JavaScript*.
Функция - *JavaScript* процедура - набор утверждений, которые выполняют определенную задачу.

Определение функции состоит из ключевого слова `function`, сопровождаемого

- именем функции,
- списком аргументов функции, приложенной в круглых скобках, и отделяемые запятыми,
- JavaScript - утверждениями, которые определяют функцию, приложенные в фигурных скобках, `{...}`

Функции, определенные в текущей странице, действуют в любом месте данной страницы. Лучше всего определять все функции в тэге `HEAD` страницы. Когда пользователь загружает страницу, сначала загружаются функции.

Метод – это функция, связанная с объектом. Вы определяете метод таким же образом, так как вы определяете стандартную функцию. Затем, используйте следующий синтаксис, чтобы связать функцию с существующим объектом:

object.methodname = function_name

где *object* - существующий объект, *methodname* - имя, которое вы присваиваете методу, и *function_name* - имя функции.

Вы можете вызывать метод в контексте объекта следующим образом:

```
object.methodname (params);
```

Кроме методов существуют и другие функции, которые больше похожи на функции из традиционных языков программирования и позволяют работать со стандартными математическими типами или управлять процессом выполнения программы.

События

События в JavaScript - это аналог программных прерываний. Эти события также ориентированы на работу в World Wide Web, например, на загрузку страницы в рабочую область Navigator'a или выбор гипертекстовой ссылки. Используя события, автор гипертекстовой страницы и программы ее отображающей может организовать просмотр динамических объектов, например, бегущая строка, или управление многооконным интерфейсом. Среди других функций, легко программируемых при помощи JavaScript, следующие:

- проверка правильности ввода данных,
- выдача на экран сообщений,
- обработка данных,
- запись в файл,
- отправка ответа пользователю.

2. Поддержка JavaScript в системе Dreamweaver

В системе Dreamweaver поддерживаются несколько технологий программирования, в том числе и технология JavaScript. Поддержка JavaScript заключается в следующем.

2.1. Поддержка файлов скриптов.

Если код JavaScript достаточно объемен, то целесообразно оформлять программы на языке JavaScript в виде отдельных файлов с расширением *.js.

Ссылка на подобный файл, например, с именем `main.js`, в гипертекстовом документе может выглядеть следующим образом:

```
<SCRIPT language=JavaScript1.2 src="main.js"></SCRIPT>
```

В данном примере подразумевается, что файл `main.js` находится в главном каталоге сайта. В противном случае в ссылке `src=` применяют маршрут к файлу.

В системе Dreamweaver имеется редактор файлов для JavaScript, включающий ряд инструментов, характерных для редакторов: форматирование текста, вставка тэгов и т.д. На рис.2 показан пример открытого файла `main.js` в редакторе Dreamweaver вместе с окном вставки тэгов и окном справочной системы, инициированной (нажатием shift-F1) для оператора `function` языка JavaScript.

Целесообразно включать в файлы скриптов функции, которые могут быть вызваны из других страниц сайта. При этом в каждой странице, где имеется вызов функции из файла скриптов, необходимо поместить ссылку на файл скриптов.

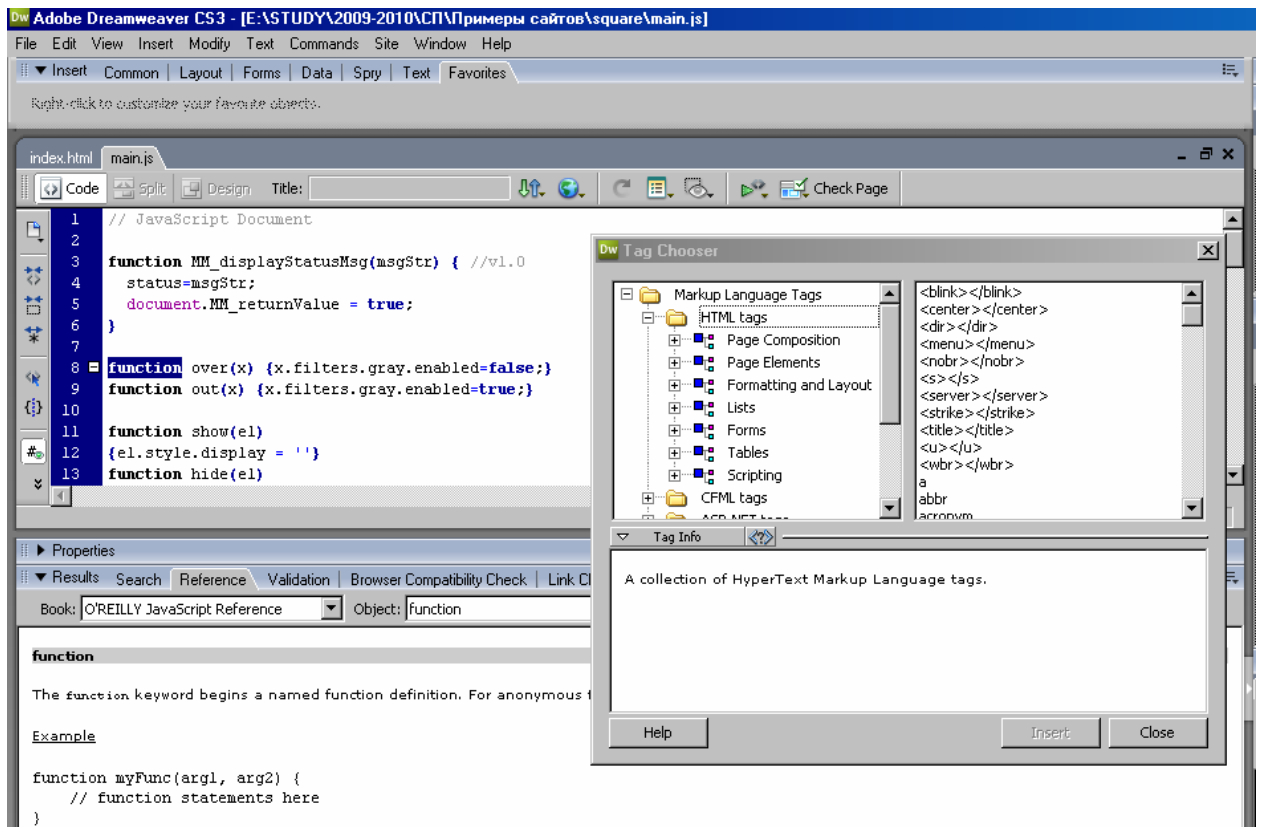


Рис. 2. Документ JavaScript и справочная система.

2.2. Использование JavaScript в гипертекстовых документах.

Код JavaScript может применяться непосредственно в гипертекстовых документах, то есть вместе с html – кодом Web-страницы. Для поддержки JavaScript в гипертекстовых документах в системе Dreamweaver имеется специальный инструмент задания действий (Behaviours) в ответ на события, связанные с конкретными объектами, используемыми в коде Web-страницы.

Действия задаются в специальной вкладке Behaviours окна обозревателя тэгов (Tag Inspector). На рис. 3 показана вкладка Behaviours.

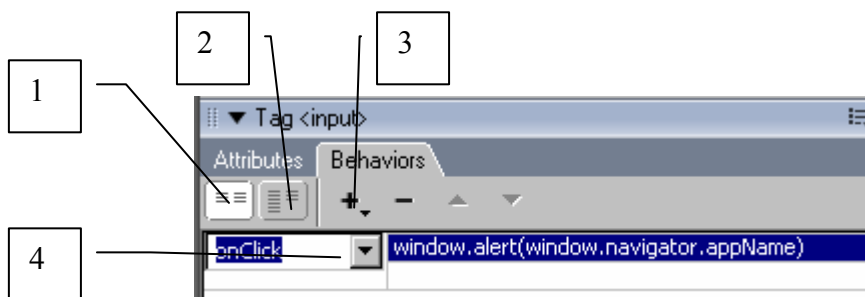


Рис. 3. Вкладка Behaviours окна обозревателя тэгов:

- 1 – список событий, установленных для выбранного объекта;
- 2 - список всех событий для выбранного объекта;
- 3 – элемент добавления/удаления скрипта реакции на события
- 4 – строка «событие – скрипт»

Алгоритм задания действий при работе с html – кодом Web-страницы состоит в следующем.

1. Выбрать объект Web-страницы, для которого будут установлена пара «событие – скрипт».
2. В строке «событие» правой кнопкой мыши инициировать выпадающий список событий; выбрать из него нужное событие.
3. С помощью элемента добавления/удаления скрипта выбрать из раздела скриптов по категориям нужный скрипт.

Для удобства все скрипты структурированы по категориям – см. рис.4

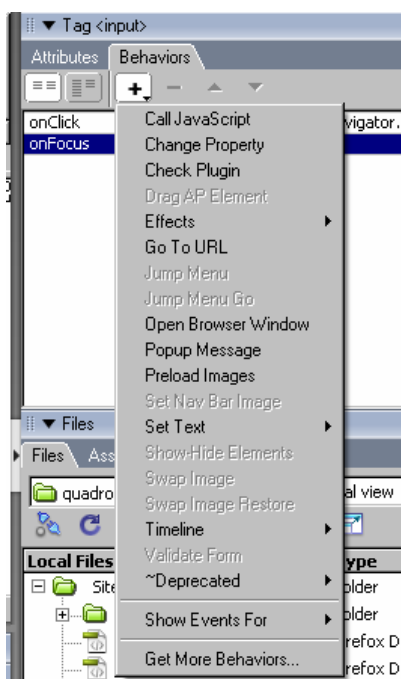


Рис. 4. Категории скриптов вкладки Behaviours.

Call JavaScript (Запрос JavaScript) - позволяет использовать панель Behaviors для указания определенных функций или строк кода JavaScript, которые должны выполняться при определенном событии. (Можно написать код JavaScript самостоятельно, а можно использовать код из библиотек JavaScript.)

Чтобы использовать действие Call JavaScript (Запрос JavaScript):

1. Выделите объект и откройте панель Behaviors (Интерактивные эффекты).
2. Щелкните на кнопке + (плюс) и выберите команду Call JavaScript (Запрос JavaScript) из раскрывающегося списка Actions (Действия). Появится диалоговое окно Call JavaScript (Запрос JavaScript).
3. Введите код JavaScript, который будет выполняться, или имя функции.
Например, чтобы создать кнопку Back (Назад), введите
`if (history.length > 0){history.back()}`
Если код такой функции уже есть, укажите только имя функции (например, `goBack()`).
4. Нажмите кнопку ОК.
5. Проверьте, что по умолчанию указано нужное вам событие.

В противном случае выберите иное событие из раскрывающегося списка. Если нужного события в списке нет, укажите другой браузер в подменю Show Events For (Показать события для).

Change Property (Изменить свойство) - изменяет значение одного из свойств объекта, например, цвет фона слоя. Свойства, которые можно изменять, определяются браузером.

Check Browser (Проверить браузер) – используется для того, чтобы посетители видели страницы, соответствующие типам и версиям своих браузеров. Например, посетитель с Netscape Navigator видит одну страницу, а посетитель с Internet Explorer - другую.

Check Plugin (Проверить дополнение) – используется для того, чтобы посетители видели страницы, соответствующие установленным у них дополнениям (plug-in). Например, посетитель с установленным приложением Shockwave видят страницу с анимацией Shockwave, а посетители не имеющие этого дополнения переходят на другую страницу.

Control Shockwave or Flash (Управление Shockwave или Flash) - используется для того, чтобы воспроизводить, останавливать, перематывать фильм Shockwave или Flash, а также переходить в них к определенному кадру.

Drag Layer (Перетащить слой) - позволяет посетителю перетаскивать слой.

Используйте его для создания головоломок, ползунков управления и других подвижных элементов интерфейса. Вы способны определить направление, в котором пользователь может перетаскивать слой (по горизонтали, вертикали или в любом другом направлении); конечный пункт, к которому слой перетаскивается; назначить привязку к некоторой точке, когда слой находится на определенном расстоянии от нее; указать дальнейшие действия, когда слой приблизился к цели и т. д. Поскольку действие Drag Layer (Перетащить слой) должно вызываться прежде, чем пользователь начнет перетаскивание слоя, удостоверьтесь, что событие, запускающее действие, происходит до попытки перетащить слой. Поэтому действие Drag Layer (Перетащить слой) лучше применять к объекту <BODY> (с событием onLoad), хотя можно также применить его к ссылке, которая заполняет весь слой (например, изображению, которое одновременно является ссылкой), используя событие onMouseOver.

Go To URL (Перейти по URL) – открывает новую страницу в текущем окне или в указанном фрейме. Это действие особенно полезно для изменения содержимого двух и более фреймов одним щелчком. Это действие может быть также вызвано из указанного кадра временной диаграммы.

Jump Menu (Меню перехода) - Чтобы создать меню перехода, выберите команду Insert (Вставка) > Form Objects (Объекты формы) > Jump Menu (Меню перехода). При этом Dreamweaver создает объект меню и назначает ему интерактивный эффект Jump Menu (Меню перехода). Обычно не требуется назначать указанное действие вручную.

Jump Menu Go (Меню перехода с кнопкой Go) - Действие Jump Menu Go (Меню перехода с кнопкой Go) позволяет связать с меню перехода с кнопкой Go (Перейти). (Чтобы использовать это действие, меню перехода должно существовать в документе.) Нажатие кнопки Go (Перейти) открывает ссылку, выбранную в меню перехода. Меню перехода обычно не нуждается в кнопке Go (Перейти), а выбор какого-либо пункта меню перехода загружает URL автоматически, не требуя дополнительных действий пользователя. Но если пользователь выбирает уже выбранный пункт, то перехода не произойдет. Кнопка Go (Перейти) полезна в тех случаях, когда для ваших посетителей очевидно, что при новом выборе перехода не происходит. В частности, добавьте кнопку

Go (Перейти) к меню перехода, когда меню появляется в рамке, а пункты меню связаны со страницами в других рамках.

Open Browser Window (Открыть в окне браузера) – используется для открытия Web-страницу в новом окне. Вы можете определить свойства нового окна, включая его размер, атрибуты (например, возможность изменения размеров окна, наличие строки меню и т. д.) и имя. Если вы не указываете атрибуты для окна, то оно открывается с такими же размерами и атрибутами, как и у окна, из которого вы его открываете. Определение любых атрибутов окна автоматически отключает все другие атрибуты, которые не указаны явно. Например, если вы не установили для окна никакие параметры, оно может иметь размер 640 x 480 пикселей и навигационную строку, панели адреса, строки состояния и строку меню. Если вы явно указываете ширину 640 пикселей, а высоту 480 пикселей и не устанавливаете никакие другие атрибуты, окно открывается с размерами 640 x 480 пикселей и не имеет навигационной строки, панели адреса, строки состояния, строки меню, маркеров изменения размера и полос прокрутки.

Play Sound (Воспроизвести звук) - используется для фонового оформления, например, когда указатель мыши наводится на ссылку или загружается страница.

Popup Message (Сообщение) - выводит предупреждающее сообщение JavaScript с заданным вами текстом. Поскольку предупреждающее диалоговое окно с сообщением JavaScript имеет только кнопку ОК, то это действие используется лишь с целью отображения информации для посетителя.

В текст можно добавлять любой допустимый запрос функции JavaScript, свойство, глобальную переменную или иное выражение. Чтобы вставить выражение JavaScript, поместите его в фигурные скобки ({}). Чтобы показать фигурную скобку, предварите ее символом обратный слэш (\ {}).

Например, можно ввести URL для этой страницы - {window.location}, и сегодня - {new Date()}

Preload Images (Предварительно загрузить изображения) – загружает в буфер браузера изображения, которые появляются на странице не сразу (например, изображения, которыми заменяются основные изображения, интерактивные эффекты или JavaScript). Таким образом предотвращаются задержки, когда необходимо использовать подобные дополнительные изображения.

Set Nav Bar Image (Настроить изображение в навигационной панели) – используется для того, чтобы превратить изображение в навигационную панель или изменить вид и действия изображений в этой панели.

Используйте вкладку Basic (Основные) в диалоговом окне Set Nav Bar Image (Настройка изображения в навигационной панели), чтобы создать или обновить изображение или набор изображений в навигационной панели, а также чтобы указать адрес URL, связанный с каждой из кнопок.

В том же диалоговом окне используйте вкладку Advanced (Дополнительно) с целью изменения состояния других изображений в документе, зависящих от состояния текущей кнопки. По умолчанию щелчок на одной кнопке в навигационной панели автоматически переключает все прочие элементы в их начальные состояния; с помощью вкладки Advanced (Дополнительно) можно установить иные состояния.

Set Text of Frame (Установить текст фрейма) - позволяет динамически изменять текст или форматирование фрейма. Содержимое может включать в себя любой допустимый код HTML, и потому это действие удобно для динамического представления информации. Хотя действие Set Text of Frame (Установить текст фрейма) заменяет форматирование фрейма, можно сохранить цвет фона и атрибуты цвета текста.

Можно вводить любые допустимые запросы JavaScript, свойства, глобальные переменные или другие текстовые выражения. Чтобы в виде текста был виден собственно

код JavaScript, поместите его в фигурные скобки ({ }). Чтобы показать фигурную скобку, предварите ее наклонной чертой влево (\{). Например,

URL для этой страницы - {window.location}, и сегодня - {new Date()}

Для создания набора фреймов выберите команду Modify (Изменить) > Frameset (Набор фреймов) > Split Frame Left (Разбить влево), Split Frame Right (Разбить вправо), Split Frame Up (Разбить вверх) или Split Frame Down (Разбить вниз).

Set Text of Layer (Установить текст слоя) - заменяет содержимое и форматирование существующего слоя на странице тем содержимым, которое вы укажете. Содержимое может иметь любой допустимый код HTML. Действие Set Text of Layer (Установить текст слоя) заменяет наполнение и форматирование слоя, но сохраняет его атрибуты, в том числе цвет.

Например:

URL для этой страницы - {window.location}, и сегодня - {new Date ()}

The URL for this page is {window.location}, and today is {new Date()}.

Set Text of Status Bar (Создать текст в строке состояния) - показывает сообщение в строке состояния в нижней левой части окна браузера. Это действие можно использовать, например, для того, чтобы описать в строке состояния назначение ссылки вместо отображения связанного с ней URL. Пример такого сообщения вы можете увидеть, если проведете мышью по любой из навигационных кнопок в справочной системе Dreamweaver.

Set Text of Text Field (Установить текст текстового поля) - заменяет содержимое текстового поля формы той строкой, которую вы укажете. Вы можете вставлять любые допустимые запросы функции JavaScript, свойства, общие переменные или другие текстовые выражения.

Show-Hide Layers (Показать/Скрыть слои) - скрывает или восстанавливает видимость одного или нескольких слоев. Оно полезно для отображения информации во время работы Web-посетителя со страницей. Например, когда пользователь перемещает указатель мыши поверх изображения растения, можно показать слой, сообщающий подробности о сельскохозяйственном сезоне, регионе, необходимом количестве солнца и т. д. Действие Show-Hide Layers (Показать/Скрыть слои) также полезно для создания предварительно загружаемого слоя, т. е. большого слоя, который затеняет содержимое страницы сначала и исчезает, когда все компоненты страницы загрузились.

Swap Image (Сменить изображение) - заменяет одно изображение другим, модифицируя атрибут src тега . Это действие используется для создания чувствительных изображений и других подобных графических эффектов. Необходимо чтобы начальное и заменяющее изображения имели одни и те же размеры. В противном случае заменяющее изображение будет автоматически сжато до размеров первоначального изображения.

Swap Image Restore (Восстановить начальное изображение) - восстанавливает предыдущий набор сменных изображений. Данное действие автоматически добавляется всякий раз, когда вы применяете действие Swap Image (Сменить изображение) к объекту. Если, применив действие Swap Image (Сменить изображение), вы установили флажок Restore Images onMouseOut (Восстановить изображения для onMouseOut), то нет необходимости применять действие Swap Image Restore (Восстановить начальное изображение) вручную.

Go to Timeline Frame (Перейти к кадру временной диаграммы) - перемещает головку воспроизведения в указанный кадр.

Это действие можно использовать в канале Behaviour (Интерактивный эффект) панели Timelines (Временные диаграммы), чтобы создать циклы из некоторого количества

кадров временной диаграммы, создать ссылку назад или кнопку перемотки, а также позволить пользователю перейти к другой части мультимедиа.

Play Timeline и **Stop Timeline (Воспроизвести / Остановить временную диаграмму)** - позволяют посетителям запускать и останавливать временную диаграмму, щелкая на объектах, которым эти действия назначены. Действие Play Timeline (Воспроизвести временную диаграмму) автоматически назначается тегу <BODY> вместе с событием onLoad.

Validate Form (Проверить заполнение формы) - анализирует содержимое указанных текстовых полей, чтобы гарантировать, что пользователь ввел правильный тип данных. Используйте это действие вместе с событием onBlur для проверки заполняемого пользователем поля или вместе с событием onSubmit к форме, чтобы проверить несколько полей сразу, когда пользователь нажимает кнопку Submit (Отправить). Применение данного действия к форме запрещает передачу данных формы на сервер, если хотя бы одно из указанных полей содержит неверные данные.

На рис. 5 показан пример применения данного инструмента к Web-странице, на которой выполняется проверка имени браузера. Код данной страницы имеет вид:

```
<HTML>
<HEAD>
<title>Test of Browser name</title>
</HEAD>
<BODY>
  <h1 align=center>Проверка имени типа браузера;</h1>
  <hr>
  Для того, чтобы получить имя вашей программы просмотра
  выберите кнопку "Browser"<br>
  <center>
<form name=kuku>
  <input name=browser type=button
  onClick="window.alert(window.navigator.appName)" value=Browser>
</form>
</BODY>
</HTML>
```

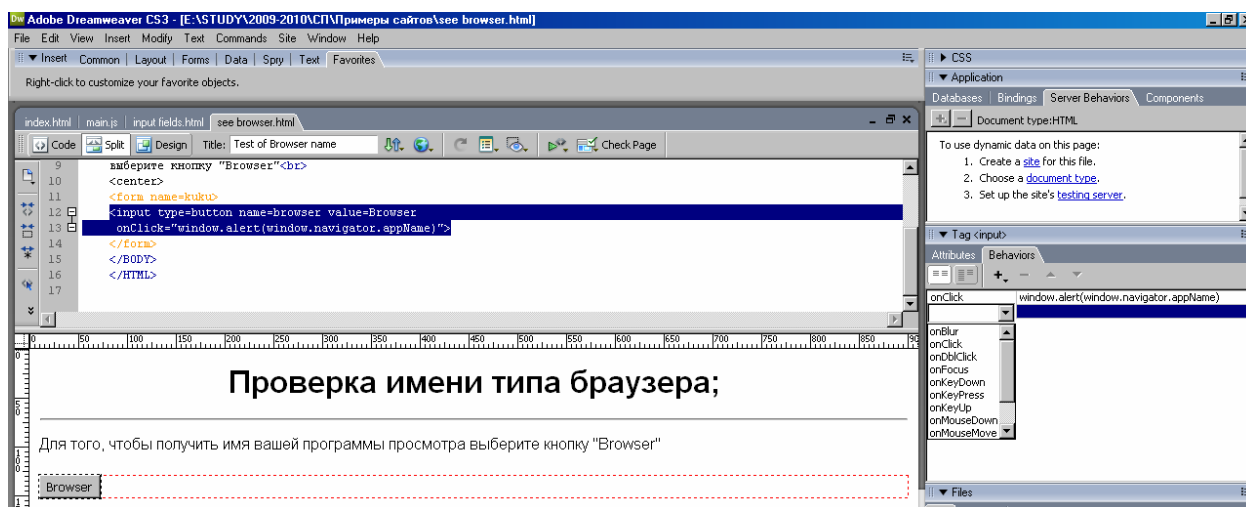


Рис. 5. Пример применения инструмента Behaviours

Программное обеспечение и оборудование для выполнения работы

Работа выполняется на современном персональном компьютере. Необходимое программное обеспечение:

- Навигатор Web-страниц (браузер)
- Система Dreamweaver
- Web-сайт в виде набора HTML-страниц и файлов JavaScript.

Порядок выполнения работы

1. Ознакомиться с теоретической справкой, спецификацией языка JavaScript.
2. Ознакомиться с примером Web-сайта. Определить, где применяется JavaScript в коде Web-страниц.
3. Запустить систему Dreamweaver.
4. Открыть в системе DreamWeaver страницу сайта, содержащую код на JavaScript.
5. Выбрать объект на странице. Используя вкладку Behaviours окна обозревателя тэгов (Tag Inspector), для выбранного объекта определить пару «событие – скрипт».
6. Выполнить действия в п. 5 для других объектов Web-страниц. Сделать выводы о способах применения JavaScript, использованных в исследуемом сайте.
7. По согласованию с преподавателем изменить один или несколько объектов, использующих JavaScript. Оценить полученные результаты.
8. Разработать и включить в сайт новые функции, применив язык JavaScript.
9. Ответить на контрольные вопросы.

Контрольные вопросы

1. В чем разница между языками Java и JavaScript ?
2. К какому типу языков относится JavaScript - это алгоритмический язык, язык скриптов или язык разметки?
3. Какие существуют способы оформления программ на JavaScript?
4. Каким образом единственный файл *.js определяет представление текстов в нескольких Web – страницах?
5. Как, не имея системы DreamWeaver, составить и применить файл JavaScript?
6. Допустимо ли применение нескольких файлов *.js для одного сайта?
7. В чем преимущества и недостатки JavaScript?

Литература и электронные ресурсы

1. Сергеев, А.П. HTML и XML. Web-дизайн и программирование для среды Internet / А.П.Сергеев .— М.и др. : Диалектика, 2004 .— 880с.
2. Холл, М. Программирование для Web / М.Холл,Л.Браун .— М. : Вильямс, 2002 .— 1264с.
3. DreamWeaver шаг за шагом. <http://www.citforum.ru/internet/dreamw/index.shtml>
4. Русская онлайн - справка по Dreamweaver
<http://www.mpgu.ru/macromedia/dreamweaver4/>
5. Обновляемые материалы по Java/JavaScript
<http://www.citforum.ru/internet/javascript/>
6. Библиотека книг по Java/JavaScript <http://docs.luksian.com/programming/java/>
7. Спецификация языка <http://synchro.net/docs/js/ref/contents.html>