**Практическая работа 11**

**Окна и динамическое управление документами в JavaScript**

**Цель:** Ознакомиться с принципами работы с окнами и динамическим управлением документами, научить использовать свойства окон при создании веб-страниц и создавать документы, изменяющие свойства других документов

**Задания:**

1. Ознакомиться с теоретическими аспектами темы.
2. Создайте веб-страницу, в которой в новое окно с помощью метода open() записывается другая страница.
3. Создайте веб-страницу, в которой производится создание нового окна фиксированного размера.
4. Создать документ с использованием методов объекта window.
5. Создать документ с использованием команд генерации нового документа.

**Необходимые приборы:** ПК, текстовый редактор Блокнот, браузер

**Методические рекомендации к выполнению лабораторной работы:**

**Методические рекомендации к выполнению задания 1**

Открытие новых окон в браузере - грандиозная возможность языка JavaScript. Можно либо загружать в новое окно новые документы, либо (динамически) создавать новые материалы. Также можно управлять самим процессом создания окна. Например, можно указать, должно ли новое окно иметь строку статуса, панель инструментов или меню. Кроме того, можно задать размер окна. Список свойств окна, которыми можно управлять:

|  |  |
| --- | --- |
| Directories | Yes / no |
| Height | количество пикселов |
| Location | Yes / no |
| Menubar | Yes / no |
| Resizable | Yes / no |
| Scrollbars | Yes / no |
| Status | Yes / no |
| Toolbar | Yes / no |
| Width | количество пикселов |

Как видите, открывая окна, мы должны использовать три аргумента:

myWin= open("cell.htm", "displayWindow", "width=400, height=300, status=no, toolbar=no, menubar=no");

Второй аргумент - это имя окна. Если оно известно, то можно загрузить туда новую страницу с помощью записи

<a href="cell.html" target="displayWindow">

При этом необходимо указать имя соответствующего окна (если же такого окна не существует, то с этим именем будет создано новое).

*myWin* - это не имя окна, но только с помощью этой переменной можно получить доступ к окну. И поскольку это обычная переменная, то область ее действия - лишь тот скрипт, в котором она определена. Имя окна (в данном случае это *displayWindow*) - уникальный идентификатор, которым можно пользоваться с любого из окон браузера.

Чтобы закрыть окно понадобится метод close(). Начало формы

Конец формы

Более того, можно таким же образом создавать и другие документы Web, такие как VRML-сцены и т.д. Для удобства можно размещать эти документы в отдельном окне или фрейме.

**Методические рекомендации к выполнению задания 2**

1. Создайте новый документ cell.html произвольного содержания.
2. Создайте новый документ html, скрипт показан ниже.

<html>

<head>

<script language="JavaScript">

<!-- hide

functionopenWin() {

myWin= open("cell.htm");

}

// -->

</script>

</head>

<body>

<form>

<input type="button" value="Открытьновоеокно" onClick= "openWin()" >

</form>

</body>

</html>

Начало формы

Конец формы

В представленном примере в новое окно с помощью метода open() записывается страница *cell.htm*.

**Методические рекомендации к выполнению задания 3**

1. Создайте новый документ html

<html>

<head>

<script language="JavaScript">

<!-- hide

function openWin2() {

myWin= open("cell.htm", "displayWindow",

"width=400,height=300,status=no,toolbar=no,menubar=no,scrollbar=no");}

// -->

</script>

</head>

<body>

<form>

<input type="button" value="Открытьновоеокно" onClick="openWin2()">

</form>

</body>

</html>

В этом скрипте открывается новое окно размером 400x300 пикселей. Оно не имеет ни строки статуса, ни панели инструментов, ни меню. Начало формы

Конец формы

Как видите, свойства окна формулируются в строке

*"width=400,height=300,status=no,toolbar=no,menubar=no"*.

Обратите внимание также и на то, что не следует помещать в этой строке символы пробела.

**Методические рекомендации к выполнению задания 4**

1. Создайтедокумент:

<html>

<script language="JavaScript">

<!-- hide

function closeIt() {

close();}

// -->

</script>

<center>

<form>

<input type=button value="Close it" onClick="closeIt()">

</form>

</center>

</html>

Если теперь в новом окне нажать на кнопку, то окно будет закрыто.   
open() и close() - это методы объекта window. Следует писать не просто *open()* и *close()*, а *window.open()* и *window.close()*. В нашем случае объект *window* можно опустить - нет необходимости писать префикс window, если нужно вызвать один из методов этого объекта (и такое возможно только для этого объекта).

**Методические рекомендации к выполнению задания 5**

1. Создайте HTML-документ:

<html>

<head>

<script language="JavaScript">

<!-- hide

function openWin3() {

myWin= open("", "displayWindow",

"width=500,height=400,status=yes,toolbar=yes,menubar=yes");

// открыть объект document для последующей печати

myWin.document.open();

// генерировать новый документ

myWin.document.write("<html><head><title>On-the-fly");

myWin.document.write("</title></head><body>");

myWin.document.write("<center><font size=+3>");

myWin.document.write("This HTML-document has been created ");

myWin.document.write("with the help of JavaScript!");

myWin.document.write("</font></center>");

myWin.document.write("</body></html>");

// закрыть документ - (но не окно!)

myWin.document.close(); }

// -->

</script>

</head><body><form>

<input type=button value="On-the-fly" onClick="openWin3()">

</form>

</body>

</html>

Начало формы

Конец формы

Давайте рассмотрим функцию openWin3(). Очевидно, мы сначала открываем новое окно браузера. Поскольку первый аргумент функции open() - пустая строка (*""*), то это значит, что мы не желаем в данном случае указывать конкретный адрес URL. Браузер должен создать дополнительно новый документ. В скрипте мы определяем переменную *myWin*. И с ее помощью можем получать доступ к новому окну. Обратите внимание, что в данном случае нельзя воспользоваться для этой цели именем окна (*displayWindow*).   
После того, как открыли окно, наступает очередь открыть для записи объект document. Делается это с помощью команды:

// открыть объект document для последующей печати

myWin.document.open();

Здесь мы обращаемся к open() - методу объекта document. Но это совсем не то же самое, что метод open() объекта window. Эта команда не открывает нового окна - она лишь готовит document к предстоящей печати. Кроме того, мы должны поставить перед *document.open()* приставку *myWin*, чтобы получить возможность писать в новом окне. В последующих строках скрипта с помощью вызова *document.write()* формируется текст нового документа:

// генерироватьновыйдокумент

myWin.document.write("<html><head><title>On-the-fly");

myWin.document.write("</title></head><body>");

myWin.document.write("<center><font size=+3>");

myWin.document.write("This HTML-document has been created ");

myWin.document.write("with the help of JavaScript!");

myWin.document.write("</font></center>");

myWin.document.write("</body></html>");

Как видно, здесь мы записываем в документ обычные тэги языка HTML. То есть мы фактически генерируем разметку HTML. При этом можно использовать абсолютно любые тэги HTML. По завершении этого нужно вновь закрыть документ. Это делается следующей командой:

// закрыть документ - (но не окно!)

myWin.document.close();

**Вопросы для самоконтроля:**

1. Перечислите свойства окна, которыми можно управлять.
2. Какие три аргумента нужно использовать при открытии окна?
3. При помощи какой переменной осуществляется доступ к окну?
4. Какой метод применяется для закрытия окна?
5. При помощи какой команды осуществляется открытие объекта document для последующей печати?
6. При помощи какой команды осуществляется закрытие документа?
7. При помощи какой команды осуществляется закрытие окна?