**Практическая работа 17**

**Использование слоев**

**Цель:** Изучить основные понятия технологии слоев

**Задание:**

1. Ознакомиться с теоретическими сведениями по теме.
2. Создайте новый документ, содержащий два слоя. В первом из них поместите изображение, а во втором - текст. Все, что нужно сделать - показать этот текст поверх данного изображения.
3. Создайте документ с использованием команд показа и скрытия слоев.
4. Создайте документ с использованием функции перемещения слоев.

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

**Методические рекомендации к выполнению лабораторной работы:**

**Методические рекомендации к выполнению задания 1**

Использование **слоев** позволяет выполнять точное позиционирование таких объектов web-страницы, как изображения. Кроме того, можно перемещать объекты по вашей HTML-странице. Вы можете также делать объекты невидимыми. Управлять слоями можно легко с помощью языка JavaScript.

**Создание слоев**

Чтобы создать слой, мы должны использовать либо тэг <layer> либо <ilayer>. Вы можете воспользоваться следующими параметрами:

|  |  |
| --- | --- |
| name="*layerName*" | Название слоя |
| left=*xPosition* | Абсцисса левого верхнего угла |
| top=*yPosition* | Ордината левого верхнего угла |
| z-index=*layerIndex* | Номер индекса для слоя |
| width=*layerWidth* | Ширина слоя в пикселах |
| clip="*x1\_offset, y1\_offset, x2\_offset, y2\_offset"* | Задает видимую область слоя |
| above="*layerName*" | Определяет, какой слой окажется под нашим |
| below="*layerName*" | Определяется, какой слой окажется над нашим |
| Visibility=show|hide|inherit | Видимость этого слоя |
| bgcolor="*rgbColor*" | Цвет фона - либо название стандартного цвета, либо rgb-запись |
| background="*imageURL*" | Фоновая картинка |
|  |  |

Тэг <layer> используется для тех слоев, которые Вы можете точно позиционировать. Если же Вы не указываете положение слоя (с помощью параметров *left* и *top*), то по умолчанию он помещается в верхний левый угол окна.

Тэг <ilayer> создает слой, положение которого определяется при формировании документа.

Рассмотрим теперь, как можно получить доступ к слоям через JavaScript. Начнем же мы с примера, где пользователь получает возможность, нажимая кнопку, прятать или показать некий слой.

Для начала мы должны знать, каким образом слои представлены в JavaScript. Как обычно, для этого имеются несколько способов. Самое лучшее - дать каждому слою свое имя. Так, если мы задаем слой

<layer ... name=myLayer>

...

</layer>

то в дальнейшем можем получить доступ к нему с помощью конструкции *document.layers["myLayer"]*. Согласно документации, предоставляемой фирмой Netscape, мы можем также использовать запись *document.myLayer* - однако в моем браузере это приводит к сбою. Конечно, это всего лишь проблема предварительной версии и в заключительном варианте будет успешно решена (сейчас я пользуюсь Netscape Navigator 4.0 PR3 на WinNT). Однако, по-видимому, нет никаких проблем с конструкцией *document.layers["myLayer"]* - поэтому мы и будем пользоваться именно такой альтернативой из всех возможных.

Доступ к этим слоям можно также получить через целочисленный индекс. Так, чтобы получить доступ к самому нижнему слою, Вы можете написать *document.layers[0]*. Обратите внимание, что индекс - это не то же самое, что параметр z-index. Если, например, Вы имеете два слоя, называемые *layer1* и *layer2* с номерами z-index 17 и 100, то Вы можете получить доступ к этим слоям через *document.layers[0]* и *document.layers[1]*, а **не** через *document.layers[17]* и *document.layers[100]*.

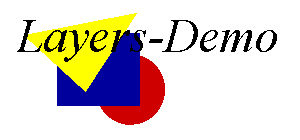
**Перемещение слоев**

Свойства *left* и *top* определяют положение данного слоя. Вы можете менять его, записывая в эти атрибуты новые значения. Например, в следующей строке задается горизонтальное положение слоя в 200 пикселов:

document.layers["myLayer2"].left= 200;

**Методические рекомендации к выполнению задания 2**

Создайте новый документ, содержащий два слоя. В первом из них поместите изображение, а во втором - текст. Все, что нужно сделать - показать этот текст поверх данного изображения.

  
Текст поверх изображения

Исходный код:

<html>

<layer name=pic z-index=0 left=200 top=100>

<img src="img.gif" width=160 height=120>

</layer>

<layer name=txt z-index=1 left=200 top=100>

<font size=+4><i> Layers-Demo </i></font>

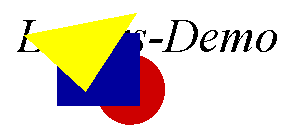
</layer>

</html>

Как видим, с помощью тэга <layer> мы формируем два слоя. Оба слоя позиционируются как 200/100 (через параметры *left* и *top*). Все, что находится между тэгами <layer> и </layer> (или тэгами <ilayer> и </ilayer>) принадлежит описываемому слою.

Кроме того, мы используем параметр *z-index*, определяя тем самым порядок появления указанных слоев - то есть, в нашем случае, Вы тем самым сообщаете браузеру, что текст будет написан поверх изображения. В общем случае, именно слой с самым высоким номером z-index будет показан поверх всех остальных. Вы не ограничены в выборе z-index лишь значениями 0 и 1 - можно выбирать вообще любое положительное число.

Так, если в первом тэге <layer> Вы напишете *z-index=100*, то текст окажется под изображением - его слой номер Z-индекса (z-index=1). Вы сможете увидеть текст сквозь изображение, поскольку я использовал в нем прозрачный фон (формат gif89a).

  
Текст под изображением

**Методические рекомендации к выполнению задания 3**

Создайте документ, содержащий скриптНачало формы

Конец формы

:

<html>

<head>

<script language="JavaScript">

<!-- hide

function showHide() {

if (document.layers["myLayer"].visibility == "show")

document.layers["myLayer"].visibility= "hide"

else document.layers["myLayer"].visibility= "show";}

// -->

</script>

</head>

<body>

<ilayer name=myLayer visibility=show>

<font size=+1 color="#0000ff"><i>This text is inside a layer</i></font>

</ilayer>

<form>

<input type="button" value="Show/Hide layer" onClick="showHide()">

</form></body></html>

Даннаякнопкавызываетфункцию*showHide()*. Можно видеть, что в этих функциях реализуется доступ к такому свойству объекта layer (*myLayer*), как **видимость**. Присвоивая параметру *document.layers["myLayer"].visibility* значения *"show"* или *"hide"*, Вы можете показать или скрыть наш слой. Заметим, что *"show"* и *"hide"* - это строки, а не зарезервированные ключевые слова, то есть Вы **не можете** написать *document.layers["myLayer"].visibility= show*.

Вместо тэга <layer> я также пользовался тэгом <ilayer>, поскольку хотел поместить этот слой в "информационный поток" документа.

**Методические рекомендации к выполнению задания 4**

Создайте документ с использованием функции перемещения слоев. Скрипт выглядит следующим образом:

<html>

<head>

<script language="JavaScript">

<!-- hide

function move() {

if (pos < 0) direction= true;

if (pos > 200) direction= false;

if (direction) pos++

else pos--;

document.layers["myLayer2"].left= pos;

}

// -->

</script>

</head>

<body onLoad="setInterval('move()', 20)">

<ilayer name=myLayer2 left=0>

<font size=+1 color="#0000ff"><i>This text is inside a layer</i></font>

</ilayer>

</body>

</html>

Мы создаем слой с именем *myLayer2*. Можно видеть, что в тэге <body>мы пользуемся процедурой *onLoad*. Нам необходимо начать прокручивание слоя, как только страница будет загружена. В процедуре обработки события *onLoad* мы пользуемся функцией *setInterval()*. Это один из новых методов версии 1.2 языка JavaScript (то есть версии JavaScript, реализованной в Netscape Navigator 4.0). Им можно пользоваться, чтобы вызывать некую функцию мвновь и вновь через определенные интервалы времени. В прошлом для этого мы пользовались функцией *setTimeout()*. Функция *setInterval()* работает почти так же, однако Вам нужно вызвать ее всего лишь один раз.

С помощью *setInterval()* мы вызываем функцию *move()* каждые 20 миллисекунд. А функция *move()*, в свою очередь, всякий раз смещает слой на новую позицию. И поскольку мы вызываем эту функцию вновь и вновь, то мы получаем быстрый скроллинг нашего текста. Все, что мы нужно сделать в функции *move()* - это вычислить новую координату для слоя и записать ее: *document.layers["myLayer2"].left= pos*.

Если Вы посмотрите исходный код этой части в онлайновом описании, то увидите, что в действительности мой код выглядит несколько иначе - я добавил некий фрагмент кода с тем, чтобы люди, работающие со старыми версиями JavaScript-браузеров, не получали из-за этого никаких сообщений об ошибках. Как этого можно достичь? Следующий фрагмент кода будет выполняться только на тех браузерах, которые воспринимают язык JavaScript 1.2:

<script language="JavaScript1.2">

<!-- hide

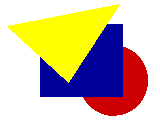
document.write("Вы используете браузер, совместимый с JavaScript 1.2.");

// -->

</script>

Та же самая проблема возникает, когда мы работаем с объектом Image. Мы можем аналогичным способом переписать кодекс. Установка переменной *browserOK* решает эту проблему.

Следующий пример демонстрирует, как может осуществляться перекрывание слоев:

*This text is inside a layer*

**Вопросы для самоконтроля:**

1. Какие возможности предоставляет использование слоев?
2. Каким образом производится создание слоев?
3. Для чего используется тег <layer>?
4. Какими командами нужно воспользоваться для того, чтобы получить доступ к слою через целочисленный индекс?
5. При помощи каких команд производится определение положения данного слоя?
6. В каких случаях используется параметр z-index?
7. В чем отличия тэгов <layer> и <ilayer>?
8. Каким образом производится перемещение слоев?
9. Каким образом производится перекрывание слоев?
10. Для чего предназначено событие onLoad?