**ОФОРМЛЕНИЕ СТРАНИЦ С ПОМОЩЬЮ ФРЕЙМОВ**

*Фрейм*, или *кадр*, *–* способ организации структуры сайта, при кото- ром web-страница дробится на ряд отдельных составляющих и собирается в главном окне браузера из нескольких независимых или вложенных пане- лей, каждая из которых выводит свой файл.

Внешне фрейм выглядит как прямоугольная область, которая может иметь границы или полосы прокрутки. В область фрейма может выводиться любой HTML-документ, содержащий текст, изображения, таблицу и т. д.

Применение фреймов позволяет более рационально использовать пространство окна браузера. Фреймы позволяют создавать фиксированные области окна, которые не меняются при прокрутке страниц. В эту область обычно помещают название, логотип, панель поиска.

Фреймы часто используются для организации навигации по сайту. Для этого в один из фреймов загружают список гиперссылок, при щелчке на одной из них ее содержимое отображается в другом фрейме.

Возможности загрузки нескольких документов в одном окне браузе- ра сделали фреймы достаточно популярными среди разработчиков web-страниц. Они могут улучшить внешний вид страниц и их функцио- нальность.

**СОЗДАНИЕ ПРОСТЫХ ФРЕЙМОВ**

При создании фреймов весь процесс можно разделить на два этапа. На первом этапе создается страница содержания (content page) – это обыч- ная страница HTML, с которыми вы работали при изучении всего преды- дущего материала. На втором этапе – страница настройки фреймов, или страница наборов фреймов (frameset page), которая предназначена разде- лять окно браузера на набор фреймов, определять размер каждого фрейма, а также определяет, какие именно страницы содержания отображаются в этих фреймах. Таким образом, страница настройки фреймов – это своеоб- разная пустая оболочка, которая делит экран браузера на два или несколь- ко фреймов, так что вам остается наполнить эти ячейки отдельными стра- ницами содержания.

В случае применения фреймов HTML-документ несколько отличается от обычного. При использовании фреймов раздел BODY меняется на раздел FRAMESET, который использует парный тег *<*FRAMESET*>* </FRAMESET*>.*

Структурный HTML-документ (тот, который определяет структуру фреймов) не может содержать никаких тегов форматирования или любых HTML-документов. Тег *<*FRAMESET*>* содержит атрибуты, указанные в табл. 4.1.

*Таблица 4.1*

**Атрибуты тега <FRAMESET>**

|  |  |  |
| --- | --- | --- |
| Атрибут | Назначение | Принимаемые значения, форма записи |
| ROWS | Определяет количество горизонтальных отрезков во фреймовом наборе | ROWS="значение", задается в пик- селах, процентах, относительных длинах  ROWS="30%, 70%" |
| COL | Определяет количество вертикальных отрезков во фреймовом наборе | COLS="значение", задается в пиксе- лах, процентах, относительных дли- нах  COLS="\*, 2\*", \* является одной ча- стью целого числа |
| FRAMEBORDER | Определяет, надо ли ото- бражать на экране поль- зователя границы фрей- мов | Может принимать значение 1 – если разделители нужны (по умолчанию), 0 – если границы невидимые FRAMEBORDER="1" |
| FRAMESPACING | Толщина разделителя | FRAMESPACING="значение" зада- ется в пикселах FRAMESPACING="2" |
| BORDER | Устанавливает ширину всех рамок | BORDER="значение" в пикселах  BORDER="5" |
| BORDERСOLOR | Устанавливает цвет ра- мок | BORDERСOLOR="цвет" BORDERCOLOR="red" |

Тег *<*FRAMESET*>* может содержать два основных атрибута ROWS и COLS*.* Эти атрибуты определяют количество горизонтальных и верти- кальных фреймов. Их можно задавать различными способами: в пикселях, процентах, относительных длинах. Причем число значений соответствует количеству фреймов. Если атрибут ROWS в теге *<*FRAMESET*>* не указан, то каждый столбец будет занимать всю длину страницы; если не указан COLS, каждая строка растянется на всю видимую ширину экрана.

Параметры фреймов задаются слева направо для горизонтальных элементов и сверху вниз – для вертикальных.

<FRAMESET ROWS="30%, 70%">

разбивает весь экран браузера на два горизонтальных фрейма, один из ко- торых (верхний) занимает 30 % рабочей области браузера, второй (ниж- ний) – 70 % (общая сумма всегда должна составлять 100 %).

<FRAMESET COLS="50%, 50%">

разделит страницу вертикально на две равные части.

Формат записи в относительных единицах выглядит следующим об- разом:

<FRAMESET COLS="\*, 2\*, 3\*">

Символ «\*» является одной частью целого числа и осуществляет пропорциональное разделение окна браузера на указанное количество фреймов (в нашем случае это три вертикальных фрейма). Чтобы подсчи- тать, какую часть окна занимает каждый фрейм, нужно сложить все циф- ры, стоящие перед звездочкой и поставить получившуюся сумму в качест- ве знаменателя от целого числа. Таким образом, 1 + 2 + 3 = 6, т. е. первый фрейм занимает 1/6 окна, второй – 2/6 (1/3) окна и третий – 3/6 (1/2) окна браузера.

Язык разметки HTML позволяет для атрибутов ROWS и COLS зада- вать смешанные значения.

<FRAMESET ROWS ="250, 10%, \*">

задает горизонтальное разделение страницы на три части: 1-я – с фиксиро- ванной высотой 250 пикселов, 2-я – высотой 10 % от оставшегося про- странства, 3-я – займет все пространство, которое останется свободным после создания первых двух окон.

Обязательного порядка записи смешанных значений не существует, однако рекомендуется в первую очередь указывать фиксированные значе- ния (пикселы), далее, желательно, должны идти проценты, напоследок – относительные единицы.

В теге *<*FRAMESET*>* можно одновременно указывать атрибуты ROWS и COLS*.* При этом каждый горизонтальный фрейм браузер разобьет на заданное вами число вертикальных фреймов.

<FRAMESET ROWS ="40%, 60%" COLS="50%, 50%">

разбивает окно браузера на фреймы, верхние и нижние из которых имеют одинаковую высоту.

Пользователь при желании может сам изменить размеры фреймов. Делается это простым перетаскиванием мышью границы между фреймами, если в теге *<*FRAMESET*>* отсутствует специальный атрибут для фиксации размеров фрейма.

Тег *<*FRAME*>* описывает свойства отдельного тега, входящего в со- став фреймовой структуры. Количеству фреймов должно соответствовать такое же количество тегов *<*FRAME*>*, которые включаются внутрь тега фреймовой структуры *<*FRAMESET*>. <*FRAME*>* не требует закрываю- щего тега и может содержать атрибуты, представленные в табл. 4.2.

Если содержимое фрейма не помещается в отведенную ему область окна браузера, то по умолчанию появляется полоса прокрутки вдоль гра- ницы фрейма. Управлять полосой прокрутки можно с помощью атрибута SCROLLING, который определяет наличие (yes), отсутствие (no) или ото- бражение по мере необходимости (auto устанавливается автоматически) полос прокрутки в текущем фрейме.

*Таблица 4.2*

**Атрибуты тега *<*FRAME*>***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Атрибут | Назначение | | | Принимаемые значения, форма записи | | |
| SRC | Адрес открываемой в дан- ном фрейме страницы | | | SRC="URL"  URL – относительный или абсо- лютный адрес страницы | | |
| NAME | Задает имя для данного конкретного фрейма | | | NAME="имя"  Имя должно начинаться с афа- витно-цифрового символа | | |
| TARGET | Указывает фрейма | имя | целевого | TARGET="имяфрейма" | | |
| SCROLLING | Определяет наличие полос прокрутки в текущем фрейме | | | Принимает три значения:  yes – полоса прокрутки присутст- вует всегда  no – запрет полосы прокрутки auto – автоматическое появление полосы прокрутки SCROLLING="yes" | | |
| FRAMEBORDER | Пользовательские грани- цы фреймов | | | Значение задается  FRAMEBORDER="5" | в | пикселах |
| BORDERCOLOR | Устанавливает цвет рамок | | | BORDERCOLOR="цвет" BORDERCOLOR="red" | | |
| NORESIZE | Запрет на изменение поль- зователем размеров окна фреймов | | | Не имеет значений | | |
| MARGINWIDTH | Задает правый и левый отступы | | | MARGINWIDTH="1cm" | | |
| WARGINHEIGHT | Задает верхний и нижний отступы | | | WARGINHEIGHT="2cm" | | |

Для запрета на изменение пользователем размеров окна фреймов применяется атрибут, который не требует указания никаких значений.

Каждый фрейм ограничен рамкой, которая устанавливается по умол- чанию для тега *<*FRAME*>.* Для управления этой рамкой используется ат- рибут FRAMEBORDER, который может принимать два значения: 0 – рам- ки не отображаются, 1 *–* наличие рамок между текущим и смежными фреймами. Для установки цвета рамки можно использовать атрибут BORDERCOLOR*.* Если в двух смежных кадрах определены собственные атрибуты BORDERCOLOR, ни один из них не будет иметь силы. Цвет их рамок будет определяться соответствующим атрибутом контейнера FRAMESET.

Для задания цвета используется стандартное имя цвета, или RGB- значение, или шестнадцатеричная константа. Наиболее удобны в исполь- зовании стандартные цвета, но браузеры их могут трактовать по-разному. Шестнадцать неизменных цветов: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, yellow.

***Пример 1.*** Структура фреймов одной высоты и разной ширины в HTML (рис. 4.1):

<HTML>

<HEAD> <TITLE> создание простых фреймов

</TITLE>

</HEAD><FRAMESET ROWS ="40%, 60%" COLS ="50%, 50%" BORDER="5">

<FRAME><FRAME><FRAME><FRAME></FRAMESET>

</HTML>

Приведенная запись разбивает окно браузера на четыре пустые рам- ки-фреймы, которые отделены друг от друга границей шириной в 5 пиксе- лов. Цвет рамки по умолчанию светло-серый.

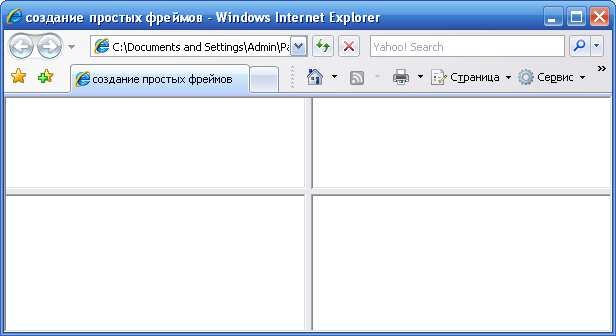


Рис. 4.1. Отображение фреймовой структуры примера 1

Полученные пустые рамки фреймов заполняем содержанием. Для этого используется атрибут SRC, который укажет адрес web-страницы. Воспользуемся текстом предыдущего примера, куда добавим адреса стра- ниц в соответствующие фреймы и просмотрим изменения (рис. 4.2).

<HTML>

<HEAD> <TITLE> создание простых фреймов

</TITLE>

</HEAD><FRAMESET ROWS ="40%, 60%" COLS ="50%, 50%" BORDER="5">

<FRAME SRC="page1.htm"><FRAME SRC="page2.htm">

<FRAME SRC="page3.htm"><FRAME SRC="page4.htm">

</FRAMESET>

</HTML>

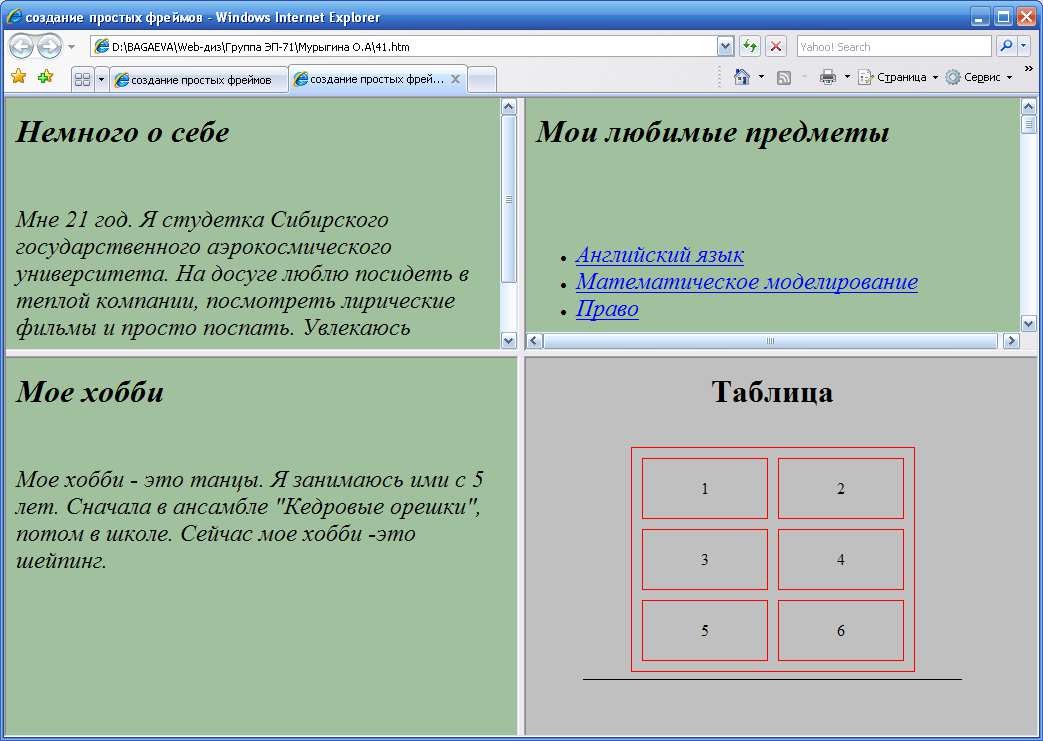


Рис. 4.2. Отображение фреймовой структуры с включенным содержанием фреймов

Атрибуты MARGINWIDTH и WARGINHEIGH предназначены для определения горизонтальных и вертикальных отступов внутри фрейма. Значения указываются в пикселах и являются равнозначными для обеих сторон.

<FRAME MARGINWIDTH="5" WARGINHEIGH="3">

В данном случае горизонтальный отступ будет равен 5 пикселам, а вертикальный – 3.

Остались нерассмотренными еще атрибуты NAME и TARGET, но к ним мы вернемся чуть позднее, когда будем рассматривать взаимодейст- вие фреймов.

**СОЗДАНИЕ ВЛОЖЕННЫХ ФРЕЙМОВ**

Каждый элемент *<*FRAMESET*>* формирует фреймы, относящиеся к одному набору (или уровню). Чтобы задать фреймы с разными свойства- ми (например, размеры или вид обрамления), применяются вложенные фреймы.

Вы познакомились с формированием на странице вертикальных и горизонтальных фреймов одного уровня, которые формируются с помо- щью одного тега *<*FRAMESET*>*. Более сложную структуру можно соз- дать, если использовать теги *<*FRAMESET*>*, вложенные друг в друга.

Попытаемся создать вертикальный и горизонтальный наборы фрей- мов (рис. 4.3).

|  |  |
| --- | --- |
| Фрейм 1 | Фрейм 2 |
| Фрейм 3 |
| Фрейм 4 |

Рис. 4.3. Фреймовая структура с вертикальными и горизонтальными фреймами

При создании такой структуры вы стремитесь сразу к двум целям. Во-первых, вам нужны два столбца. Во-вторых, правый столбец вам нужно разделить на три строки. Вот в такой последовательности и следует раз- местить описание фреймов: внутри описания правого столбца поместите описание трех строк. Это на языке HTML будет выглядеть так, как указано ниже, результат отображения в браузере см. на рис. 4.4.

<HTML>

<HEAD><TITLE> создание вложенных фреймов

</TITLE>.

</HEAD>

<FRAMESET COLS="40%, 60%">

<FRAME SRC="page2fl1.htm">

<FRAMESET ROWS="\*, \*, \*">

<FRAME SRC="page2fl2.htm">

< FRAME SRC="page2fl3.htm">

<FRAME SRC="page2fl4.htm">

</ FRAMESET>

</FRAMESET>

</HTML>

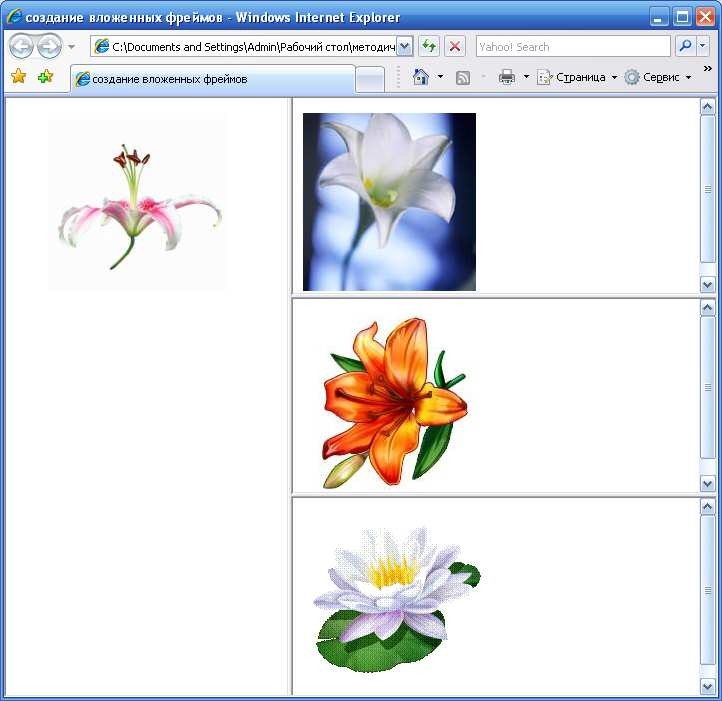


Рис. 4.4. Результат отображения вложенных фреймов

Итак, в данном примере второй *<*FRAMESET*>* вложен в первый. На самом деле он заменяет собой второй тег <FRAME>, требующийся первому набору фреймов. В нем и определяются три строки, на которые делится правый столбец.

Рассмотрим еще один пример на вложенные фреймы с использованием тех же самых web-страниц (рис. 4.5, 4.6).

Рис. 4.5. Структура вложенных фреймов второго примера

<HTML>

<HEAD>

<TITLE> создание вложенных фреймов

</TITLE>

</HEAD>

<FRAMESET ROWS="25%, 50%, 25%">

<FRAME SRC="page2fl1.htm">

<FRAMESET COLS="25%, 75%">

<FRAME SRC="page2fl2.htm">

<FRAME SRC="page2fl3.htm">

</FRAMESET>

<FRAME SRC="page2fl4.htm">

</FRAMESET>

</HTML>

В этом документе используется вложение одного кадра в другой. Внешний контейнер *<*FRAMESET*>* определяет три горизонтально распо- ложенных кадра, занимающих всю ширину окна браузера по вертикали (25, 50 и 25 %). Верхний и нижний кадры содержат документы page2fl1.htm и page2fl4.htm. Средний кадр имеет вложенный контейнер *<*FRAMESET*>.*

<FRAMESET COLS="25%, 75%">

<FRAME SRC="page2fl2.htm">

<FRAME SRC="page2fl3.htm">

</FRAMESET>

Определяющие – два вертикальных кадра, высота которых задается атрибутом внешнего контейнера. Левый кадр занимает 25 %, а правый – 75 % ширины окна браузера, они содержат документы page2fl2.htm и page2fl3.htm*.*

Однако на странице, как правило, не должно быть более трех или че- тырех кадров одновременно. Если вам хочется иметь больше 3–4 вертикальных или горизонтальных кадров одновременно, вам следует восполь- зоваться табличным представлением информации. Кадры в основном нужны для включения в вашу страницу элементов управления областями окна браузера или для организации динамического обновления информации. Размещение статичных данных по строкам и колонкам удобнее всего в обыкновенной таблице.

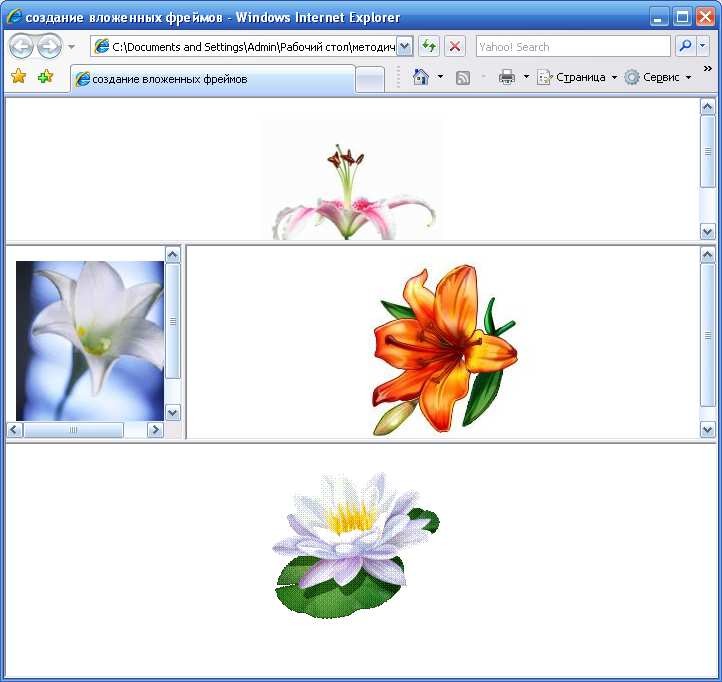


Рис. 4.6. Вложенные фреймы второго примера

Директива *<*NOFRAMES></NOFRAMES*>*, включающая вложенные теги <BODY> </BODY>, позволяет записать произвольное количество HTML-кода, который будет динамически преобразован в самостоятельную web-страницу, в случае если клиентский браузер не поддерживает фреймы.

**ВЗАИМОДЕЙСТВИЕ ФРЕЙМОВ**

После того как мы узнали о методах создания фреймов, пришло вре- мя познакомиться с их главным предназначением – управлением навига- цией по сайту. Работа с фреймовыми структурами имеет специфические особенности, которые следует учитывать при использовании данного сред- ства навигации для HTML-документов.

Взаимодействие между отдельными фреймами происходит посред- ством загрузки документа по соответствующей ссылке в установленное окно. Чтобы при организации ссылки выполнялась загрузка страницы в определенный фрейм, этому фрейму (он называется целевым) нужно при- своить уникальное имя с помощью атрибута NAME. Имена целевых фрей- мов могут начинаться с любой строчной или прописной буквы латинского алфавита. Всем фреймам, содержание которых планируется менять, необ- ходимо дать правильные имена: они должны начинаться с алфавитно- цифрового символа, но не с символа подчеркивания, так как он является первым символом зарезервированных имен фреймов. Все остальные име- на, начинающиеся указанным символом, игнорируются.

Чтобы назначить фрейму имя, следует вставить атрибут NAME в тег

*<*FRAME*>*.

<FRAME SRC="info.htm" NAME="perv">

устанавливает, что содержимым фрейма с именем "perv" после начальной загрузки будет файл info.htm*.* Реализация операции загрузки осуществля- ется с помощью параметра TARGET тега *<*A*>.*

<A HREF="moreinfo.htm" TARGET="perv"> текст ссылки </A>

Атрибут TARGET содержит имя фрейма. При активизации этой ссылки содержимое фрейма perv, т. е. файл info.htm, размещенный в нем при создании, будет заменен файлом moreinfo.htm.

Если при создании фрейма ему не присваивается имя, то по умолча- нию вызванный по гиперссылке документ будет загружаться в текущий фрейм, из которого была активизирована ссылка. Этого можно избежать и сделать так, чтобы страница появлялась в соседнем окне. Существуют спе- циальные зарезервированные имена, по которым происходит загрузка до- кументов на узлах с фреймовой структурой:

\_blank – загружает указанный файл в новое окно без названия;

\_self – загружает указанный файл в кадр, откуда делается вызов;

\_parent – загружает указанный файл в старший (родительский) кадр сетки кадров; если такой кадр не определен, то этот результат аналоги- чен self;

\_top – загружает указанный файл в полное окно, разрушая всю структуру кадров.

Приведем несколько примеров:

<A HREF="stuft.htm" TARGET="\_blank">

активизация ссылки приведет к открытию нового, не имеющего названия окна браузера и выводу в него содержимого файла stuft.htm. Это может быть новый HTML-документ либо определение новой сетки кадров.

<A HREF="stuft.htm" TARGET="\_self">

указывает на то, что при активизации ссылки кадр, в котором она распо- ложена, будет очищен и в него будет выведен файл stuft.htm.

<A HREF="stuft.htm" TARGET="\_parent">

при активизации ссылки сетка кадров, которой принадлежит вызывающий кадр, будет заменена содержимым stuft.htm.

<A HREF="stuft.htm" TARGET="\_top">

означает, что при активизации ссылки все окно браузера будет заменено содержимым stuft.htm.

Рассмотрим пример по взаимодействию между фреймами и откры- тию отдельных окон браузера с помощью атрибута TARGET.

Создадим файл со следующей фреймовой структурой (рис. 4.7).

|  |  |
| --- | --- |
| Меню  (список гиперссылок) | |
| Левый | Правый |

Рис. 4.7. Пример фреймовой структуры

<HTML>

<HEAD>

<TITLE> взаимодействие между фреймами

</TITLE>

</HEAD>

<FRAMESET ROWS="2\*, \*">

<FRAME NAME="menu" SRC="menu.htm"» NORESIZE FRAMEBORDER="1">

<FRAMESET COLS="50%, 50%">

<FRAME NAME="left" SRC="page2fl2.htm">

<FRAME NAME="right" SRC="page2fl3.htm">

</FRAMESET>

</FRAMESET>

</HTML>

Верхний фрейм будет содержать перечень гиперссылок, а два ниж- них предназначены для открытия в них содержимого конкретной ссылки, но предварительно в них загружается фоновое изображение.

Содержимое файла верхнего фрейма:

<HTML><HEAD>

<TITLE> взаимодействие между фреймами

</TITLE></HEAD>

<BODY BGCOLOR="#ffffff" TEXT="black" LINK="#ff0000" ALINK="#00ff00" VLINK="blue">

<H3>Ссылки верхнего фрейма></H3>

<UL TYPE="square">

<LI><A HREF="page2fl4.htm" TARGET="left">водяная лилия в левом фрейме</A>

<LI><A HREF="page2fl4.htm" TARGET="right"> водяная лилия в правом фрейме</A>

<LI><A HREF="page1121.htm" TARGET="left">ромашки в левом фрейме</A>

<LI><A HREF="page1121.htm" TARGET="right">ромашки в правом фрейме</A></UL></BODY></HTML>

После открытия файла с фреймовой структурой вы увидите экран браузера, разделенный на две горизонтальные части (рис. 4.8). В верхней части приведен список еще не использованных гиперссылок, которые ото- бражены синим цветом. А в нижней части, в свою очередь разделенной тоже на две части, увидите фоновое содержание этих фреймов. После использования гиперссылок видно, что содержимое этих частей меняется, а активная гиперссылка отображается зеленым цветом.

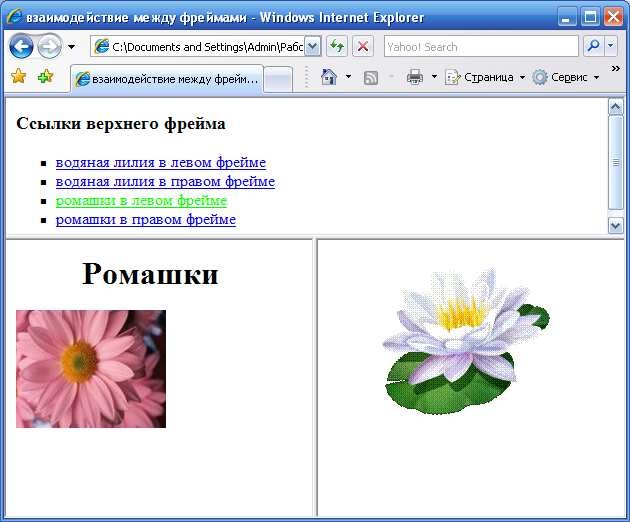
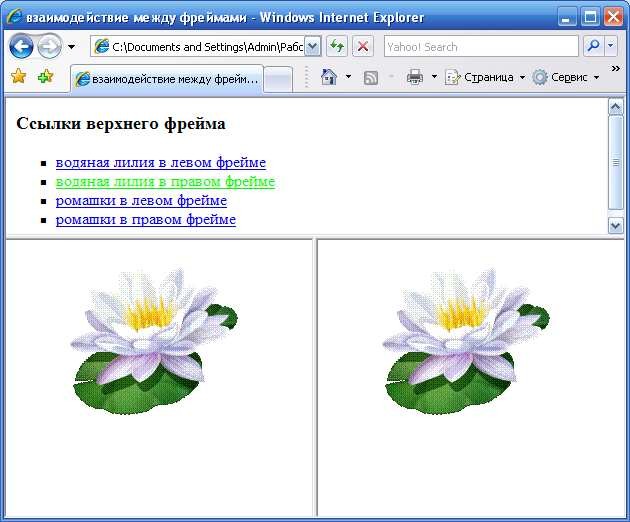
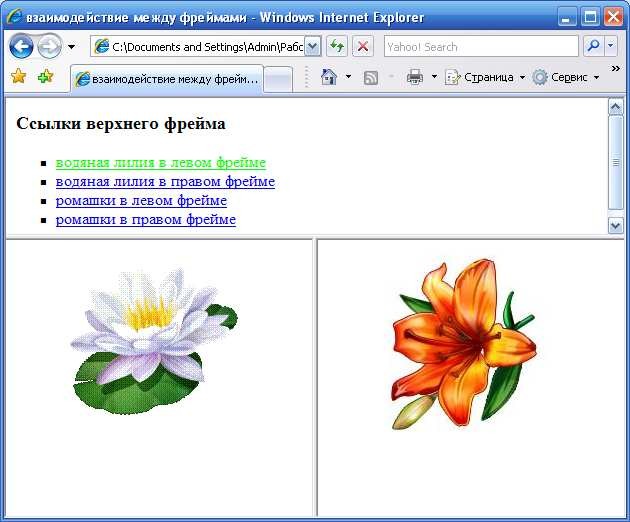
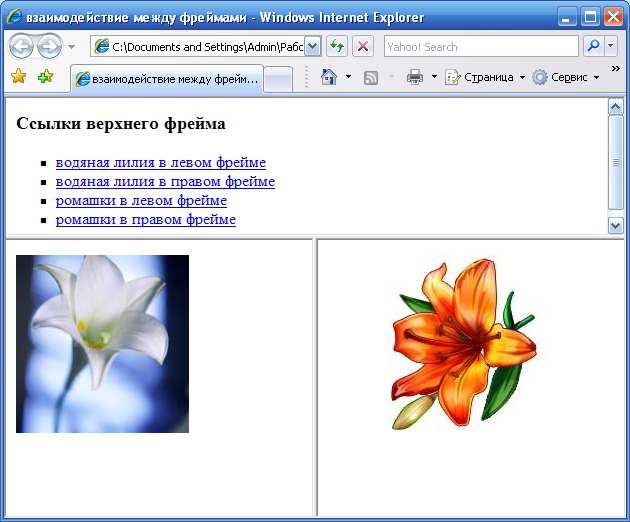


Рис. 4.8. Основная фреймовая структура и структура с использованием гиперссылок

Итак, мы рассмотрели документы, которые по специальной гипер- ссылке загружались в целевой фрейм. При создании целевого фрейма использовались атрибуты NAME и TARGET*.* В рассмотренном примере были использованы три целевых фрейма. Но иногда бывает ситуация, когда используется один целевой фрейм, а все файлы по нескольким гиперссылкам должны открываться в этом фрейме. Чтобы организовать такую загрузку, не нужно записывать атрибут TARGET в каждый элемент привязки, достаточно для этого использовать тег *<*BASE*>*, который раз- мещается в заголовочной части HTML-файла.

Фрейм, в который производится загрузка по гиперссылкам, называ- ется базовым фреймом и задается значением атрибута TARGET в одиноч- ном теге *<*BASE*>.* Например, начало HTML-файла, где располагаются гиперссылки, будет выглядеть следующим образом:

<HTML>

<HEAD>

<TITLE> список меню

</TITLE>

<BASE TARGET="right" >

</HEAD>

**ПЛАВАЮЩИЕ ФРЕЙМЫ**

Фреймы, которые размещаются в обычном HTML-документе в преде- лах тега *<*BODY*>* </BODY*>* с использованием тега *<*IFRAME*>* </IFRAME*>*, называются плавающими, его также называют строчным от сокращения английского термина «inline frame». Задание плавающего фрейма анало- гично описанию тега *<*IMG*>*. Кратко перечислим только допустимые атрибуты тега*:*

*SRC* – задает URL-адрес загружаемого файла;

*WIDTH*, *HEIGHT* – определяют ширину и высоту плавающего фрейма;

*ALIGN* – устанавливает выравнивание фрейма относительно окна браузера;

*MARGINWIDTH*, *MARGINHEIGHT*, *FRAMEBORDER SCROLLING* –

назначают размеры свободных полей, вывод рамки фрейма и полос про- крутки;

*NAME* – задает имя фрейма, используемого при определении целево- го фрейма.

В отличие от обычных фреймов, размеры плавающего фрейма не мо- гут быть изменены посетителем страницы, поэтому атрибут NORESIZE не входит в число допустимых атрибутов тега *<*IFRAME*>*

Пример схемы HTML-документа, содержащего плавающий фрейм, приведен ниже, а его отображение в браузере представлено на рис. 4.9.

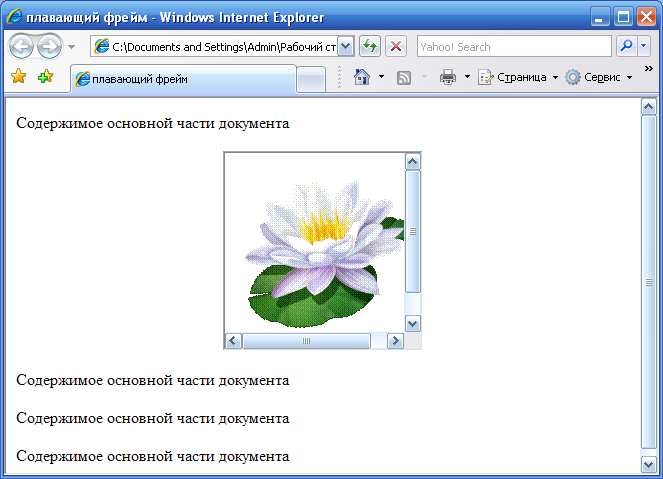


Рис. 4.9. HTML-документ с плавающим фреймом

<HTML><HEAD>

<TITLE> плавающий фрейм

</TITLE></HEAD>

<BODY>

<P> Содержимое основной части документа</P>

<IFRAME NAME="plav" SRC="page2fl4.htm" MARGINWIDTH="30" FRAMEBORDER="1" WIDTH="200" HEIGHT="200" ALIGN="center">

</IFRAME>

<P> Содержимое основной части документа</P>

<P> Содержимое основной части документа</P>

<P> Содержимое основной части документа</P>

</BODY>

</HTML>

*Практическое задание*

РАЗМЕТКА WEB-СТРАНИЦЫ С ПОМОЩЬЮ ФРЕЙМОВ

1. **Разметка web-страницы с использованием фреймов простой организации.** Наберите в простом редакторе «Блокнот» структуру гипер- текстовой программы.

<HEAD>

<TITLE>

Фреймы простой структуры

</TITLE>

</HEAD>

</HTML>

После закрытия заголовка </HEAD*>* вставьте тег *<*FRAMESET*>* для создания шести фреймов на странице (две строки и три столбца) – значе- ния атрибутов ROWS и COLS задайте в процентах. В этом же теге исполь- зуйте атрибуты задания ширины рамок BORDER и цвет рамок BORDERCOLOR. Сохраните изменения в файле и просмотрите исходный файл в браузере. Есть ли разделительные границы между фреймами? И ес- ли нет, то почему?

Затем внутрь тега <FRAMESET> вставьте 6 тегов <FRAME>

</FRAME>, соответственно количеству фреймов. Сохраните файл и про- смотрите в браузере. Отметьте, что изменилось сейчас. Задайте значения для атрибута SRС (для каждого фрейма задайте название файла, содержа- щего страницы page1.htm, page2.htm и т. д., созданные в предыдущих практических заданиях). Сохраните изменения и просмотрите в браузере.

Величину фреймов задайте сначала в относительных единицах, затем задайте в пикселах. Просмотрите разницу при задании этих значений в браузере.

Поэкспериментируйте: задайте различные цвета рамок для каждого фрейма.

1. **Разметка web-страницы с использованием фреймов сложной организации.** Используйте предыдущую страницу в качестве исходного текста программы. Для этого сначала сохраните его в другом файле, а за- тем внесите изменения согласно рис. 4.10.

Рис. 4.10. Структура фреймов для второго задания

1. **Создание навигации по сайту.** Создайте самостоятельно web-

страницу, оформленную с использованием фреймов (рис. 4.11).

|  |  |
| --- | --- |
| Заголовок страницы | |
| Гиперссылки | Содержание |
| Дополнительная информация или адрес | |

Рис. 4.11. Фреймовая структура для навигации по сайту

Содержание файла с гиперссылками выводится во фрейме «Гипер- ссылки». При активизации гиперссылки содержимое вызываемого файла должно открываться во фрейме «Содержание». Для привязки файлов к ги- перссылкам используйте готовые страницы из предыдущих практических заданий.

**Контрольные вопросы и задания**

1. Что такое фрейм?
2. Для чего предназначен фрейм?
3. Какие этапы используются при создании простой организации фреймов?
4. В чем отличие простого HTML-документа от документа с фрей- мовой организацией?
5. Какие теги используют для оформления фреймов?
6. Какие атрибуты тега *<*FRAMESET*>* определяют количество вер- тикальных и горизонтальных отрезков во фреймовом наборе?
7. Какие атрибуты предназначены для изменения параметров рамок фреймов?
8. Какая существует последовательность для задания параметров фреймов на странице?
9. Какие единицы используются для задания размеров фрейма?
10. Для чего предназначен тег <FRAME>?
11. Для чего используется атрибут NORESIZE тега <FRAME>*?*
12. Перечислите остальные атрибуты тега <FRAME> и их назна- чения.
13. Как происходит оформление web-страницы фреймами простой организации?
14. Как происходит оформление web-страницы с использованием сложной организации фреймов?
15. Для чего используются атрибуты NAME и TARGET тега

<FRAME>?

1. Перечислите зарезервированные имена фреймов.
2. Что такое целевой и базовый фрейм?
3. Назовите способ задания базового фрейма.
4. Какой фрейм является плавающим?
5. Как и где создается плавающий фрейм?

**Задания для самостоятельной работы**

Создайте вложенную фреймовую структуру на web-странице соглас- но варианту, указанному ниже.

# 1 2

**3 4**

# 5 6

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | |  | |
|  | |
|  | |  | |  |

**7 8**

**9 10**

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | |
|  |  | |
|  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |
|  |  |
|  |

# 11 12

**13 14**

|  |  |  |  |
| --- | --- | --- | --- |
|  | | |  |
|  | | |  |
|  |  |  | |

**15 16**

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | |  |
|  | |  | |
|  |  | |  |