

ПРАКТИЧЕСКАЯ РАБОТА 4

1. Запустите программу Paint.
2. Создайте рисунок, который вы в дальнейшем будете использовать в

качестве фонового изображения.

3. Сохраните этот рисунок под именем «Фон.gif».
4. Создайте домашнюю страничку, состоящую из нескольких фреймов (структуру фреймов выбрать самостоятельно).

5. Оформите, Web-страницу используя полученные знания.
6. Пример создания домашней странички приведен на рис. 17.
7. Переход по ссылке «Моё хобби» на рис. 18.

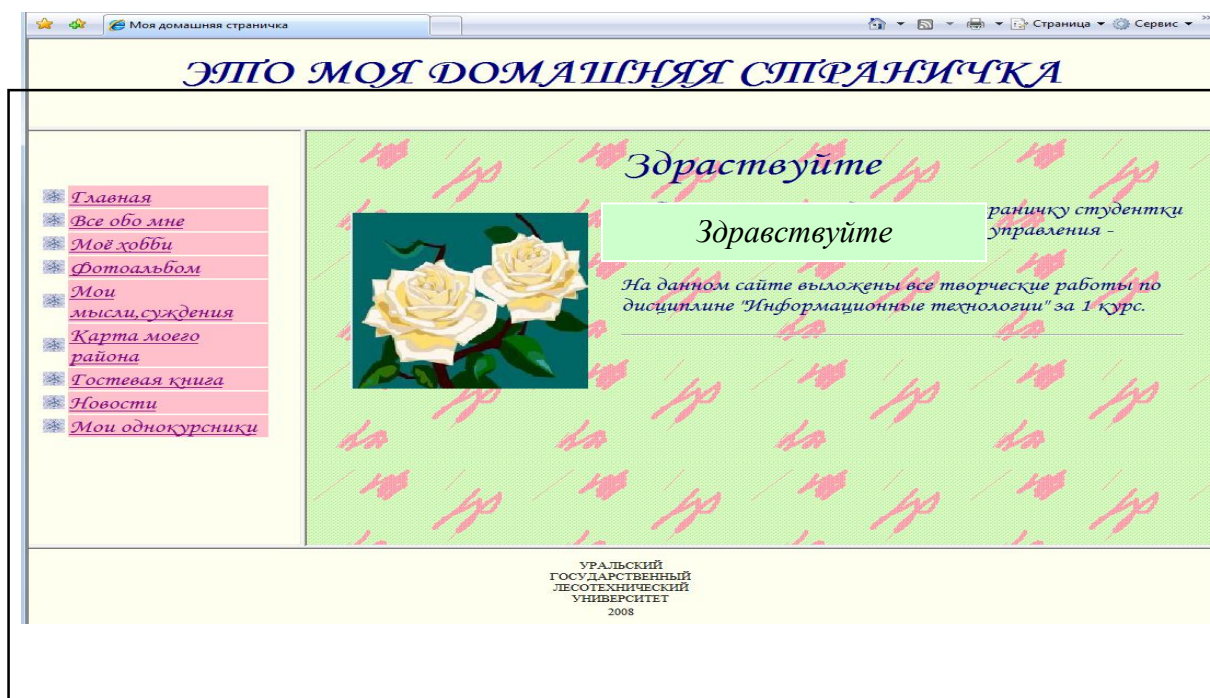


Рис.17. Пример оформления домашней странички



Рис. 18. Переход по ссылке «Моё хобби»

Элемент NOFRAMES

Фреймы являются весьма привлекательным средством оформления сайтов. Однако фреймы появились не сразу. У достаточно большого количества пользователей до сих пор могут стоять старые версии браузеров, НЕ поддерживающие фреймов вообще.

Если важно, чтобы при просмотре сайта, использующего фреймы, пользователь хоть что-то увидел, то следует применять HTML-элемент **NOFRAMES**. Этот элемент задается парными тегами **<NOFRAMES>** **</NOFRAMES>**. Он помещается в один файл с описанием набора фреймов. Между этими тегами помещается текст, который браузер показывает в том случае, если он не поддерживает фреймы (или если пользователь отключил в браузере поддержку фреймов).

ФОРМЫ

Назначение форм состоит в том, чтобы дать возможность посетителю управлять содержимым Web-страницы, вызывать на выполнение сценарии, а также вводить собственные данные и отправлять их на Web-сервер или по адресу электронной почты.

Создание формы

Для вставки формы в HTML-документ используется элемент **FORM**. Он задается парными тегами **< FORM >** **</ FORM >**. Между этими тегами помещаются описания элементов управления формы. При создании формы используются следующие атрибуты:

- **action** – обязательный для каждой формы параметр, URI программы-обработчика данных формы;
- **method** – задает способ отправки данных, введенных в форму, может принимать значения **get** (используется по умолчанию) или **post**;
- **enctype** – задает тип данных формы, если используется метод отправки **post**; по умолчанию имеет значение **application/x-www-form-urlencoded**; при необходимости передачи файлов используется значение **multipart/form-data**;
- **accept-charset** – применяется при передаче файлов, позволяет указать, какие кодировки используются для каждого из файлов (список строковых значений – названий кодировок), по умолчанию устанавливается значение **UNKNOWN** (приложение на сервере должно само определять кодировку);
- **accept** – описывает типы файлов (**MIME** – типы), передаваемые серверу; если этот параметр не использовать, то серверное приложение должно уметь само определять типы передаваемых ему файлов.

Стандартные элементы управления

Элементы управления, которые используются чаще всего – однострочное текстовое поле, поле для ввода пароля, флажки, переключатели, кнопки (как пользовательские, так и выполняющие стандартные действия), поля имен файлов. Все упомянутые элементы управления отображаются браузером **Internet Explorer**. Для обозначения всех этих элементов управления используется один **HTML**-элемент – **INPUT**. Этот элемент задается одиночным тэгом **<INPUT>** и имеет следующие атрибуты:

- **type** – принимает строку, задающую тип элемента управления (по умолчанию используется строка **text** и создается соответственно поле для ввода текста. Значения атрибута **type** приведены в табл. 3.

Таблица 3

Значения атрибута **type**

Значение атрибута	Особенности
text	Создается однострочное текстовое поле. Значение атрибута value отображается в качестве текста по умолчанию. Атрибут size воспринимается как количество символов, которое может отображаться в поле без необходимости горизонтальной прокрутки текста. Атрибут maxlength задает максимальное количество символов, которые могут быть введены в поле
password	Действие этого значения аналогично действию значения text , но дополнительно скрывается вводимый пользователем текст (заменяется символами «*» или другими). Используется для ввода конфиденциальной информации типа паролей

Значение атрибута	Особенности
checkbox	Создает элемент управления флажок. Задавать значение атрибута value нужно обязательно, так как именно это значение отправляется серверу, если флажок установлен. Можно использовать несколько элементов управления checkbox с одинаковым значением атрибута name для обеспечения возможности задания нескольких значений одного свойства
radio	Создает переключатель. Имеет смысл использовать как минимум два этих элемента управления вместе, объединенных в группу, с заданием одного и того же значения атрибута name. Тогда у пользователя появляется возможность выбора одного из нескольких вариантов (одновременно можно установить только один переключатель в группе). Значение атрибута value задавать также обязательно, оно используется аналогично элементу управления checkbox
file	Создает поле для ввода имени файла с возможностью выбора файла с помощью диалогового окна открытия файла или без него, что зависит от браузера. Содержимое выбранного файла или файлов пересылается вместе с формой. Для корректной работы необходима установка значения атрибута method формы в post, а enctype — в multipart/form-data. Значение атрибута value используется как имя файла по умолчанию. Значение атрибута size задается и работает аналогично элементу управления text
submit	Создает кнопку, при нажатии которой браузер отправит форму. Значение атрибута value задает надпись на кнопке
image	Создает изображение, при щелчке на котором произойдет отправка формы браузером. При этом на сервер передаются также координаты указателя мыши относительно левого верхнего угла изображения. Атрибут src задает расположение используемого изображения
reset	Создает кнопку, при нажатии которой значения всех элементов управления будут заменены значениями по умолчанию. Значение value задает надпись на кнопке
button	Создает пользовательскую кнопку. При нажатии или другом действии с кнопкой может выполняться ассоциированный сценарий. Значение атрибута value используется как надпись на кнопке. Подробно создание сценариев, ассоциирование их с элементами управления и не только будет рассмотрено далее
hidden	Создает скрытый элемент управления. Для него задаются атрибуты name и value. Пользователь не видит и не может изменить содержимое этого элемента управления. Однако данные скрытого элемента управления отправляются серверу вместе с остальными данными формы

- **name** – используется для задания имени элементу управления (строка, которая помимо идентификации элемента управления добавляется в данные, отсылаемые серверу);
- **checked** – булев атрибут, если он установлен, то флажок или переключатель считается (и отображается браузером);
- **size** – задает размер элемента управления (единицы измерения и действие специфичны для разных элементов управления);
- **value** – начальное значение для полей ввода текста и полей для указания имен файлов, также используется как надпись таких элементов управления, как кнопки; необязательно для всех элементов управления, кроме флажков и переключателей;
- **disabled** – булев атрибут, установка которого не позволяет пользователю работать с элементом управления;
- **readonly** – булев атрибут, позволяет запретить изменение состояния элемента управления (работает только для текстовых полей и поля выбора файла, так что для остальных элементов управления лучше использовать атрибут **disabled**, однако при этом данные деактивированных элементов управления не отправляются серверу);
- **maxlength** – задает максимальную длину текста, который может быть введен в текстовые поля (положительное численное значение);
- **src** – для элемента управления **image** задает расположение используемого изображения;
- **title** – описание элемента управления (может отображаться браузерами как всплывающая подсказка);
- **align** – задает горизонтальное выравнивание элемента управления, работает так же, как и для любого другого HTML-элемента, поддерживающего этот атрибут;
- **tabindex** – номер элемента управления при навигации при помощи табуляции;
- **accesskey** – «горячая» клавиша для элемента управления (для перехода к элементу управления нужно нажать **Alt** и заданную клавишу).

Многострочное текстовое поле

Обозначается этот элемент управления HTML-элементом **TEXTAREA** и задается парными тэгами **<TEXTAREA>** и **</TEXTAREA>**.

Списки

Другой способ дать возможность пользователю выбрать то или иное значение состоит в использовании списков. Различают два вида списков: обычные с полосой прокрутки и раскрывающиеся. Раскрывающийся список удобен в тех случаях, когда нужно сэкономить место на странице и уместить много элементов управления в пределах окна обозревателя.

Несмотря на различия во внешнем виде и функциональности, списки обоих типов создаются с помощью одних и тех же дескрипторов. `<SELECT>...</SELECT>` — создает пустой объект списка.

Установка атрибутов в дескрипторе `<SELECT>` определяет вид и функциональность списка.

SIZE — устанавливает число строк в поле списка. Если этому атрибуту присвоено значение 1, обозреватель показывает раскрывающийся список, при любом другом положительном значении — обычный список с полосой прокрутки.

MULTIPLE — позволяет выбирать в списке сразу несколько элементов. По умолчанию атрибут не установлен.

При выборе какого-либо элемента снимается выделение с элемента, который был выбран ранее.

Выбор нескольких элементов невозможен в раскрывающемся списке, поэтому установка атрибута **multiple** преобразует список в обычный даже в том случае, если атрибуту **size** присвоено значение 1.

NAME — имя объекта списка для обращения к списку в коде сценария.

`<OPTION>` — создает элемент списка. Данные дескрипторы добавляются в список между парой дескрипторов `<SELECT>...</SELECT>`. Любой текст за дескриптором `<OPTION>` отображается обозревателем как текст элемента списка. В данном дескрипторе устанавливаются следующие атрибуты:

– **VALUE** — значение элемента списка. Данное значение никак не отображается в окне обозревателя, но именно оно, а не подпись выбранного элемента списка, добавляется в данные формы;

– **SELECTED** — установка этого атрибута выделяет элемент списка по умолчанию.

Если атрибут **selected** установлен в нескольких дескрипторах `<option>`, тогда как в дескрипторе списка `<select>` атрибут **multiple** не был установлен, выделен будет последний элемент в списке с установленным атрибутом **select**.

Упражнение 13

1. Запустите программу Блокнот.
2. Наберите в окне редактора текст файла HTML:

```
<HTML>
  <HEAD>
    <TITLE>Переключатели и флажки</TITLE>
  </HEAD>
  <BODY>
    <FORM>
      <!-- Поле ввода -->
      <BR>Фамилия:
      <BR><INPUT TYPE='text' VALUE='Иванов'><BR>
      <!-- Группа переключателей -->
```

```

<H2>Группа переключателей</H2>
<BR><INPUT TYPE='radio' NAME='color'>Белый
<BR><INPUT TYPE='radio' NAME='color'>Красный
<BR><INPUT TYPE='radio' NAME='color'>Желтый
<BR><INPUT TYPE='radio' NAME='color'>Зеленый<BR>
<!-- Группа флажков -->
<H2>Группа флажков</H2>
<BR><INPUT TYPE='checkbox' NAME='fruit'>Груша
<BR><INPUT TYPE='checkbox' NAME='fruit'>Персик
<BR><INPUT TYPE='checkbox' NAME='fruit'>Яблоко<BR>
<!-- Кнопки -->
<BR><INPUT TYPE='button' VALUE='Пишите мне' ON-
CLICK=window.open('mailto:MyAddress@mail.ru');>
<BR><INPUT TYPE='button' VALUE='Посмотрите мои фотографии'
ONCLICK=window.open('http:\\\\www.MySite.org\\MyFoto.htm');> </FORM>
</BODY>
</HTML>

```

3. Сохраните файл под именем «Упражнение 13. html». Загрузите свой документ. На экране вы увидите результат своей работы, изображенный на рис. 19.

The screenshot shows a web browser window with the title "Переключатели и флажки". Inside the browser, there is a form with the following elements:

- A text input field labeled "Фамилия:" containing the text "Иванов".
- A section header "Группа переключателей" followed by four radio buttons labeled "Белый", "Красный", "Желтый", and "Зеленый".
- A section header "Группа флажков" followed by three checkboxes labeled "Груша", "Персик", and "Яблоко".
- Two buttons at the bottom: "Пишите мне" and "Посмотрите мои фотографии".

Рис.19. Форма с основными элементами управления

Упражнение 14

1. Запустите программу Блокнот.
2. Наберите в окне редактора текст файла HTML:
<HTML>

<HEAD>

<TITLE>Раскрывающиеся и обычные списки</TITLE>

</HEAD>

<BODY>

<FORM>

<TABLE>

<CAPTION> Выберите страну:</CAPTION>

<TR><TD VALIGN='top'>

<!--Раскрывающийся список-->

<SELECT NAME='COUNTRY' SIZE='1'>

<OPTION VALUE='Беларусь'>Беларусь

<OPTION VALUE='Казахстан'>Казахстан

<OPTION VALUE='Россия'>Россия

<OPTION VALUE='Украина'>Украина

</SELECT></TD>

<!--Обычный список-->

<TD VALIGN='top'>

<SELECT NAME='COUNTRIES' SIZE='3' MULTIPLE>

<OPTION VALUE='Беларусь'>Беларусь

<OPTION VALUE='Казахстан'>Казахстан

<OPTION VALUE='Россия' SELECTED>Россия

<OPTION VALUE='Украина'>Украина

</SELECT></TD></TR></TABLE></FORM></BODY></HTML>

3. Сохраните файл под именем «Упражнение 14.html». Загрузите свой документ. На экране вы увидите результат своей работы, изображенный на рис. 20.

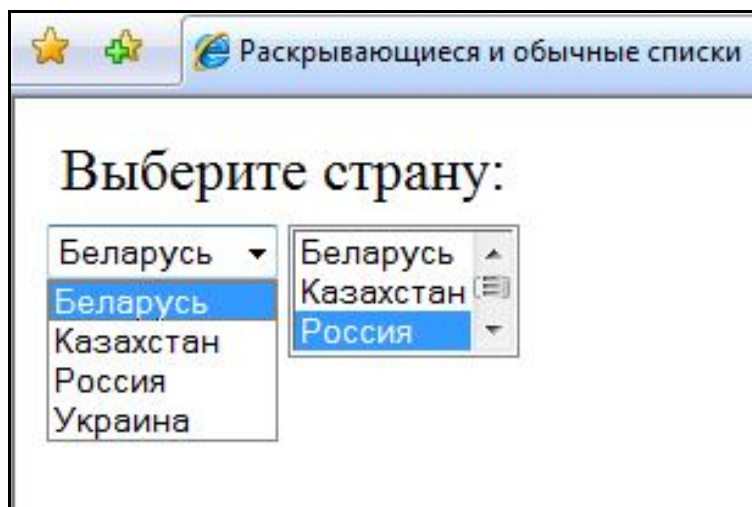


Рис.20. Группы флажков и переключателей