

**Цели занятия:***Учебные:*

1. Знать основные принципы построения современных ЭВМ и вычислительных систем.

*Воспитательные:*

1. Стимулировать активную познавательную деятельность обучаемых, способствовать формированию творческого мышления, добросовестного отношения к освоению учебного материала, настойчивости и целеустремленности в овладении знаниями.

**В ходе занятия формируются следующие компетенции (части компетенций):**

Код компетенции	Формируемые компетенции (части компетенций)
ДПК-3	Способность владеть основными методами, способами и средствами получения, хранения, переработки информации и работы в глобальных компьютерных сетях

**Учебные вопросы:**

1. Создание Web-страниц.
2. Разработка интерактивных страниц.
3. Взаимодействие сервера и клиента.

**Время:** 4 академических часа.

**Материально-техническое обеспечение:**

1. Используемая УМБ: Мультимедийный проектор типа BenQ-232.
2. СИО: УММ к ПЗ №3.

**Литература:**

1. А.С. Шаламов Интегрированная логистическая поддержка наукоемкой продукции. Университетская книга, 2008. – 463 с.
2. Е.В. Судов Интегрированная информационная поддержка жизненного цикла машиностроительной продукции. Принципы. Технологии. Методы. Модели. ООО Издательский дом «МВМ», 2003. – 263 с.
3. Е.В. Судов, А.И. Левин, В.В. Барабанов, А.Н. Давыдов Концепция развития CALS-технологий в промышленности России // М.: ВИМИ, 2002.

**Фонд оценочных средств для текущего контроля успеваемости:***Перед занятием:*

1. Перечислите способы подключения стилей к сайту.
2. Дайте краткую характеристику каждому стилю.
3. Дайте определение термину «Блочная верстка».
4. Укажите основной принцип Блочная верстка.

*В ходе и по окончании занятия:*

1. Порядок создания гиперссылки?

2. Порядок создания форм?
3. Порядок создания бегущей строки?
4. Для чего нужен тег <BODY>?
5. Как поменять цвет фона?
6. Порядок создания фреймов?
7. Порядок создания форм?
8. Порядок создания бегущей строки?
9. Как поменять цвет фона?
10. Порядок создания динамического баннера?

**Оценивание осуществляется по следующей шкале:**

100-балльная шкала	Результат освоения
Менее 40	Критерий не сформирован
41-70	Критерий четко не выражен
71-100	Критерий выражен четко

**Для оценивания ситуационных заданий используется следующая шкала:**

100-балльная шкала	Результат освоения
Менее 30	Обучающийся не может сформулировать проблему, представленную в задании
31-50	Обучающийся формулирует поставленную задачу, у него сформированы изолированные знания и умения, однако отсутствуют интегрированные понятия и навыки, в результате чего допущены ошибки в решении и задание не выполнено
51-80	Задание выполнено, обучающийся применяет знания для решения поставленной проблемы, однако не сформированы компетенции, вследствие чего обучающийся испытывает затруднения в демонстрации способов решения задачи
81-100	Задание выполнено как в теоретическом, так и в практическом плане, обучающийся легко демонстрирует свою компетентность по данному вопросу

## Введение

Веб-сайт - это набор логически связанных между собой веб-страниц, расположенных в Сети под единым именем - доменом.

Веб-страницы - это документы, которые написаны на языке разметки HTML и доступны в вебе. Эти страницы содержат контент - например, статьи, фото или видео. Сайты просматривают с помощью браузера.

Основные компоненты сайта - файлы кода, хостинг, доменное имя, функциональность, дизайн и контент.

Создать сайт самостоятельно можно с помощью конструктора или CMS.





Конструктор - платформа с визуальным редактором, позволяющим создавать страницы из готовых элементов.

CMS - платформа для управления контентом: CMS сложнее конструкторов, но дают больше возможностей.

## Вопрос №1. Создание Web-страниц

### 1.1. Гиперссылки

Ссылки могут вести на:

-  другую страничку,
-  на картинку (удобно не показывать большую картинку, а показать её уменьшенную копию, щелкнув на которую можно увидеть большое изображение в высоком качестве),
-  на любой файл,
-  на адрес электронной почты.





Для перехода на другую страничку используется конструкция:

`<A href="URL" target="Окно" title="Подсказка"> Название ссылки </A>`,

где:

**URL** (унифицированный локатор ресурсов) - адрес любого файла в Интернете. Может быть абсолютными, то есть указывается полный адрес странички (например, `http://web.holm.ru/avtor.htm`) и относительным, как видно из названия указывается файл относительно текущего (например, `avtor.htm`).

**Окно** определяет, в каком окне (фрейме) загрузить гиперссылку. Может иметь значения:

-  конкретное имя окна (фрейма) в котором нужно загрузить гиперссылку.
-  `_top` - загружает гиперссылку на всем пространстве окна браузера (если до этого существовало разбиение на фреймы, то оно исчезнет).
-  `_blank` - загружает гиперссылку в новом окне браузера.
-  `self` – загружает содержимое страницы, в окно, которое содержит эту ссылку (и так используется по умолчанию, так что если вам надо именно так загрузить ссылку, то параметр `target` вообще можно не использовать).

✚ `_parent` - загружает содержимое страницы, заданной ссылкой, в окно, являющееся непосредственным владельцем набора фреймов.

**Подсказка** - текст подсказки, который будет появляться при наведении мышки на гиперссылку.

**Название ссылки** - текст самой ссылки, например "Домашняя страничка Николая".

**ЗАДАНИЕ 1.** Создать простую гиперссылку на Web-страницу.

```
<a href="Doc1.htm" target="_top">Моя Web- страничка</a >
```

Часто используют не текстовую ссылку, а картинку, щелкнув на которую вы перейдете в другое место. Для этого используется следующая конструкция:

```
< a href="URL" target="Окно">
```

```

```

где:

**картинка** - это имя файла картинки. Если она находится в той же директории, что и ваша страничка, то это просто имя файла (\*.jpf, \*.gif, \*.png). Если картинка находится в ином месте (например, где-то в Интернете), то указывайте имя с полным путем.

**ширина** - ширина изображения в пикселях.

**высота** - высота изображения в пикселях.

**отступ1** - горизонтальный отступ от графического изображения в пикселях.

**отступ2** - вертикальный отступ в пикселях.

**окантовка** – так как это не картинка, а гиперссылка, то она может быть выделена окантовкой того же цвета, который определен для цвета гиперссылок. Итак, окантовка это число, определяющее толщину окантовки. Значение его, как правило, равно 0, а другие значения применяются довольно редко.

**Подсказка** - это сообщение, которое выводится над гиперссылкой вместо картинки, если она не показывается (не найдена или пользователь настроил свой браузер так, что тот не показывает картинки). Кроме того, вы увидите этот текст в виде подсказки, когда курсор мыши находится над картинкой-гиперссылкой. (!) В данной конструкции именно этот параметр определяет подсказку, а параметр `title` – не используется.

**ЗАДАНИЕ 2.** Создать гиперссылку на документ, используя в качестве гиперссылки картинку.

## 1.2. Атрибуты тега `body`

Из предыдущих занятий мы узнали, что структура Web-документа имеет вид:

```
<HTML>
```

```
<HEAD><TITLE>DesignLAB</TITLE></HEAD>
```

```
<BODY>
</BODY>
</HTML>
```

### Фон страниц

Рассмотрим тег `<BODY>` с которого и начинается тело документа. Тег парный, имеет вид:

**`<BODY>`** и **`</BODY>`**.

С его помощью можно изменить стиль страницы, перед тем как набрать текст. Например, необходимо сделать так, чтобы текст был белым на черном фоне, а ссылки (посещенные, непосещенные или локальные) - светло-зелеными. В этом случае, необходимо дополнить тэг `<BODY>` следующими параметрами:

```
файл: index.html
<HTML>
...
<BODY BGCOLOR="BLACK" TEXT="WHITE" LINK="LIME" ALINK="LIME" VLINK="LIME">
...
</BODY>
</HTML>
```

- **BGCOLOR="BLACK"** - цвет фона - черный.
- **TEXT="WHITE"** - цвет текста - белый.
- **LINK="LIME"** - цвет непосещенной ссылки - салатový.
- **VLINK="LIME"** - цвет посещенной ссылки - салатový.
- **ALINK="LIME"** - цвет локальной ссылки - салатový.

**ЗАДАНИЕ 3.** Создайте Web-страницу. Цвет текста **черный**, цвет фона **синий** (См. рис.1.)

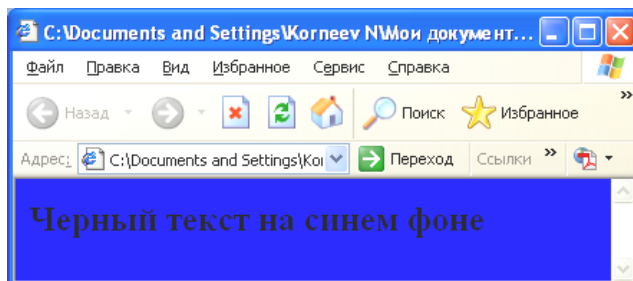


Рис.1. Черный текст на синем фоне

Очень часто в качестве фона используется заранее подготовленный, выполненный в графическом редакторе файл, подключение которого выполняет атрибут **BackGround**. Значением атрибута является имя файла (при необходимости указывается полный путь к файлу).

- **BACKGROUND="U1.GIF"** - цвет фона – графическая картинка.

**ЗАДАНИЕ 4.** Создайте Web-страницу Цвет текста **черный**, цвет фона графическая картинка (см. рис. 2.).

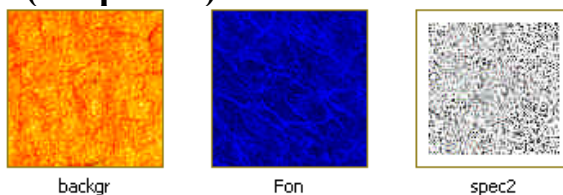


Рис. 2. Цвет фона - графическая картинка

Выберите файл для фона из предлагаемых.

**ЗАДАНИЕ 5.** Создать в **PhotoShop** файл для использования в качестве фона.

**Алгоритм выполнения.**

1. Откройте программу **PhotoShop**. Создайте новый файл размером 60 x 60 пикселей, в режиме **RGB** цветов (рис.3.).

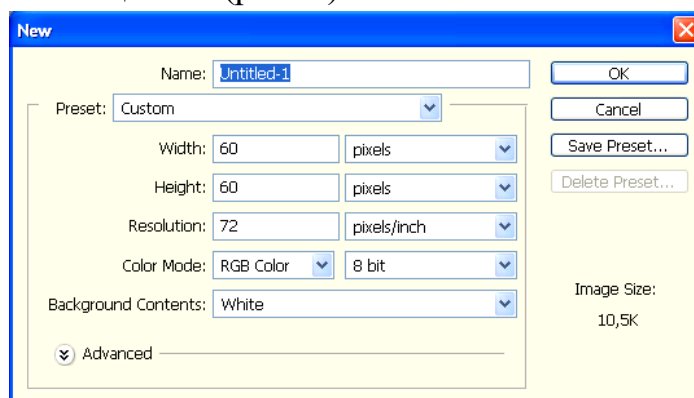


Рис.3. Окно создания нового файла.

2. Выполните **Filter/Noise/Add Noise**. Выберите **Gaussian**, и 80%.

3. Выберите **Filter/Styleize/Emboss**. Установите угол (**angle**) в -50, высоту (**Height**) в 2 пикселя, и Суммирование (**Amount**) в 70%. Теперь мы имеем серый рельефный бумажный фон (рис.4.).

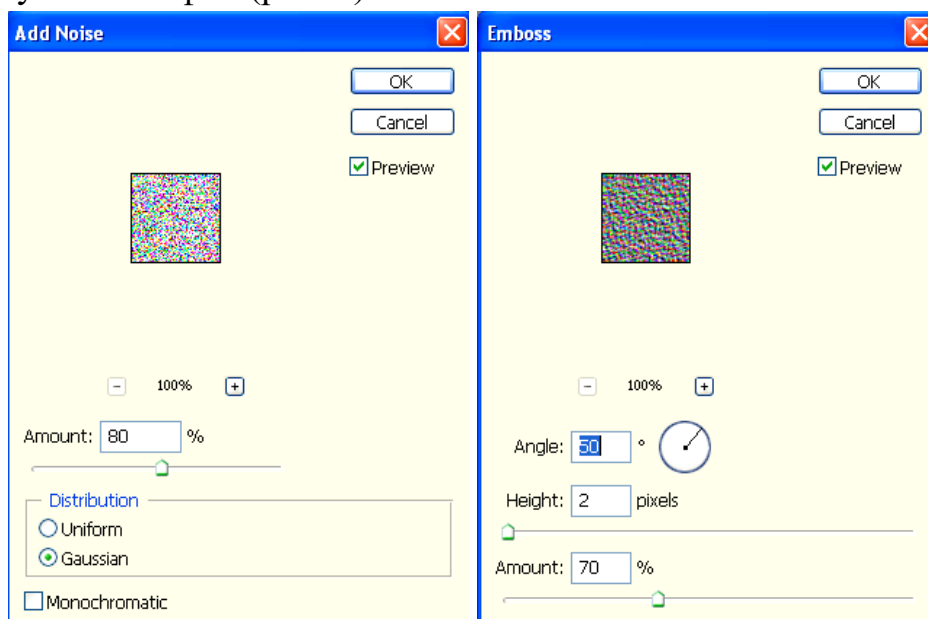


Рис.4. Применение фильтров

4. Чтобы изменить цвет, выберите **Image/Adjustments/Hue-Saturation**. Установите **Colorize**, и сделайте цвет (**Hue**) равным 40, также насыщенность (**Saturation**) равным 100 (рис.5.).

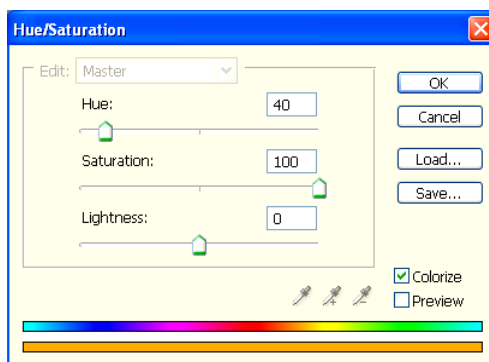


Рис. 5. Изменение цветонасыщения.

5. Если нравится другой цвет, установите в цвете другие установки. Если плитка получится очень темная, можно изменить контрастность используя **Image/Adjust/Levels** перетаскивая треугольник, пока не получите нужный цвет фона (рис.6.).

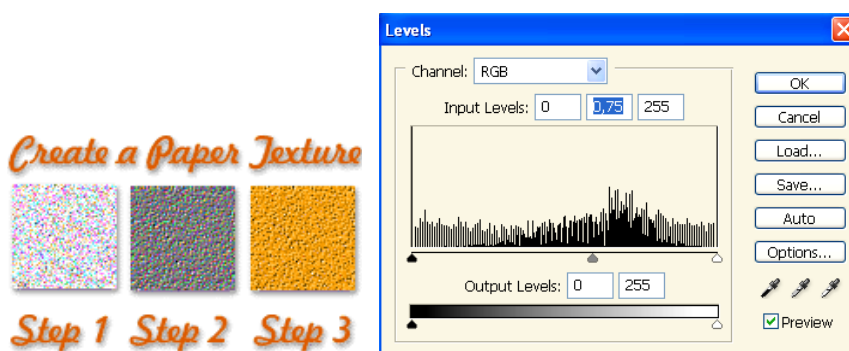


Рис.6. Изменение контрастности

6. Сохраните файл в **gif** формате для вашей странички, для этого выполните **Image/Mode/Indexed**, затем сохраните как **gif** файл.

7. Создайте web-страницу с заливкой созданной текстурой. Код представлен на рис. 7.

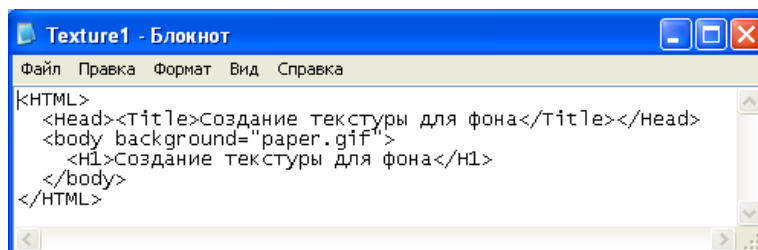


Рис.7. HTML –код Web страницы

8. Выполните просмотр данной страницы (рис.8.)

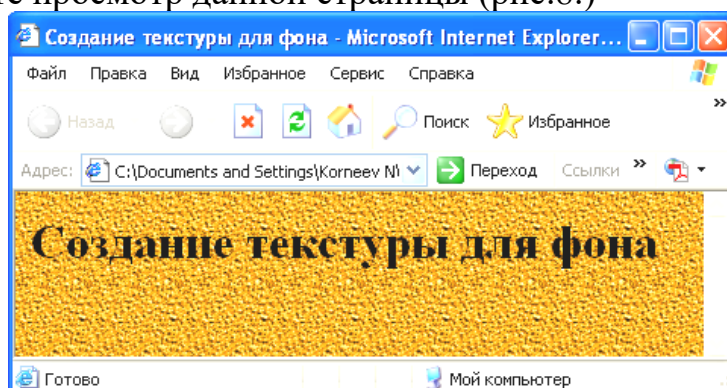


Рис.8. Web -страница

## Вопрос №2. Разработка интерактивных страниц

### 2.1. Верхние и нижние индексы. Специальные символы.

В последних версиях HTML появилось два новых тега - теги нижнего и верхнего индексов. Они очень удобны, когда нужно отобразить на странице такую специфическую информацию как химические формулы (H<sub>2</sub>O) или математические формулы (a<sup>2</sup>=b+c<sup>2</sup>).

Чтобы текст выводился в виде нижнего индекса, его следует пометить между тегами `<sub>` и `</sub>`, в виде верхнего - `<sup>` и `</sup>`. Например:

Химическая формула воды - H<sub>2</sub>O

Используемая нами формула - a<sup>2</sup>=b+c<sup>2</sup>

Вот что мы получим в результате:

Химическая формула воды - H<sub>2</sub>O

Используемая нами формула - a<sup>2</sup>=b+c<sup>2</sup>

**ЗАДАНИЕ 6.** Используя надстрочные и подстрочные теги, написать химическое уравнение

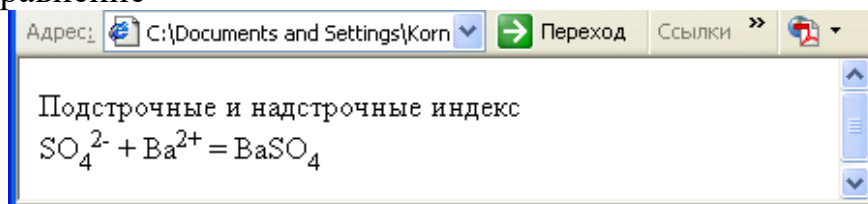


Рис. 9. Управление верхними и нижними индексами

### Специальные символы

Иногда могут возникнуть проблемы при попытке отобразить некоторые символы на веб-странице. Такие символы как **&**, **<**, **>**, **®**, **©**, **™**, используются при написании некоторых тегов. Другие символы не содержатся в обычном алфавите. Кроме того, существуют символы, используемые в иностранных языках, но не содержащиеся в английском - например, чтобы правильно отобразить на веб-странице слово français вам понадобится набрать **fran** (буквы до сидиля), набрать **&ccedil;**, и **ais** (буквы после сидиля). Имеем:

fran&ccedil; ais

Вот что получим в результате:

français

Таким же образом можно вставлять любые символы.

Таблица наиболее распространенных символов.

Символ	HTML - тег	Символ	HTML - тег
<	&lt;	>	&gt;



&	&amp;	"	&quot;
©	&copy;	®	&reg;
™	&trade;	¢	&cent;
£	&pound;	¤	&curren;
«	&laquo;	¬	&not;
—	&macr;	°	&deg;
±	&plusmn;	²	&sup2;
³	&sup3;	´	&acute;
μ	&micro;	¶	&para;
¸	&cedil;	¹	&sup1;
º	&ordm;	»	&raquo;
¼	&frac14;	½	&frac12;
¾	&frac34;	¿	&iquest;
À	&Agrave;	Á	&Aacute;
Â	&Acirc;	Ã	&Atilde;
Ä	&Auml;	Å	&Aring;
Æ	&AElig;	È	&Egrave;
É	&Eacute;	Ê	&Ecirc;
Ë	&Euml;	Ì	&Igrave;
Í	&Iacute;	Î	&Icirc;
Ï	&Iuml;	Ð	&ETH;
Ñ	&Ntilde;	Ò	&Ograve;
Ó	&Oacute;	Ô	&Ocirc;
Õ	&Otilde;	Ö	&Ouml;
×	&times;	Ø	&Oslash;
Ù	&Ugrave;	Ú	&Uacute;
Û	&Ucirc;	Ü	&Uuml;
Ý	&Yacute;	Þ	&THORN;
ß	&szlig;		&nbsp;

**ЗАДАНИЕ 7.** Создать простейшую Web-страницу с использованием специальных символов (рис.10.)

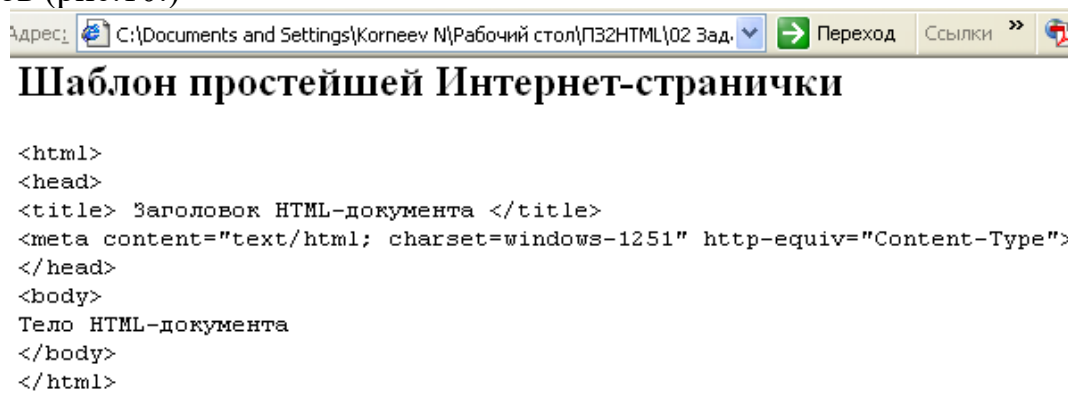


Рис. 10. Web-страница с использованием специальных символов

## Линии и их параметры

Тег **<HR>** рисует тонкую линию на всю ширину экрана  
Основные атрибуты тега **<HR>** приведены на рис.11.

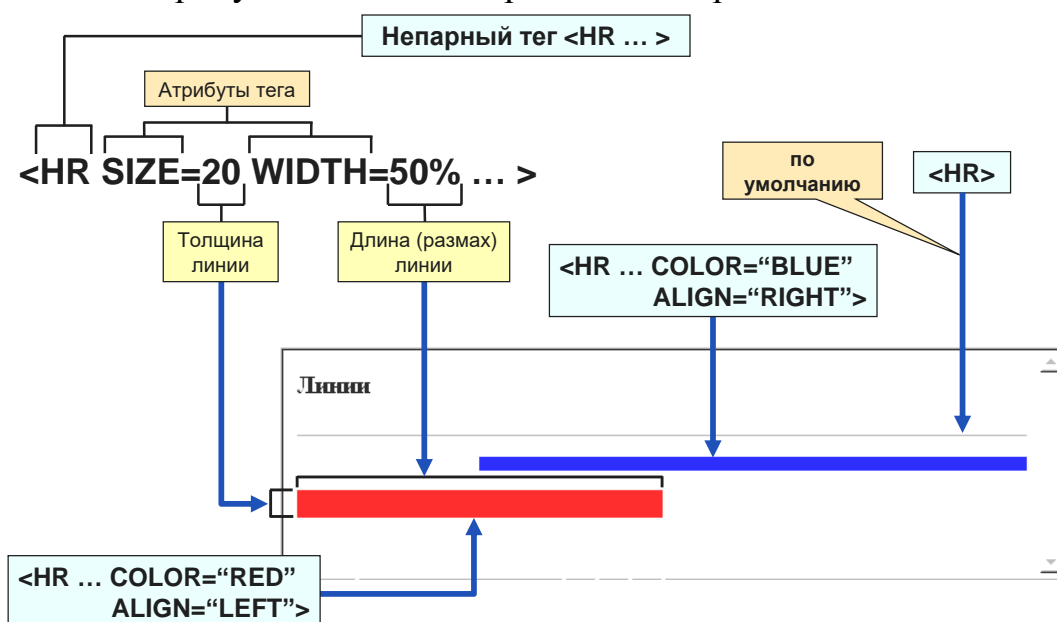


Рис.11. Синтаксическая диаграмма тега линия (**<HR>**)

Помимо тонкой линии поперек страницы можно применить и другой тип линии, используя расширения тега **<hr>** (атрибуты). Рассмотрим несколько примеров:

Простая линия <code>&lt;hr&gt;</code>
Линия без тени <code>&lt;hr noshade&gt;</code>
Линия размером в 10px <code>&lt;hr size="10"&gt;</code>
Линия шириной в 80% <code>&lt;hr width="80%"&gt;</code>
Оранжевая линия <code>&lt;hr color="orange"&gt;</code>

### Параметры линий:

● `<hr noshade>` - По умолчанию, линия отбрасывает трудно-видимую тень. Для того чтобы тени от линии не было, необходимо использовать такой синтаксис.

● `<hr size="10">` - По умолчанию, линия имеет размер 1 пиксел. Но мы можем его менять. Для этого необходимо вписать `size="ваше число"`, где **ваше число** - число пикселей.

● `<hr width="80%">` - По умолчанию, линия имеет ширину равную 100%, т.е. во всю длину окна браузера. В данном случае мы изменили этот параметр на 80%, т.е. ширина линии равна не 100, а 80 % занимаемой ширины окна.

● `<hr color="orange">` - По умолчанию, линия имеет прозрачный цвет. Но из-за тени, нам кажется, что линия серая. Мы можем изменить цвет линии.

Все эти параметры могут сочетаться и использоваться параллельно.

**ЗАДАНИЕ 8.** Сгармонируйте все параметры в одну линию.



**ЗАДАНИЕ 9.** Создать Web-страничку, используя картинки, вставленные в ячейки таблицы в качестве ссылок

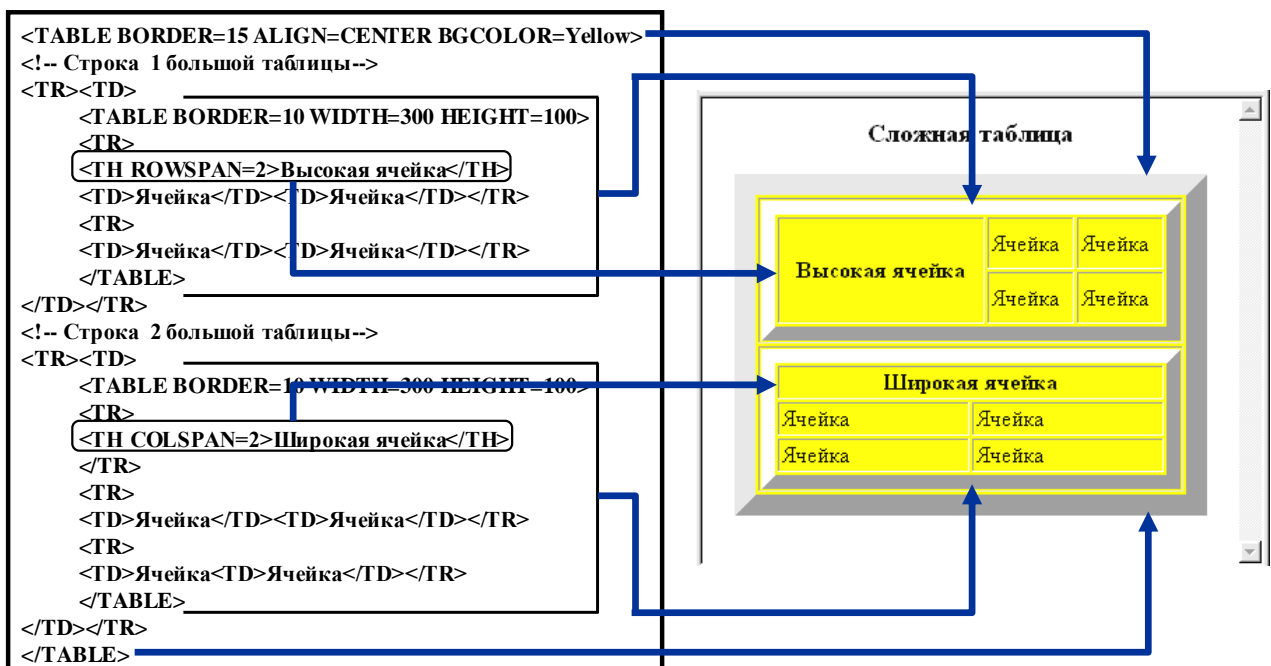


```

F20 - Блокнот
Файл Правка Формат Вид Справка
<TABLE BORDER=4 CELSPACING=6 CELLPADDING=3>
<!-- Заглавие таблицы и размер заглавия -->
<CAPTION ALIGN=TOP></CAPTION>
<TR>
<TH>ЛЕС</TH>
<TD><A HREF="0028b.jpg"><IMG SRC="1.jpg"></A></TD>
<TD><A HREF="0029b.jpg"><IMG SRC="2.jpg"></A></TD>
<TD><A HREF="0030b.jpg"><IMG SRC="3.jpg"></A></TD>

```

### СЛОЖНЫЕ ТАБЛИЦЫ



Примеры сложных таблиц:

1	2
	3
4	5

1	3
2	
4	5

```
<table width="200" border="1" cellpadding="0" cellspacing="0">
<tr>
  <td rowspan="2">1</td>
  <td>2</td>
</tr>
<tr align="center">
  <td>3</td>
</tr>
<tr>
  <td>4</td>
  <td>5</td>
</tr>
</table>
```

Все отличие такого построения таблиц заключается в появлении параметра **rowspan="2"**, цифра в котором соответствует количеству "подстрок", т.е. буквально то, сколько строк слилось в этой ячейке:

**ЗАДАНИЕ 10.** Построить таблицу, приведенную на рисунке

1	3	3b
2		
4	5	

### КАРТЫ-ИЗОБРАЖЕНИЯ. БЕГУЩАЯ СТРОКА.

Карты изображений. Используя карту-изображение можно создать графическую гиперссылку, которая будет срабатывать при нажатии на определенный участок рисунка. Причем не на огромный кусок, а на участок любой формы. Карты распознают рабочие поля трех форм: круг, прямоугольник и многоугольник (рис. 12.).

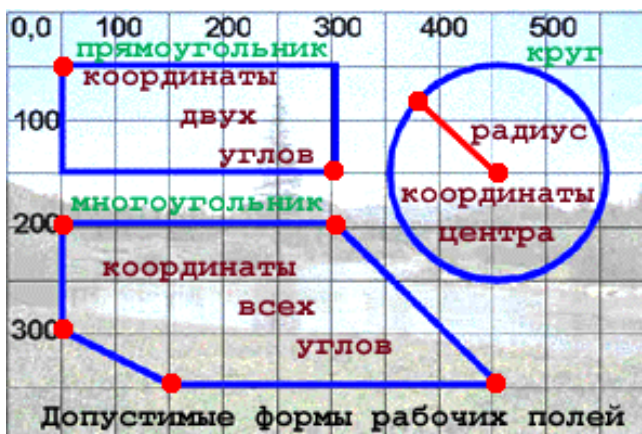


Рис.12. Виды полей карты изображений

Для начала необходимо:

## 1. Включить в Web - документ картинку.

Пример:

```
<IMG height=308 width=400 alt="Карта Европы" src="photo2.jpg"
align=right border=0 useMap=#photo >
```

Изучить все атрибуты тега <IMG> и выпишите их тетрадь

Height -

Width -

alt -

src -

align

border -

useMap -

## 2. Установить на карте (картинке) области, являющиеся гиперссылками

2.1. Участок, где будет расположен рисунок обозвать картой с некоторым именем:

```
<MAP name=photo>
```

2.2. Обозначить координаты и адреса всех будущих ссылок: (см. ниже)

2.3. Закрыть участок данной карты: </MAP>

```
<MAP name=photo>
  <AREA href="Куда уйти" shape=CIRCLE coords=x1,y1,R
      alt=Окружность share="circle">
  <AREA shape=RECT coords=x1,y1,x2,y2 share="rect">
  .....
</MAP>
```

Разберем подробно п.2.2.:

**Круглое поле:** После описания в специальном теге **area** самой ссылки, задается тип рабочего поля(circle) и даются координаты центра круга по осям координат (два первых числа) а затем его радиус (все в пикселах): `<area href="zz.htm" shape="circle" coords="350, 200, 100">`

**Прямоугольник:** Задаются координаты сперва верхнего левого, а затем, так же через запятую, противоположного угла :

```
<area href="pp.htm" shape="rect" coords="34,67,125,280">
```

**Многоугольное поле:** Координаты всех точек поочереды (в порядке их соединения (последняя точка соединится с первой)) просто перечисляются через запятую.

```
<area href="bb.htm" shape="polygon" coords="56, 175, 347, 61, 98,5 6, 11, 37,
57, 27, 250, 99">
```

### Определение координат:

Возьмите чистый лист бумаги, откройте необходимую фотографию в Adobe Photoshop.

Откройте палитру **Info**, выполните настройку (укажите координаты в пикселях) и считываете соответствующие координаты (рис.13.)

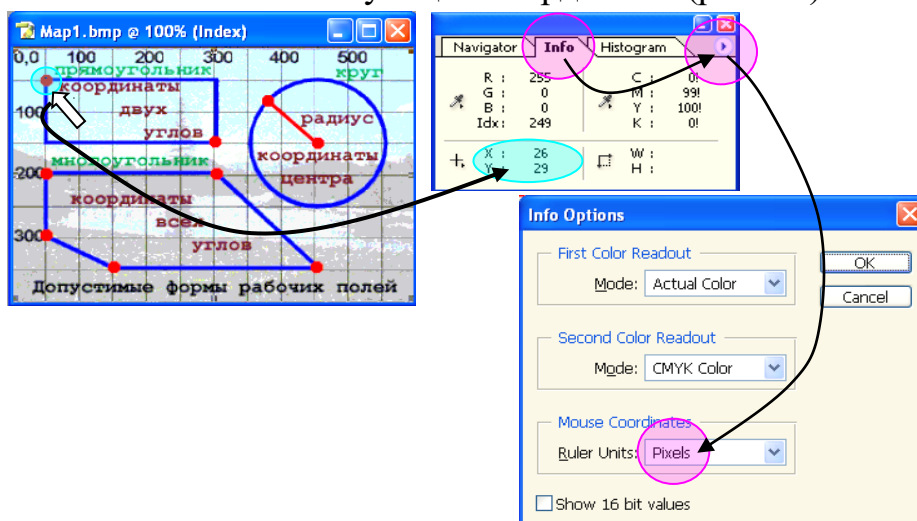






Рис.13. Определение координат, используя программу PhotoShop

**<MAP NAME="map\_name">** Данный тег определяет начало описания карты с именем `map_name`


**<AREA...>** Описывает участок изображения и ставит ему в соответствие URL.


#### Параметры:


**SHAPE** - Необязательный параметр, указывающий на форму определяемой области изображения. Может принимать значения:

-  default - по умолчанию (обычно прямоугольник)
-  rect - прямоугольник
-  circle - круг
-  poly - многоугольник произвольной формы

**COORDS** - Координаты в пикселях описываемой области.

 Для прямоугольника это четыре координаты левого верхнего и правого нижнего углов,

 Для круга - три координаты (две - центр круга, третья - радиус).

 Для многоугольника это описание каждого угла в двух координатах - соответственно число координат равно удвоенному количеству углов.

**ЗАДАНИЕ 7.** Создать карту изображений, выделив области (рис. 14.):




-  Берлин – окружность
-  Франция – прямоугольник
-  Испания - многоугольник



Рис.14. Выделение областей на карте изображений

## Бегущая строка

Структура тега <MARQUEE> приведена на рис. 15.

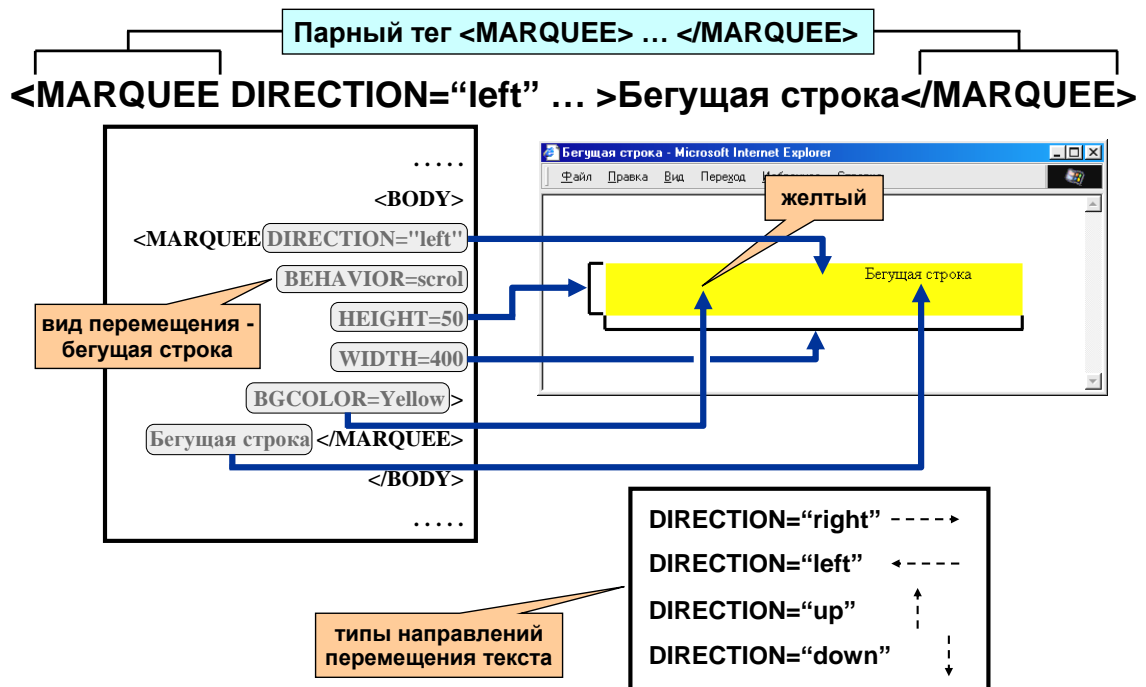


Рис.15. Структура тега <MARQUEE>

Для создания бегущей строки в IE напишите в исходнике следующее:

<marquee>

**Произвольный текст** (например – Бегущая строка)

</marquee>

Оформить строку можно следующим образом:

<marquee behavior="alternate" bgcolor="yellow" scrolldelay="145"  
 width="50%">

< font size="5" color="red">Текст бегущей строки. </font>

</marquee>



- **behavior="alternate"** -команда заставляющая строку болтаться туда-сюда. По умолчанию данной команды, строка ползет справа налево.
- **BGCOLOR** -цвет фона. Все что касается цвета
- **Scrollldelay** - скорость перемещения.
- **Width** - длина бегущей строки (необязательно в процентах).

**ЗАДАНИЕ 8.** Создать бегущую строку в соответствии с заданием (рис.16.)

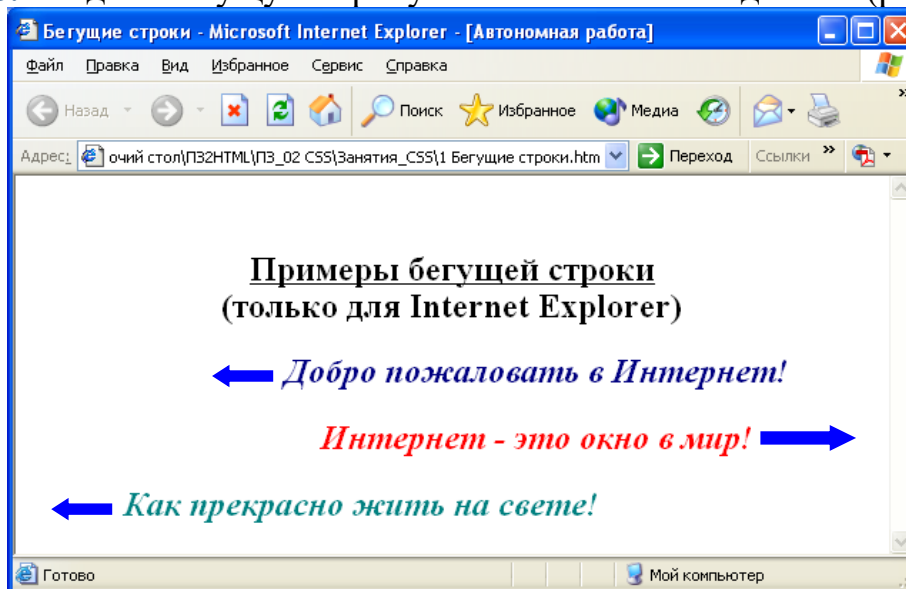


Рис.16. Вариант задания «Бегущая строка»

**Контрольные вопросы:**

1. Порядок создания гиперссылки?
2. Порядок создания форм?
3. Порядок создания бегущей строки?
4. Для чего нужен тег <BODY>?
5. Как поменять цвет фона?



## Вопрос №2. ТЕХНОЛОГИЯ РАЗРАБОТКИ WEB-САЙТА ФРЕЙМЫ.

**Фреймы** (Frames-рамки, кадры) - множественные скроллируемые окна, позволяющие выводить в одном окне браузера, несколько страниц одновременно. Использование фреймов позволяет получить интересные интерфейсные решения оформления страниц.

Фреймы задаются тэгом **<FRAMESET></FRAMESET>**

### Атрибуты тэга <FRAMESET>:

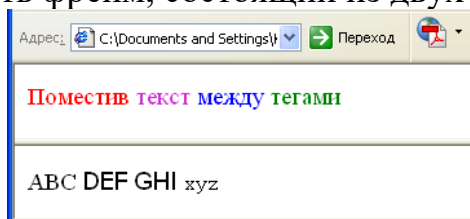
<i>rows</i> ="разделенный запятыми список пикселей, процентов и относительных длин".	Расположение горизонтальных фреймов. По умолчанию равен 100%, что означает одну строку.
<i>cols</i> ="разделенный запятыми список пикселей, процентов и относительных длин".	Расположение вертикальных фреймов. По умолчанию равен 100%, что означает один столбец.

### СТРУКТУРА HTML ДОКУМЕНТОВ ДЛЯ ФРЕЙМОВ

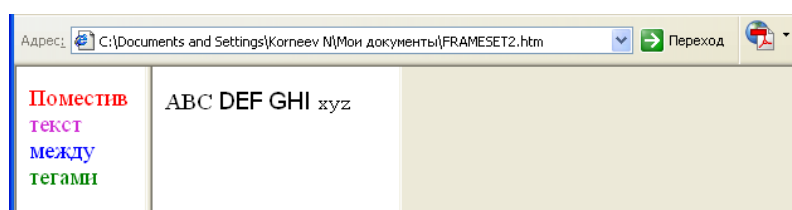
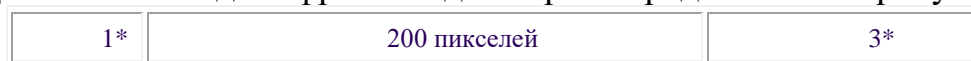
```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD> <TITLE>Фреймы</TITLE>
<FRAMESET FRAMEBORDER="1" FRAMESPACING="15" COLS="40%,*">
  <FRAME SRC="file1.htm">
  <FRAMESET ROWS="150,*">
    <FRAME SRC="file2.htm" MARGINWIDTH="30">
    <FRAME SRC="file3.htm" MARGINWIDTH="10">
  </FRAMESET>
</FRAMESET>
</HTML>
  
```

**ЗАДАНИЕ 1.** Создать фрейм, состоящий из двух строк.



**ЗАДАНИЕ 2.** Создать фрейм вид которого представлен на рисунке.



**ЗАДАНИЕ 3.** Создать сложную структуру фреймов с помощью вложения набора фреймов друг в друга. при открытии которой страница разбивается на две строки, высотой 30% и 70% от всей высоты окна браузера, и три столбца, левый задан с шириной 33% от ширины экрана браузера, средний - 37%, а правый получит все что осталось т.е.  $100\% - (33\% + 37\%) = 30\%$

Страничка разобьется приблизительно так:

Название странички	
Главное меню	Здесь будем отображать информацию, выбранную в главном меню
copyright	

Здесь три строки, причем средняя имеет два столбца. Сначала задаем строки. Верхняя и нижняя будут иметь высоту по 5%, а средняя то что останется.

```
<FRAMESET rows="5%,*,5%" >
```

Затем задаем столбцы. Левый будет 30% от ширины браузера, а правый, соответственно, 70%.

```
<FRAMESET cols="30%,70%" >
```

```
</FRAMESET>
```

```
</FRAMESET>
```

Теперь осталось задать сами фреймы. Фреймы определяются тэгом

```
<FRAME></FRAME>.
```

Атрибуты тэга <FRAME>:	
<b><i>src="URL"</i></b>	Описывает URL документа, который будет отображен внутри ланного фрэйма. Если он отсутствует, то будет отображен пустой фрэйм.
<b><i>name="frame_name"</i></b>	Имя фрейма, которое должно обязательно начинаться с символа. Содержимое поименованных фрэймов может быть задействовано из других документов при помощи специального атрибута TARGET, описываемого ниже.
<b><i>marginwidth="value"</i></b>	Ширина боковых разделительных полос между фреймами в пикселях.
<b><i>marginheight="value"</i></b>	Ширина верхних и нижних разделительных полос между фреймами в пикселях.
<b><i>scrolling="yes", "no", "auto"</i></b>	Наличие полос прокрутки у фреймов где: <ul style="list-style-type: none"> <li>• yes - полосы прокрутки присутствуют.</li> <li>• no - полосы прокрутки отсутствуют.</li> <li>• auto - полосы прокрутки присутствуют только при их необходимости (по умолчанию).</li> </ul>
<b><i>noresize</i></b>	Отмена возможности изменения размеров фреймов. Если у одного фрэйма установлен атрибут noresize, то у соседних фрэймов тоже не может быть изменен размер со стороны данного.

Исходя из всего этого допишем программу:

```
<FRAMESET rows="5%,*,5%" >
```

Задаем параметры верхнего фрейма, который будет фиксированной ширины, без полос прокрутки

```
<FRAME src="top.htm" name="top_page" scrolling="no" noresize>  
<FRAMESET cols="30%,70%" >
```

Задаем параметры левого фрейма, фиксированной ширины, с возможностью вывода полос прокрутки

```
<FRAME src="main_mnu.htm" name="main_mnu" scrolling="auto" noresize>
```

Задаем параметры правого фрейма, фиксированной ширины, с возможностью вывода полос прокрутки

```
<FRAME src="content.htm" name="content" scrolling="auto" noresize>  
</FRAMESET>
```

И наконец, задаем параметры нижнего фрейма, фиксированной ширины, без полос прокрутки.

```
<FRAME src="bottom.htm" name="bottom_page" scrolling="no" noresize>  
</FRAMESET>
```

Полностью код программы будет выглядеть так:

```
<HTML>  
<TITLE>Титул странички</TITLE>  
<FRAMESET rows="5%,*,5%" >  
  <FRAME src="top.htm" name="top_page" scrolling="no" noresize>  
  <FRAMESET cols="30%,70%" >  
    <FRAME src="main_mnu.htm" name="main_mnu" scrolling="auto"  
      noresize>  
    <FRAME src="content.htm" name="content" scrolling="auto" noresize>  
  </FRAMESET>  
  <FRAME src="bottom.htm" name="bottom_page" scrolling="no" noresize>  
</FRAMESET>
```

Документ, описывающий набор фреймов, не содержит тэга **<BODY>**. Он должен задавать только структуру фреймов и больше ничего.

Чтобы страничка была видна в браузерах не поддерживающих фреймы, следует указать тэг **<NOFRAMES></NOFRAMES>**, внутри контейнера **<FRAMESET>**.

К таким браузерам относятся браузер NSCA Mosaic, IE2.0 и NN версии ниже, чем 2.0.

Например:

```
<FRAMESET>  
<NOFRAMES>  
<BODY bgcolor="red"> Ваш браузер не поддерживает фреймы. </BODY>
```

</NOFRAMES>

</FRAMESET>

## РАБОТА С ФОРМАМИ

Форма на Web-странице представляет собой лишь набор *полей*, которые можно также рассматривать как *элементы управления*.

### ОПИСАНИЕ ФОРМЫ

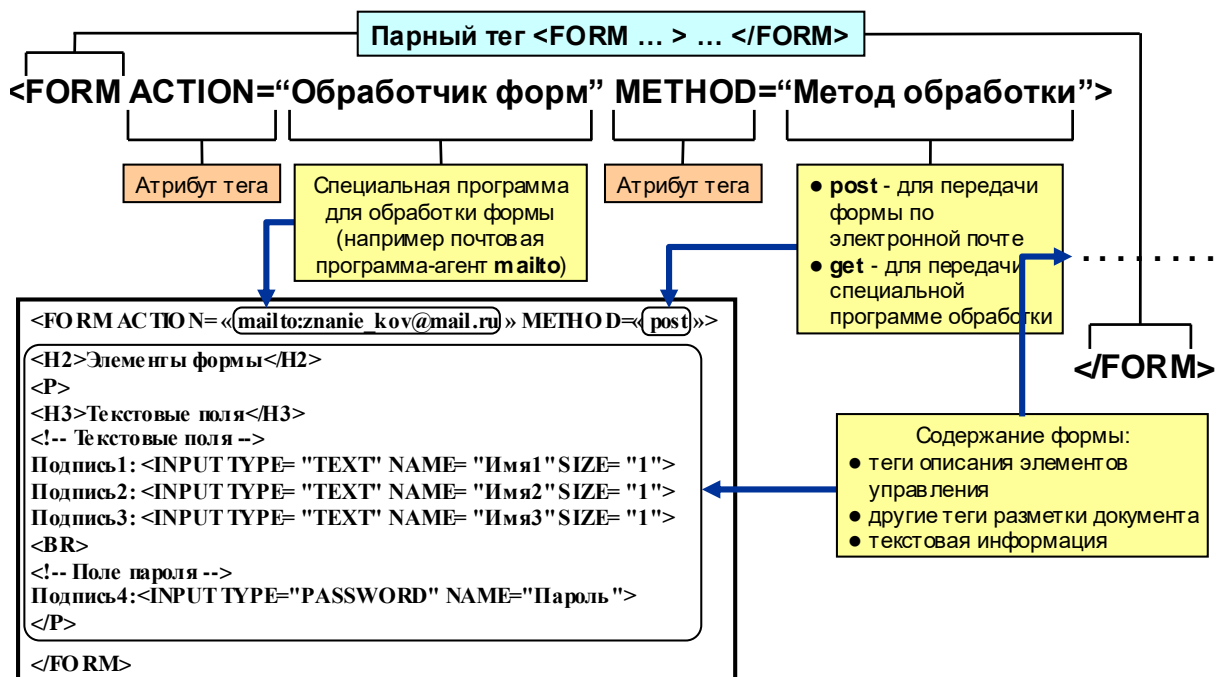


Рис. Пример формы на Web-странице

Форма - это блочный элемент, описываемый парным тегом `<FORM>`.

Атрибут **METHOD=** определяет способ передачи представленных пользователем данных. Он может иметь два значения. Значение "GET" указывает, что данные будут переданы программе (или сценарию) CGI. В этом случае атрибут ACTION= содержит адрес URL этой программы. Если указано **METHOD="POST"**, то данные, введенные пользователем, отправляются по электронной почте.

Внутри элемента формы располагаются поля формы. Они задаются при помощи различных тегов.

**Непарный тег `<INPUT>`** позволяет создавать различные элементы управления, в том числе текстовые поля и командные кнопки.

**Парный тег `<TEXTAREA>`** определяет текстовые области.

**Парный тег `<SELECT>`** позволяет создавать обычные и раскрывающиеся списки. Отдельные пункты задаются при помощи парного тега `<OPTION>`, который допустим только внутри данного элемента (закрывающий тег `</OPTION>` можно опускать).

Тег `<INPUT>` должен содержать обязательный атрибут **TYPE=**,

определяющий конкретный тип элемента управления.

Основные возможные значения этого атрибута:

- ✚ **"TEXT"** — создается текстовое поле;
- ✚ **"PASSWD"** — создается текстовое поле, но вводимая информация не отображается на экране («текстовое поле для ввода пароля»);
- ✚ **"CHECKBOX"** — создается флажок, который может быть установлен или сброшен;
- ✚ **"RADIO"** — создается переключатель (из группы переключателей может быть включен только один);
- ✚ **"SUBMIT"** — создается *кнопка отправки* формы,
- ✚ **"IMAGE"** — создается графическая кнопка отправки;
- ✚ **"RESET"** — создается кнопка очистки формы, щелчок на которой возвращает форму к ее исходному состоянию;

Организация передачи данных, введенных в форму, осуществляется следующим образом. Каждый из тегов, задающих поля формы, должен иметь атрибуты NAME= и VALUE=. Атрибут NAME= определяет имя поля формы, атрибут VALUE= — значение поля. Для текстового поля и текстовой области атрибут VALUE= приобретает значение, соответствующее содержимому этого поля, заданному пользователем для элемента SELECT атрибут VALUE= соответствует содержимому выбранного элемента OPTION. Все переключатели одной группы должны иметь одинаковые значения атрибута NAME= и разные значения атрибута VALUE=.

По щелчку на кнопке отправки данные из формы передаются в виде пар текущих значений атрибутов NAME= и VALUE=, соединенных знаком равенства. Информация о флажке передается только в том случае, если он установлен. Если атрибуты NAME= и VALUE= определены для использованной кнопки отправки, соответствующие данные также передаются (это позволяет включать в форму несколько кнопок отправки). Данные, поступающие в таком виде, удобны как для ручной, так и для автомата" ческой обработки.

**ЗАДАНИЕ 13.** Построить документ, содержащий форму, представленную выше.

Код формы приведен ниже.

```
<HTML>
  <HEAD><TITLE>From Sample</TITLE></HEAD>
  <FORM ACTION="ваш t-mail_адрес" METHOD="POST">
  <BODY>
    <H4>Моя анкета</H4>
    <P>Ваше имя:<BR><INPUT size=30 name=Имя><BR>
      Адрес:<BR><INPUT size=30 name=Адрес><BR>
      Телефон:<BR><INPUT size=10 name=Телефон>
    <P>Пол:<BR>
```

```

<INPUT type=radio value=Мужчина name=Выбор1> Мужчина<BR>
<INPUT type=radio value=Женщина name=Выбор1> Женщина
<P>Образование:<BR>
<INPUT type=checkbox value=Школа name=Выбор2> Школа<BR>
<INPUT type=checkbox value=Техникум name=Выбор2>
    Техникум<BR>
<INPUT type=checkbox value=Институт name=Выбор2>
Институт<BR>
<INPUT type=checkbox value=Аспирантура name=Выбор2>
    Аспирантура
<P>Знание иностранных языков <SELECT name=Язык>
    <OPTION selected>Английский
    <OPTION>Французский
    <OPTION>Немецкий
    <OPTION>Испанский
</OPTION></SELECT>
<P>Комментарий:<BR>
    <TEXTAREA name=Комментарий rows=7 cols=50> </TEXTAREA>
<P><INPUT type=submit value=Послать>
    <INPUT type=reset value=Очистить>
</FORM>
</P></BODY></HTML>

```

## СОЗДАНИЕ ЗАГОЛОВКА САЙТА

В большинстве случаев сайт состоит из трех составных частей

- Заголовка;
- Меню;
- Контента.

Заголовок, который нам предстоит создать представлен на рис.1.



Рис.1. Вид заголовка сайта

Заголовок будем выполнять, используя технологию создания таблиц. Общий вид заголовка в виде таблицы приведен на рис.2.

Ячейка 1 таблицы 1	Ячейка 2 таблицы 1	Ячейка 3 таблицы 1
--------------------	--------------------	--------------------

Рис.2. Общий вид таблицы заголовка

Приступим к созданию таблицы

### Шаг 1. Создание заголовочной части документа



```

<HTML>
<HEAD><TITLE>Общие сведения</TITLE>
<LINK href="style.css" rel=stylesheet>
</HEAD>
<BODY vLink=#154eaa aLink=#154eaa link=#154eaa bgColor=#ffffff>

```

**Шаг 2.** Создание общей таблицы, которая будет в дальнейшем заголовочной частью Web- страницы

```

<TABLE cellSpacing=0 cellPadding=0 width="100%" bgColor=#bec5d5 border=0>
<TR>
<TD style="BACKGROUND-REPEAT: no-repeat" align=middle width="32%"
height=92 background="logo.jpg" >
<TD vAlign=center width="68%" </TD>
</TR>
</TABLE>

```



**Шаг 3.** Создадим в ячейке 2 Таблицы 1 внутреннюю таблицу 2 состоящую из одной строки с двумя ячейками (см. рис.3.)

```

<TABLE cellSpacing=0 cellPadding=0 width="100%" bgColor=#bec5d5 border=0>
<TR>
<TD style="BACKGROUND-REPEAT: no-repeat" align=middle width="32%"
height=92 background="logo.jpg" >
<TD vAlign=center width="68%" </TD>
</TR>
</TABLE>

```

**Шаг 4.** В первую ячейку таблицы 2 вставим таблицу 3 с текстом, а во вторую ячейку картинку

```

<TABLE cellSpacing=0 cellPadding=0 width="100%" align=right border=0>
<TR>
<TD style="BACKGROUND-REPEAT: no-repeat" vAlign=center width="90%" background="headbgr1.jpg">
<TABLE cellSpacing=0 cellPadding=0 width="100%" border=0>
<TR>
<TD width="100%">
<DIV class=logo align=center>Московский государственный университет<BR>
имени<BR>М. <BR>В. <BR>Ломоносова
</DIV>
</TD>
</TR>
</TABLE>
</TD>
<TD vAlign=top align=left width=78 bgColor=#bec5d5 height=92>
<IMG height=92 src="headbgr2.jpg" width=77 border=0>
</TD>
</TR>
</TABLE>

```

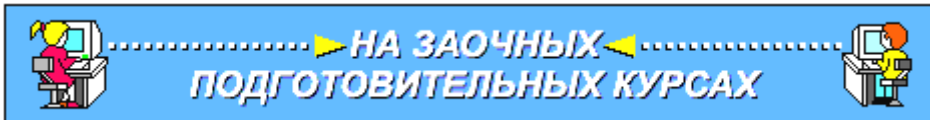
Тогда на шаге 4 мы будем иметь заголовочную часть документа, представленную на рис.4.







Во вторую ячейку первой строки вставлена анимационная картинка



Кнопки, определяющие факультеты института



Кнопки, определяющие ссылки на страницы сайта (пока не рабочие)



### Шаг 1. Заголовочная часть документа

```
<HTML><HEAD><TITLE>ИМПЭ</TITLE>  
<STYLE type=text/css>INPUT {FONT-SIZE: x-small; FONT-FAMILY: Arial}  
                                INPUT.but {FONT-SIZE: x-small; BACKGROUND-COLOR: transparent}  
</STYLE></HEAD>
```

### Шаг 2. Создание заголовка в виде таблицы, состоящей из одной строки и двух столбцов

```
6 <BODY bgColor=#ffffff leftMargin=1 topMargin=1 marginwidth="1" marginheight="1">  
7 <CENTER>  
8 <TABLE cellSpacing=0 cellPadding=0 width="100%" border=0>  
9 <TR valign=center align=middle>  
10 <TD width=63><IMG height=65 src="IMP/logoup.gif" width=200</TD>  
11 <TD width="100%" bgColor=#ccccff>  
12 <A href="start_class_distant.html">  
13 <IMG src="IMP/distant_classes.gif" border=0</A>  
14 </TD>  
15 </TR>
```

### Шаг 3. Добавить в таблицу еще одну строку из двух ячеек

```
<TR bgColor=#6666cc>  
<TD height=14><IMG src="IMP/logo_middle.gif" width=200</TD>  
<TD background="IMP/back_line.gif">  
<IMG height=14 src="IMP/blank.gif" width=100 border=0</TD>  
</TR>
```



## Шаг 4. Выполнить построение третьей строки

```
<TR>
  <TD vAlign=top align=right rowSpan=2>
    <IMG height=58 src="IMP/logo_down.gif" width=188></TD>
  <TD vAlign=top align=right>
    <TABLE cellSpacing=0 cellPadding=4 width="100%" border=0>
      <TR>
        <TD><A href="http://www.iile.ru/"><IMG height=24 alt="Главная страница"
          src="IMP/icon_home.gif" width=24 border=0>
          </A></TD>
      </TR></TABLE></TD>
</TR>
```

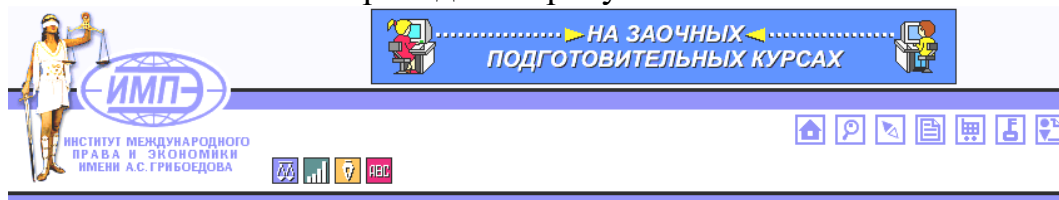


## Шаг 4. Подчеркивание

```
<TABLE cellSpacing=0 cellPadding=0 width="100%" border=0>
  <TR bgColor=#6666cc>
    <TD background="IMP/back_line.gif">
      <IMG height=14 src="IMP/blank.gif" width=100 border=0></TD>
  </TR>
</Table>
```



Итоговый заголовок приведен на рисунке



## СОЗДАНИЕ АНИМИРОВАННОГО БАННЕРА

Баннер - это картиночка, часто - анимированная, которая приглашает вас кликнуть по ней и зайти туда, куда просят. Основные форматы баннеров: **468x60**, **120x120**, **100x100** или **88x31** (в просторечии - кнопка). Все эти форматы мы будем условно называть одним словом - баннер.

Будем делать анимированный баннер размером 468x60 пикселей, к примеру, вот такой:



Отметим, что анимация складывается из последовательного показа двух статичных картинок (составляющих) баннера, а именно:



1.



2.

Сделаем одну составляющую баннера, а для этого откроем "Photoshop" Открыв программу, идем в меню "Файл" - "Создать". Откроется диалоговое окно (рис.1):

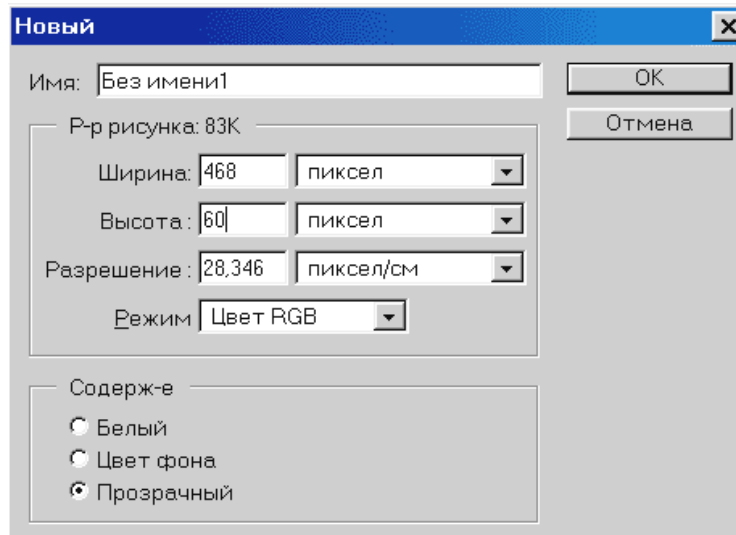


Рис.1

Выставляем требуемые ширину и высоту баннера, задаем цвет фона "прозрачный", нажимаем **ОК** и получаем первооснову нашей картинки (рис.2):

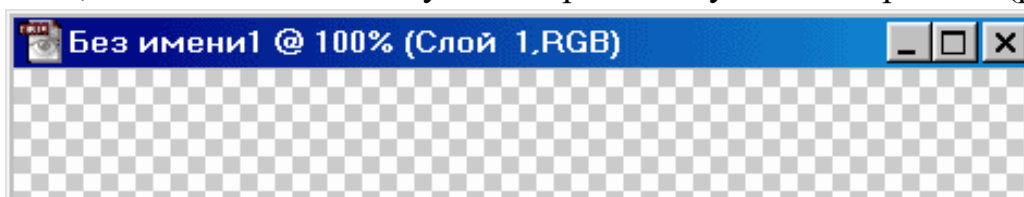


Рис.2

Выберем цвет для основной заливки подготовленной основы. Для этого щелкаем по левому окошку с цветом (всплывающую подсказку вы видите на рис.3), и перед нами открывается очередное диалоговое окно с выбором цвета (рис.4):

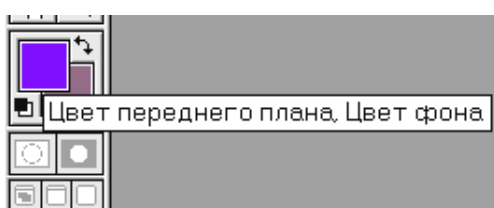


Рис.3

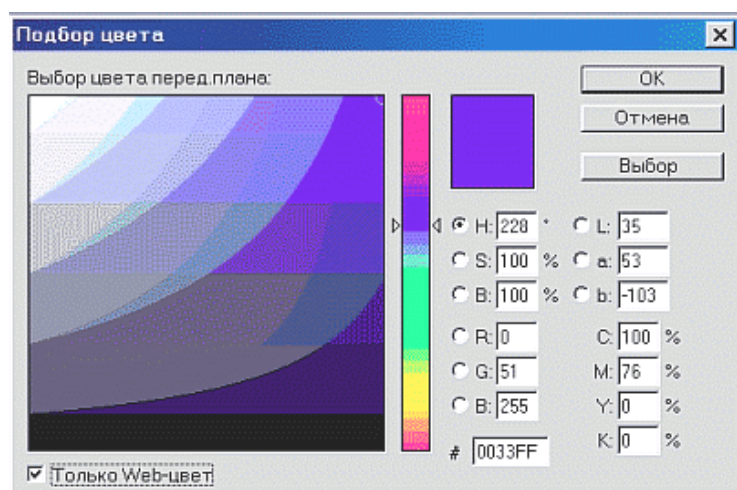


Рис.4

Щелкаем на понравившемся (а еще лучше - на нужном нам) оттенке цвета, и получаем искомый результат, показанный на рис.3.

При помощи инструмента "Ведро", показанном на рис.5 зальем созданную



Рис.5

нами "клетчатую" заготовку искомым синим цветом, для чего активизируем наше "ведро" (щелкнем по нему мышкой), подводим это ведерко к заготовке нашего будущего баннера, и снова щелкаем левой кнопкой мыши на заготовке (рис.6).

Теперь сохраним то, что мы создали: "Файл" - "Сохранить как...". В типах файлов мы увидим только одно расширение - \*.PSD, в нём и сохранимся.

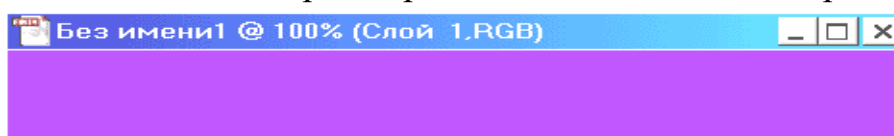


Рис.6

При помощи инструмента "Область прямоугольного выделения" (рис.7) выделяем на нашем синем прямоугольнике область определенной высоты, для чего мышкой аккуратно растягиваем эту область по всей ширине синего прямоугольника, после чего получим область, выделенную пунктиром (рис.8):

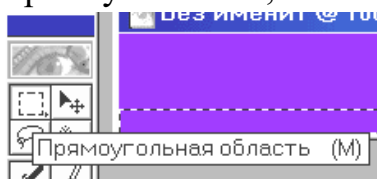


Рис.7

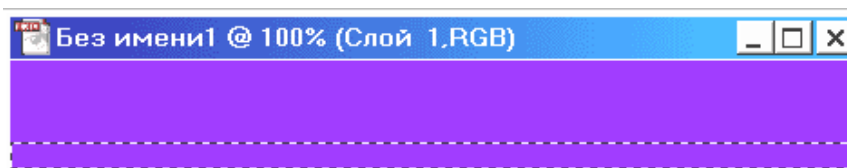


Рис.8

Если вы неправильно выделили область, или вам просто не понравилось, не огорчайтесь: идем в меню Фотошопа "Выделить" - "Снять выделение", после чего неудачная область исчезнет, и можно попробовать сделать заново. Зальем выделенную полосу желтым цветом (рис.9).

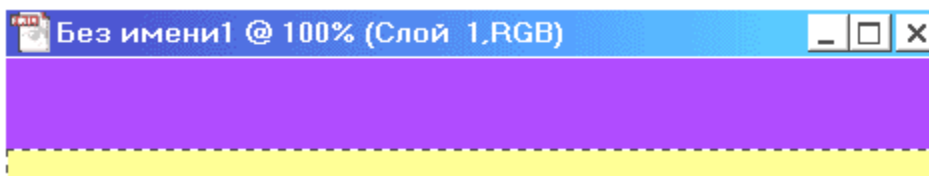


Рис.9

Выполнить команду: "Выделить" - "Снять выделение", после чего пунктирные линии исчезнут (рис.10):

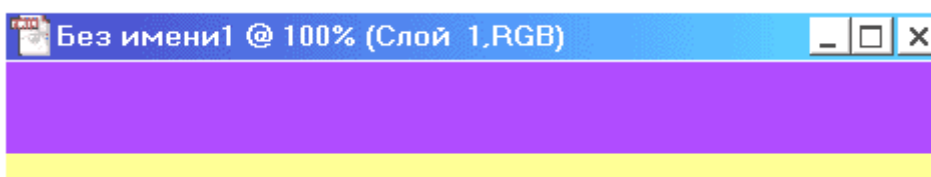


Рис.10

Сохраним то, что получилось, в формате \*PSD для этого просто выберем команду "Сохранить" в меню "Файл".

Поместим в заготовку баннера фотографию молодого человека вот с этой фотографии (рис.11):



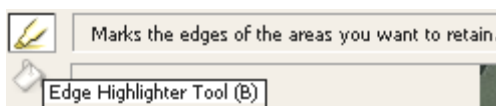
Рис.11



Рис.12

Нам нужна не вся фотография парня, а его изображение от головы до пояса без всякого фона.

1. Выполним команду **Filter/Extract**. Откроется окно редактора фильтра **Extract**



2. Маркер краев используем для обводки объекта. При обводке прихватываем как объект, так и фон.

На панели опций – **Brush Size** выбираем из следующего принципа: чем четче граница – тем тоньше линия маркера.

т.о. нам надо сделать замкнутую линию или довести до границы холста.

3. С помощью инструмента **Fill Tool** заливаем объект. **ОК**. Результат - на рис. 13.



Рис.13

**Итак**, мы сделали заготовку для нашего будущего баннера, прямоугольник синего цвета с желтой полоской размером 468x60 пикселей и фотографию молодого человека. Теперь нам нужно поместить в заготовку баннера фотографию юноши.

Изображение такого размера в наш прямоугольник с размерами 468 на 60 пикселей просто не влезет. Нужно уменьшить изображение ("Изображение" - "Размер изображения"). В открывшемся окне поставим галочку в "Ограничить пропорции" и зададим высоту нашей картинке согласно высоте баннера. Поставим 65 пикселей, чтобы потом был небольшой запас пространства для манёвра (рис.14).



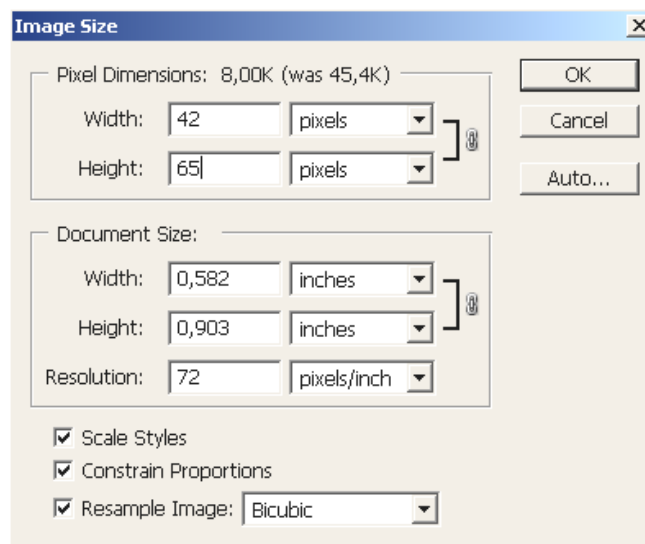


Рис.14

И, как всегда, нажимаем нашу любимую кнопку **ОК**. Что же произошло? Ничего страшного, разумеется. Просто наша картинка уменьшилась, и стала вот такой (рис. 15):

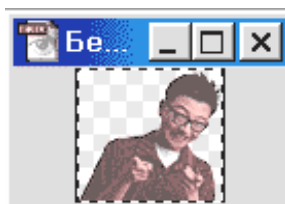


Рис.15

После изменения размеров картинки выделим изображение при помощи команд **"Выделить" - "Всё"**. Поскольку наша картинка имеет прозрачную подложку или фон (он же слой), то в дальнейшем, при помещении этой картинке в заготовку нашего баннера, сквозь прозрачный фон картинке будет великолепно "проглядывать" основной фон нашей заготовки-прямоугольника. Дальше при открытом файле с парнишкой мы дополнительно откроем файл с нашим сине-желтым прямоугольником. Теперь у нас открыты два файла. Переходим на картинку с парнишкой, и делаем то, о чем я говорил чуть выше : **"Выделить" - "Всё"**. Картинка обрамится характерной пунктирной рамкой (посмотрим еще раз на рис. 3). Затем комбинацией **"Правка" - "Вырезать"** скопируем парнишку в буфер обмена. Перейдем на рисунок с синим прямоугольником. Теперь снова знакомая комбинация: **"Правка" - "Вставить"**, и наш юноша послушно залезет в сине-желтый прямоугольник (рис. 16).

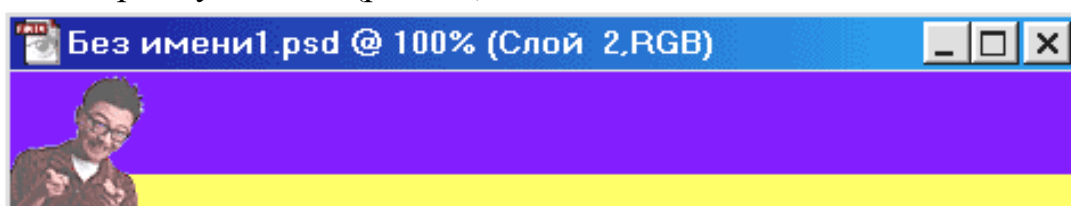


Рис.16

При помощи инструмента "Перемещение" (рис.17): мы можем мышью поставить картинку с юношей туда, куда захотим.



Рис. 17

Поместим эту картинку в правый угол заготовки. Теперь еще раз сохраним получившийся рисунок в формате \*PSD, Текст для баннера мы придумали, и теперь снова посмотрим в правый нижний угол интерфейса программы "Фотошоп". Щелкнем по опции "Слой" (крайняя левая кнопка окошка). На ней, и увидим следующее (рис. 18):

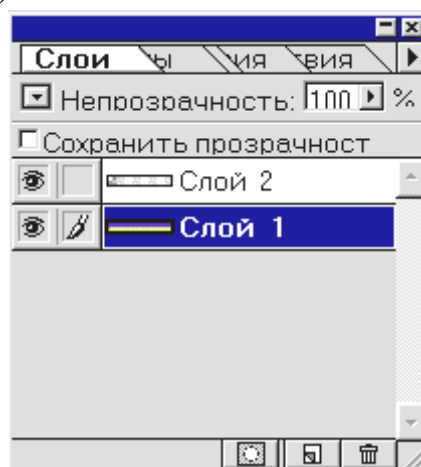


Рис.18

Первый слой - прозрачный, с фотографией молодого человека. Второй слой - сине-желтая заготовка для баннера. Теперь наша задача - добавить текст.



рис.19

Делаем активным слой с сине-желтым прямоугольником, для чего кликнем мышью на пиктограмме соответствующего слоя плавающего окна. Затем активизируем кнопку инструмента "Текст" (рис. 19) на панели инструментов "Фотошопа". Далее, подводим курсор мыши к нашему синему прямоугольнику, и щелкаем в этой области левой кнопкой мыши, после чего появится диалоговое окно параметров текста, где мы сможем выбрать тип шрифта, его размер, начертание и цвет (рис. 20):



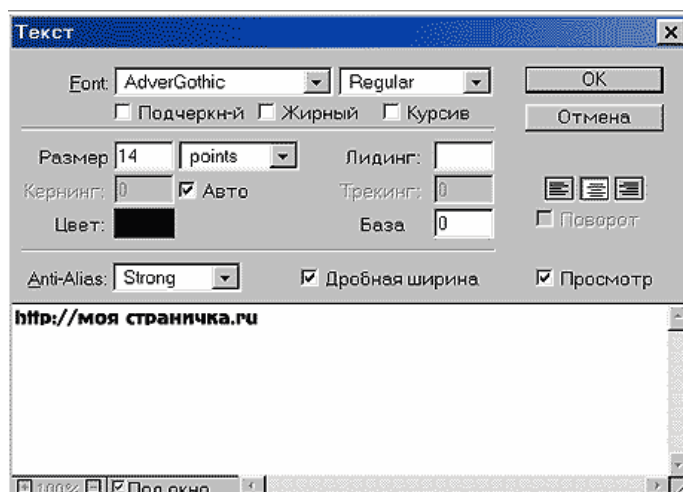


Рис.20

Наберем в русской раскладке аналогичный текст, и при помощи инструмента "Перемещение" отбуксируем его на желтую полоску. Теперь сделаем те же операции для синей области. Там мы введем текст: **"Не умеешь делать баннер ?"**. Сделаем цвет текста белым, размер кегля - на свой вкус, и при помощи инструмента "Перемещение" расположим его там, где нам удобно. Так же сделаем текст: **"Научим"**, только цвет сделаем красным, и поместим текст рядом с предыдущим. Вот что в итоге у нас должно получиться (рис. 21):



Рис.21

Сохраним полученное изображение в формате \*PSD, а также в формате \*GIF, для чего при помощи команды "Сохранить для WEB" выберем адаптивную палитру с 32 или 64 цветами (посмотрим на соотношение величины файла и качества картинки). Как видите, у нас уже получился статичный баннер в формате \*GIF "весом" в 7 Кб. Можно использовать и такой вариант, но мы хотели сделать баннер анимированный. Для этого сделаем вторую составляющую, для чего откроем наш файл, сохраненный в \*PSD, и активизировав слой с надписью "Моя страничка.ру", удалим его. Кликаем **правой** кнопкой мыши на активном слое, и в контекстном меню выбираем опцию "Удалить слой". Вот что у нас получилось в итоге (рис. 22) :



Рис.22

Сохраняем и это изображение в \*GIF.

Теперь все готово для анимации и размещении на сайте баннера.

### **Контрольные вопросы:**

1. Порядок создания фреймов?
2. Порядок создания форм?
3. Порядок создания бегущей строки?
4. Как поменять цвет фона?
5. Порядок создания динамического баннера?

### **ЗАКЛЮЧИТЕЛЬНАЯ ЧАСТЬ**

Подводятся итоги занятия по степени достижения поставленных целей занятия, активности обучаемых. При возникновении вопросов дается краткий ответ или же назначается время консультации. Объявляются оценки за занятие. Объявляется тема следующего занятия. Выдается задание на самостоятельную работу. Оформляется журнал учебной группы.

#### **Задание на самостоятельную работу:**

1. Изучить и доработать вопросы занятия по рекомендованной литературе.
2. Темой следующего практического занятия №3 будет «Исследование взаимодействия HTML, CSS и JavaScript».

Преподаватель кафедры С-20  
кандидат технических наук

Р. Татарников