

# Internet- технологии

ЛЕКЦИЯ №3  
ЯЗЫК HTML. СОЗДАНИЕ ФОРМ И УПРАВЛЯЮЩИХ  
ЭЛЕМЕНТОВ

# Содержание

```
<!DOCTYPE HTML>
<html>
<head>
  <title>Содержание лекции №3</title>
  <meta charset="utf-8">
</head>
<body background="back1.jpg">
  <h2><i>Содержание</i></h2>
  <ul>
    <li>Использование графики в веб.
Баннеры, фон.</li>
    <li>Виды компьютерной графики.
<br>Форматы графических файлов.</li>
    <li>Изображения, карты изображений.</li>
    <li>Формы и управляющие элементы.</li>
    <li>Фреймы.</li>
    <li>Особенности HTML5.</li>
  </ul>
</body>
</html>
```

## Содержание

- Использование графики в веб. Баннеры, фон.
- Виды компьютерной графики. Форматы графических файлов.
- Изображения, карты изображений.
- Формы и управляющие элементы.
- Фреймы.
- Особенности HTML5.



# Использование графики в веб

Правильно подобранная и размещенная на Web-странице графика облегчает восприятие контента и делает страницу визуально привлекательней.

Направления использования графики:

- различные рисунки, диаграммы и графики;
- фон страницы и оформление ее элементов;
- рекламные объявления (банеры);
- логотипы и т.д.

Например, некоторые изображения с официального сайта университета:



**Думай смело, креативно!**

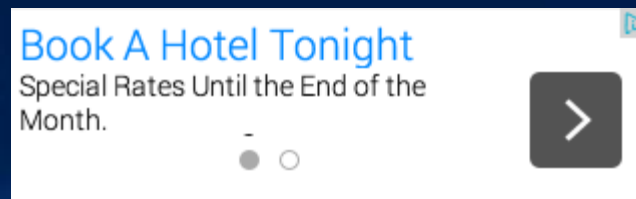
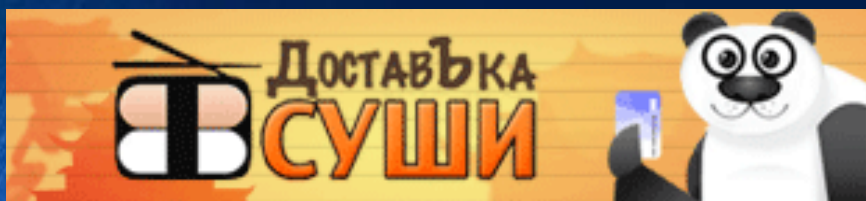




# Использование графики в веб. Баннеры

**Баннер** (англ. banner – транспарант) – графическое изображение рекламного характера, являющееся гиперссылкой на сайт рекламодателя или страницу с дополнительной информацией. Один из преобладающих форматов интернет-рекламы.

Частым явлением являются веб-страницы, направленные исключительно на получение прибыли от баннерной рекламы.



# Использование графики в веб. Баннеры

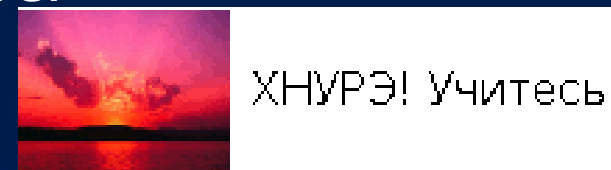
Можно выделить следующие типы баннеров:

**Статичные баннеры** – одиночное графическое изображение.

**GIF-баннеры** – представляют собой последовательность кадров, которые сменяют друг друга. Смена кадров происходит последовательно с учетом запрограммированной задержки каждого кадра. Плавность движений в таком баннере может достигаться только за счет множества промежуточных кадров, что дает визуальный эффект движения, но это существенно увеличивает объем баннера.

**Flash-баннеры** – используют векторную графику, что позволяет получать анимационные эффекты при небольшом размере баннера. Также технология Flash предусматривает возможность создания интерактивного баннера.

Типичные баннеры, созданные на лабораторной работе:



# Использование графики в веб. Фон

Одним из наиболее важных аспектов графического оформления сайта является использование **фона**. Для создания фона, состоящего из одного сплошного цвета, используется атрибут **bgcolor** со hex-значением цвета. Для использования изображения в качестве фона используется атрибут **background** с адресом фонового рисунка. *Данные атрибуты устарели, используйте CSS!*

```
body {  
    background-image: url(background.png);  
    background-position: left top;  
    background-repeat: repeat-x;  
}
```



## Содержание

- Использование графики в веб. Баннеры, фон.
- Виды компьютерной графики. Форматы графических файлов.
- Изображения, карты изображений.
- Формы и управляющие элементы.
- Фреймы.
- Особенности HTML5.

Источником фона может служить рисунок небольшого размера, тиражированный так, чтобы заполнить все пространство окна.

Фон не должен снижать контрастность страницы и мешать чтению текста.





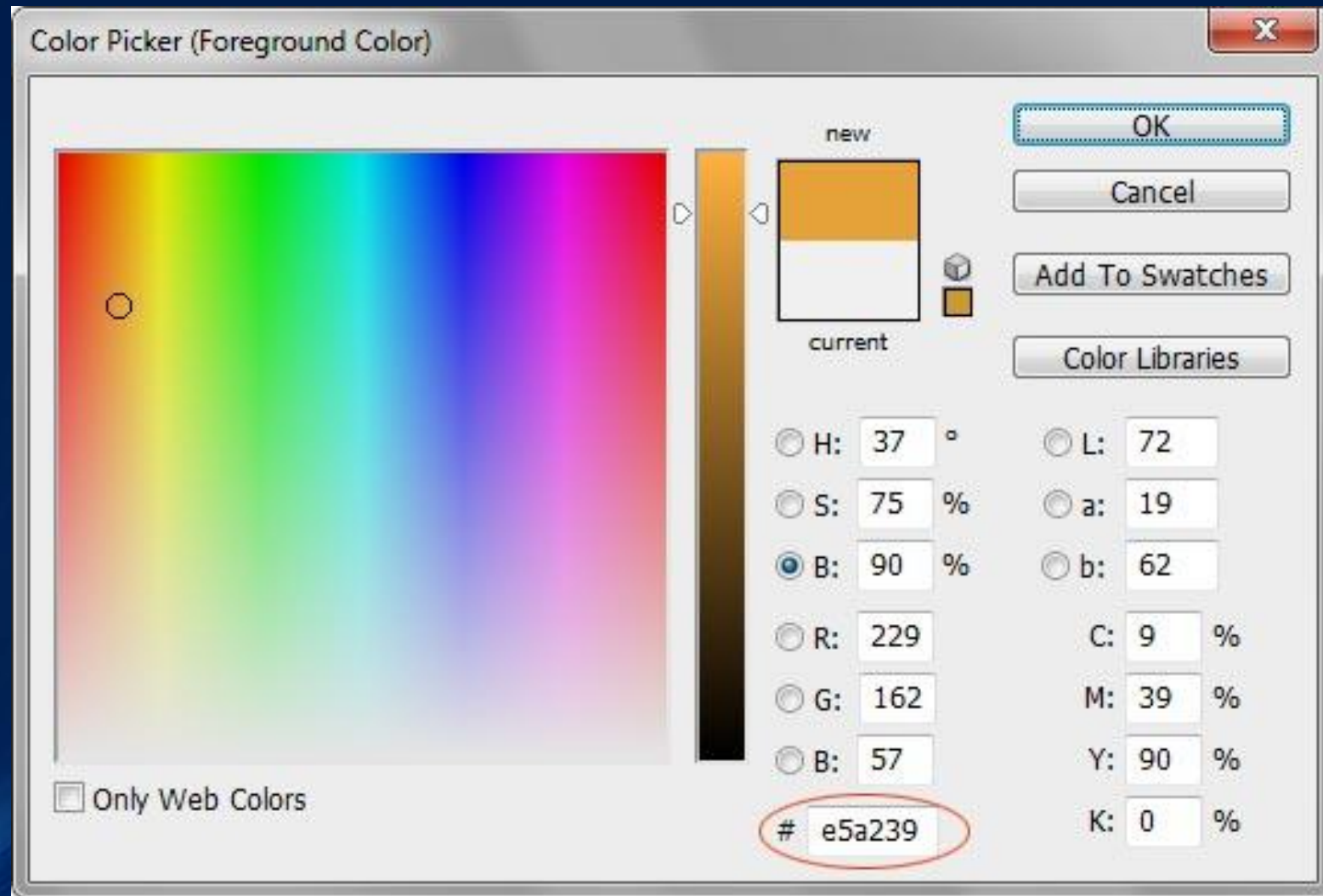
# Указание цвета

Веб-цветами называются такие цвета, для каждой составляющей – красной, зелёной и синей, устанавливается одно из шести значений – 0 (00), 51 (33), 102 (66), 153 (99), 204 (CC), 255 (FF). Общее количество веб-цветов 216. Пример веб-цвета – #33ff66.

Имя цвета	Цвет	Описание	Шестнадцатеричное значение
black		Черный	#000000
gray		Серый	#808080
green		Зеленый	#008000
navy		Темно-синий	#000080
olive		Оливковый	#808000
purple		Фиолетовый	#800080
red		Красный	#ff0000
silver		Светло-серый	#c0c0c0
teal		Сине-зеленый	#008080
white		Белый	#ffffff



# Указание цвета





# Виды компьютерной графики

Критерий сравнения	Растровая графика	Векторная графика
Способ представления изображения	Растровое изображение строится из <b>множества пикселей</b> .	Описывается в виде последовательности команд (набора <b>графических примитивов</b> ).
Области применения	Используется для представления реальных образов.	Схемы, карты, шрифты, логотипы.
Качество при редактировании изображения	При масштабировании и вращении возникают искажения	Векторные изображения могут быть легко преобразованы без потери качества.



# Графические форматы

## Растровые:

BMP, JPEG, GIF, PNG, APNG, ICO, TIFF, PCX, Raw, TGA, WBMP, XCF, WebP.



## Векторные:

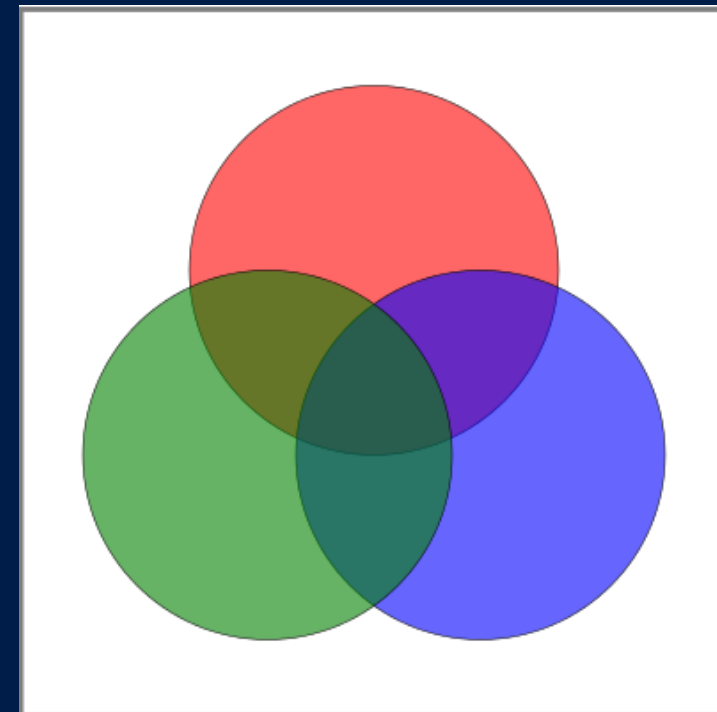
SVG, WMF, SWF, AI, CDR, EMF, PS, EPS, XPS.



# Векторная графика. Пример

Пример svg-рисунка:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<svg version = "1.1"
  baseProfile="full"
  xmlns = "http://www.w3.org/2000/svg"
  xmlns:xlink = "http://www.w3.org/1999/xlink"
  xmlns:ev = "http://www.w3.org/2001/xml-events"
  height = "400px" width = "400px">
  <rect x="0" y="0" width="400" height="400"
    fill="none" stroke="black" stroke-width="5px" stroke-opacity="0.5"/>
  <g fill-opacity="0.6" stroke="black" stroke-width="0.5px">
    <circle cx="200px" cy="200px" r="104px" fill="red" transform="translate( 0,-52)" />
    <circle cx="200px" cy="200px" r="104px" fill="blue" transform="translate( 60, 52)" />
    <circle cx="200px" cy="200px" r="104px" fill="green" transform="translate(-60, 52)" />
  </g>
</svg>
```



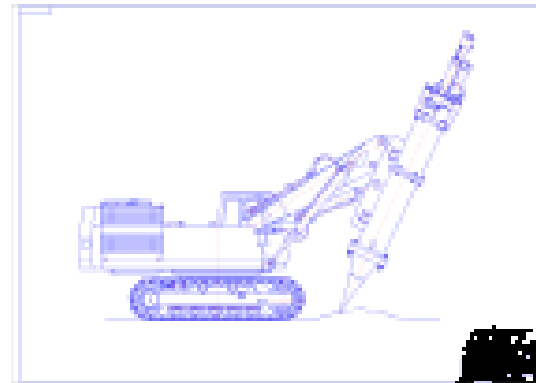


# Использование изображений в веб

Для вставки изображения используются одинарные теги `<img>`. Адрес файла с картинкой задаётся через атрибут `src`. Другим обязательным атрибутом является атрибут `alt`, который устанавливает альтернативный текст для изображений, позволяющий получить текстовую информацию о рисунке при отключенной в браузере загрузке изображений, а также поисковыми системами.

Если необходимо, то рисунок можно сделать ссылкой, поместив `<img>` в контейнер `<a>`.

```
<a href="index.php"></a>
```



file:///C:/Documents and Settings/IGS/Мои документы/index.php



# Карты изображений

Рисунки также могут применяться в качестве карт-изображений. Такая карта по внешнему виду ничем не отличается от обычного изображения, но при этом оно может быть разбито на невидимые области произвольной формы, где каждая из зон служит отдельной ссылкой.

Элемент `<map>` служит контейнером для элементов `<area>`, которые определяют активные области для карт-изображений. Связь элемента `<img>` с клиентской картой-изображением выполняется при помощи единого идентификатора как в `<img>`, задаваемого атрибутом `usemap`, так и в `<map>`, устанавливаемого атрибутом `name`.

Синтаксис:

```

```

```
<map name="mapid">
```

```
  <area href="path1.html" shape="rect" coords="12,34,56,78">
```

```
</map>
```



# Карты изображений. Пример

```
  
<map name="hmap">  
  <area href="path1.html" shape="rect"  
coords="12,34,73,103">  
  <area href="path2.html" shape="poly"  
coords="100,100,160,100,140,40">  
  <area href="path3.html" shape="circle"  
coords="240,80,45">  
</map>
```



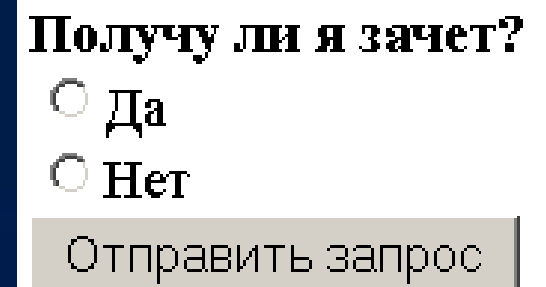


# Формы и управляющие элементы

Форма – это инструмент, с помощью которого HTML-документ может послать обработчику некоторую информацию (в том числе и введенную пользователем) в виде пар «имя=значение».

Форма характеризуется **адресом обработчика** (адрес программы на веб-сервере, которая будет обрабатывать содержимое данных формы) и включает в себя набор **элементов формы**, которые представляют собой стандартные поля для ввода информации. В частности, кнопка отправки данных формы на сервер (кнопка SUBMIT).

```
<form action="handler.php">  
  <b>Получу ли я зачет?</b><br>  
  <input type="radio" name="answer" value="a1">Да<br>  
  <input type="radio" name="answer" value="a2">Нет<br>  
  <input type="submit">  
</form>
```



Получу ли я зачет?

☐ Да

☐ Нет

Отправить запрос



# Формы. Атрибуты тега <form>

Атрибут	Описание
accept-charset	Устанавливает кодировку, в которой сервер может принимать и обрабатывать данные.
<b>action</b>	Адрес программы или документа, который обрабатывает данные формы. Различают два метода – get и post.
autocomplete	Включает автозаполнение полей формы.
enctype	Способ кодирования данных формы.
<b>method</b>	Метод протокола HTTP.
name	Имя формы.
novalidate	Отменяет встроенную проверку данных формы на корректность ввода.
target	Имя окна или фрейма, куда обработчик будет загружать возвращаемый результат.



# Формы. Методы отправки

Значения атрибута method:

**get** – предназначен для передачи данных в **адресной строке**. Пары «имя=значение» присоединяются в этом случае к адресу после вопросительного знака и разделяются между собой амперсандом (символ &).  
handler.php?answer=a1&name=SomeName

**post** – посылает на сервер данные в HTTP-запросе браузера. Это позволяет отправлять большее количество данных, чем доступно методу get, поскольку у get установлено ограничение в 4 Кб. Большие объёмы данных используются в почтовых службах, при заполнении базы данных, пересылке файлов и др.





# Формы и управляющие элементы

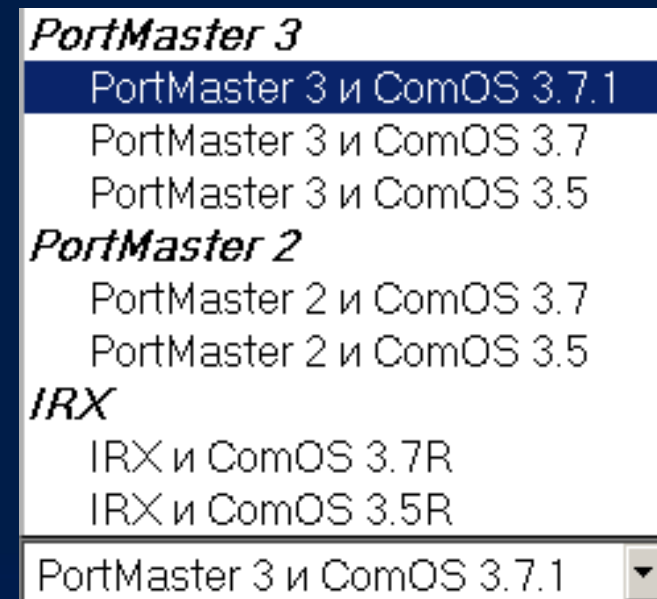
Тег	Описание
<b>&lt;form&gt;</b>	Устанавливает форму на веб-странице.
<b>&lt;button&gt;</b>	Создаёт на веб-странице кнопку.
<b>&lt;input&gt;</b>	Позволяет создавать разные элементы интерфейса.
<b>&lt;textarea&gt;</b>	Создаёт поле для многострочного текста.
<b>&lt;keygen&gt;</b>	Используется для генерации ключей.
<b>&lt;fieldset&gt;</b>	Предназначен для группирования элементов формы.
<b>&lt;label&gt;</b>	Устанавливает метку для элемента формы.
<b>&lt;legend&gt;</b>	Создаёт заголовок группы элементов формы.
<b>&lt;select&gt;</b>	Создаёт раскрывающийся список.
<b>&lt;option&gt;</b>	Определяет отдельные пункты списка, создаваемого с помощью контейнера select.
<b>&lt;optgroup&gt;</b>	Группирует элементы списка.



## Управляющие элементы. <select>

<select> позволяет создать элемент интерфейса в виде **раскрывающегося списка**, а также список с одним или множественным выбором. Каждый пункт создаётся с помощью элемента <option>, который должен быть вложен в контейнер <select>. Если планируется отправлять данные списка на сервер, то требуется поместить <select> внутри формы.

```
<select name="ComOS">
  <optgroup label="PortMaster 2">
    <option label="3.7" value="pm2_3.7">PortMaster 2 и ComOS 3.7
  </option><option label="3.5" value="pm2_3.5">PortMaster 2 и ComOS 3.5
  </option>
</optgroup>
<optgroup label="IRX">
  <option label="3.7R" value="IRX_3.7R">IRX и ComOS 3.7R
  </option><option label="3.5R" value="IRX_3.5R">IRX и ComOS 3.5R
  </option>
</optgroup>
</select>
```



# Управляющие элементы. <input>. Типы

Тип	Описание	Вид
text	Текстовое поле. Предназначено для ввода символов с помощью клавиатуры.	<input type="text"/>
button	Кнопка.	<input type="button" value="Кнопка"/>
checkbox	Флажки. Позволяют выбрать более одного варианта из предложенных.	<input type="checkbox"/> Чай <input type="checkbox"/> Кофе
file	Поле для ввода имени файла, который пересылается на сервер.	<input type="text"/> <input type="button" value="Обзор..."/>
hidden	Скрытое поле. Оно никак не отображается на веб-странице.	<input type="hidden"/>
password	Скрытое текстовое поле, все символы показываются звездочками.	<input type="password"/>
radio	Переключатели. Используются, когда следует выбрать один вариант из нескольких предложенных.	<input type="radio"/> Чай <input type="radio"/> Кофе
reset	Кнопка для возвращения данных формы в первоначальное значение.	<input type="button" value="Сброс"/>
submit	Кнопка для отправки данных формы на сервер.	<input type="button" value="Отправить запрос"/>





# Управляющие элементы. `<input>`. Новые типы

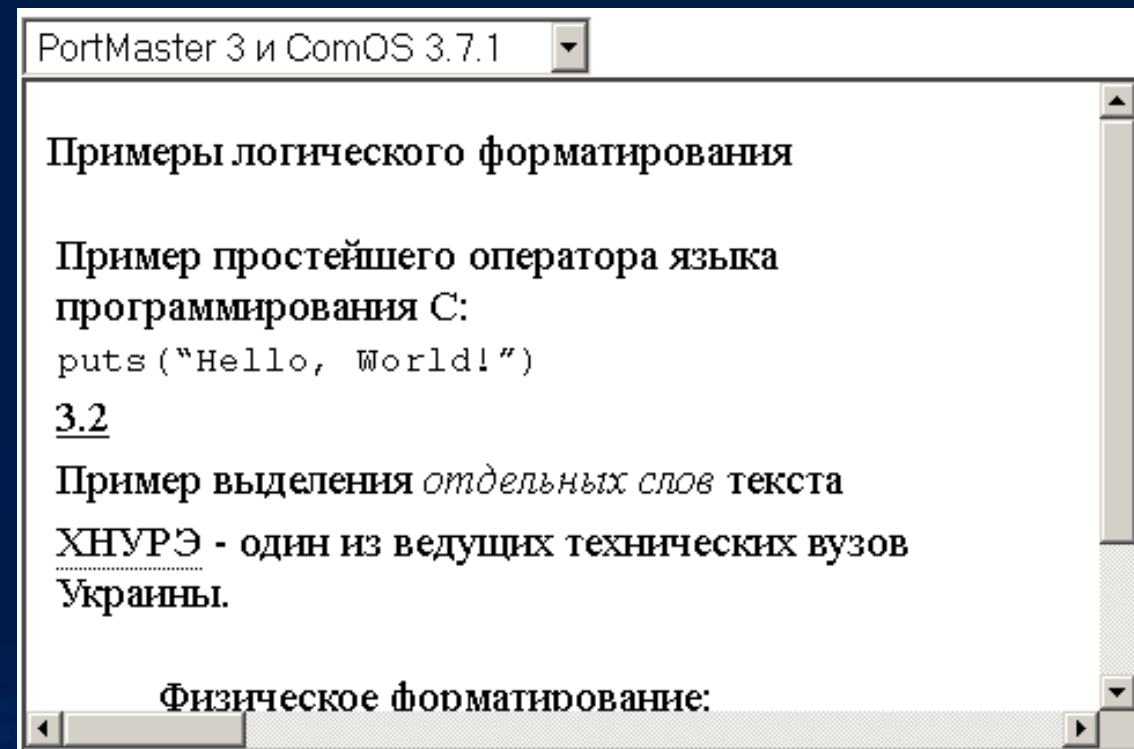
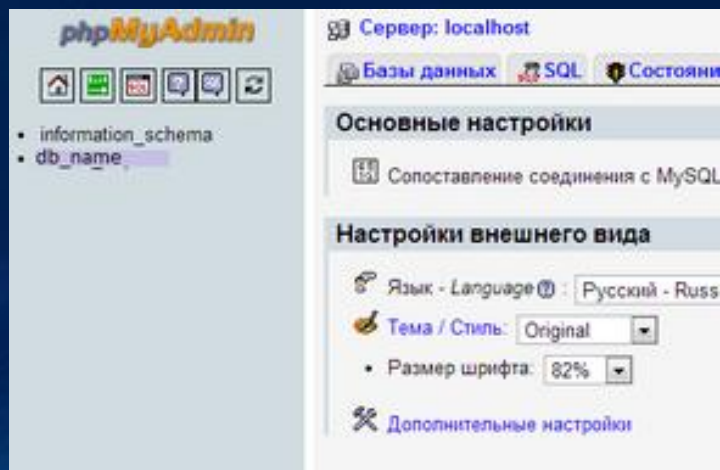
Тип	Описание
color	Виджет для выбора цвета.
date	Поле для выбора календарной даты.
datetime	Указание даты и времени.
datetime-local	Указание местной даты и времени.
email	Для адресов электронной почты.
number	Ввод чисел.
range	Ползунок для выбора чисел в указанном диапазоне.
search	Поле для поиска.
tel	Для телефонных номеров.
time	Для времени.
url	Для веб-адресов.
week	Выбор недели.



# Фреймы

Фреймы разделяют окно браузера на отдельные области, расположенные рядом друг с другом. В каждую из таких областей загружается отдельная веб-страница. Практически вышли из употребления, за исключением плавающего (встроенного) фрейма, который можно добавлять в любое место веб-страницы с помощью тега `<iframe>`.

```
<iframe src="Lb1_1.html" width="400"
height="240">Ваш браузер не
поддерживает встроенные фреймы!
</iframe>
```



# HTML5

Некоторые отличия HTML5:

- Изменён синтаксис.
- Встраивание SVG и MathML в text/html.
- Новые элементы: **<article>**, **<aside>**, **<audio>**, **<canvas>**, **<datalist>**, **<footer>**, **<header>**, **<hgroup>**, **<keygen>**, **<nav>**, **<progress>**, **<section>**, **<source>**, **<summary>**, **<time>**, **<video>** и др.
- Новые элементы управления.
- Элементы, которые будут исключены: **<acronym>**, **<applet>**, **<basefont>**, **<big>**, **<center>**, **<dir>**, **<font>**, **<frame>**, **<frameset>**, **<isindex>**, **<noframes>**, **<strike>**, **<tt>**
- Новые атрибуты: **charset** (в **<meta>**), **async** (в **script**).
- Пользовательские атрибуты.



# Вопросы

- Способы указания фона для веб-документа.
- Недостатки использования растровой графики.
- Форматы графических файлов для векторной графики.
- Что такое карта изображений?
- Для чего используются формы?
- Как устроены формы?
- Назовите основные управляющие элементы.
- Отличия использования элементов `<input type="button">` и `<button>`.
- Недостатки использования фреймов. Плавающие фреймы.

