

# WEB-СТРАНИЦЫ. ЯЗЫК HTML

# ЧТО ТАКОЕ WEB-СТРАНИЦЫ?

**WWW (World Wide Web)** – «всемирная паутина», служба Интернет для обмена информацией в виде гипертекста (и гипермедиа).

**Web-страница** – текстовый файл, в котором описано размещение материала на экране.



**Гипертекст** – текст, содержащий гиперссылки.

**Гиперссылка** – «активная»ссылка на другой документ.

**Гипермедиа-документ** – документ, включающий текст, рисунки, звуки, видео, в котором любой элемент может быть гиперссылкой.

**Браузер** – программа для просмотра Web-страниц на экране (*Internet Explorer, Mozilla Firefox, Opera*).

# Какие бывают Web-страницы?

- **статические** – существуют на сервере в виде готовых файлов: **\*.htm, \*.html**
- **динамические** – полностью или частично создаются на сервере в момент запроса (выбор информации из базы данных)  
**\*.shtml, \*.asp, \*.pl, \*.php**
-  позволяют выбирать информацию из базы данных по заранее неизвестным запросам
-  дополнительная нагрузка на сервер
- загружаются медленнее

# Язык HTML

HTML = *Hypertext Markup Language*  
(язык разметки гипертекста)

**!** HTML – это не язык программирования!

HTML-страница – это текстовый файл (*Блокнот*):

index.html

```
<HTML>
<HEAD>
<TITLE>Моя страница</TITLE>
</HEAD>
<BODY>
Привет!
...
</BODY>
</HTML>
```

images



sail.jpg



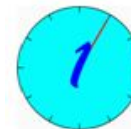
man.jpg



sunset.jpg



bee.wav



clock.avi



ball.swf

# Тэги

**Тэг** – это команда языка HTML, которую выполняет браузер:

- **непарные** тэги

вставить  
рисунок

```
<IMG SRC = "vasya.jpg">
```

- **парные** тэги (*контейнеры*)

открывающий

```
<TABLE>  
...  
</TABLE>
```

область действия  
тэга: описание  
таблицы

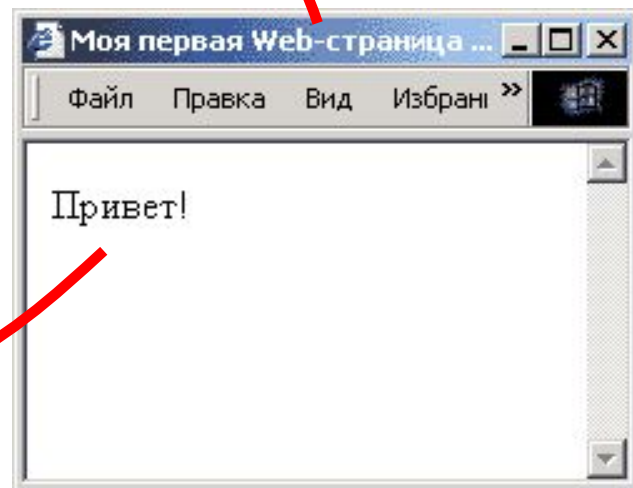
закрывающий

# Простейшая Web-страница

`first.html`

шапка («голова»)

```
<HTML>
<HEAD>
<TITLE>Моя первая
  Web-страница</TITLE>
</HEAD>
<BODY>
Привет!
</BODY>
</HTML>
```



основная часть  
(«тело»)

# Заголовки: H1 ... H6

<BODY>

<H1>Заголовок документа</H1>

<H2>Заголовок раздела</H2>

<H3>Заголовок подраздела</H3>

<H4>Заголовок параграфа</H4>

<H5>Комментарий</H5>

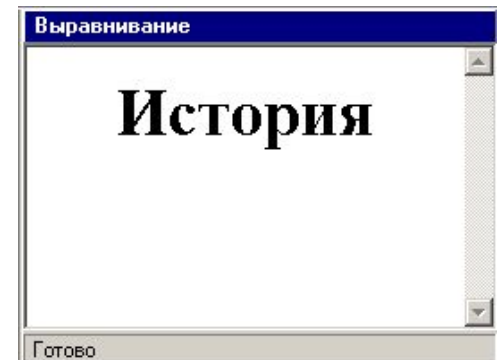
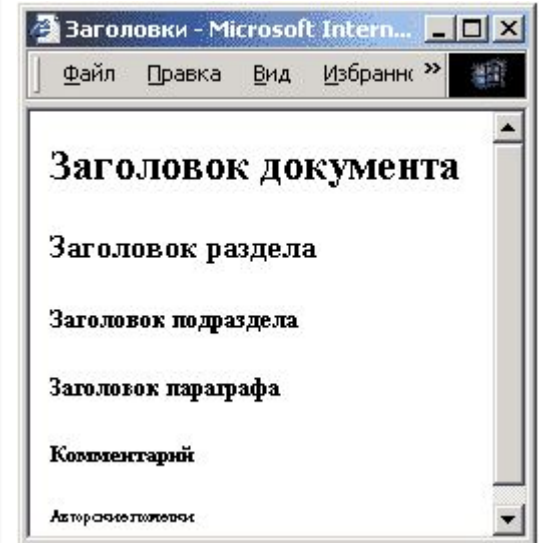
<H6>Авторские пометки</H6>

</BODY>

**выравнивание:**

left,  
center,  
right

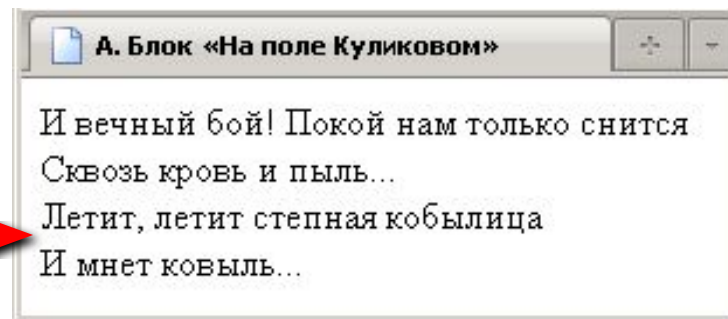
<H1 **ALIGN**=center>История</H1>



# Абзацы

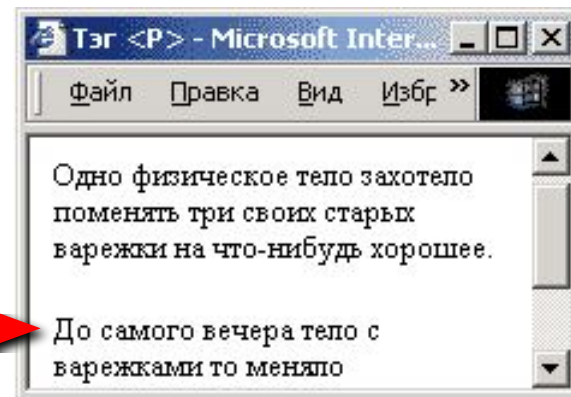
- переход на новую строку

И вечный бой! Покой  
нам только снится<BR>  
Сквозь кровь и пыль...<BR>  
Летит, летит степная  
кобылица<BR>  
И мнет ковыль...



- абзац (с отступами)

<P>  
Одно физическое тело захотело  
поменять три своих старых варежки  
на что-нибудь хорошее.  
</P>  
<P>  
До самого вечера тело с  
варежками ...  
</P>





# Выравнивание

атрибут тэга <P>

```
<P ALIGN="center">
```

Этот текст выровнен по центру.

```
</P>
```

```
<P ALIGN="justify">
```

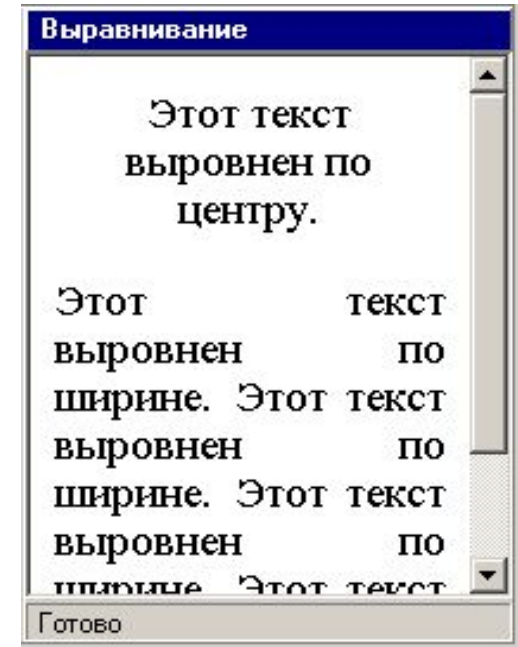
Этот текст выровнен по ширине.

Этот текст выровнен по ширине.

Этот текст выровнен по ширине.

Этот текст выровнен по ширине.

```
</P>
```



<b>left</b>	по левой границе
<b>right</b>	по правой границе
<b>center</b>	по центру
<b>justify</b>	по ширине



**Не используйте выравнивание по ширине для узких столбцов!**

# Специальные символы

<b>Символ</b>	<b>HTML-код</b>	<b>Название</b>
–	&mdash;	(длинное) тире
	&nbsp;	неразрывный пробел
§	&sect;	параграф
©	&copy;	символ авторского права
«	&laquo;	левая русская кавычка
»	&raquo;	правая русская кавычка
®	&reg;	зарегистрированная торговая марка
°	&deg;	градус
2	&sup2;	квадрат
3	&sup3;	куб
1/4	&frac14;	четверть
1/2	&frac12;	половина
3/4	&frac34;	три четверти
×	&times;	знак умножения
÷	&divide;	знак деления

# Специальные символы

А.С.&nbsp;Пушкин&nbsp;&mdash; солнце  
русской поэзии.

Из дома вышел А.С.  
Пушкин – солнце  
русской поэзии.

Вышел А.С. Пушкин  
– солнце русской  
поэзии.

Дом сдали в 2011&nbsp;году.

Пёс весил 12&nbsp;кг.

# Маркированные списки

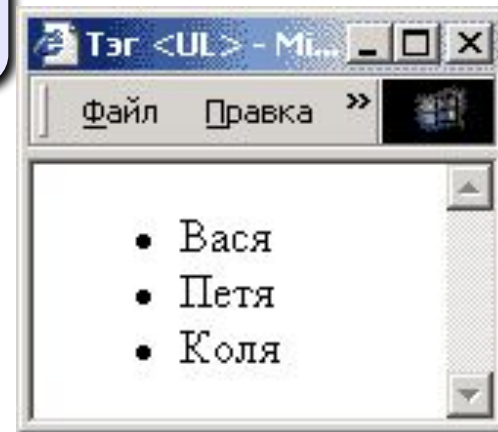
*unordered list*

(неупорядоченный список)

*list item*

(элемент списка)

```
<UL>  
<LI>Вася  
<LI>Петя  
<LI>Коля  
</UL>
```



изменение маркера:

```
<UL TYPE="disc">  
...  
</UL>
```

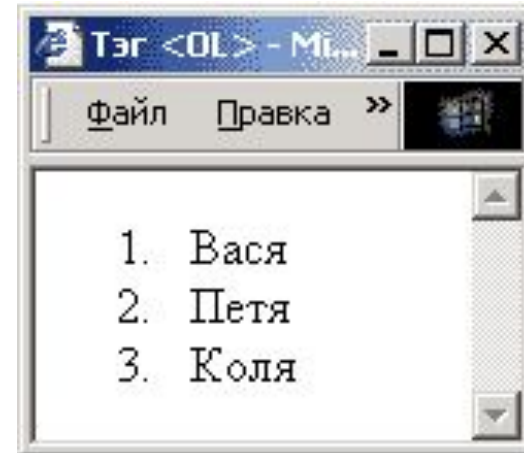
**disc**     ·  
**circle**    ○  
**square**    ■

# Нумерованные списки

*ordered list*

(упорядоченный список)

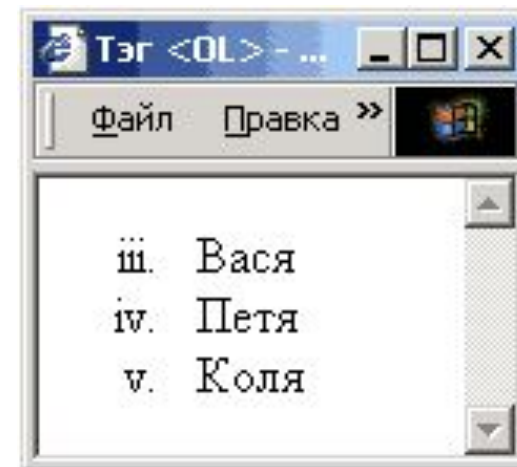
```
<OL>  
<LI>Вася  
<LI>Петя  
<LI>Коля  
</OL>
```



изменение нумерации:

1, i, l, a, A

```
<OL TYPE=i  
START=3>  
...  
</OL>
```



# Многоуровневые списки

```
<UL>
```

```
<LI>Города России
```

```
<OL>
```

```
<LI>Москва
```

```
<LI>Санкт-Петербург
```

```
</OL>
```

```
<LI>Города Украины
```

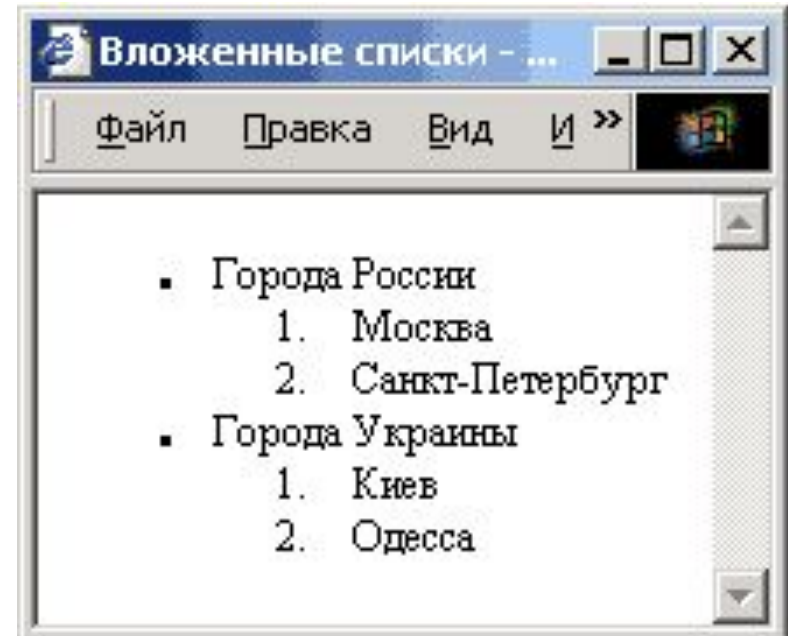
```
<OL>
```

```
<LI>Киев
```

```
<LI>Одесса
```

```
</OL>
```

```
</UL>
```

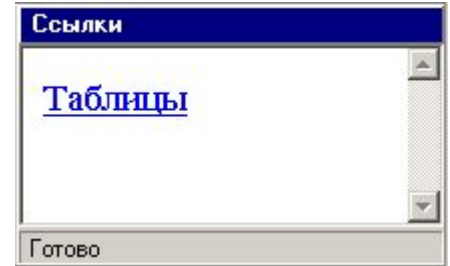


# Ссылки на другие страницы сайта

- страница в той же папке

*anchor* (якорь)

```
<A HREF="table.htm">Таблицы</A>
```



*hyper reference*  
(гиперссылка)

- страница во вложенной папке

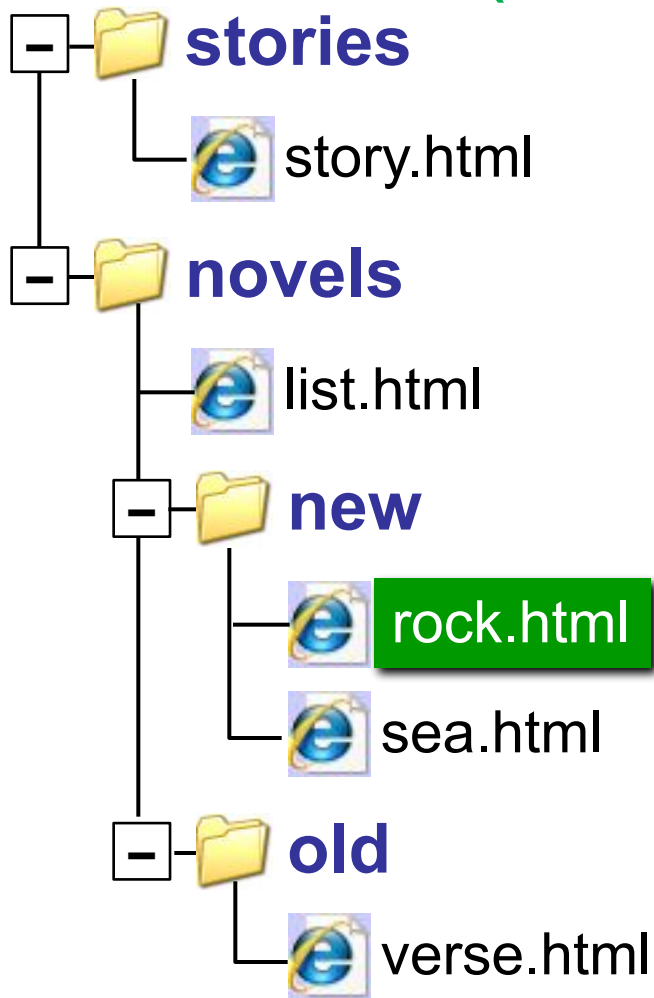
```
<A HREF="example/ex1.htm">Пример</A>
```

- страница в соседней папке

```
<A HREF=" ../texts/text1.htm">Текст</A>
```

ВЫЙТИ ИЗ ТЕКУЩЕЙ ПАПКИ

## (ссылки из файла *rock.html*)





# Ссылки на другие сайты

- на главную страницу сайта

```
<A HREF="http://www.mail.ru">Почта</A>
```

index.htm, index.html, default.asp, ...

- на конкретную страницу сайта (URL)

```
<A HREF="http://www.vasya.ru/text/a.htm">  
Васин текст</A>
```

- на файл для скачивания

```
<A HREF="http://www.vasya.ru/prog.zip">  
Скачать</A>
```

# Ссылки внутри страницы

```
<A NAME="up"></A>
```

переход на метку

```
<A HREF="#chap1">Глава 1</A><br>
```

```
<A HREF="#chap2">Глава 2</A><br>
```

```
<A NAME="chap1"></A>
```

```
<H1>Глава 1</H1>
```

Это текст главы 1. Это текст главы 1.

метка (якорь)

Это текст главы 1. Это текст главы 1.<BR>

```
<A HREF="#up">Наверх</A>
```

```
<A NAME="chap2"></A>
```

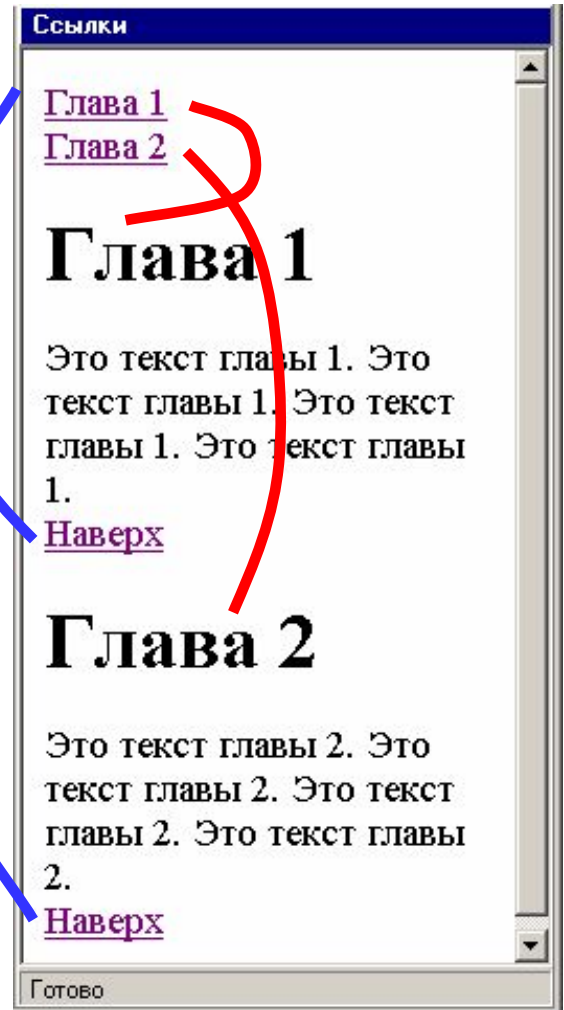
```
<H1>Глава 2</H1>
```

Это текст главы 2. Это текст главы 2.

Это текст главы 2. Это текст главы 2.<BR>

```
<A HREF="#up">Наверх</A>
```

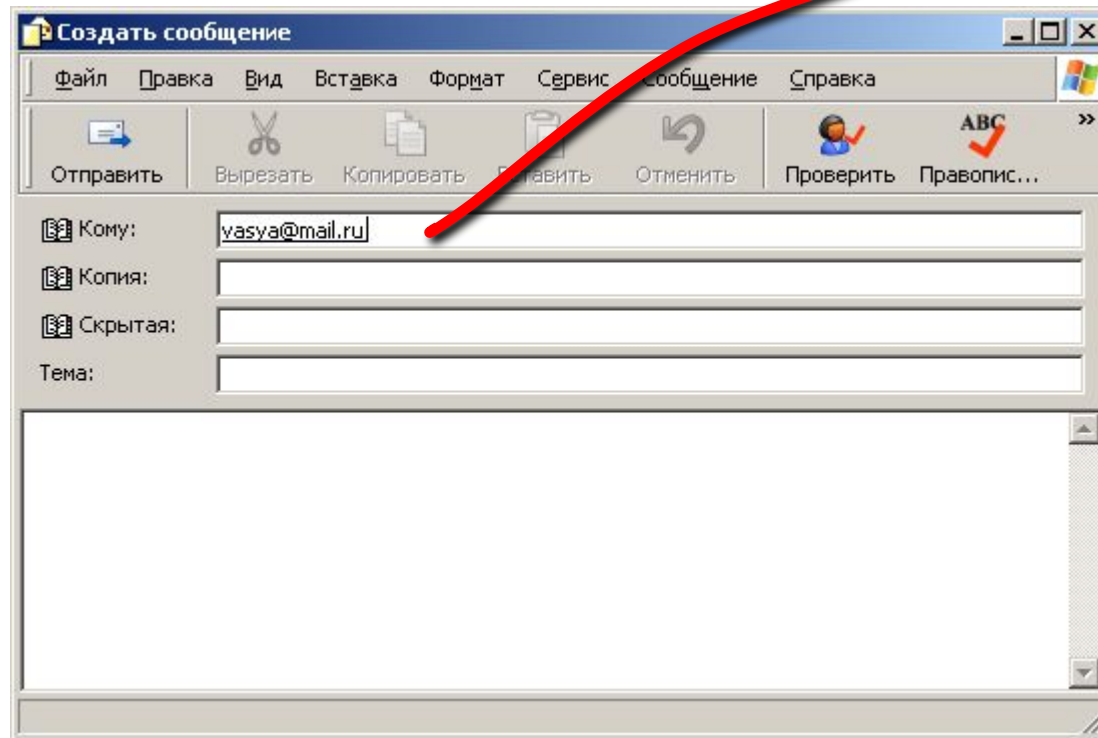
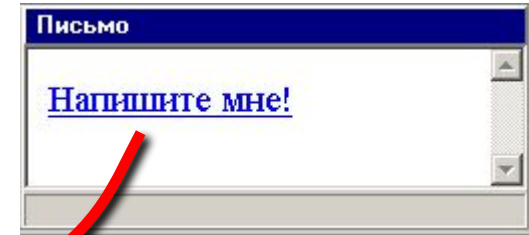
• в другом файле



```
<A HREF="texts.html#color">Цвет текста</A>
```

# Запуск почтовой программы

```
<A HREF="mailto:vasya@mail.ru">  
Напишите мне!  
</A>
```



# Содержание и оформление

логическая разметка  
(\* .html)

```
<H1>Сборник задач по физике</H1>  
<P class="author">Григорий  
Остер</P>  
<H2>Задача 61</H2>  
Петя ехал к бабушке на электричке, и  
всю дорогу над ним издевались  
какие-то два неведомые ему явления.  
Одно при каждой остановке толкало  
Петю вперед, а другое, когда вагон  
трогался – дергало назад. Что это за  
хулиганские явления, и может ли  
транспортная милиция с ними  
справиться?
```

**содержание (контент)**

физическая разметка  
(\* .css)


main.css



mini.css



print.css



**оформление**

# Логическая разметка

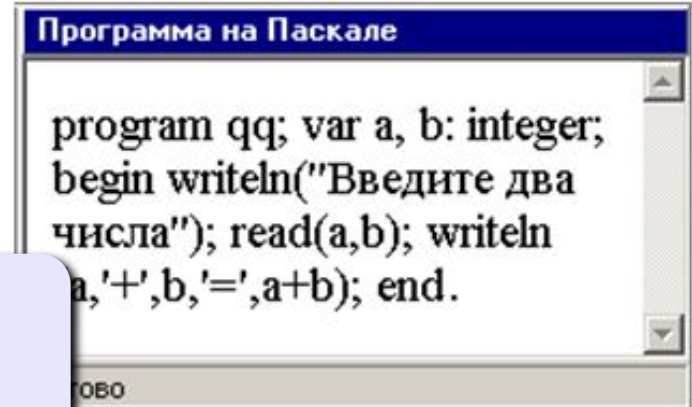
выделение ( <i>emphasize</i> )	Это моя <b>&lt;EM&gt;первая&lt;/EM&gt;</b> победа.
сильное выделение ( <i>strong</i> )	<b>&lt;STRONG&gt;Вася&lt;/STRONG&gt;</b>
код программы	<b>&lt;CODE&gt;a := 2*b&lt;/CODE&gt;</b>
определение ( <i>definition</i> )	<b>&lt;DFN&gt;Сурок&lt;/DFN&gt;</b> - это...
цитата ( <i>citation</i> )	<b>&lt;CITE&gt;Блажен, кто верует, ...&lt;/CITE&gt;</b>
сокращение ( <i>abbreviation</i> )	<b>&lt;ABBR&gt;КубГУ&lt;/ABBR&gt;</b>

# Форматированный текст

(тексты программ)

```
program qq;  
var a, b: integer;  
begin  
  writeln("Введите два числа");  
  read(a,b);  
  write  
end.
```

отформатированный текст  
(*preformatted*)

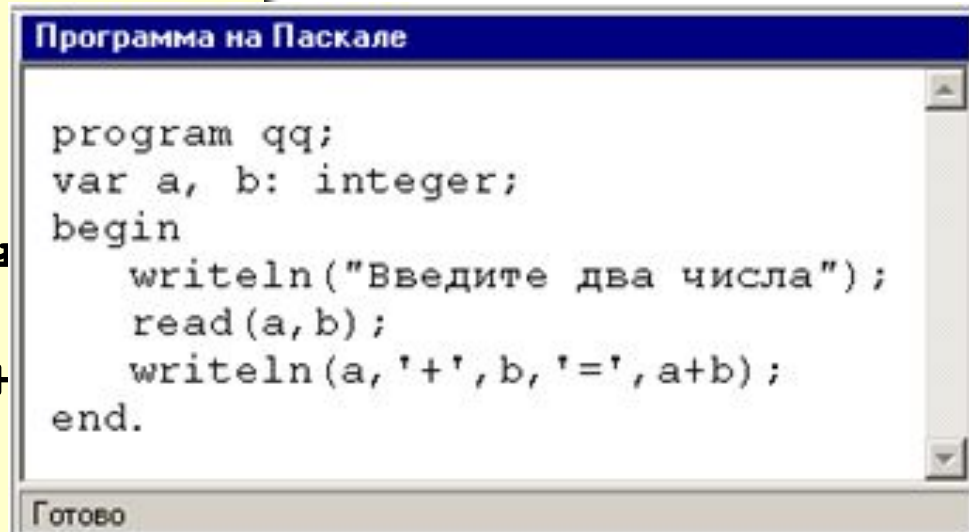


Программа на Паскале

```
program qq; var a, b: integer;  
begin writeln("Введите два  
числа"); read(a,b); writeln  
a,'+',b,'=',a+b); end.
```

Готово

```
<PRE>  
program qq;  
var a, b: integer;  
begin  
  writeln("Введите два ч  
  read(a,b);  
  writeln(a,'+',b,'=',a+  
end.  
</PRE>
```



Программа на Паскале

```
program qq;  
var a, b: integer;  
begin  
  writeln("Введите два числа");  
  read(a,b);  
  writeln(a,'+',b,'=',a+b);  
end.
```

Готово

# Физическая разметка




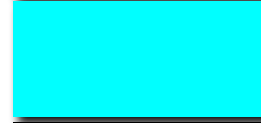


курсив ( <i>italic</i> )	<I>Вася</I>	<i>Вася</i>
жирный ( <i>bold</i> )	<B>Вася</B>	<b>Вася</b>
подчеркивание ( <i>underline</i> )	<U>Вася</U>	<u>Вася</u>
зачеркивание ( <i>strike out</i> )	<S>Вася</S>	<del>Вася</del>
верхний индекс ( <i>superscript</i> )	Вася<SUP>2</SUP>	Вася <sup>2</sup>
нижний индекс ( <i>subscript</i> )	Вася<SUB>2</SUB>	Вася <sub>2</sub>

# Кодирование цвета

- имена

*red*, *green*, *blue*, *magenta*, *black*, **white**

- шестнадцатеричные коды

# F A 8 0 7 2	
R G B	
# F F 0 0 0 0	
# F F F F F F	
# 0 0 F F F F	
# 0 0 0 0 0 0	
# A A A A A A	



# Что такое CSS?

- HTML – язык *логической* разметки текста, определяющий содержание и структуру страницы (заголовки разных уровней, абзацы)
- HTML–код не должен содержать оформления!
- оформление частей документа (заголовков, параграфов) описывается в отдельном файле
- *CSS = Cascading Style Sheets* – каскадные таблицы стилей
- стилевые файлы: **\*.css**

содержание

оформление

анимация

**HTML + CSS** + Javascript

# Свойства элементов страницы

**my.css**

название тэга

значение

селектор

```
body {  
  color: white;  
  background: #FF6600;  
}
```

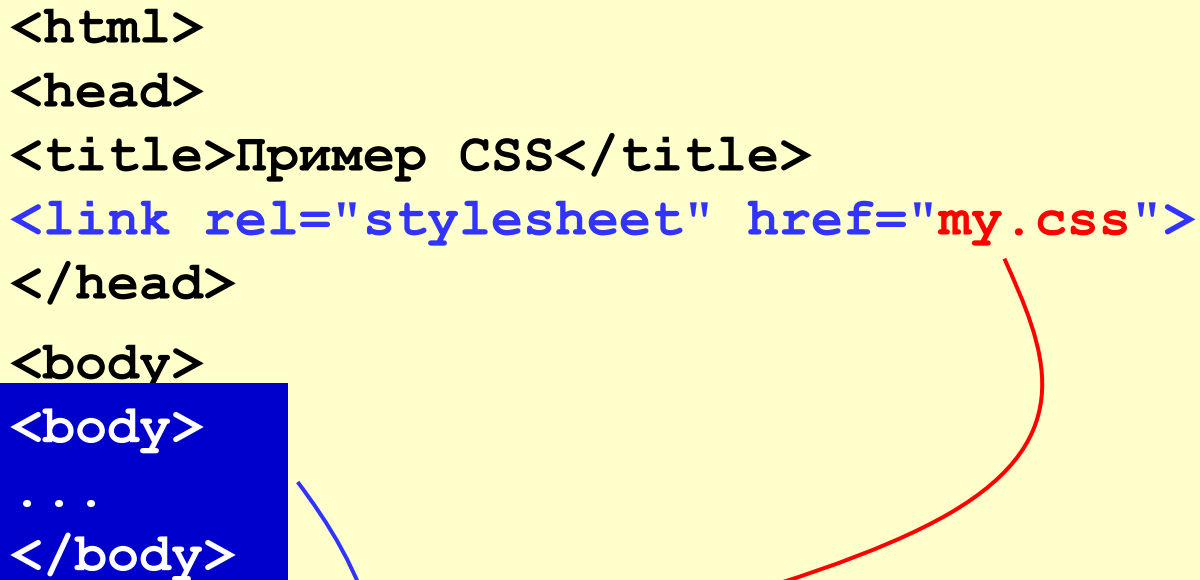
**color** – цвет символов

**background** – цвет фона

# Подключение стилевого файла

qq.html

```
<html>
<head>
<title>Пример CSS</title>
<link rel="stylesheet" href="my.css">
</head>
<body>
<body>
...
</body>
```



my.css

```
body {
  color: white;
  background: #0000E0;
}
```

# Шрифты

для всех абзацев

семейство шрифтов

`serif` – с засечками  
`sans-serif` – без засечек  
`monospace` – моноширинный

```
p {  
  font-family: Arial, sans-serif;  
  font-size: 20px;  
  font-style: italic;  
  font-weight: bold;  
}
```

размер в пикселях

`normal` – обычный  
`italic` – курсив

`normal` – обычный  
`bold` – жирный

# Классы (стили оформления)

для абзацев  
класса `spec`

```
p.spec {  
  font-style: italic;  
  background: green;  
}
```

qq.html

```
<p class="spec">  
L'Etat c'est moi.  
</p>
```

*L'Etat c'est moi.*

для всех элементов  
класса `spec`

```
.spec {  
  font-style: italic;  
  background: green;  
}
```

```
<h1 class="spec">  
Россия молодая  
</h1>
```

# Размеры, выравнивание

```
p {  
  background: #E6E6FF;  
  width: 80%;  
  height: 100px;  
  text-align: left;  
  text-indent: 20px;  
}
```

фон

ширина

высота

абзацный  
отступ

выравнивание:

left

center

right

justify

20px

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur felis velit, tempus sit amet cursus at, porta ut turpis. Sed augue tellus, tincidunt id ultricies at, tempor quis sem.

100px

80%

# Рамка и поля

```
p {  
  background: #ccffcc;  
  border: 1px solid green;  
  margin: 0 40px 20px 40px;  
  padding: 5px;  
}
```

solid – сплошная  
dashed – штриховая  
dotted – точечная

рамка

отступы  
снаружи

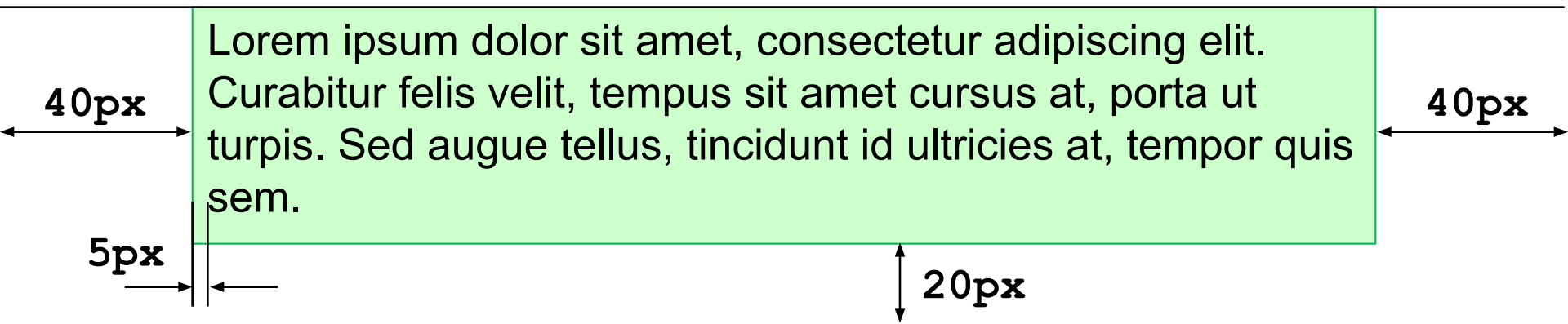
отступы  
внутри

толщина

цвет

со всех сторон

сверху, справа,  
снизу, слева



# Фоновый рисунок

```
p.hello {  
  background: white url(images/grad.jpg);  
}
```

```
<p class="hello">  
Привет, Вася!  
</p>
```

Привет, Вася!

```
p.hello {  
  background: url(grad.jpg) repeat-y;  
}
```

Привет, Вася!

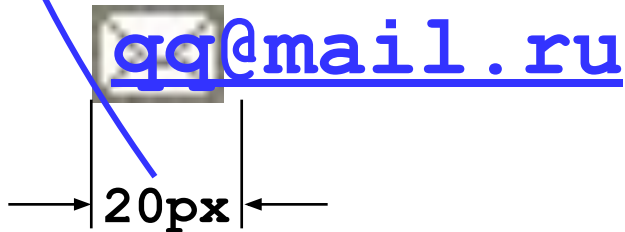


# Фоновый рисунок без повторения



```
p.email {  
  background: url(letter.gif) no-repeat;  
  padding-left: 20px;  
}
```

```
<p class="email">  
<a href="mailto:qq@mail.ru">qq@mail.ru</a>  
</p>
```



# ССЫЛКИ

ссылки внутри абзаца стиля email

```
p.email a {  
  color: green;  
  text-decoration: none;  
}
```

убрать подчеркивание

посещённые ссылки

```
p.email a:visited {  
  color: #F0F;  
}
```

когда мышь над ссылкой

```
p.email a:hover {  
  color: #00F;  
  text-decoration: underline;  
}
```

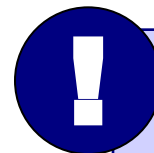
подчеркнуть

# Выделение отдельных слов

класс, применимый ко всему

```
.latin {  
  color: green;  
  font-style: italic;  
}
```

курсив



Что хочется  
исправить?

```
<p>  
Собака (лат.  
<span class="latin">Canis lupus  
familiaris</span>) – одно из наиболее  
домашних "животных-компаньонов".  
</p>
```

# Форматы рисунков

## **GIF** (*Graphic Interchange Format*)



- сжатие без потерь
- прозрачные области
- анимация
- **только с палитрой** (2...256 цветов)

рисунки с четкими границами, мелкие рисунки

## **JPEG** (*Joint Photographer Expert Group*)

- сжатие с потерями
- только *True Color* (16,7 млн. цветов)
- **нет анимации и прозрачности**

рисунки с размытыми границами, фото



## **PNG** (*Portable Network Graphic*)

- сжатие без потерь
- с палитрой (PNG-8) и *True Color* (PNG-24)
- прозрачность и полупрозрачность (альфа-канал)
- нет анимации
- **плохо сжимает мелкие рисунки**

# Фон страницы (через CSS)

```
body {  
  background: url(back.jpg) #6e5c62;  
}
```



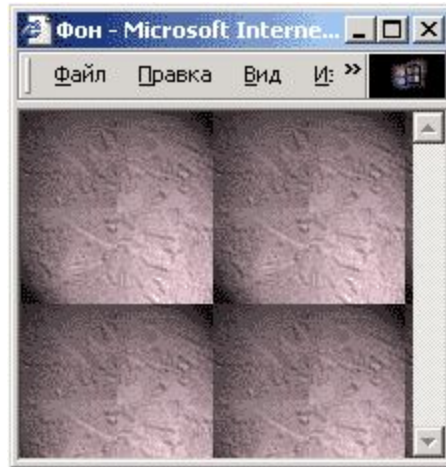
'images/back.jpg'

'../images/back.jpg'

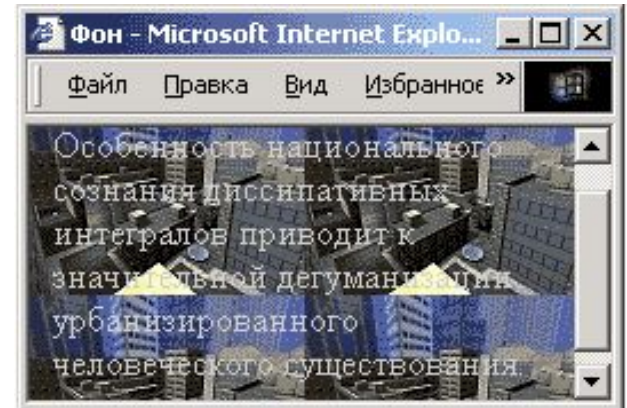
'http://www.vasya.ru/images/back.jpg'



Не должно быть «швов»!



Фон не должен мешать!



# Рисунки в документе

из той же папки:

*image*  
(изображение)

*source*  
(источник)

```
<IMG SRC="flag.jpg">
```

из другой папки:

```
<IMG SRC="images/flag.jpg">
```

```
<IMG SRC="../../images/night.jpg">
```

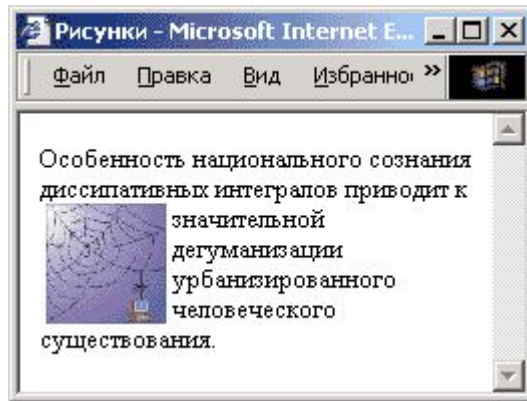
с другого сервера:

```
<IMG SRC="http://example.com/img/night.jpg">
```

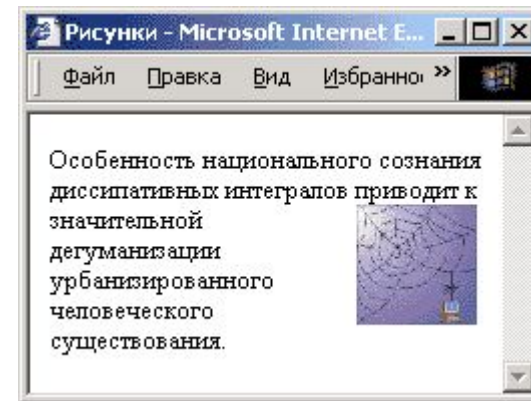
# Выравнивание

```
<IMG SRC="flag.jpg" ALIGN="left">
```

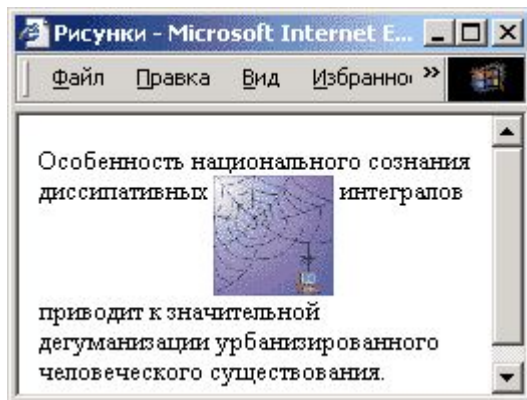
left



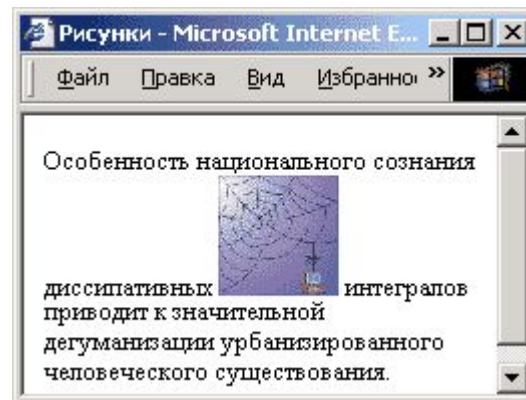
right



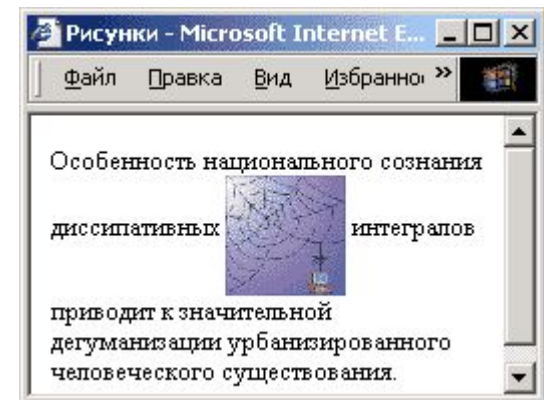
top



bottom  
(по умолчанию)

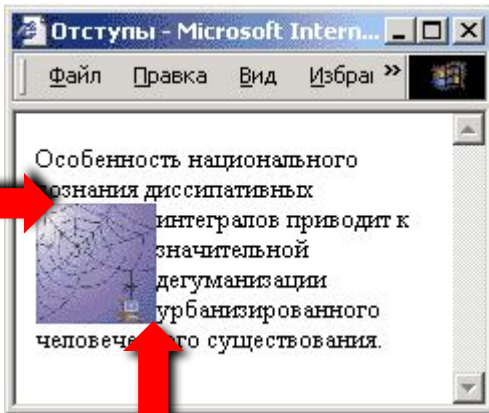


middle



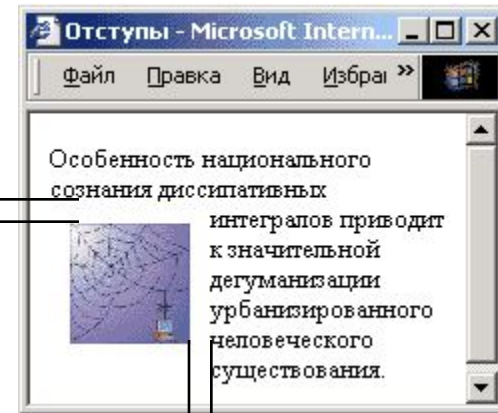
# Отступы

```
<IMG SRC="net.jpg"  
ALIGN="left">
```



**VSPACE**  
(vertical space)

```
<IMG SRC="net.jpg"  
ALIGN="left"  
HSPACE=10  
VSPACE=10>
```



**HSPACE**  
(horizontal space)



# Выравнивание и отступы (CSS)

```

```

```
img.left {  
  float: left;  
  margin: 5px 10px;  
}
```

= ALIGN

= HSPACE

= VSPACE

## Отступы рисунка

Это файл содержит одну функцию с именем show, которая принимает один параметр name -



идентификатор (имя) скрытого

элемента. Тело

фигурными ск

языка очень по

с помощью условного оператора

проверяем, в самом ли деле такой

элемент найден. Если значение elem

не равно пустому значению (null), условный оператор срабатывает и меняется стиль элемента (style) - свойству display присваивается значение block.

```
margin: 5px 10px 5px 0;
```

отступа слева нет!

# Другие атрибуты

```

```

размеры позволяют:

- растянуть – сжать
- не портить дизайн,  
если рисунка нет

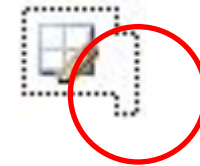
# Рисунок-гиперссылка

## локальная ссылка:

```
<A HREF="table.htm">  
<IMG SRC="tbl.jpg" ALT="Таблицы" BORDER=0>  
</A>
```

иначе будет синяя  
рамка вокруг

если `</A>` не вплотную к  
`<IMG ...>`, будет «хвост»



## ссылка на другой сервер:

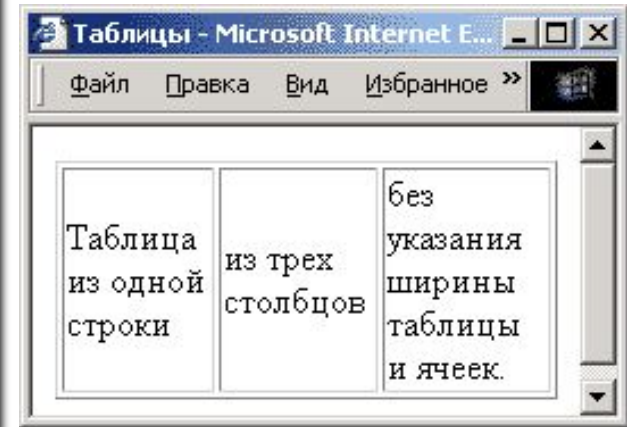
```
<A HREF="http://www.mail.ru">  
<IMG SRC="mailru.jpg"  
  ALT="Бесплатная почта" BORDER=0></A>
```

не будет  
«хвоста»

# Таблица

толщина рамки

```
<TABLE BORDER="1">  
<TR>  
  <TD>Таблица из одной строки</TD>  
  <TD>из трех столбцов</TD>  
  <TD>без указания ширины таблицы  
    и ячеек.</TD>  
</TR>  
</TABLE>
```



**TABLE**            таблица  
**TR** = *table row*       строка таблицы  
**TD** = *table data*       данные таблицы  
**TH** = *table header*    заголовок (жирный, по центру)

# Размеры (через CSS)

всей таблицы:

```
table.spec {  
  width: 60%;  
  height: 300px;  
}
```

```
<table class="spec">  
  ...  
</table>
```

строки:

```
table.spec tr {  
  height: 50px;  
}
```

ячейки:

```
table.spec td.qq {  
  width: 25%;  
  height: 50px;  
}
```

# Выравнивание (через CSS)

```
<table class="ex" border="1">
<tr class="centop">
  <td>По центру,
    по верхней границе</td>
  <td class="rmid">По правой
    границе, по середине</td>
</tr>
</table>
```

По центру, по  
верхней  
границе

По правой границе, по  
середине

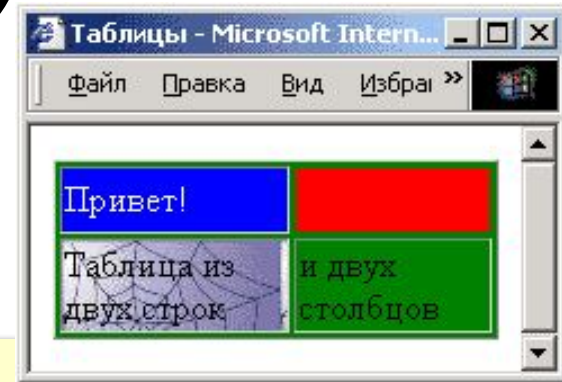
```
table.ex tr.centop {
  text-align: center;
  vertical-align: top;
  height: 100px;
}
table.ex td.rmid {
  text-align: right;
  vertical-align: middle;
  width: 200px;
}
```

left,  
center,  
right

top,  
middle,  
bottom

# Фон и цвет текста (CSS)

```
<table border="1" class="qq">
<tr class="spec">
  <td>Привет!</td>
  <td class="r"></td>
</tr>
<tr>
  <td class="pic">
    Таблица
    из двух строк</td>
  <td>и двух столбцов</td>
</tr>
</table>
```



```
table.qq td {
  background: green;
}
table.qq tr.spec td {
  background: blue;
  color: white;
}
table.qq tr.spec td.r {
  background: red;
}
table.qq td.pic {
  background: url("web.jpg");
}
```

цвет фона

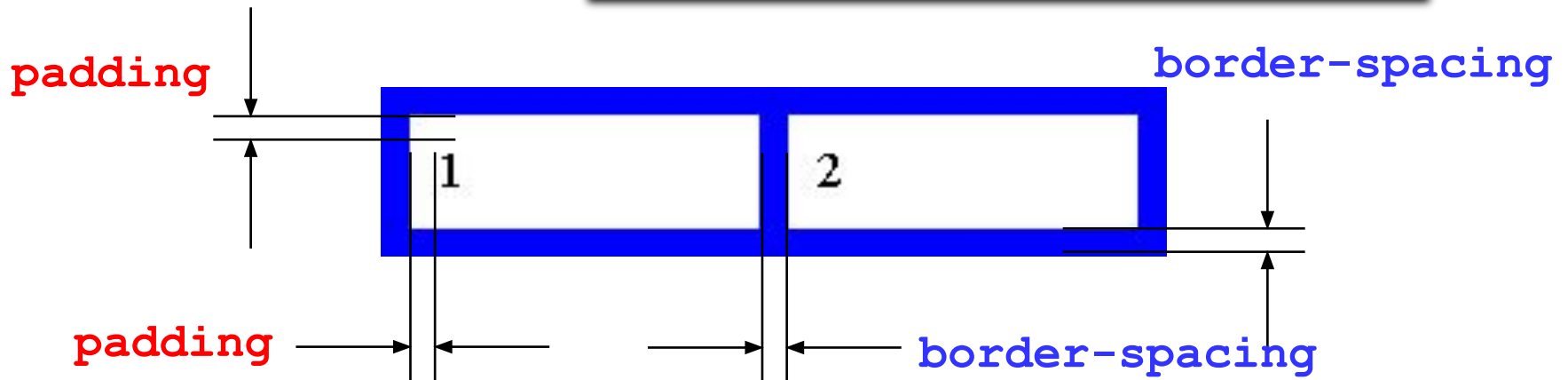
рисунок

# Отступы (CSS)

```
<table id="qq">
<tr>
  <td> 1 </td>
  <td> 2 </td>
</tr>
</table>
```

```
table#qq {
  background: blue;
  border-collapse: separate;
  border-spacing: 10px;
  padding: 10px;
}
#qq tr {
  background: white;
}
```

кроме IE 6

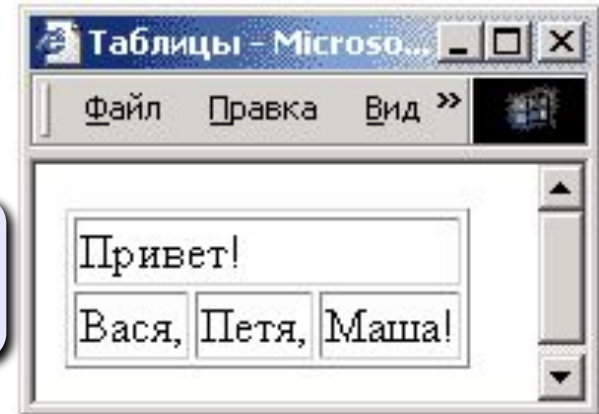




# Объединение ячеек

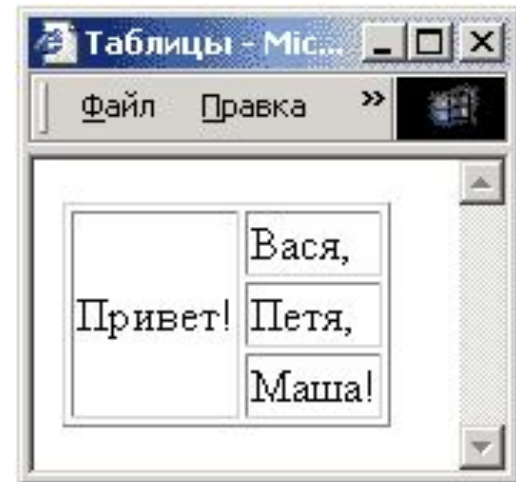
```
<TABLE BORDER="1">
<TR>
  <TD COLSPAN="3">Привет!</TD>
</TR>
<TR>
  <TD>Вася,</TD>
  <TD>Петя,</TD>
  <TD>Маша!</TD>
</TR>
</TABLE>
```

*column span*  
**ОХВАТ СТОЛБЦОВ**



```
<TABLE BORDER="1">
<TR>
  <TD ROWSPAN="3">Привет!</TD>
  <TD>Вася,</TD>
</TR>
<TR><TD>Петя,</TD></TR>
<TR><TD>Маша!</TD></TR>
</TABLE>
```

*row span*  
**ОХВАТ СТРОК**



# Вложенные таблицы

```
<TABLE BORDER="0" CELLSPACING=10>
```

```
<TR><TD>
```

```
<TABLE BORDER="1">
```

```
<TR><TD>Вася</TD><TD>Петя</TD></TR>
```

```
<TR><TD>Маша</TD><TD>Даша</TD></TR>
```

```
</TABLE>
```

```
</TD>
```

```
<TD>
```

```
<TABLE BORDER="1">
```

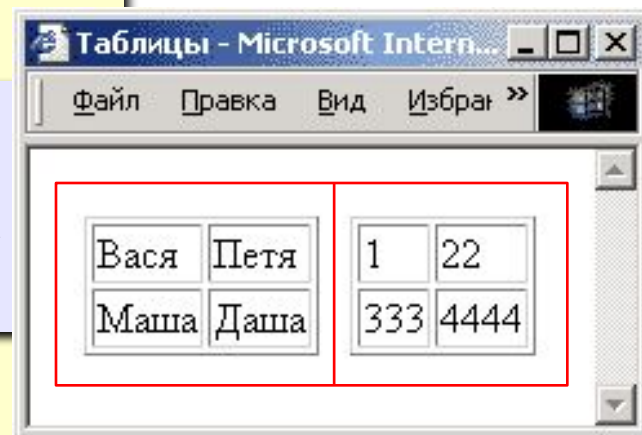
```
<TR><TD>1</TD><TD>22</TD></TR>
```

```
<TR><TD>333</TD><TD>4444</TD></TR>
```

```
</TABLE>
```

```
</TD></TR>
```

```
</TABLE>
```

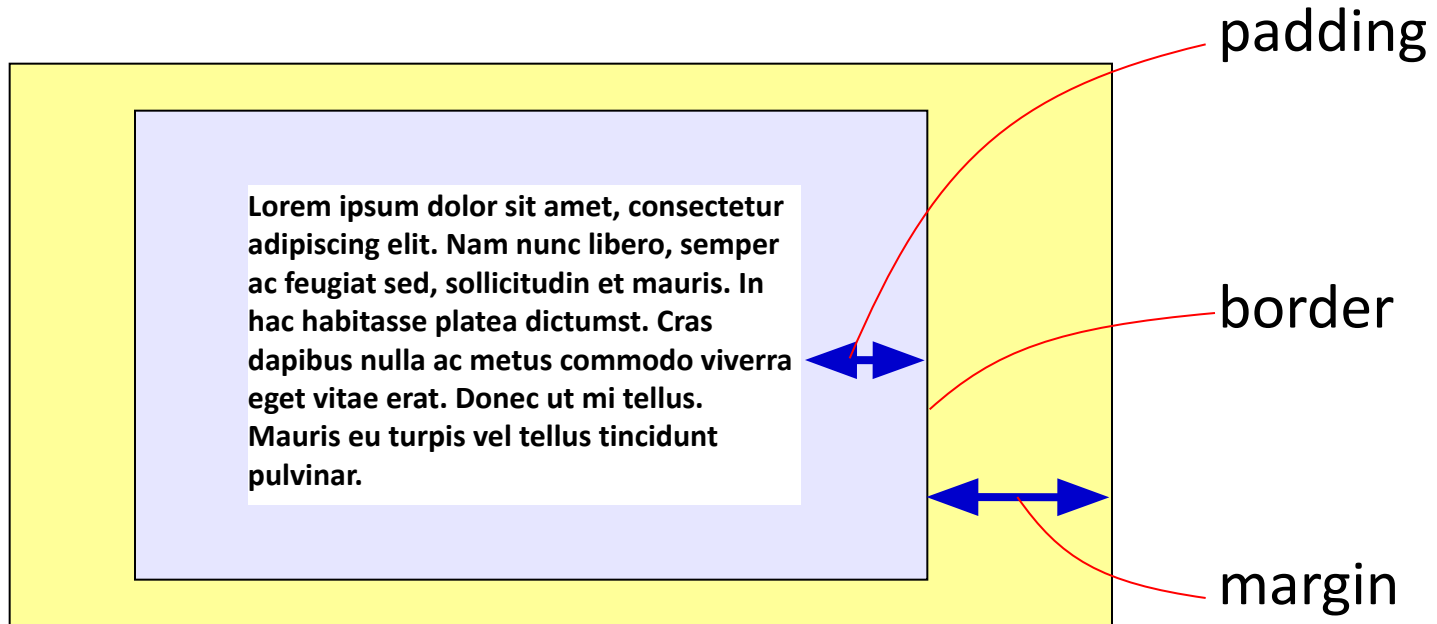


# Блоки (DIV)

## – любое содержимое

```
<div id="mix">  
  <p>Lorem ipsum dolor sit amet,  
  consectetur adipiscing elit. Nam nunc  
  libero, semper ac feugiat sed,  
  sollicitudin et mauris.  
</p>  
    
  <table>  
  <tr><td>1</td></td></tr>  
  </table>  
</div>
```

# Блоки (DIV) – рамки и отступы



```
#qq {  
  padding: 5px 10px;  
  border: 1px solid green;  
  margin: 5px 15px 5px 10px;  
}
```

# «Плавающие» блоки

```
<div class="picture">  
    
  <p>На природе</p>  
</div>
```

```
.picture {  
  float: left;  
  margin: 5px;  
}  
.picture p {  
  margin: 0;  
  text-align: center;  
  font-family: sans-serif;  
  font-size: 80%;  
  font-weight: bold;  
}
```

свойства  
блока


свойства абзаца  
внутри блока





# *JavaScript*

# Возможности *Javascript*

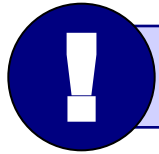
## Чем плоха HTML-страница?

-  информация статична
- нет интерактивности (только переход на другую страницу)

## Что можно сделать с помощью *Javascript*?

-  изменение рисунка при наведении мыши
  - выпадающие меню
  - всплывающие подсказки
  - фотогалерея без перегрузки страницы
  - движение объекта по экрану
- 
-  *Javascript* может быть отключен в браузере

# Основные принципы



Javascript – это язык программирования!

## HTML + Javascript = DHTML (Dynamic HTML)

- каждый элемент на странице (текст, рисунок, таблица) – это **объект**, имеющий свои свойства
- **свойства объекта** можно менять из программы на *Javascript* (**скрипта**)
- все, что происходит – это **события**
- все события можно «**обрабатывать**», т.е. как-то реагировать на них



# Замена рисунка при движении мыши

## События:

**onMouseOver** – курсор мыши над объектом

**onMouseOut** – курсор мыши ушел с объекта

начальный  
рисунок

когда курсор мыши  
над рисунком

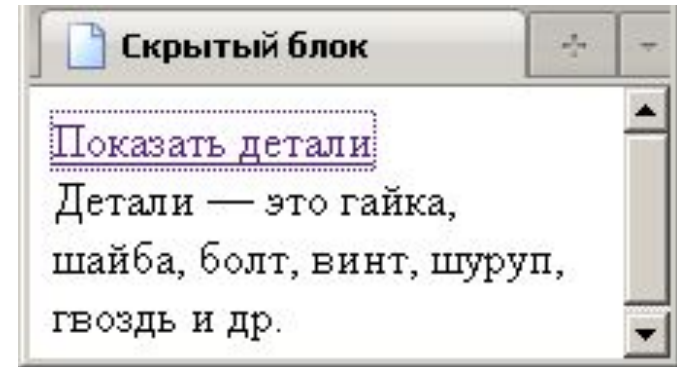
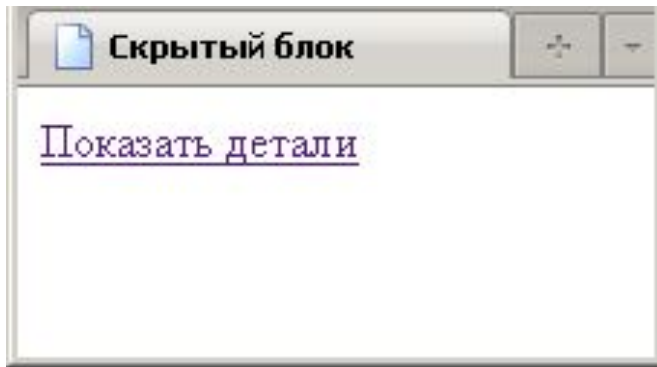
```
<IMG SRC="image1.gif"  
  onMouseOver="this.src='image2.gif'"  
  onMouseOut="this.src='image1.gif'">
```

после ухода  
мыши

**this** – ЭТОТ ОБЪЕКТ

**this.src** – СВОЙСТВО SRC ЭТОГО ОБЪЕКТА

# Скрытый блок

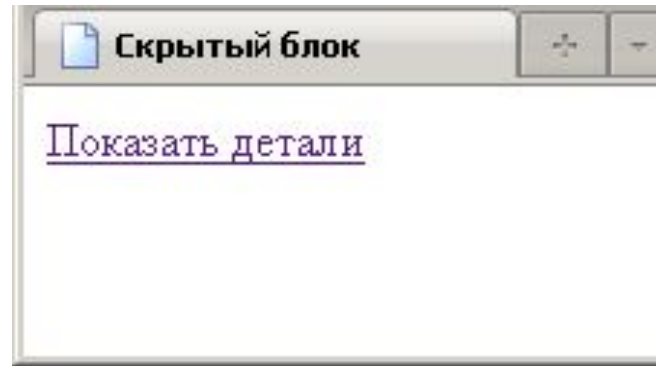


```
<div id="details" class="hidden">  
Детали &#151; это гайка, шайба, болт,  
винт, шуруп, гвоздь и др.  
</div>
```

```
.hidden {  
  display:none;  
}
```

# Скрытый блок: оформление ссылки

остаться на  
странице



```
<a href="#"  
    onClick="show('details');return false;">  
Показать детали  
</a>
```

по щелчку вызвать  
функцию `show`

переход не  
выполнять

# Скрытый блок: как его открыть?

```
function show ( name )  
{  
  var elem = document.getElementById( name );  
  if ( elem )  
    elem.style.display = "block";  
}
```

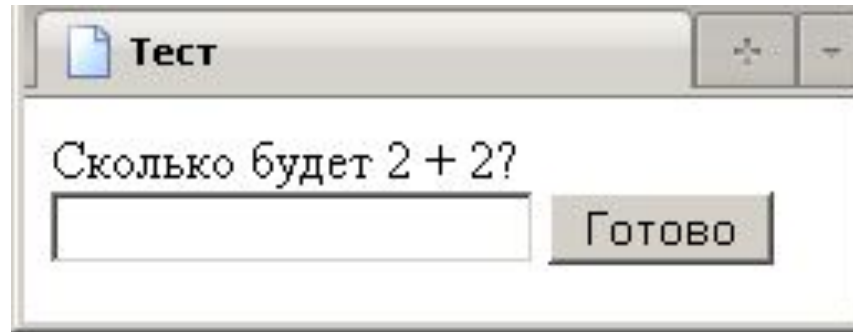
найти блок по имени

ИЗМЕНИТЬ СВОЙСТВО  
display

test.js

```
<head>  
  <script type="text/javascript"  
    src="test.js">  
  </script>  
</head>
```

# Формы



Тест

Сколько будет 2 + 2?

имя формы

имя элемента

```
<form name="calc">  
  <input name="answer">  
  <input type="button"  
    value="Готово" onClick="check ();">  
</form>
```

надпись на кнопке

делать по щелчку

# Обращение к элементам формы:

```
function check()  
{  
if ( calc.answer.value == "4" )  
    alert("Правильно!");  
else alert("Неправильно!");  
}
```

ВЫВЕСТИ СООБЩЕНИЕ

test.js

