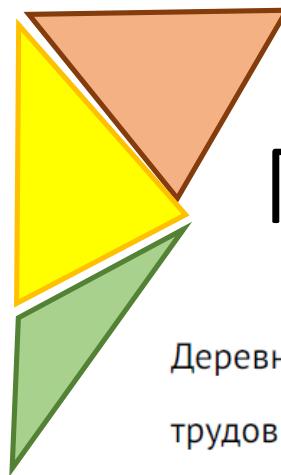


# Язык HTML

- от англ. *HyperText Markup Language* – «язык разметки гипертекста»
- служит для создания веб-страниц
- HTML-документы представляют собой файлы с текстом и дополнительными инструкциями (**тегами**)
- Официальной спецификации HTML 1.0 не существует. До 1995 года существовало множество неофициальных спецификаций HTML, появившихся в ходе браузерных войн
- RFC 1866 — HTML 2.0, одобренный как официальный стандарт 22 сентября 1995 года
- HTML 3 (март 1996) - не нашла поддержки у разработчиков
- HTML 3.2 — 14 января 1997 года
- HTML 4.0 — 18 декабря 1997 года (многие унаследованные элементы были отмечены как устаревшие и нерекомендованные к использованию)
- HTML 4.01 — 24 декабря 1999 года (версия включала малозаметные, но существенные изменения по сравнению с предыдущей)
- ISO/IEC 15445:2000 (ISO HTML, основан на HTML 4.01 Strict) — 15 мая 2000 года
- HTML 5 — разработан и принят W3C в 2014



# Пример

Деревня Приветствую тебя, пустынnyй уголок, приют спокойствия, трудов и вдохновенья, где льется дней моих невидимый поток на лоне счастья и забвенья. Я твой – я променял порочный двор Цирцей, роскошные пиры, забавы, заблужденья на мирный шум дубров, на тишину полей, на праздность вольную, подругу размышленья.

Деревня

Приветствую тебя, пустынnyй уголок,  
Приют спокойствия, трудов и вдохновенья,  
Где льется дней моих невидимый поток  
На лоне счастья и забвенья.

Я твой – я променял порочный двор Цирцей,  
Носкошные пиры, забавы, заблужденья  
На мирный шум дубров, на тишину полей,  
На праздность вольную, подругу размышленья.

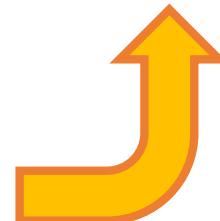
[начало заголовка] Деревня [конец заголовка]

[начало абзаца] Приветствую тебя, пустынnyй уголок, [перенос]

Приют спокойствия, трудов и вдохновенья, [перенос]

Где льется дней моих невидимый поток [перенос]

На лоне счастья и забвенья. [конец абзаца]

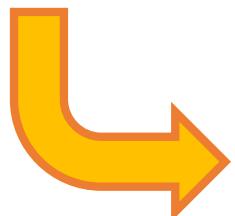


[начало абзаца] Я твой – я променял порочный двор Цирцей, [перенос]

Носкошные пиры, забавы, заблужденья [перенос]

На мирный шум дубров, на тишину полей, [перенос]

На праздность вольную, подругу размышленья. [конец абзаца]



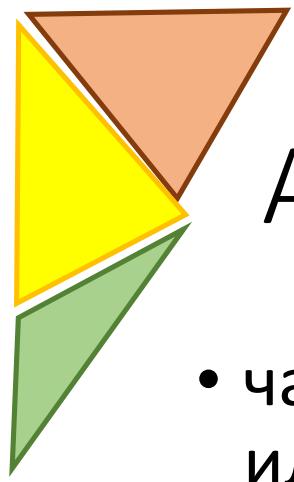


# Тег

- Элемент языка разметки, заключённый между символами < и >
- Виды:
  - Двойные. Например: <html></html>
  - Одиночные. Например: <img/>
- Регистронезависимы  
<Img>, <img>, <IMG>, <iMg> - один и тот же тег
- Необходимо соблюдать правильную вложенность тегов

01. <i>очень <b> длинный текст </b> очень </i> – Правильно

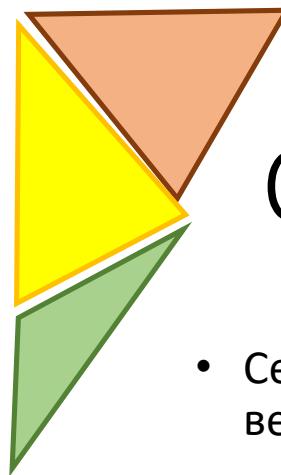
02. <i>очень<b> длинный текст </i> очень</b> – Неправильно



# Атрибуты тегов

- часть языка разметки, используемая для настройки поведения или отображения элемента HTML

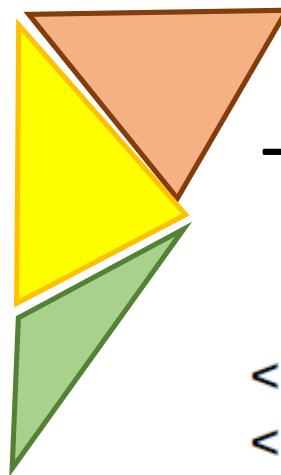
01. `<div class="he'a'der">` – в двойных кавычках
02. `<input checked='checked'>` – в апострофах
03. `<div class=footers>` – без кавычек
04. `<input checked>` – одиночные атрибуты



# Структура HTML документа

- Секция DOCTYPE указывает браузеру тип документа и версию использованного языка разметки
- Мета-тег HTML — это элемент разметки html, описывающий свойства документа как такового (метаданные). Назначение мета-тега определяется набором его атрибутов, которые задаются в теге <meta>. Мета-теги размещают в блоке <head>...</head> веб-страницы. Они не являются обязательными элементами, но могут быть весьма полезны
- В блоке <head>...</head> также происходит подключение скриптов, указание заголовка страницы
- Блок <body>...</body> представляет содержимое веб-страницы

```
<!DOCTYPE html>
<html lang="ru">
  <head>
    <!-- Служебная информация -->
  </head>
  <body>
    <!-- Содержание -->
  </body>
</html>
```



# Теги для работы с текстом

**< b > ... < / b >** – выделение текста жирным

**< i > ... < / i >** – выделение текста курсивом

**< u > ... < / u >** – подчеркивание текста

**< sub > ... < / sub >** – форматировать текст как подстрочный индекс

**< sup > ... < / sup >** – форматировать текст как надстрочный индекс

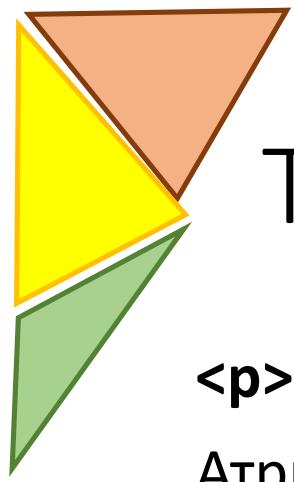
Пример:

HTML-код:  $101<\sub>2</sub> = 5$

В браузере:  $101_2 = 5$

HTML-код:  $2<\sup>8</sup> = 256$

В браузере:  $2^8 = 256$



# Теги для работы с текстом

`<p>...</p>` – задает начало и конец параграфа

Атрибут:

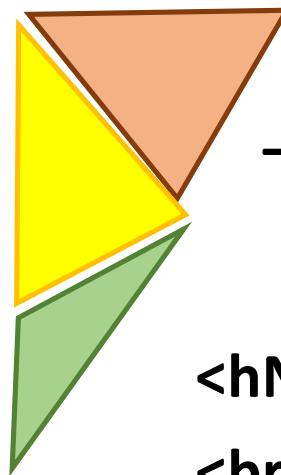
`align="..."` – определяет режим выравнивания текста

`left` – по левому краю (по умолчанию)

`center` – по центру

`right` – по правому краю

`justify` – по ширине



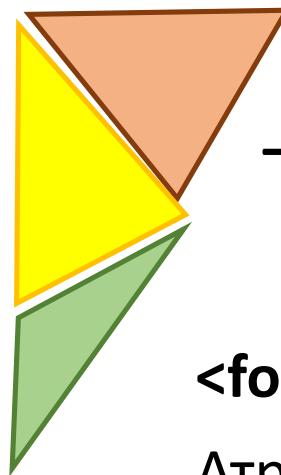
# Теги для работы с текстом

**<hN>...</hN>** – вложенный текст, является заголовком документа уровня N из [1..6]

**<br>** – перенос строки

**<hr>** – выводит горизонтальную разделительную линию

Атрибуты:



# Теги для работы с текстом

**<font>...</font>** – устанавливает размер и цвет текста

Атрибуты:

color="цвет" – задает цвет текста

face="шрифт" – определяет шрифт текста

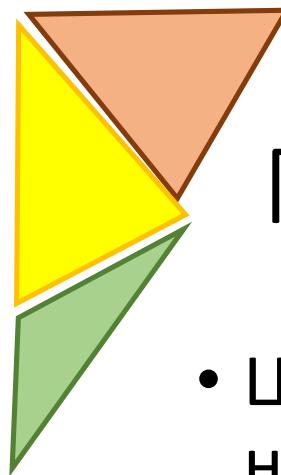
size="1-7" – устанавливает размер шрифта (1–7)

Пример:

HTML-код:

```
<FONT face="Tahoma" size="2" color="gray">текст</FONT>
```

В браузере:      текст



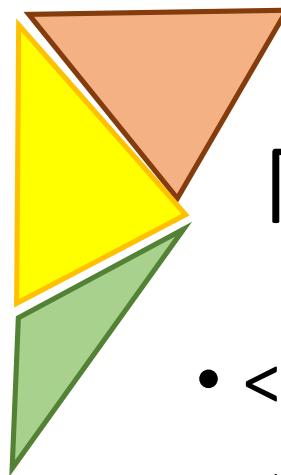
# Представление цвета в HTML

- Цвет в HTML может быть задан ключевыми словами – названиями цветов на английском языке

Пример:

Стандартные цвета HTML

| Название в HTML | Название на русском | Код в RGB |
|-----------------|---------------------|-----------|
| aqua            | морская волна       | #00ffff   |
| black           | черный              | #000000   |
| blue            | синий               | #0000ff   |
| fuchsia         | фуксия              | #ff00ff   |
| grey            | серый               | #808080   |
| green           | зеленый             | #008000   |



# Примеры задания цвета

- <body bgcolor="chucknorris"> test </body>
- <body bgcolor="arnoldschwarzenegger"> test </body>
- <body bgcolor="sarcasm"> test </body>
- <body bgcolor="dontspeak"> test </body>
- <body bgcolor="webbrowser"> test </body>

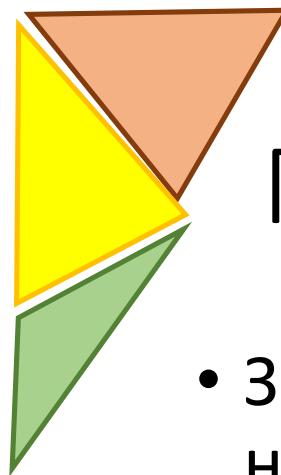
test

test

test

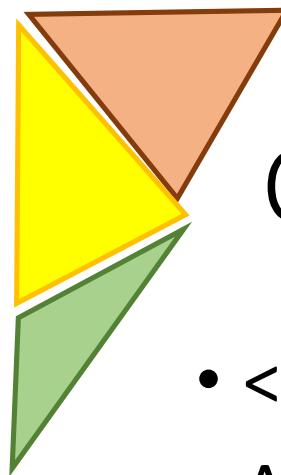
test

test



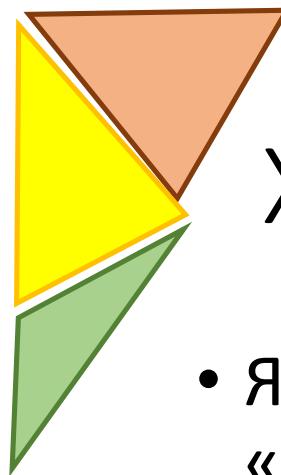
## Пример «**chucknorris**»

- Заменяем все неправильные шестнадцатеричные символы нулями 0, **chucknorris** превращается в **c00c00000000**
- “Добавляем” число нулями до количества знаков, делимого на 3 (11 -> 12), получаем **c00c 0000 0000**
- Разбиваем на три группы, каждая из которых отвечает за одну компоненту RGB:  
**RGB (c00c, 0000, 0000)**
- Обрезаем каждый из аргументов справа, оставляя 2 символа (8 бит), и получаем результат  
**RGB (c0, 00, 00) = #C00000 или он же RGB(192, 0, 0)**



# ССЫЛКИ

- <a href="...">...</a>
- Атрибут href задает адрес куда будет выполняться переход после перехода по ссылке
- Примеры:
  - <a href=".">виды ссылок</a>
  - [протокол]://ya.ru
  - /maps
  - #name
  - ../monkey.zip
  - mailto:akuv@ya.ru



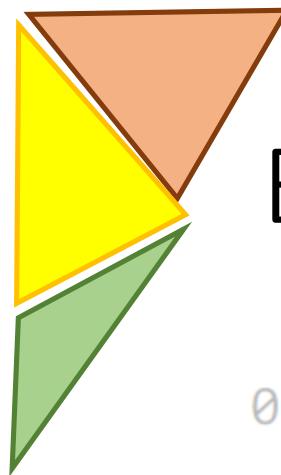
# ХЭШ ССЫЛКИ

- Якорь (**хеш-ссылка**) – это **ссылка**-закладка, нажатие на которую «заставляет» браузер прокрутить страницу до нужного «заякоренного» места
- Пример:

01. <a href="#line">Хэш-ссылки</a>

02. очень много текста

03. <a name="line">нужная строка</a>



# Вложенные ссылки

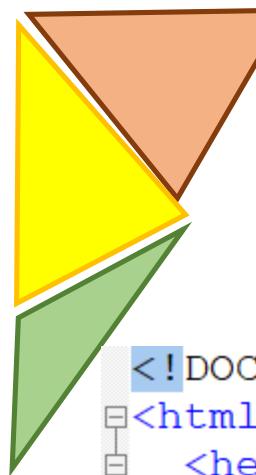
01. <a href="map">Ya<a href="navigator">nd</a>ex</a>

02. <!--Браузер это поймет так:-->

03. <a href="map">Ya</a><a href="navigator">nd</a>ex

Yandex

Вкладывать ссылки в ссылки нельзя!



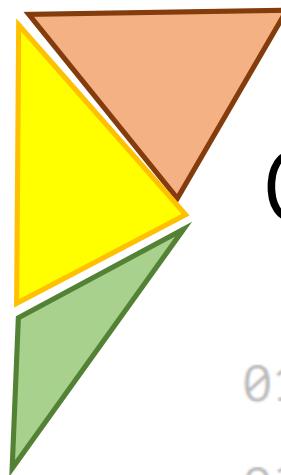
# СПИСКИ

```
<!DOCTYPE html>
<html lang="ru">
  <head>
    <title>Test page</title>
  </head>
  <body>
    <ul>
      <li>Свёрточные</li>
      <li>Рекурентные</li>
      <li>Трансформерные
          <ul>
            <li>BERT</li>
            <li>GPT</li>
            <li>T5</li>
          </ul>
        </li>
    </ul>
  </body>
</html>
```

- Свёрточные
- Рекурентные
- Трансформерные
  - BERT
  - GPT
  - T5

```
<!DOCTYPE html>
<html lang="ru">
  <head>
    <title>Test page</title>
  </head>
  <body>
    <ol>
      <li>Свёрточные</li>
      <li>Рекурентные</li>
      <li>Трансформерные
          <ol type='a'>
            <li>BERT</li>
            <li>GPT</li>
            <li>T5</li>
          </ol>
        </li>
    </ol>
  </body>
</html>
```

1. Свёрточные
2. Рекурентные
3. Трансформерные
  - a. BERT
  - b. GPT
  - c. T5

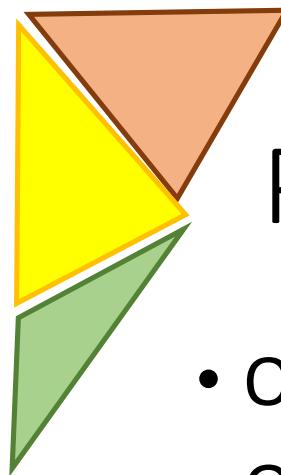


# Список определений

01. <dl> <!--definition list-->
02. <dt>HTML-документ</dt> <!--definition term-->
03. <dd>Обычный текстовый файл, который может содержать в себе текст, теги и стили. Изображения и другие объекты хранятся отдельно.</dd> <!--definition description-->
04. </dl>

HTML-документ

Обычный текстовый файл, который может содержать в себе текст, теги и стили. Изображения и другие объекты хранятся отдельно.



# Работа с изображениями

- Осуществляется непарным тегом `<img>`
- Обязательный атрибут `src` указывает абсолютный или относительный путь до изображения
- Стандартными форматами изображений являются GIF, PNG и JPEG

Пример:

```

```



# Работа с изображениями

Атрибуты:

`align="..."` – определяет режим выравнивания изображения относительно текста в строке:

top – по верхнему краю

middle – по центру строки

bottom – по нижнему краю (по умолчанию)

left – по левому краю окна

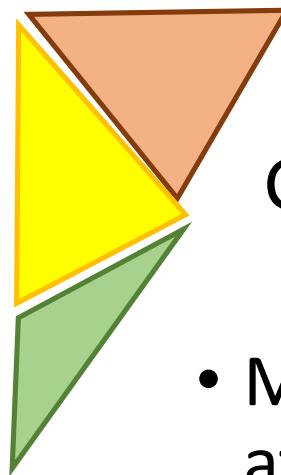
right – по правому краю окна

`alt="..."` – определяет текст, описывающий изображение для браузеров с отключенной графикой

`border="N"` – устанавливает толщину рамки вокруг изображений, равной N пикселей (0 – отключение рамки)

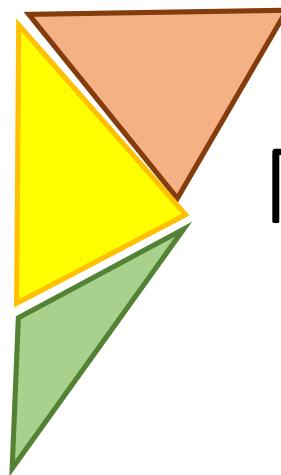
`height="N"` – высота изображения в пикселях или процентах

`width="N"` – ширина изображения в пикселях или процентах



# Фоновое изображение страницы

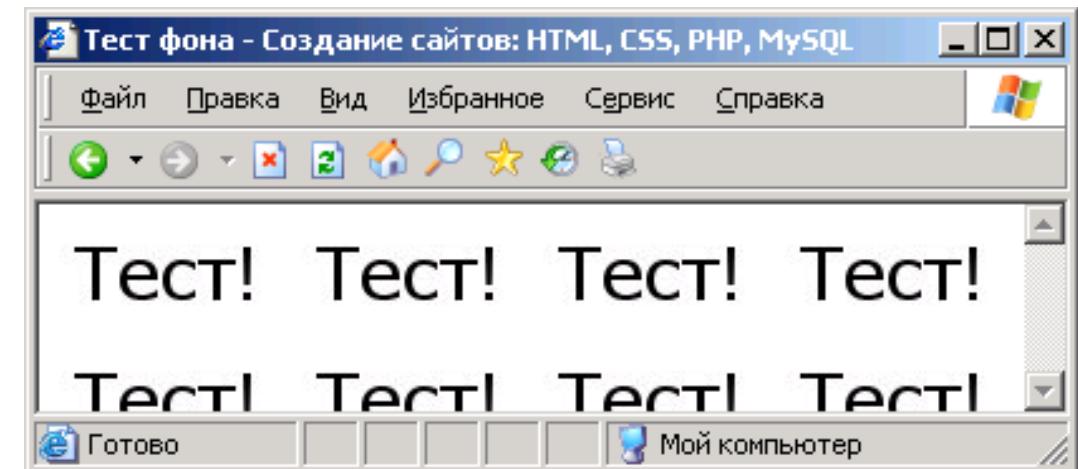
- Можно задавать адрес фонового изображения для страницы в атрибуте `background` тега `<body>`
- Фоновое изображение отображается в натуральную величину
- Если размер изображения меньше размера окна браузера, то рисунок повторяется по горизонтали вправо и по вертикали вниз
- Чтобы сделать неповторяющийся фон, необходимо выбрать картинку заведомо большую, чем размер страницы по ширине и высоте или использовать повторяющийся фон

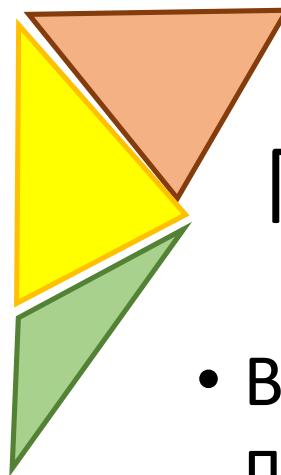


# Пример

```
<html>
<head>
<title>Тест фона</title>
</head>
<body background="bg1.jpg">
</body>
</html>
```

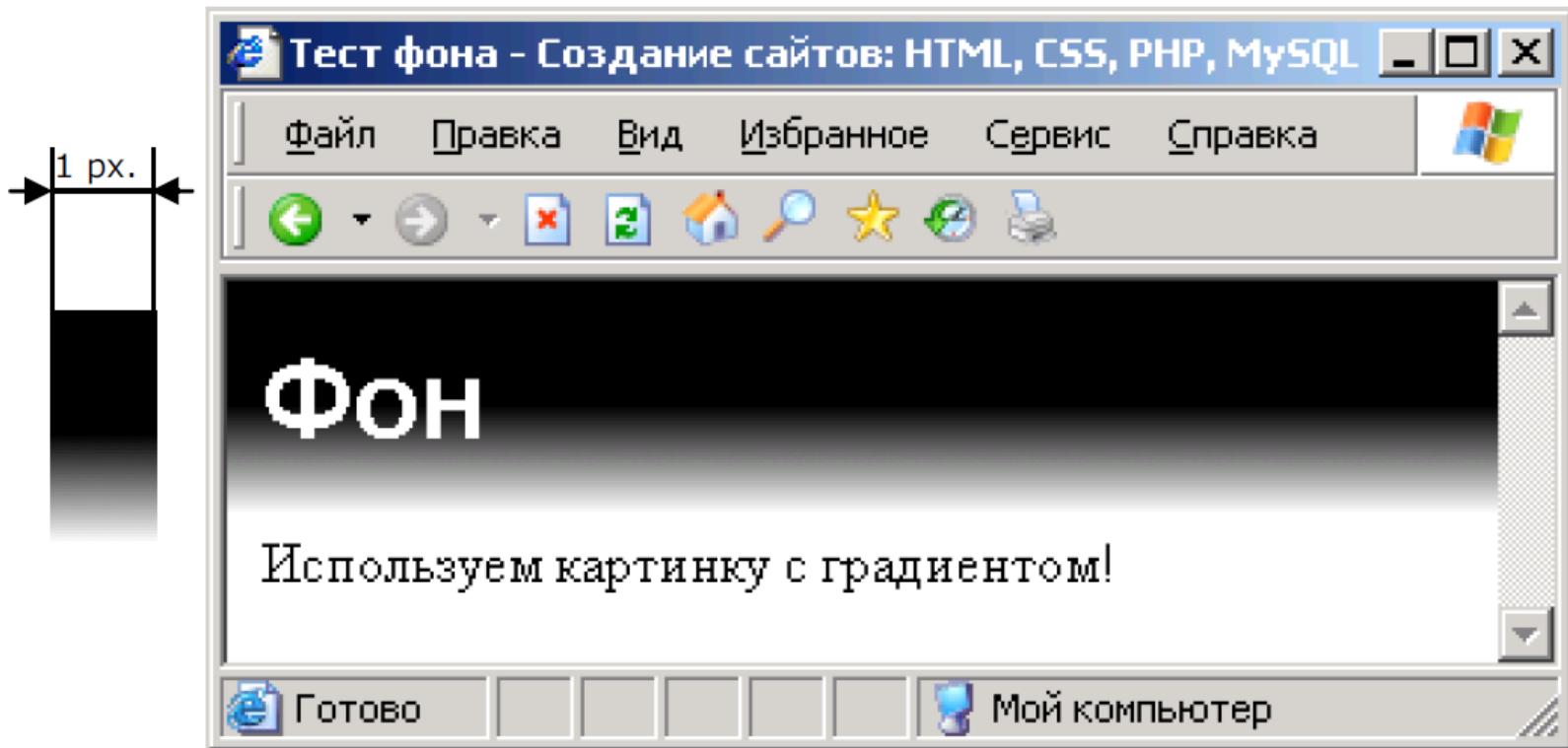
Тест!

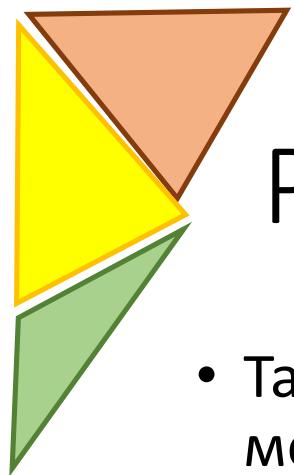




# Пример

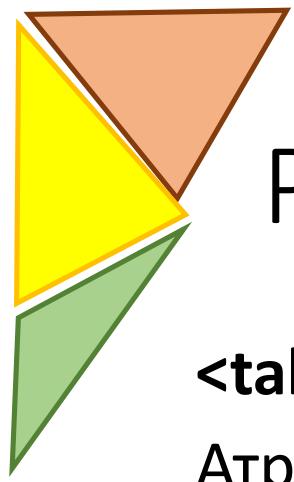
- Взять картинку шириной 1 пиксель и высотой, например, 2000 пикселей. На экране она будет копироваться только по горизонтали





# Работа с таблицами

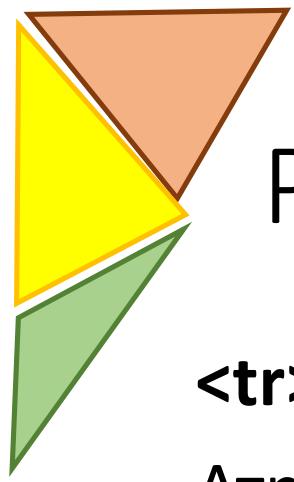
- Таблица в HTML – это совокупность данных, расположенных и связанных между собой при помощи ячеек, размещаемых в строках и колонках
- Таблица заполняется данными построчно
- Для вставки таблиц определено 3 основных тега:
  - **<table>...</table>** – тег таблицы
  - **<tr>...</tr>** – теги строк
  - **<td>...</td>** – содержимое ячейки



# Работа с таблицами

**<table>...</table>** – определяет начало и конец таблицы

Атрибуты:



# Работа с таблицами

`<tr>...</tr>` – определяет строку ячеек таблицы

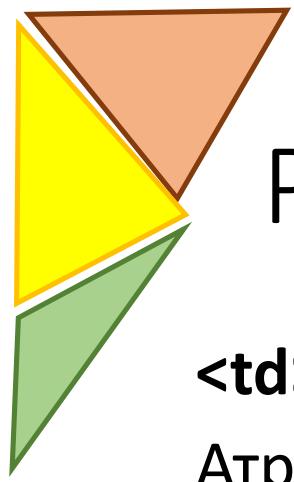
Атрибуты:

`align="..."` – определяет режим выравнивания содержимого ячеек строки (`left`, `right`, `center`, `justify`)

`background="URL"` – URL изображения, которое заполнит фон ячеек строки

`bgcolor="цвет"` – цвет фона ячеек строки

`valign="..."` – определяет режим выравнивания содержимого ячеек строки по вертикали (`top`, `middle` – по умолчанию, `bottom`)



# Работа с таблицами

**<td>...</td>** – определяет ячейку данных таблицы

Атрибуты:

`align="..."` – определяет режим выравнивания содержимого ячейки (left, right, center)

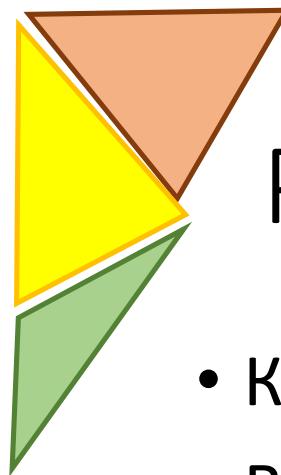
`background="URL"` – URL изображения, которое заполнит фон ячейки

`bgcolor="цвет"` – цвет фона ячейки

`valign="..."` – определяет режим выравнивания содержимого ячейки по вертикали (top, middle, bottom)

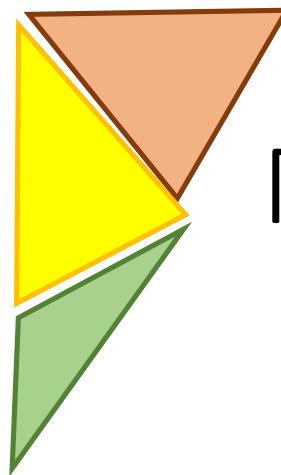
`height="N"` – высота ячейки

`width="N"` – ширина ячейки



# Работа с таблицами

- Количество тегов `<tr>...</tr>` определяет количество строк
- В каждом теге строки должно быть одно и то же число тегов `<td>...</td>`, равное числу столбцов (иначе таблица отобразится неправильно)
- Можно создавать вложенные таблицы: вкладывать таблицу в ячейку другой таблицы

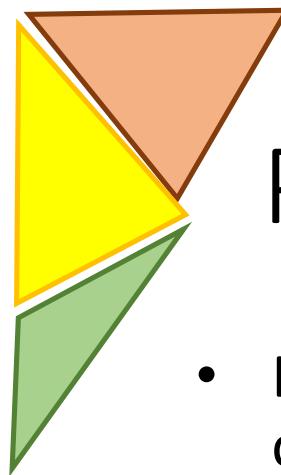


# Пример

```
<table>
  <tr> <td>1</td> <td>2</td> <td>3</td> </tr>
  <tr> <td>4</td> <td>5</td> <td>6</td> </tr>
</table>
```

В браузере:

|   |   |   |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |

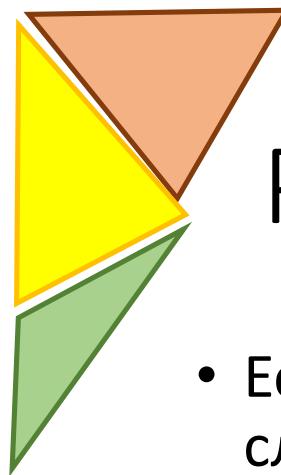


# Работа с таблицами

- Если ширина таблицы изначально не задана, то она вычисляется исходя из содержимого ячеек
- Максимальная ширина таблицы в таком случае равна ширине окна

```
<table border="1">  
<tr><td>Ширина таблицы не задана!</td></tr>  
</table>
```

Ширина таблицы не задана!

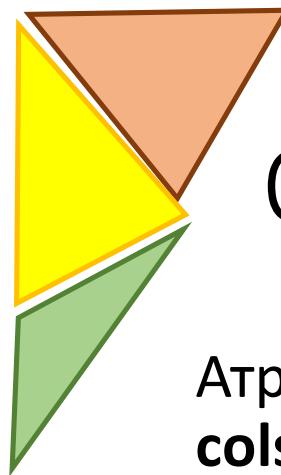


# Работа с таблицами

- Если же ширина задана атрибутом width, то браузер расставляет переносы слов в тексте ячеек таким образом, чтобы сохранить заданный размер

```
<table width="100" border="1">  
  <tr><td>Если задать атрибут width, текст начинает  
    переноситься по словам</td></tr>  
</table>
```

Если задать  
атрибут  
width, текст  
начинает пе-  
реноситься  
по словам



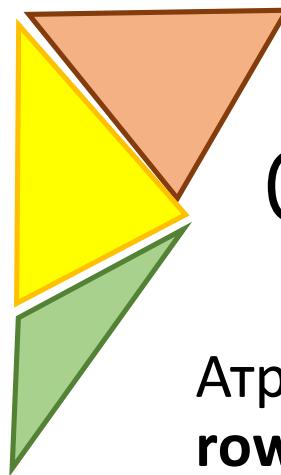
# Объединение ячеек

Атрибут  
**colspan="N"** –  
растягивает  
ячейку на N  
столбцов влево

```
<TABLE cellpadding="15" border="1">
<TR><TD colspan="2">1</TD></TR>
<TR><TD>2</TD><TD>3</TD></TR>
</TABLE>
```

В браузере:

|   |   |
|---|---|
| 1 |   |
| 2 | 3 |



# Объединение ячеек

Атрибут  
**rowspan="N"** –  
растягивает  
ячейку на N  
строк вниз

```
<TABLE cellpadding="15" border="1">
<TR><TD rowspan="2">1</TD><TD>2</TD></TR>
<TR><TD>3</TD></TR>
</TABLE>
```

В браузере:

|   |   |
|---|---|
| 1 | 2 |
|   | 3 |