

Министерство образования Республики Беларусь  
Учреждение образования  
Гомельский государственный университет им. Ф. Скорины

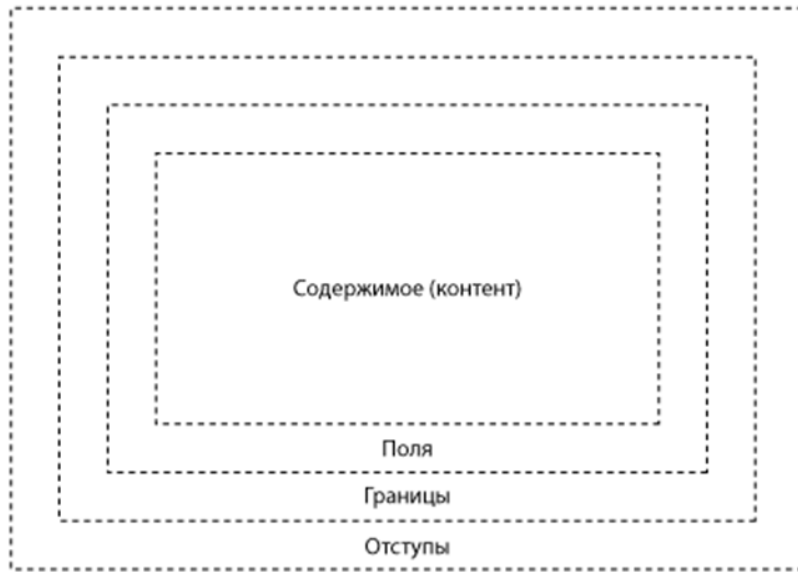
Физический факультет

«Компьютерные сети и Web-дизайн»

# Лекция - Основы блочной верстки

Лектор – ст. преподаватель Грищенко В.В.

# Блочная концепция верстки



Блочный элемент `<div></div>` состоит из набора свойств накладываемых друг на друга.

Основой любого блочного элемента выступает его контент, определяемый шириной (**width**), высотой (**height**), значением размеров полей вокруг содержимого блока (**padding**), границы контентной области блока (**border**) и отступы (**margin**) от внешнего края видимой области окна браузера.

# Поля, границы и отступы в блочной модели

**Поле (padding)** - это расстояние от внутреннего края границы или края блока до воображаемого прямоугольника, ограничивающего содержимое блока.

**Основное предназначение полей** — создать пустое пространство вокруг содержимого блочного элемента, например текста, чтобы он не прилегал плотно к краю элемента, тем самым повышая читабельность текста и улучшая внешний вид страницы.

**Границы (border)** - это линии вокруг полей элемента на одной, двух, трёх или всех четырёх его сторонах, для которых можно задать толщину, стиль и цвет.

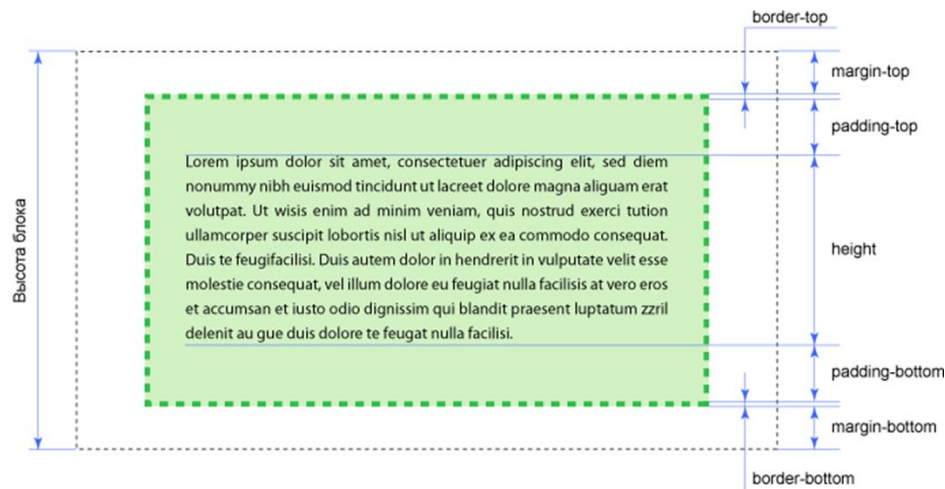
Для создания линий на отдельных сторонах элемента можно воспользоваться свойствами **border-left**, **border-top**, **border-right** и **border-bottom**, устанавливающих границу слева, сверху, справа и снизу.

**Отступ (margin)** - это пустое пространство от внешнего края границы, полей или содержимого блока.

Для отступов характерны следующие особенности:

- Отступы прозрачны, на них не распространяется цвет фона или фоновая картинка, заданная для блока. Однако если фон установлен у родительского элемента, он будет заметен и на отступах.
- Отступы в отличие от полей могут принимать отрицательное значение, это приводит к сдвигу всего блока в указанную сторону. Так, если задано **margin-left: -10px**, это сдвинет блок на десять пикселей влево.
- Для отступов характерно явление под названием «схлопывание», когда отступы у близлежащих элементов не суммируются, а объединяются меж собой.
- Отступы, заданные в процентах, вычисляются от ширины контента блока. Это касается как вертикальных, так и горизонтальных отступов.

# Ширина и высота блока



**Ширина/высота блока** это комплексная величина и складывается из нескольких значений свойств:

**width/height** — ширина/высота контента, т.е. содержимого блока;

**padding-left/padding-top** и **padding-right/padding-bottom** — поле слева/сверху и справа/снизу от контента;

**border-left/border-top** и **border-right/border-bottom** — толщина границы слева/сверху и справа/снизу;

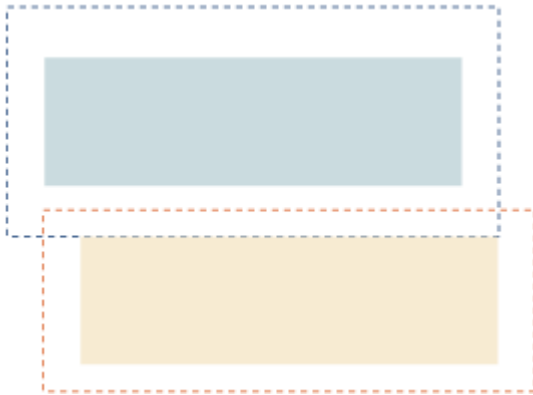
**margin-left/margin-top** и **margin-right/margin-bottom** — отступ слева/сверху и справа/снизу.

Если значение **width/height** не задано, то оно по умолчанию устанавливается как **auto**. В этом случае ширина/высота блока будет занимать всю доступную ширину/высоту при сохранении значений полей, границ и отступов. Под доступной шириной/высотой подразумевается ширина/высота контента у родительского блока, а если родителя нет, то ширина/высота контента браузера.

# Схлопывающиеся отступы и наложение слоев

**Эффект схлопывания** наблюдается в случае, когда блочные элементы расположенные рядом друг с другом по вертикали, при этом их отступы не суммируются, а объединяются между собой. Этот эффект работает только для блоков, у которых не заданы поля и границы. Для отступов слева и справа схлопывание никогда не применяется.

**Эффект схлопывания** имеет практическое значение и в первую очередь предназначено для корректного отображения текста и гарантирует, что расстояние в абзацах везде будет одинаковым. Правила вычисления единого отступа меняются в зависимости от ряда условий, так, есть разница между положительным и отрицательным значением отступа, родительским и дочерним элементом.



```
H1 {  
  background: #F0BA7D;  
  margin-bottom: 20px; }  
  
P { background: #CADADD;  
  margin: 5px 0; }
```

```
H1 {  
  background: #F0BA7D;  
  margin-bottom: 20px; }  
  
P {  
  background: #CADADD;  
  margin: -10px 0 5px; }
```

```
H1 {  
  background: #F0BA7D;  
  margin-bottom: -1em; }  
  
P {  
  background: #CADADD;  
  margin: -10px 0 5px; }
```

# Особенности использования эффекта схлопывания

При верстке страниц использование эффекта схлопывания отступов между блоками «ломает» дизайн.

## Схлопывание не срабатывает:

- для элементов, у которых установлено свойство **padding**.
- для элементов, у которых на стороне схлопывания задана граница;
- на элементах с абсолютным позиционированием, т.е. таких, у которых **position** установлено как **absolute**;
- на плавающих элементах (для них свойство **float** задано как **left** или **right**);
- для строчных элементов;
- для **<html>**.

Эффект схлопывания не срабатывает при следующих условиях:

- для элементов, у которых значение **overflow** задано как **auto**, **hidden** или **scroll** схлопывание не действует для их дочерних элементов;
- у элементов, к которым применяется свойство **clear**, не схлопывается верхний отступ с нижним отступом родительского элемента.

# Позиционирование элементов

**Позиционированием** - называется положение элемента в системе координат.

Различают четыре типа позиционирования: **нормальное**, **абсолютное**, **фиксированное** и **относительное**, которое устанавливается через свойство **position**, изменяя при этом систему координат.

## Нормальное позиционирование

При нормальном позиционировании элементы отображаются на странице в том порядке, как они идут в исходном коде HTML, при этом свойство **position** для отображаемых элементов не задано или его значение **static**.

## Абсолютное позиционирование

При абсолютном позиционировании элемент не существует в потоке документа и его положение задаётся относительно краёв браузера.

Задание такого типа позиционирования осуществляется через значение **absolute** свойства **position**, при этом координаты размещаемых элементов указываются относительно краёв окна браузера его **«видимой области»**

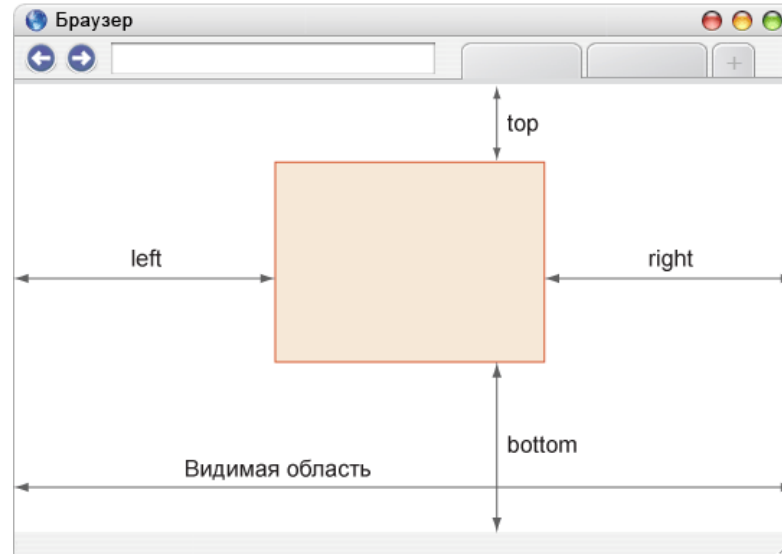
## Фиксированное позиционирование

Фиксированное положение слоя задаётся значением **fixed** свойства **position** не меняя при этом в точке на экране своего положения при прокрутке веб-страницы в отличие от абсолютного. Применяется такой тип позиционирования для создания меню, вкладок, заголовков, в общем, любых элементов, которые должны быть закреплены на странице и всегда видны посетителю.

## Относительное позиционирование

Относительное позиционирование элемента задается значение **relative** свойства **position**, при этом его положение устанавливается относительно его первоначального расположения.

# Абсолютное позиционирование



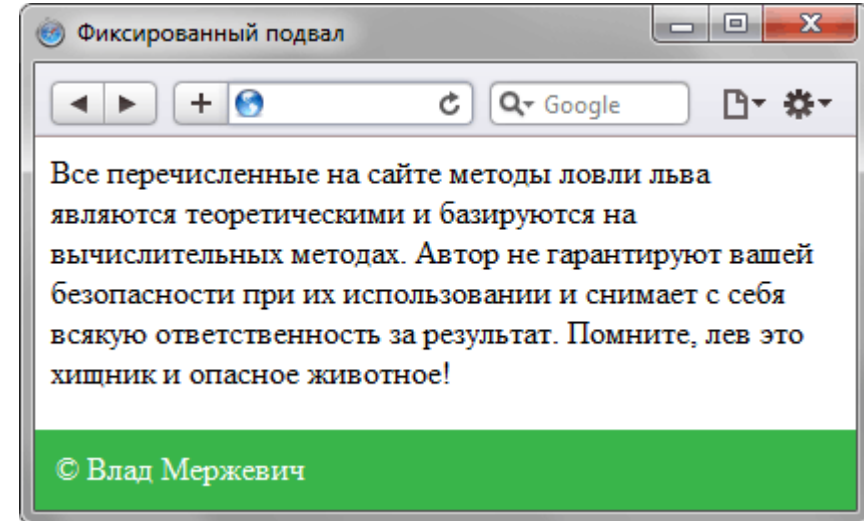
Особенности абсолютного позиционирования:

- Ширина слоя, если она не задана явно, равна ширине контента плюс значения полей, границ и отступов.
- Слой не меняет своё исходное положение, если у него нет свойств **right**, **left**, **top** и **bottom**.
- Свойства **left** и **top** имеют более высокий приоритет по сравнению с **right** и **bottom**.
- Если **left** и **right** противоречат друг другу, то значение **right** игнорируется.
- При отрицательном значении **left** позиционируемый элемент уйдёт за левую границу видимой области браузера, при этом полоса прокрутки будет отсутствовать
- При значении **left** большем чем ширина видимой области или **right** с отрицательным значением, появится горизонтальная полоса прокрутки.
- Одновременно указанные свойства **left** и **right** формируют ширину слоя, в случае если **width** не указано.
- Элемент с абсолютным позиционированием перемещается вместе с документом при его прокрутке.

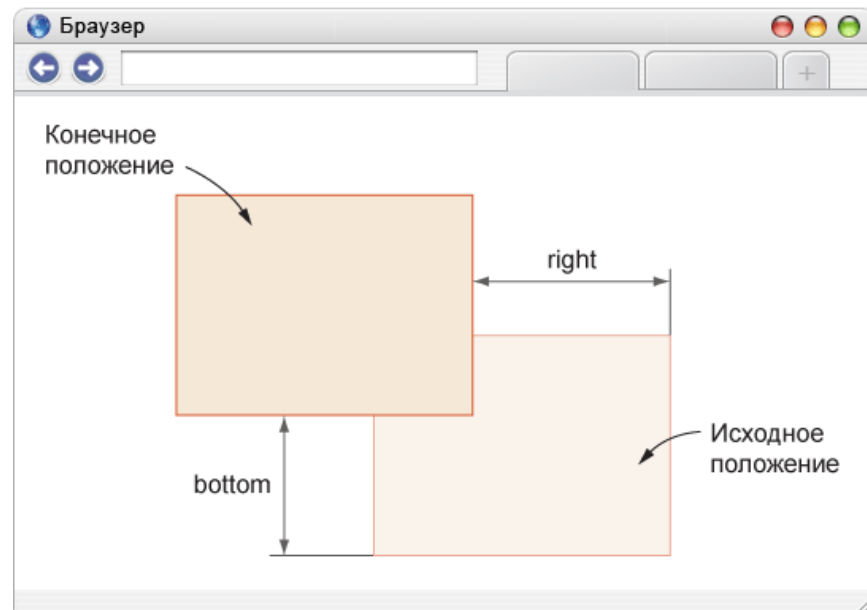
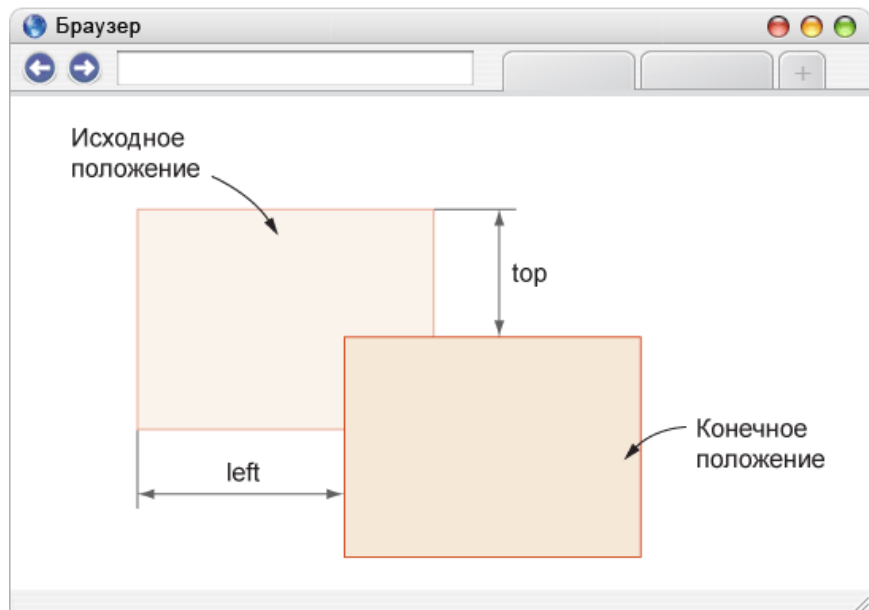


# Фиксированное позиционирование

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Фиксированный подвал</title>
<style type="text/css">
BODY {
margin-bottom: 50px;
}
#footer {
position: fixed; /* Фиксированное положение */
left: 0; bottom: 0; /* Левый нижний угол */
padding: 10px; /* Поля вокруг текста */
background: #39b54a; /* Цвет фона */
color: #fff; /* Цвет текста */
width: 100%; /* Ширина слоя */
} </style>
</head>
<body>
<div id="content"> Все перечисленные на сайте методы ловли льва
являются теоретическими и базируются на вычислительных методах.
Автор не гарантируют вашей безопасности при их использовании и
снимает с себя всякую ответственность за результат. Помните, лев это
хищник и опасное животное! </div>
<div id="footer">&copy; Влад Мержевич</div>
</body>
</html>
```



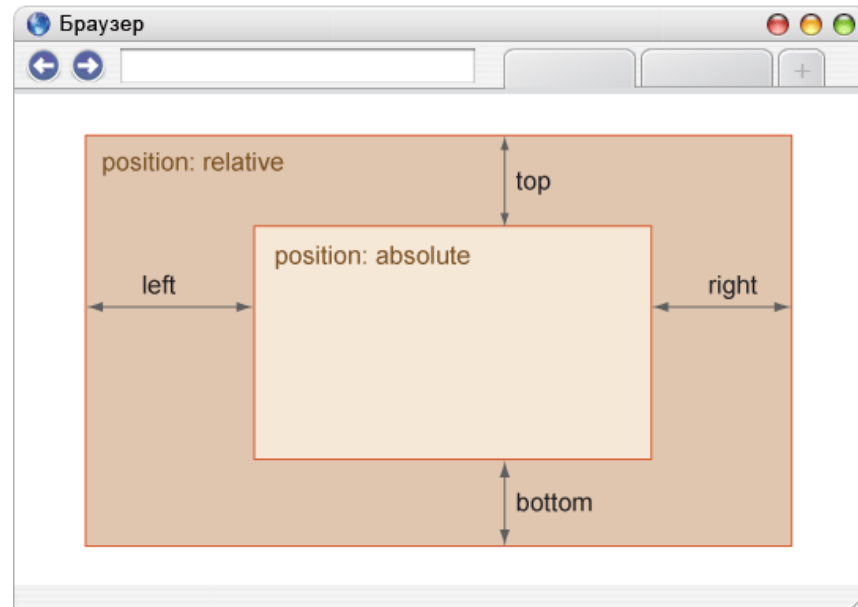
# Относительное позиционирование



Особенности относительного позиционирования:

- Относительное позиционирование не применимо к элементам таблицы
- При смещении элемента относительно исходного положения, место, которое занимал элемент, остаётся пустым и не заполняется ниже или вышележащими элементами.

# Позиционирование вложенных элементов



Для вложенных элементов одним из удобных и практичных приёмов вёрстки является использование сочетания разных типов позиционирования.

В случае если для родительского элемента задать **relative**, а для дочернего **absolute**, то произойдёт смена системы координат и положение дочернего элемента при этом указывается относительно его родителя.

Отсчёт координат ведётся от внутреннего края границы, значения полей не учитываются. В следующем примере текст располагается в правом нижнем углу слоя возле границы, игнорируя свойство **padding**.

# Блочные элементы

**Блочным** - называется элемент, который отображается на веб-странице в виде прямоугольника, который занимает всю доступную ширину, высота элемента определяется его содержимым, и он всегда начинается с новой строки.

К блочным элементам относятся теги **<address>**, **<blockquote>**, **<div>**, **<fieldset>**, **<form>**, **<h1>**,...,**<h6>**, **<hr>**, **<ol>**, **<p>**, **<pre>**, **<table>**, **<ul>**.

Особенности блочных элементов:

- Блоки располагаются по вертикали друг под другом.
- На прилегающих сторонах элементов действует эффект схлопывания отступов.
- Запрещено вставлять блочный элемент внутрь строчного (Например, `<a><h1>Заголовок</h1></a>`).
- По ширине блочные элементы занимают всё допустимое пространство.
- Если задана ширина контента (свойство **width**), то ширина блока складывается из значений **width**, полей, границ, отступов слева и справа.
- Высота блочного элемента вычисляется браузером автоматически, исходя из содержимого блока.
- Если задана высота контента (свойство **height**), то высота блока складывается из значения **height**, полей, границ, отступов сверху и снизу. При превышении указанной высоты контент отображается поверх блока.
- На блочные элементы не действуют свойства, предназначенные для строчных элементов, вроде **vertical-align**.
- Текст по умолчанию выравнивается по левому краю.

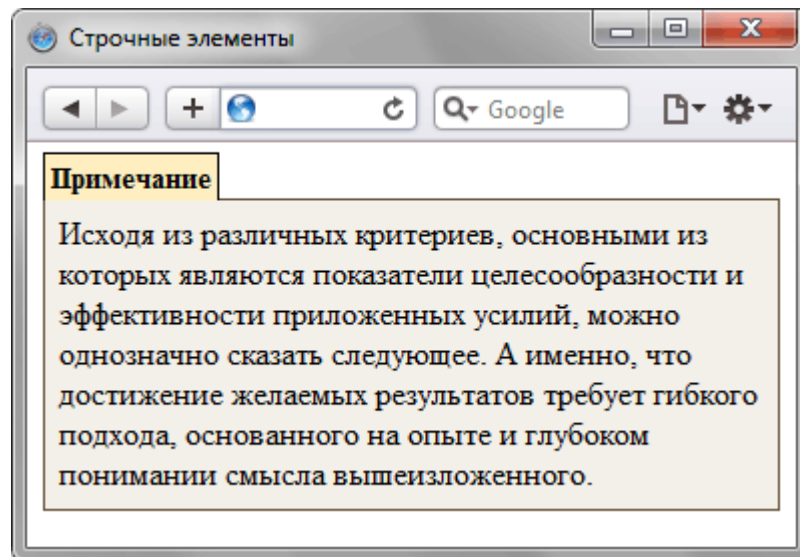
# Строчные элементы

**Строчными** - называются такие элементы документа, которые являются непосредственной частью строки. К строчным элементам относятся теги `<img>`, `<span>`, `<a>`, `<q>`, `<code>`, а также элементы, у которых свойство `display` установлено как `inline`. выделения.

Особенности строчных элементов:

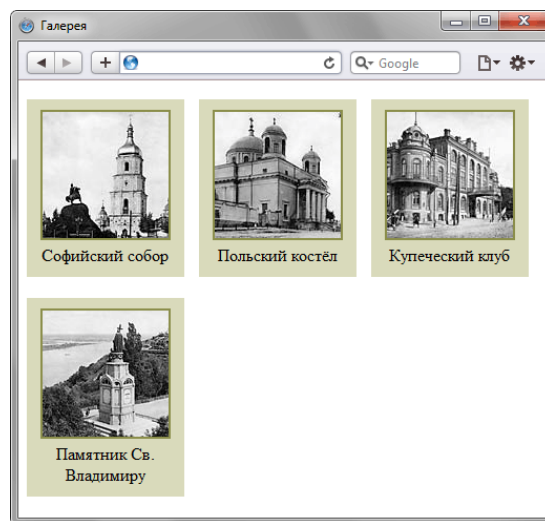
- Внутри строчных элементов допустимо помещать текст или другие строчные элементы. Вставлять блочные элементы внутри строчных запрещено.
- Эффект схлопывания отступов не действует.
- Свойства, связанные с размерами (**width, height**) не применимы.
- Ширина равна содержимому плюс значения отступов, полей и границ.
- Несколько строчных элементов идущих подряд располагаются на одной строке и переносятся на другую строку при необходимости.
- Можно выравнивать по вертикали с помощью свойства **vertical-align**.

```
<style type="text/css">
.notetitle {
border: 1px solid black; /* Параметры рамки */
border-bottom: none; /* Убираем границу снизу */
padding: 3px; /* Поля вокруг текста */
display: inline; /* Устанавливаем как строчный элемент */
background: #ffeebf; /* Цвет фона */
font-weight: bold; /* Жирное начертание */
font-size: 0.9em; /* Размер шрифта */
margin: 0; /* Убираем отступы */
white-space: nowrap; /* Запрещены переносы текста */
}
.note {
border: 1px solid #634f36; /* Параметры рамки */
background: #f3f0e9; /* Цвет фона */
padding: 7px; /* Поля вокруг текста */
margin: 0 0 1em 0; /* Значение отступов */
}
</style>
```



**Результат преобразования блочного  
элемента в строчный**

# Строчно-блочные элементы



В HTML нет тега, который относится к строчно-блочным элементам, его можно определить, задав элементу свойство **display** со значением **inline-block**.

```
div {  
  display: inline-block;  
}
```

Особенности строчно-блочных элементов:

- Внутри строчно-блочных элементов допустимо помещать текст, строчные или блочные элементы.
- Высота элемента вычисляется браузером автоматически, исходя из содержимого блока.
- Ширина равна содержимому плюс значения отступов, полей и границ.
- Несколько элементов идущих подряд располагаются на одной строке и переносятся на другую строку при необходимости.
- Можно выравнивать по вертикали с помощью свойства `vertical-align`.
- Разрешено задавать ширину и высоту.
- Эффект схлопывания отступов не действует.