

КАСКАДНЫЕ ТАБЛИЦЫ СТИЛЕЙ (ПРОДОЛЖЕНИЕ)

Модели отображения html-элементов.

CSS 2.1 определяет 4 модели макетов страницы:

- *block* – блоковая модель;
- *inline* – встроенные в общий поток отображения элементы;
- *table* – табличная модель;
- *positioned* – фиксированное отображение элементов.

```
display: [ <display-outside> || <display-inside> ] |  
  <display-listitem> | <display-internal> |  
  <display-box> | <display-legacy>
```

```
<display-outside> = block | inline | run-in ;
```

```
<display-inside> = flow | flow-root | table | flex | grid | ruby
```

```
<display-listitem> = list-item &&  
  <display-outside>? && [ flow | flow-root ]?
```

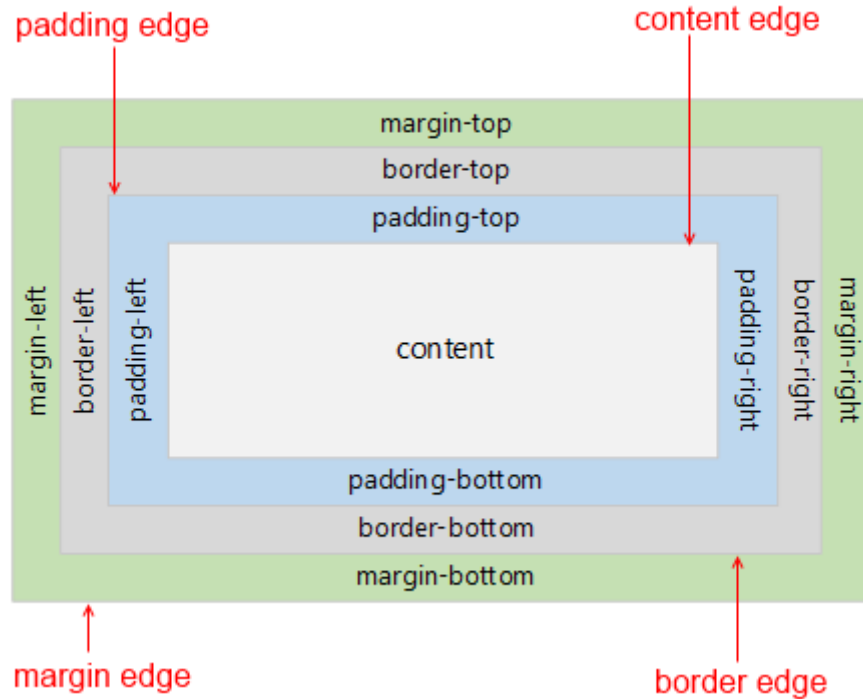
```
<display-internal> = table-row-group | table-header-group |  
  table-footer-group | table-row | table-cell |  
  table-column-group | table-column |  
  table-caption |  
  ruby-base | ruby-text | ruby-base-container |  
  ruby-text-container
```

```
<display-box> = contents | none
```

```
<display-legacy> = inline-block | inline-list-item |  
  inline-table | inline-flex | inline-grid
```

Value set	Value
Basic values (CSS 1)	none, inline, block, list-item
Extended values (CSS 2.1)	inline-block
Table model values (CSS 2.1)	inline-table table, table-caption, table-cell, table-column, table-row, table-column-group, table-footer-group, table-header-group, table-row-group
Flexbox model values (CSS3)	flex, inline-flex
Grid box model values (CSS3)	grid, inline-grid
Ruby formatting model values (CSS3)	ruby, ruby-base, ruby-text, ruby-base-container, ruby-text-container
Experimental values	run-in, contents

Блочная модель отображения элементов.



display: none | block | inline | inline-block

Поля, рамки и отступы являются частью блочной модели CSS.

- **float:** none | left | right
- **clear:** none | left | right | both
- **margin-top:** px | % | cm
- **margin-bottom:** px | % | cm
- **margin-left:** px | % | cm
- **margin-right:** px | % | cm
- **margin:** 4числа
- **padding:** 4числа - **padding-top**, **padding-right**, **padding-bottom** и **padding-left**.
- **width** и **height**
- **box-sizing:**
 - border-box - width = content (padding, border, margin не учитываются)
 - padding-box - width = content + padding. Поддерживается только FireFox.
 - content-box - width = content + padding + border (margin не учитывается).

Позиционирование элементов.

- **position:**

static - статичный элемент. Такие элементы не могут позиционироваться с помощью свойств *top*, *left*, *right*, *bottom* и *z-index*.

relative - Результат применения этого свойства к *table-*-group*, *table-row*, *table-column*, *table-cell* и *table-caption* элементам - не определён.

absolute | fixed | sticky

Пример 1.

```
<div style="position: absolute; top: 10px; left: 10px;
```

```
        width: 10px; height: 10px; background-color: blue">
</div>
```

Пример 2.

```
<div style="position: absolute; left: 25%; top: 25%;
        width: 50%; height: 50%;
        border: 2px solid black">
</div>
```

- **z-index:** auto | число

Пример 3. Что увидит пользователь?

```
<span style="text-shadow: 3px 3px 1px #888">Текст </span>
<span style="position: relative;">
    Текст
    <span style="position: absolute; top: 3px; left: 3px;
            z-index: -1; color: gray">
        Текст
    </span>
</span>
```

Пример 4.

```
.элемент { box-shadow: px px px Цвет }
.f1 { box-shadow: inset 0 0 30px #000000; }
.f3 { box-shadow: inset 0 0 30px hsl(0, 0%, 0%),
        inset 0 0 70px hsla(0, 97%, 53%, 1); }
```

Видимость элементов.

- **visibility:** visible | hidden | collapse
- **overflow:** visible | hidden | auto | scroll
- **clip:** auto | clip(top right bottom left)
- **display:**
- **opacity:** число

«Гибкая» модель отображения элементов.
(flex layout mode)

Пример 1. Из www.w3.org (в [браузере](#))

```
#deals {
  display: flex;
  flex-flow: row wrap-reverse;
}
.sale-item {
  display: flex;
  flex-flow: column;
}
.sale-item > img {
  order: -1;
  align-self: center;
}
.sale-item > button {
  margin-top: auto;
}
```

```
<div id="deals">
  <div class="sale-item">
    <h1>Computer 1</h1>
    <p>This is the best computer you can buy.</p>
    <ul>
      <li>Computer
      <li>Monitor
    </ul>
    
    <button>BUY NOW</button>
  </div>
  <div class="sale-item"> ... </div>
</div>
```



Некоторые возможности отображения блоков не применимы к flex-контейнерам:

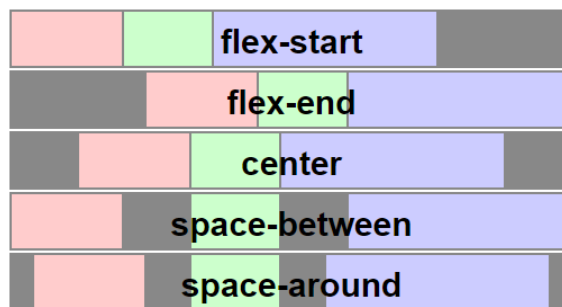
- *column-**
- *float, clear, vertical-align*
- *::first-line*
- *::first-letter*

Свойства *flex-контейнера*: (css-tricks.com)

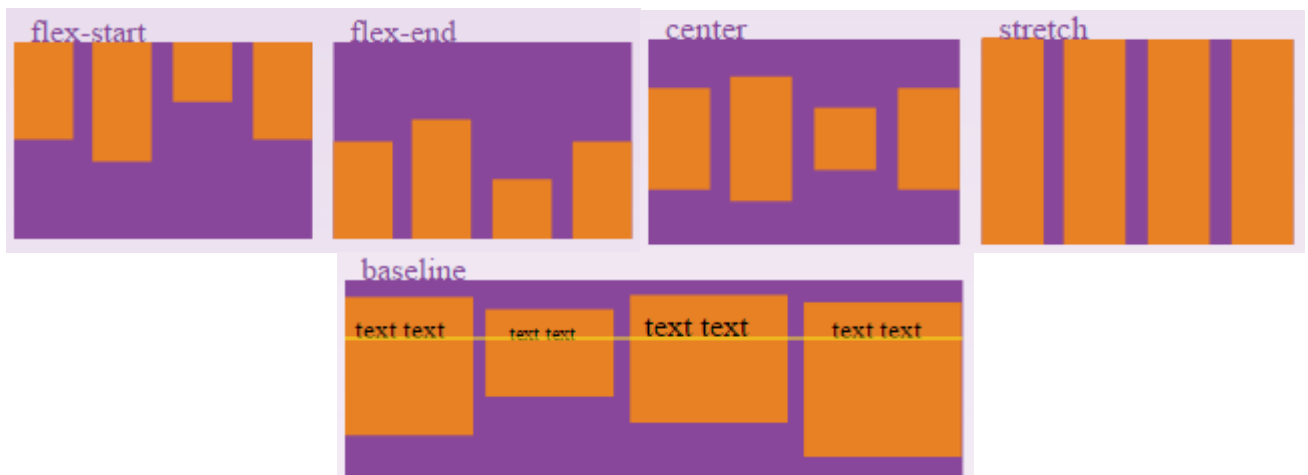
- **display:** flex | inline-flex
- **flex-direction:** row | row-reverse | column | column-reverse



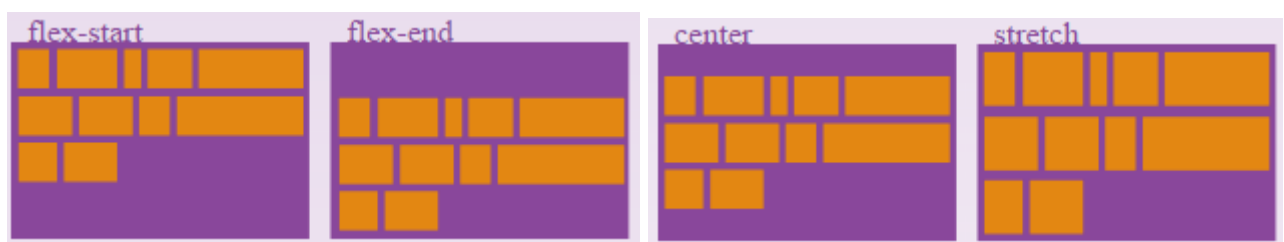
- **flex-wrap:** nowrap | wrap | wrap-reverse
- **flex-flow**
- **justify-content:** flex-start | flex-end | center | space-between | space-around

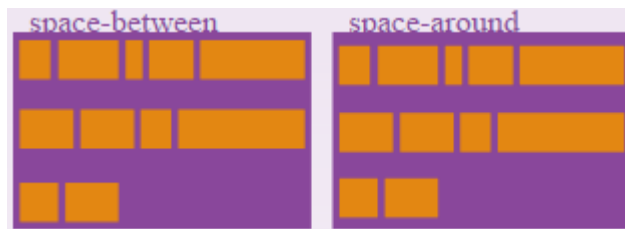


- **align-items:** flex-start | flex-end | center | baseline | stretch



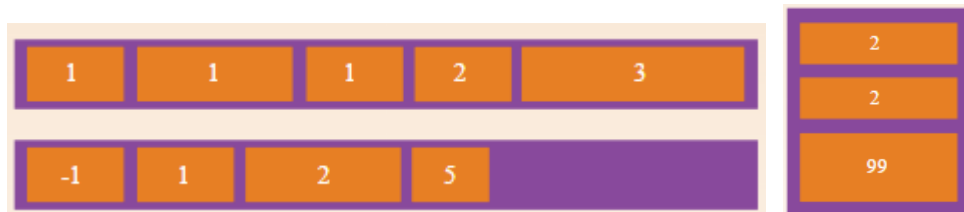
- **align-content:** flex-start | flex-end | center | space-between | space-around | stretch



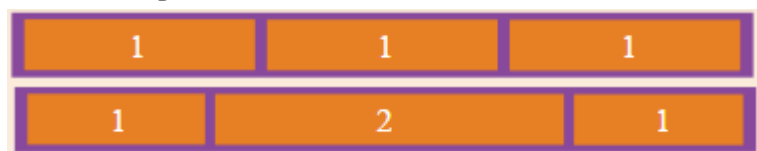


Свойства *flex-элементов*:

- **order**: число



- **flex-grow**: число > 0



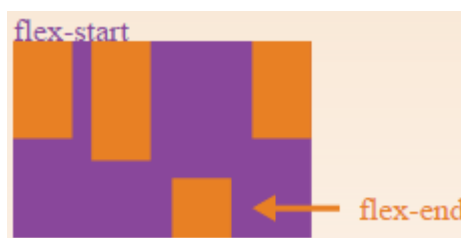
- **flex-shrink**: число > 0

- **flex-basis**: число | auto | content

- **flex**: none |

[<'flex-grow'> <'flex-shrink'>? || <'flex-basis'>]

- **align-self**: auto | flex-start | flex-end | center | baseline | stretch



Правила форматирования содержимого страницы.

Группы стилевых правил.

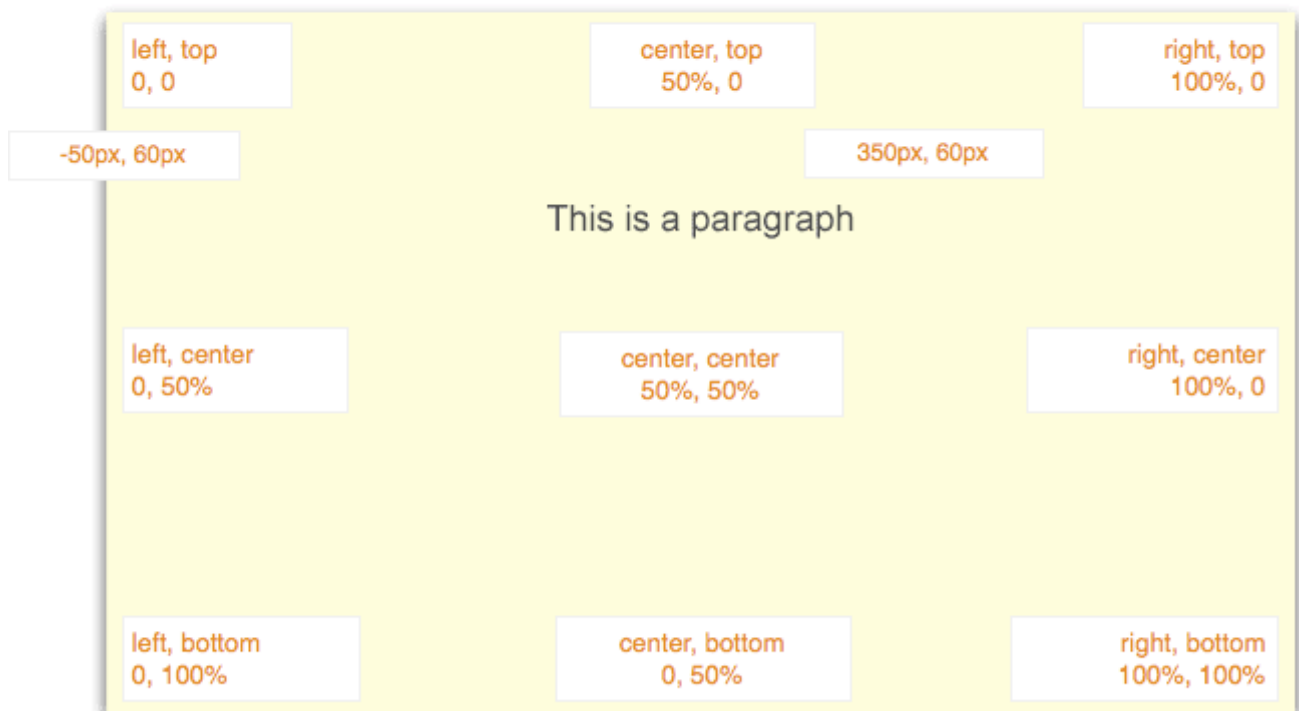
Цвет и фон	Шрифты	Переходы
Рамки	Таблицы	Колонки
Блочные элементы	Списки	Фильтры
Flexible элементы	Анимация	
Текст	Трансформации	

Стандартные значения свойств: *inherit* | *initial*

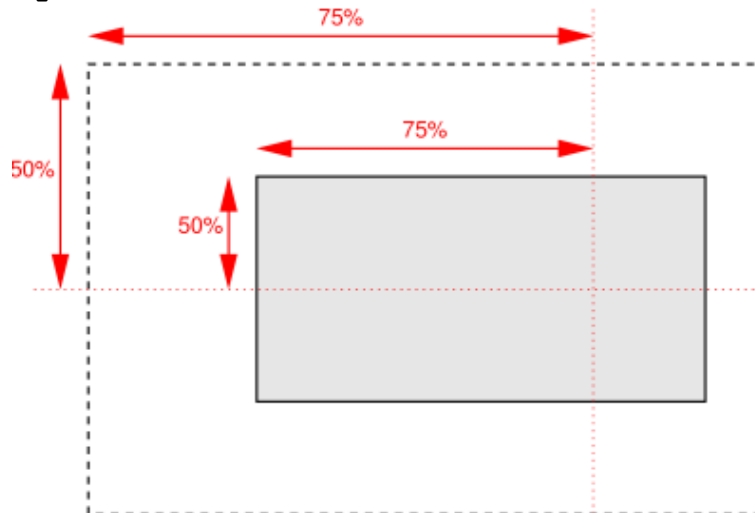
Цвет и фон.

- **color:** имя цвета | #RRGGBB | rgb(R,G,B) (*CSS1*)
- **background-color:** цвет | transparent (*CSS1*)
- **background-image:** url(адрес рисунка) (*CSS1*)
множественные фоновые изображения (CSS3):
background: url('1.png'), url('2.png'), url('3.png'), green;
- **background-size:** w h | auto | cover | contain (*CSS3*)
background-size: 100% 50%, 300px 400px, auto;
- **background-repeat:** repeat | repeat-x | repeat-y | no-repeat (*CSS1*)
- **background-attachment:** fixed | scroll | local (*CSS1*)
- **background-position:** left | center | right | top | bottom

350px, -60px



background-position: cm|% cm|% (CSS1)



CSS3:

```
background: url('1.png') center, url('2.png'),  
           url('3.png') left bottom;
```

```
background-image: url(flower.png), url(ball.png), url(grass1.png);
```

```
background-position: center center, 20% 80%, top left;
```

```
background-repeat: no-repeat, no-repeat, no-repeat;
```

- **background-clip:** border-box | padding-box | content-box (CSS3)



- **background-origin:**

padding-box | border-box | content-box (CSS3)



- **background-blend-mode:**

режим наложения слоёв

normal | multiply | screen | overlay | darken |
lighten | color-dodge | saturation | color |
luminosity (CSS3)

- **background:** свойство1 ... свойствоN (CSS1)

```
.bg {  
  background: url(back.gif) no-repeat 10% 20%  
}
```


Градиенты: Линейные и Радиальные

- *Линейный* фоновый градиент:

```
linear-gradient (  
    направление,  
    цвет_1, позиция_цвета_1,  
    ...  
    цвет_N, позиция_цвета_N  
)
```

Пример.

```
body { background: linear-gradient( 90deg,  
                                    #ffffff 0%,  
                                    #e4e4e4 50%,  
                                    #ffffff 100%); }
```

- *Радиальный* фоновый градиент:

```
radial-gradient (  
    X Y,                                     // начало градиента  
    [circle | ellipse]                       // форма  
    [closest-side | contain | closest-corner |  
    farthest-side | farthest-corner | cover], // размеры  
    цвет_1, позиция_цвета_1,  
    ...  
    цвет_N, позиция_цвета_N  
)
```

Пример.

```
background: radial-gradient(center, ellipse cover, #ffffff 72%,  
                             #dddddd 100%);
```

- *Повторяющиеся* фоновые градиенты.

```
repeating-linear-gradient (...);  
repeating-radial-gradient (...);
```

- «Сложный» градиент. Примеры с lea.verou.me/css3patterns/



```
body {  
    background-color:white;  
    background-image:
```

```

radial-gradient(midnightblue 9px, transparent 10px),
repeating-radial-gradient(
  midnightblue 0, midnightblue 4px, transparent 5px,
  transparent 20px, midnightblue 21px, midnightblue 25px,
  transparent 26px, transparent 50px
);
background-size: 30px 30px, 90px 90px;
background-position: 0 0;
}

```



```

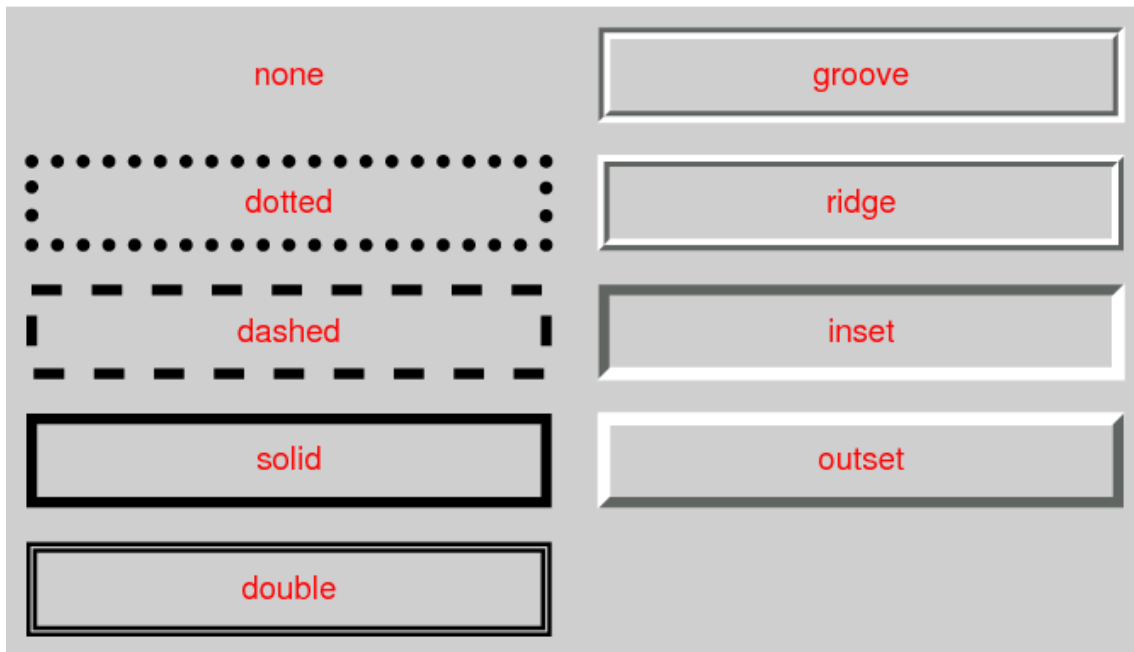
body {
  background:
    linear-gradient(45deg, #dca 12%, transparent 0,
      transparent 88%, #dca 0),
    linear-gradient(135deg, transparent 37%, #a85 0,
      #a85 63%, transparent 0),
    linear-gradient(45deg, transparent 37%, #dca 0,
      #dca 63%, transparent 0) #753;
  background-size: 25px 25px;
}

```



Рамки.

- **border-width:** px | thin | medium | thick (CSS1)
- **border-color:** цвет (CSS1)
- **border-style:** none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset (CSS1)



- **border-radius:** число {1, 4} (CSS3)

Эквивалентные стили:

`border-radius: 4em;`

`border-top-left-radius: 4em;`

`border-top-right-radius: 4em;`

`border-bottom-right-radius: 4em;`

`border-bottom-left-radius: 4em;`

`border-radius:`

`2em 1em 4em / 0.5em 3em;`

`border-top-left-radius: 2em 0.5em;`

`border-top-right-radius: 1em 3em;`

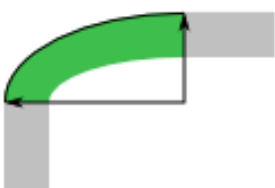
`border-bottom-right-radius: 4em 0.5em;`

`border-bottom-left-radius: 1em 3em;`

Пример 1.

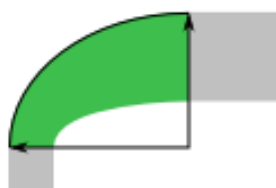
A

`border-width: 20px;`
`border-radius:`
`80px / 40px;`



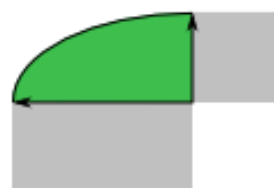
B

`border-width:`
`40px 20px;`
`border-radius:`
`80px / 60px;`



C

`border-width:`
`40px 80px;`
`border-radius:`
`80px / 40px;`



D

`border-width:`
`60px 40px;`
`border-radius:`
`80px / 40px;`



Пример 2.

```
p { width: 70px; height: 70px; border: solid 30px;
border-color: orange orange silver silver;
border-top-right-radius: 100%; }
```



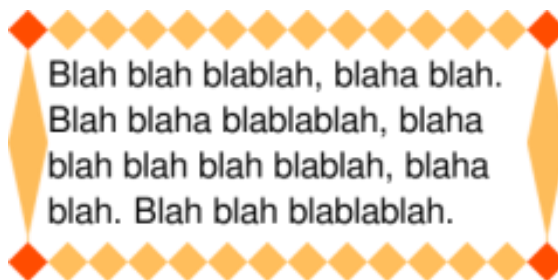
- **border-image** (CSS3)
- **border-image-source**: url
- **border-image-outset**: px {1, 4}
- **border-image-repeat**: stretch | repeat | round | space
- **border-image-slice**: px {1, 4}
- **border-image-width**: [% | px | auto]{1,4}

Пример 1 из W3C.

```
DIV {
border: double orange 1em;
border-image: url("border.png") 27 round stretch;
}
```

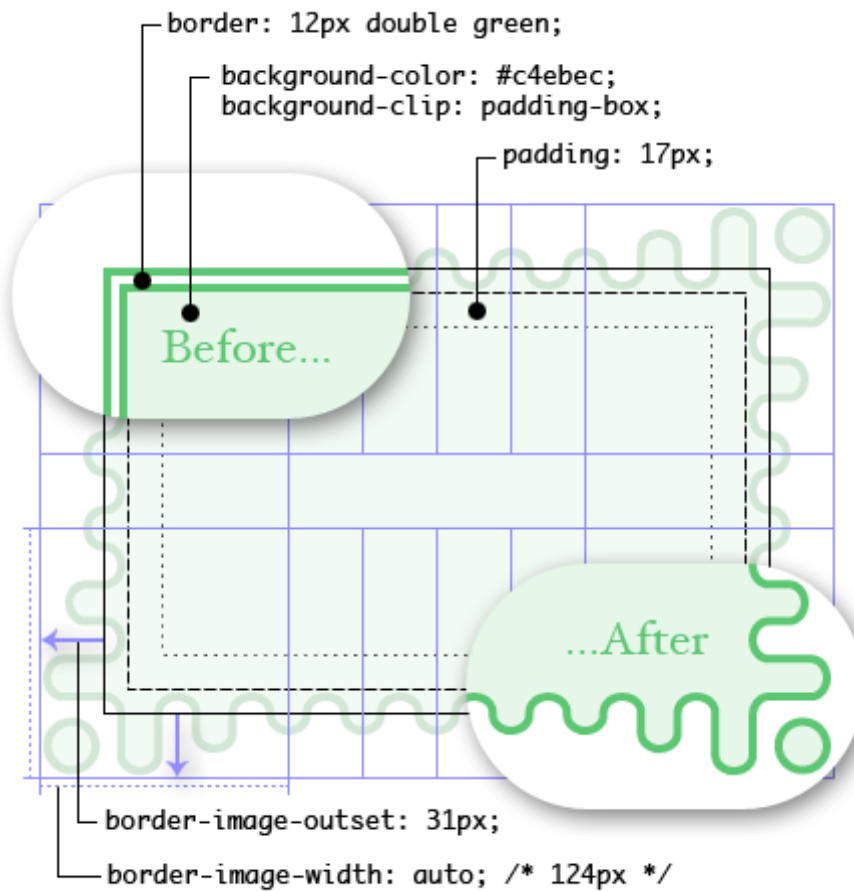
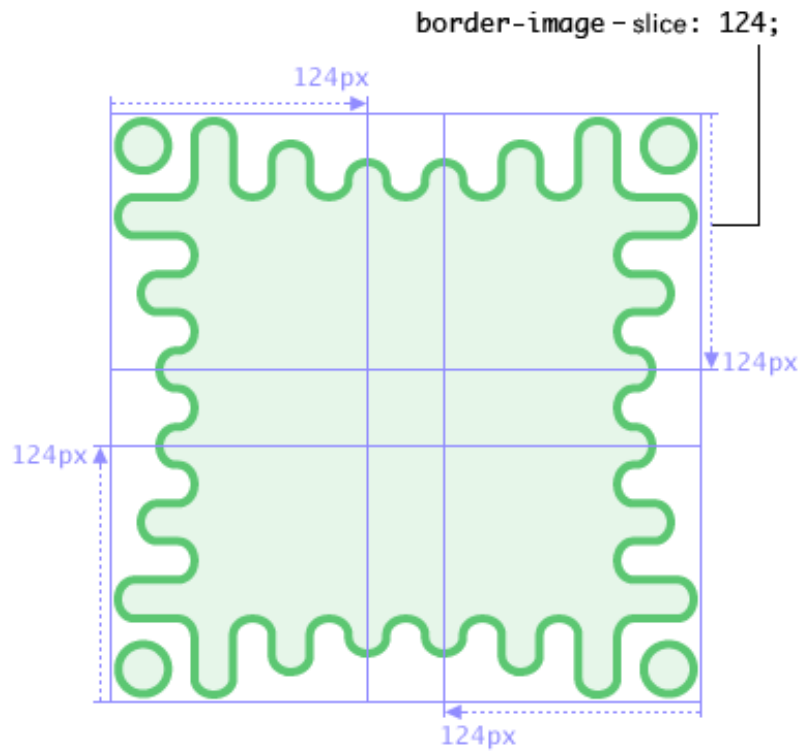


91x91px.



DIV: 12em x 5em

Пример 2 из W3C.



— Border Box - - - - - Padding Box
- - - - - Image Parts (Slices) ······ Content Box

- border-top: border- right: (CSS1)

border-left: **border-bottom:**
.red2 {border-top: 20cm red}

- **border:** (*CSS1*)
.red2 {border: 20cm red double}

Шрифт.

- **font-size:**
pt | % | em | ex |
xx-small, x-small, small, medium, large, xlarge, xx-large
smaller | larger
// Масштабирующий множитель: CSS1 - 1.5, CSS2 - 1.2.
- **font-weight:** число 100..900 | normal | bold | bolder | lighter
- **font-style:** normal | italic | oblique
- **font-variant:** normal | small-caps
- **font-family:**
serif | sans-serif | cursive | fantasy | monospace

Serif

Sans-serif

Cursive

FANTASY

Monospace

F

Sans-serif

F

Serif

F

Serif
(red serifs)

Примеры:

```
.sserif { font-family: "Times New Roman", sans-serif }  
.sserif { font-family: Verdana, sans-serif }
```

- **font:**
.curs {font: serif 12 italic}

«Нестандартные» шрифты.

@font-face появилось в CSS 2, в CSS 2.1 оно отсутствовало, появилось в CSS3

Форматы шрифтов:

Embedded OpenType (EOT) – Internet Explorer

TrueType (TTF)

Scalable Vector Graphics (SVG)

Web Open Font Format (WOFF).

Пример. Файл stylesheet.css для *Bebas Neue*:

```
@font-face {
  font-family: 'BebasNeueRegular';
  src: url('BebasNeue-webfont.eot');
  src: url('BebasNeue-webfont.eot?#iefix')
    format('embedded-opentype'),
    url('BebasNeue-webfont.woff') format('woff'),
    url('BebasNeue-webfont.ttf') format('truetype'),
    url('BebasNeue-webfont.svg#BebasNeueRegular')
    format('svg');
  font-weight: normal;
  font-style: normal;
}
```

Изменяем пути:

















```
@font-face {
  font-family: 'BebasNeueRegular';
  src: url('../fonts/BebasNeue-webfont.eot');
  src: url('../fonts/BebasNeue-webfont.eot?#iefix')
  ...
}
```

































































Стилевое правило для использования шрифта:

```
#content h1 {
  font-family: 'BebasNeueRegular';
}
```

Иконочные шрифты.

Шрифт *awesome*

-  Новые иконки в 4.5
-  Веб-приложения
-  Иконки жестов
-  Транспорт
-  Гендерные
-  Типы файлов
-  Вращающиеся
-  Элементы форм
-  Платежные системы
-  Диаграммы
-  Валюты
-  Редактирование
-  Направление
-  Видео-плеер
-  Бренды
-  Медицина

- | | | | |
|---|---|--|--|
|  battery-half |  battery-quarter |  battery-three-quarters |  bed |
|  beer |  bell |  bell-o |  bell-slash |
|  bell-slash-o |  bicycle |  binoculars |  birthday-cake |
|  bluetooth |  bluetooth-b |  bolt |  bomb |
|  book |  bookmark |  bookmark-o |  briefcase |
|  bug |  building |  building-o |  bullhorn |
|  bullseye |  bus |  cab (синоним) |  calculator |
|  calendar |  calendar-check-o |  calendar-minus-o |  calendar-o |
|  cloud-upload |  code |  code-fork |  coffee |
|  cog |  cogs |  comment |  comment-o |
|  commenting |  commenting-o |  comments |  comments-o |
|  compass |  copyright |  creative-commons |  credit-card |
|  crop |  crosshairs |  cube |  cubes |
|  cutlery |  dashboard (синоним) |  database |  desktop |
|  diamond |  dot-circle-o |  download |  edit (синоним) |
|  ellipsis-h |  ellipsis-v |  envelope |  envelope-o |

Пример в [браузере](#).

Переходы.

Пример. В [браузере](#).

- **Стили по-умолчанию:**

```
#content {
  ...
  border: 1px solid #bfbfbf;
  color: #ffffff;
  text-shadow: 0px 1px black;
}
```

- **Активное («включенное») состояние:**

```
#content:hover {
  border: 1px solid #000000;
  color: #000000;
  text-shadow: 0px 1px white;
}
```

- **Плавный переход от состояния 1 к состоянию 2:**

```
#content {
```



```

...
transition: all 1s ease 0s;
}

```

Свойства переходов:

- **transition-property:** CSS-свойство | all
- **transition-duration:** сек

Пример,

```

#content a {
...
transition-property: border, color;
transition-duration: 2s, 3s;
}

```

- **transition-timing-function:** linear | ease | ease-in | ease-out | ease-in-out | step-start | step-end | steps(int, start|end) | cubic-bezier(n, n, n, n)



<http://cubic-bezier.com>

- **transition-delay:** сек
- **transition:** property duration timing-function delay

2D-трансформации CSS3.

```

a:hover { transform: scale(3.0); }

```

- **transform:** вид_2D_трансформации

Виды трансформаций:

- `scale(v)` – масштабирование
- `translate(x, y)` – сдвиг по *x* и *y*
- `rotate(deg)` – поворот
- `skewX(deg)` – скашивание по *X*
- `skewY(deg)` – скашивание по *Y*
- `skew(degX, degY)` – скашивание по *X* и *Y*
- `matrix(scaleX, skewY, skewX, scaleY, translateX, translateY)`

- **transform-origin:** *x y* – центр трансформации

3D-трансформации CSS3.

- **transform:** вид_3D_трансформации
 - `matrix3d(n,n,n,n,n,n,n,n,n,n,n,n,n,n,n,n)`
 - `translate3d(x,y,z)`
 - `translateX(x) | translateY(y) | translateZ(z)`
 - `scale3d(x,y,z)`
 - `scaleX(x) | scaleY(y) | scaleZ(z)`
 - `rotate3d(x,y,z,angle)`
 - `rotateX(angle) | rotateY(angle) | rotateZ(angle)`
 - `perspective(n)`
- **transform-style:** `flat|preserve-3d`
- **perspective-origin:** `x-axis y-axis`
- **perspective:** `length|none`
- **backface-visibility:** `visible|hidden`

CSS3-анимация.

CSS3-анимация состоит из двух компонентов:

- объявление `@keyframes`

```
@keyframes GreenLight {  
  0%    { text-shadow: 0px 0px 4px green; }  
  50%   { text-shadow: 0 0 20px green; }  
  100%  { text-shadow: 0px 0px 4px green; }  
}
```

Или:

```
@keyframes GreenLight {  
  from { text-shadow: 0px 0px 4px green; }  
  50%   { text-shadow: 0 0 20px green; }  
  to   { text-shadow: 0px 0px 4px green; }  
}
```
- свойство `animation`

```
ЭЛЕМЕНТ {  
  animation: GreenLight 1.5s infinite ease-in  
}
```

Пример

Параметры свойства `animation`:

- **animation-name:** `ИМЯ | none`
- **animation-duration:** `время`
- **animation-timing-function:** `linear | ease | ease-in | ease-out | ease-in-out | step-start | step-end | steps(int,start|end) | cubic-bezier(n,n,n,n)`
- **animation-iteration-count:** `число | infinite`

- **animation-direction:** normal|reverse|alternate|alternate-reverse
- **animation-play-state:** running | paused
- **animation-delay:** время
- **animation-fill-mode:** none|forwards|backwards|both
- **animation:** name duration timing-function delay
iteration-count direction fill-mode play-state

Текст.

- **text-shadow:** px px px Цвет
- **text-transform:** capitalize | uppercase | lowercase
- **text-decoration:** none | underline | overline |
line-through | blink
- **text-align:** left | right | center | justify
- **text-indent:** см | %
- **vertical-align:** baseline | middle | _{sub} | ^{super} | text-top |
text-bottom | top | bottom | %
- **line-height:** pt | % | normal
- **letter-spacing:** число
- **word-spacing:** число
- **white-space:** normal | pre | nowrap
- **direction:** ltr | rtl

Отображение списков.

- **list-style-type:** none | disc | circle | square |
decimal | lower-roman | upper-roman |
lower-alpha | upper-alpha
- **list-style-image:** url("адрес изображения") | none;
- **list-style-position:** inside | outside
- **list-style**

Множественные колонки для адаптивных веб-дизайнов

- **column-width:** 12em;
- **column-count:** 4;
- **column-gap:** 2em;
- **column-rule:** thin dotted #999;
- **column-width:** 12em;

Фильтры.

- filter:** none | blur() | brightness() | contrast() |
drop-shadow() | grayscale() | hue-rotate() |
invert() | opacity() | saturate() | sepia() | url()

[Пример](#) в браузере.

Курсор.

cursor: url("url") | auto | crosshair | default |
pointer | progress | move | e_resize |
ne_resize | nw_resize | n_resize | se_resize |
sw_resize | s_resize | w_resize | text | wait | help



[Все курсоры](#)

Префиксы производителей ПА.

```
.round{  
  -ms-border-radius: 10px;    /* Microsoft */  
  -o-border-radius: 10px;     /* Opera */  
  -moz-border-radius: 10px;   /* Mozilla (Firefox) */  
  -webkit-border-radius: 10px; /* Webkit (Safari и Chrome) */  
  border-radius: 10px;        /* W3C */  
}
```