

CSS (каскадные таблицы стилей)

— формальный язык описания внешнего вида документа, написанного с использованием языка разметки

Преимущества

- централизованное управление отображением множества документов;
- упрощенный контроль внешнего вида веб-страниц;
- наличие разработанных дизайнерских техник;
- возможность использования различных стилей для одного документа, в зависимости от устройства.

Общий вид CSS-правила

```
селектор {  
    свойство: значение;  
    свойство: значение;  
    свойство: значение;  
}  
селектор, селектор, селектор {  
    свойство: значение;  
}
```

Селектор (selector)

это часть CSS-правила, которая сообщает браузеру, к какому элементу (или элементам) веб-страницы будет применён стиль

Простейшие селекторы

Селектор "тег"

```
div{}
```

Селектор "класс"

```
.test{}
```

```
<div class="test"></div>
```

Основные единицы измерения

Единица измерения	Обозначение
Пиксели	px
Размер литеры 'm' текущего шрифта	em
Корневой em	rem
Процент от размера родительского элемента	%
0.01 ширины или высоты окна	vw, vh

Единицы измерения

Единица измерения	Обозначение
Применяются для печати	
Пункты, пики, сантиметры, миллиметры, дюймы	pt, pc, cm, mm, in
Применяются редко	
Размер литеры 'x' текущего шрифта, корневой	ex, rex
Проблемы поддержки	
0.01 от меньшей (большей) стороны окна	vmin, vmax

Цвет

- `#00FF00`
- `#00FF007F`
- `rgb(0,255,0)`
- `green`
- `hsl(120,100%,100%)`
- `rgba(0,255,0,.5)`
- `hsla(120,100%,100%,0.1)`

Цвет

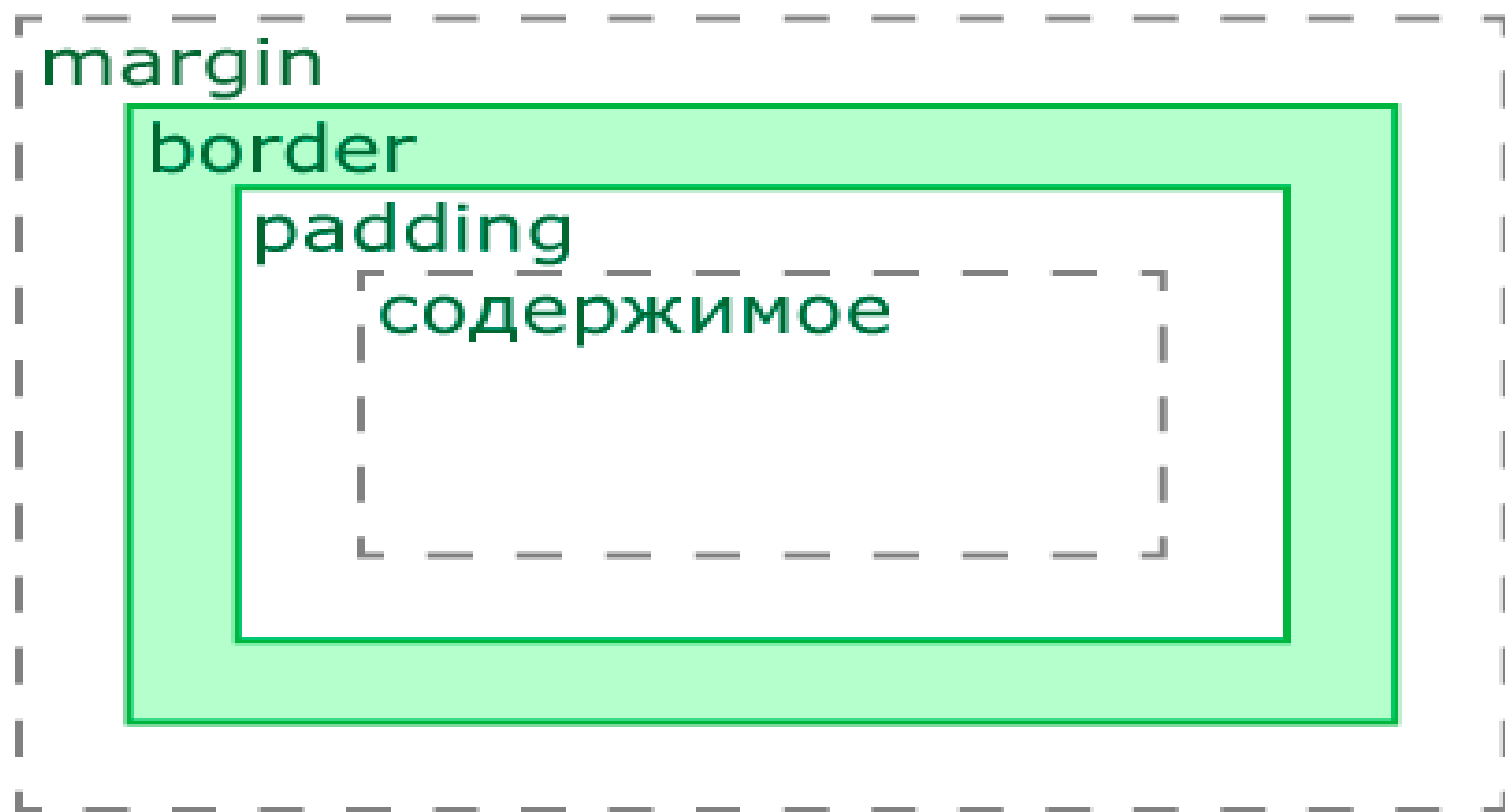
- green
- #00FF00
- rgb(0,255,0)
- hsl(120,100%,100%)
- #00FF007F
- rgba(0,255,0,.5)
- hsla(120,100%,100%,0.5)

Блок - блочный элемент разметки

Блочный элемент - элемент разметки, который обладает следующими свойствами:

- занимает всю ширину страницы
- не обтекается другими элементами
- имеет регулируемые размеры и отступы

2.1 Блочная модель



Простой способ подключения CSS

```
<head>
  ...
  <style>
    .test{
      color: blue;
    }
  </style>
  ...
</head>
<body>
  <div class="test">
    ZZZZZZ ZZZZZZZZZ zzzzzzz zzzzzzz zzzz .test
  </div>
</body>
```

ZZZZZZ ZZZZZZZZZ zzzzzzz zzzzzzz zzzz .test

Цвет фона background, цвет текста color

```
.test{  
  color: blue;  
}
```

ZZZZZZZ ZZZZZZZZZZ zzzzzzzz zzzzzzzz zzzzz .test

Границы блока

```
.test{  
  color: blue;  
  border-style: solid;  
}
```

ZZZZZZZ ZZZZZZZZZZ zzzzzzzz zzzzzzzz zzzz .test

`border-style: dotted;`

`border-style: dashed;`

`border-style: double;`

`border-style: groove;`

`border-style: ridge;`

`border-style: inset ;`

`border-style: outset;`

Границы блока 1 значение

```
.test{  
  color: blue;  
  border-style: solid;  
  border-color: red;  
  border-width: 30px;  
}
```

ZZZZZZZ ZZZZZZZZZ zzzzzzz zzzzzzz zzzz .test

Границы блока 1 значение с указанием сторон

```
.test{  
    color: blue;  
    border-style: solid;  
    border-color: red;  
    border-left-width: 30px;  
    border-right-width: 10px;  
}
```

zzzzzzzz zzzzzzzzzz zzzzzzzz zzzzzzzz zzzzz .test

Несколько значений свойства для разных сторон



```
border-color: red;
```



```
border-color: red green;
```



```
border-color: red green blue;
```



```
border-color: red green blue cyan;
```

Справедливо для других блочных свойств

Сокращенная запись границ

```
.test{  
  color: blue;  
  border: 30px 2px solid red;  
}
```

ZZZZZZZ ZZZZZZZZZ zzzzzzz zzzzzzz zzzz .test

Не допускает нескольких значений для разных свойств, но можно заменить на border-left, border-right ...



`border-top: 30px solid red;`

`border-right: 30px solid red;`



`border-left: 30px solid red;`

`border-bottom: 30px solid red;`



Справедливо для других блочных свойств

Внутренний отступ

```
.test{  
  color: blue;  
  border: 30px 2px solid red;  
  padding: 40px 30px 10px;  
}
```



zzzzzz zzzzzzzz zzzzzz zzzzzz zzzz .test

Зелёным цветом выделена область занимаемая текстом!!!
Допустимы padding-left, padding-right ...

Внешний отступ

```
.test{  
  color: blue;  
  border: 30px 2px solid red;  
  padding: 40px 30px 10px;  
  margin: 20px 30px;  
}
```

ZZZZZZZ ZZZZZZZZZ zzzzzzzz zzzzzzzz zzzzz .test

Также допустимы margin-left, margin-right ...

Размеры

```
.test{  
  color: blue;  
  border: 30px 2px solid red;  
  padding: 40px 30px 10px;  
  margin: 20px 30px;  
  width: 600px;  
  height: 300px;  
}
```

ZZZZZZZ ZZZZZZZZZZ zzzzzzzz zzzzzzzz
zzzz .test

margin: X auto

```
.test{  
  color: blue;  
  border: 30px 2px solid red;  
  padding: 40px 30px 10px;  
  margin: 20px auto;  
  width: 600px;  
  height: 300px;  
}
```

ZZZZZZZ ZZZZZZZZZZ zzzzzzzz zzzzzzzz
zzzz .test

Визуальный размер элемента

`width: 700px;`



`width: 700px; border: 15px solid red;`



`width: 700px; border: 15px solid red; padding: 15px;`



border-radius

```
.test{  
  color: blue;  
  border: 30px 2px solid red;  
  padding: 40px 30px 10px;  
  margin: 20px auto;  
  width: 600px;  
  height: 300px;  
  border-radius: 30px;  
}
```

ZZZZZZZ ZZZZZZZZZZ zzzzzzzz zzzzzzzz
zzzz .test



`border-radius: 30px;`



`border-radius: 60px;`



`border-radius: 30px 60px;`



`border-top-left-radius: 30px 60px;`

overflow

```
.test{  
  color: blue;  
  border: 30px 2px solid red;  
  padding: 40px 30px 10px;  
  margin: 20px auto;  
  width: 600px;  
  height: 170px;  
  border-radius: 30px;  
}
```

без указания

zzzzzzzz zzzzzzzzzz zzzzzzzz zzzzzzzz

zzzz .test

overflow: hidden;

777777 77777777 777777 777777

overflow: scroll;

777777 77777777 777777 777777

overflow: auto;

777777 77777777 777777 777777

- `visible` - значение, используемое браузером по умолчанию
- `scroll` - добавляет к элементу полосы прокрутки по вертикали и горизонтали
- `auto` - добавляет полосы прокрутки при необходимости
- `hidden` - скрывает часть содержимого, которое выходит за границы блочного элемента

```
overflow-x: scroll
```

Добавляет пролосу прокрутки только для
горизонтальной оси

```
overflow-y: scroll
```

Добавляет пролосу прокрутки только для
вертикальной оси

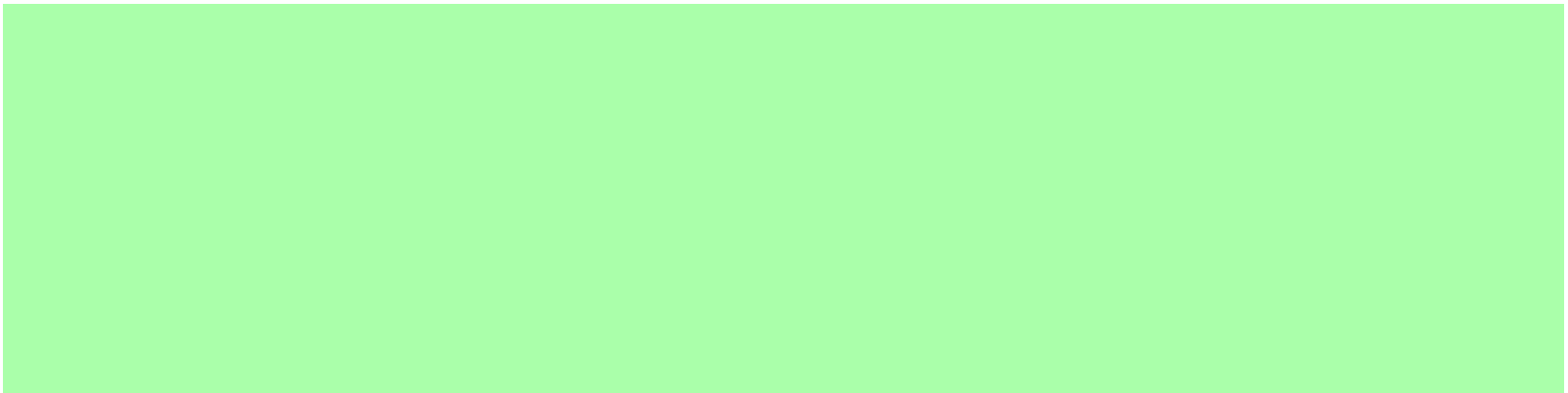
2.2 Фоны

- background-color
- background-image
- background-repeat
- background-position

Применяется не только для блочных элементов

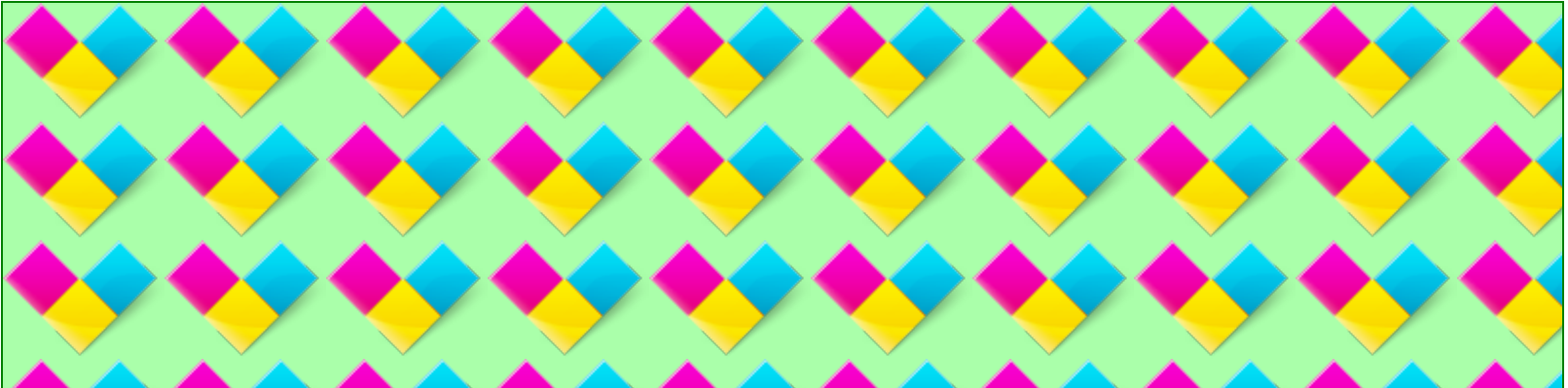
background-color

```
.test{  
  background-color: #aaffaa;  
  height: 200px;  
}
```



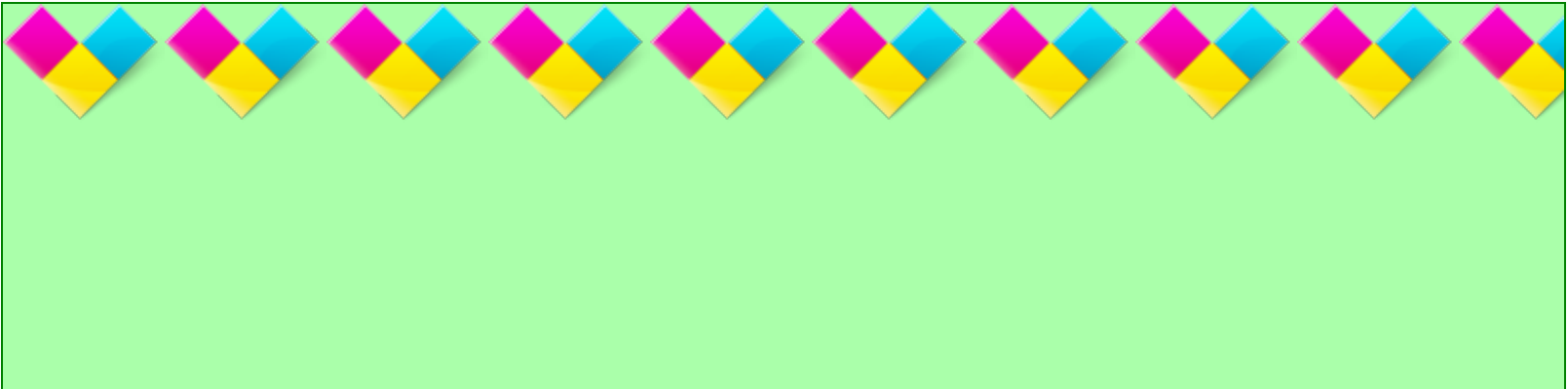
background-image

```
.test{  
  background-color: #aaffaa;  
  background-image: url('images/2-logo.png');  
  height: 200px;  
  border: 1px solid green;  
}
```



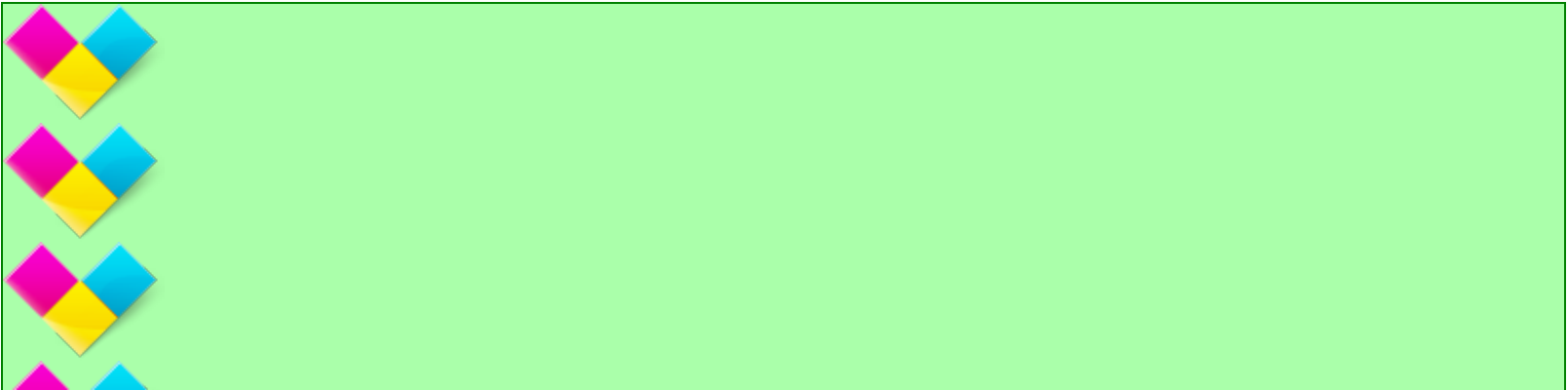
background-repeat: repeat-x;

```
.test{  
  background-color: #aaffaa;  
  background-image: url('images/2-logo.png');  
  background-repeat: repeat-x;  
  height: 200px;  
  border: 1px solid green;  
}
```



background-repeat: repeat-y;

```
.test{  
  background-color: #aaffaa;  
  background-image: url('images/2-logo.png');  
  background-repeat: repeat-y;  
  height: 200px;  
  border: 1px solid green;  
}
```



background-repeat: no-repeat;

```
.test{  
  background-color: #aaffaa;  
  background-image: url('images/2-logo.png');  
  background-repeat: no-repeat;  
  height: 200px;  
  border: 1px solid green;  
}
```



background-position

```
.test{  
  background-color: #aaffaa;  
  background-image: url('images/2-logo.png');  
  background-repeat: no-repeat;  
  height: 200px;  
  border: 1px solid green;  
  background-position: 50% 40px;  
}
```



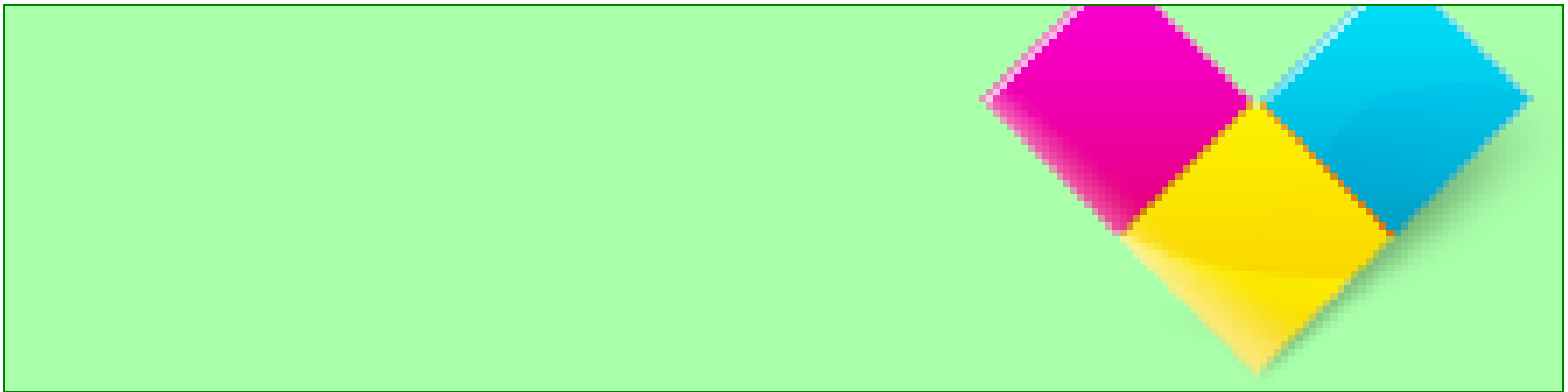
background-position

```
.test{  
  background-image: url('images/2-logo.png');  
  background-repeat: no-repeat;  
  height: 200px;  
  border: 1px solid green;  
  background-position: bottom right;  
}
```



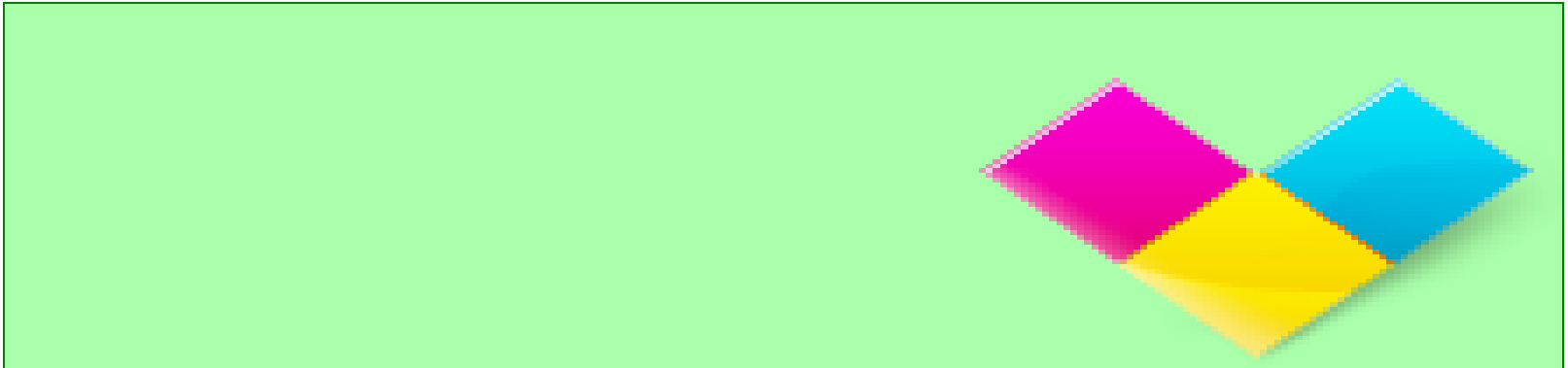
background-size

```
.test{  
  background-color: #aaffaa;  
  background-image: url('images/2-logo.png');  
  background-repeat: no-repeat;  
  height: 200px;  
  border: 1px solid green;  
  background-position: bottom right;  
}
```



background-size

```
.test{  
  background-color: #aaffaa;  
  background-image: url('images/2-logo.png');  
  background-repeat: no-repeat;  
  height: 200px;  
  border: 1px solid green;  
  background-position: bottom right;  
  background-size: 300px 150px;  
}
```



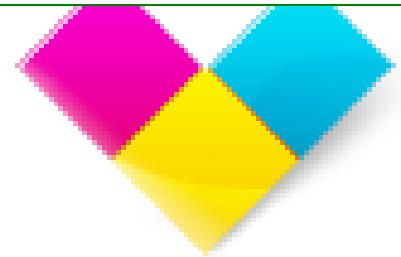
background (сокращенная запись)

```
.test{  
  background: #aaffaa url('images/2-logo.png')  
             no-repeat bottom right;  
  height: 200px;  
  border: 1px solid green;  
  background-size: 200px;  
}
```



background (сокращенная запись)

```
.test{  
  background: url('images/2-logo.png')  
             no-repeat bottom right,  
             url('images/2-logo.png')  
             no-repeat bottom left;  
  height: 200px;  
  border: 1px solid green;  
  background-size: 200px, 100px;  
}
```



background-attachment

- `fixed` - делает фоновое изображение элемента неподвижным.
- `scroll` - Позволяет перемещаться фону вместе с содержимым.
- `local` - Фон фиксируется с учётом поведения элемента. Если элемент имеет прокрутку, то фон будет прокручиваться вместе с содержимым, но фон выходящий за рамки элемента остаётся на месте.

2.3 Базовая работа с текстом

"Красная строка"

```
.test{  
  background-color: #aaffaa;  
  padding: 15px;  
  text-indent: 50px;  
}
```

Стандартный вид текста не всегда подходит, как по внешнему виду, так и по цвету, CSS предоставляет обширные возможности, с помощью которых, вы можете кардинально изменить ваш текст, например задав ему цвет, меняя межстрочный интервал, расстояние между буквами или размер текста и многое другое.

Выключка

text-align: right; Стандартный вид текста не всегда подходит, как по внешнему виду, ...

text-align: center; Стандартный вид текста не всегда подходит, как по внешнему виду,...

text-align: left; Стандартный вид текста не всегда подходит, как по внешнему виду, ...

text-align: justify; Стандартный вид текста не всегда подходит, как по внешнему виду, ...

Размер и цвет

```
.test{  
  background-color: #aaffaa;  
  padding: 15px;  
  color:red;  
  font-size: 50px;  
}
```

Стандартный вид текста...

Тень

```
.test{  
  background-color: #aaffaa;  
  padding: 15px;  
  color:red;  
  font-size: 50px;  
  text-shadow: 5px 7px 3px #000;  
}
```

Стандартный вид текста...

- 5px - смещение вправо
- 7px - смещение вниз
- 3px - смещение размытие
- #000 - цвет

Шрифт

```
.test{  
  background-color: #aaffaa;  
  padding: 15px;  
  color:red;  
  font-size: 50px;  
  font-family: 'Times New Roman', Times, serif;  
}
```

Стандартный вид текста...

Наборы шрифтов

- sans-serif - шрифты без засечек, считается что они лучше читаются на экране компьютера, чем шрифты семейства serif
- serif - семейство шрифтов с засечками
- monospace - моноширинные шрифты
- cursive - рукописные шрифты
- fantasy - художественные и декоративные шрифты

Стиль шрифта

font-style:italic;

font-style:oblique;

font-style:none;

Начертание шрифта

font-weight: normal; (по умолчанию)

font-weight: bold;

font-weight: 900;

font-weight: 100;

Дополнительное оформление текста

text-decoration: none;

text-decoration: underline;

~~text-decoration: line-through;~~

text-decoration: overline;

Регистр букв

TEXT-TRANSFORM: UPPERCASE;

text-transform: lowercase;

Text-Transform: Capitalize;

Высота строки (интервал)

```
.test{  
  
}
```

Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna aliqua.

Высота строки (интервал)

Множитель

```
.test{  
  line-height:2;  
}
```

Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna aliqua.

Высота строки (интервал)

В пикселях

```
.test{  
    line-height:100px;  
}
```

Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna aliqua.

Ширина между символами

```
.test{  
}
```

Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna aliqua.

Ширина между символами

```
.test{  
  letter-spacing:2px;  
}
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Ширина между символами

```
.test{  
  letter-spacing:-1px;  
}
```

Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna aliqua.

2.4 Изменение расположения и взаимодействия объектов

```
<div class="container">  
  <div class="block1">block1</div>  
  <div class="block2">block2</div>  
  <div class="block3">block3</div>  
</div>
```

block1

block2

block3

```
.container{  
    background:#cccccc;  
}  
.block1{  
    width:120px;  
    height:120px;  
    background:green;  
}
```

```
.block2{  
    width:150px;  
    height:60px;  
    background:blue;  
}  
.block2{  
    width: 300px;  
    height:40px;  
    background:yellow;  
}
```

block1

block2

block3

Обтекание

```
.block1{  
    }  
.block2{  
    }
```

```
.block3{  
    }  
.container{  
    }
```

block1

block2

block3

Обтекание

z-index не работает

<pre>.block1{ float:left; } .block2{ }</pre>	<pre>.block3{ } .container{ }</pre>
--	---



Обтекание

```
.block1{  
    float:right;  
}  
.block2{  
}
```

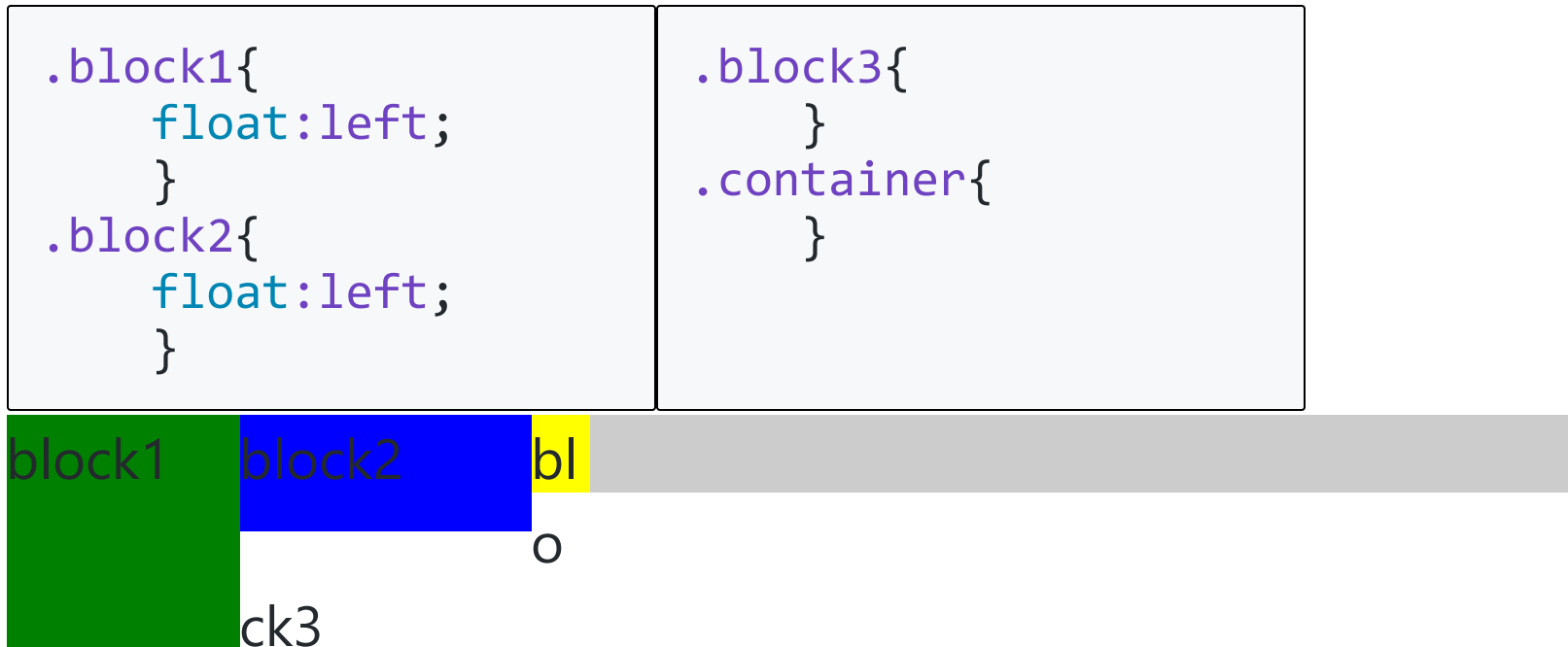
```
.block3{  
}  
.container{  
}
```

block2

block3

block1

Обтекание



Обтекание

```
.block1{  
  float:left;  
}  
.block2{  
  float:left;  
}
```

```
.block3{  
  clear:both}  
.container{  
}
```

block1

block2

block3

Обтекание

```
.block1{  
    float:right;  
}  
.block2{  
    clear:both;  
}
```

```
.block3{  
    }  
.container{  
    }
```

block1

block2

block3

Позиционирование

```
.block1{  
    }  
.block2{  
    }
```

```
.block3{  
    }  
.container{  
    }
```

block1

block2

block3

Позиционирование

```
.block1{  
    position:relative;  
}  
.block2{  
}
```

```
.block3{  
}  
.container{  
}
```

block1

block2

block3

Позиционирование

```
.block1{  
  position:relative;  
  left: 20px;  
  top:10px;  
}  
.block2{  
}
```

```
.block3{  
  }  
.container{  
  }
```



Позиционирование

```
.block1{  
    position:relative;  
    right: 20px;  
    bottom:10px;  
}  
.block2{  
}
```

```
.block3{  
}  
.container{  
}
```

block1

block2

block3

Позиционирование

```
.block1{  
  position:absolute;  
}  
.block2{  
}
```

```
.block3{  
}  
.container{  
}
```



block1

block1

Позиционирование

```
.block1{  
  position:absolute;  
  left: 20px;  
  top:10px;  
}  
.block2{  
}
```

```
.block3{  
}  
.container{  
}
```

block2

block3

Позиционирование

```
.block1{  
    position:absolute;  
    right: 20px;  
    bottom:10px;  
}  
.block2{  
}
```

```
.block3{  
}  
.container{  
}
```

block2

block3

block1

Позиционирование

```
.block1{  
    position:absolute;  
    right: 0;  
    bottom:0;  
}  
.block2{  
}
```

```
.block3{  
}  
.container{  
    position:relative;  
}
```

block2

block3

block1

Позиционирование

```
.block1{  
  position:absolute;  
  bottom:0;  
}  
.block2{  
}
```

```
.block3{  
}  
.container{  
  position:relative;  
}
```

block1



The diagram illustrates the CSS positioning of the elements. A large gray rectangle represents the container. Inside the container, on the left side, there is a green rectangle labeled 'block1'. To the right of 'block1' is a small blue rectangle. Below 'block1' is a yellow rectangle. The green rectangle 'block1' is positioned at the bottom of the container, as indicated by the CSS code '.block1{position:absolute;bottom:0;}'.

Позиционирование

```
.block1{  
  position:fixed;  
}  
.block2{  
}
```

```
.block3{  
}  
.container{  
}
```



block1

Позиционирование

```
.block1{  
    position:scroll;  
}  
.block2{  
}
```

```
.block3{  
}  
.container{  
}
```

block1

block2

block3

Наложение

```
.block1{  
  position:relative;  
  top:20px;  
}  
.block2{  
  position:relative;  
}
```

```
.block3{  
  }  
.container{  
  }
```

block1

block2

block3

Наложение

```
.block1{  
  position:relative;  
  top:20px;  
  z-index:10;  
}  
.block2{  
  position:relative;  
}
```

```
.block3{  
  }  
.container{  
  }
```

block1

block2

block3

Видимость

```
.block1{  
  visibility:hidden;  
}  
.block2{  
}
```

```
.block3{  
}  
.container{  
}
```

block2

block3

Свойства отображения

```
.block1{  
  display: none;  
}  
.block2{  
}
```

```
.block3{  
}  
.container{  
}
```

block2

block3

Свойства отображения

```
.block1{  
  display: inline-block;  
}  
.block2{  
}
```

```
.block3{  
}  
.container{  
}
```

block1

block2

block3

Свойства отображения. Flexbox

Flexbox

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display: flex;  
}
```



Свойства отображения. Flexbox

Flexbox

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display: inline-flex;  
}
```

block1

block2

block3

Свойства отображения. Flexbox

Направление

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display: flex;  
  flex-direction: row;  
}
```

block1

block2

block3

Свойства отображения. Flexbox

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display: flex;  
  flex-direction: row-reverse;  
}
```

block3

block2

block1

Свойства отображения. Flexbox

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display: flex;  
  flex-direction: column;  
}
```

block1

block2

block3

Свойства отображения. Flexbox

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display: flex;  
  flex-direction: column-reverse;  
}
```

block3

block2

block1

Свойства отображения. Flexbox

При нехватке места блоки равномерно уменьшаются их ширина

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display:flex;  
  width:400px;  
}
```



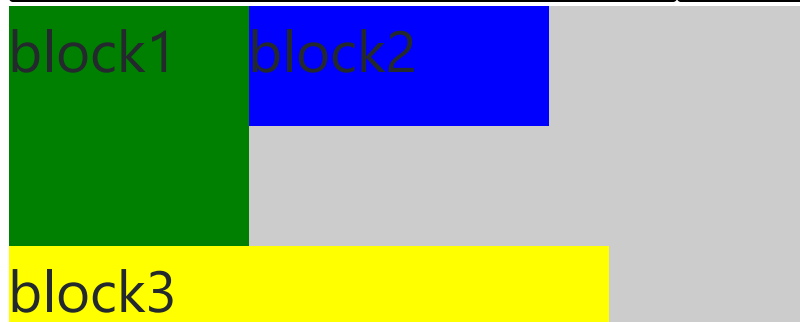
block1 block2 block3

Свойства отображения. Flexbox

Режим переноса блоков

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display:flex;  
  width:400px;  
  flex-wrap:wrap;  
}
```

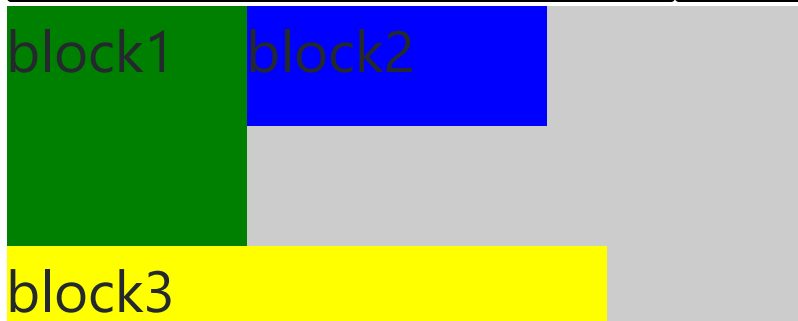


Свойства отображения. Flexbox

flex-direction + flex-wrap

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display:flex;  
  width:400px;  
  flex-flow: row wrap;  
}
```



Свойства отображения. Flexbox

Изменение порядка

```
.block1{  
  order: 1}  
.block2{  
}
```

```
.block3{  
}  
.container{  
  display: flex;  
}
```

block2

block3

block1

Свойства отображения. Flexbox

Изменение порядка

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  order: -1;  
.container{  
  display: flex;  
}
```

block3

block1

block2

Свойства отображения. Flexbox

Изменение порядка

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  order: -1;  
.container{  
  display: flex;  
}
```

block3

block1

block2

Свойства отображения. Flexbox

Распределение элементов вдоль направления

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display:flex;  
  justify-content: flex-start;  
}
```



Свойства отображения. Flexbox

Распределение элементов вдоль направления

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display:flex;  
  justify-content: flex-end;  
}
```

block1

block2

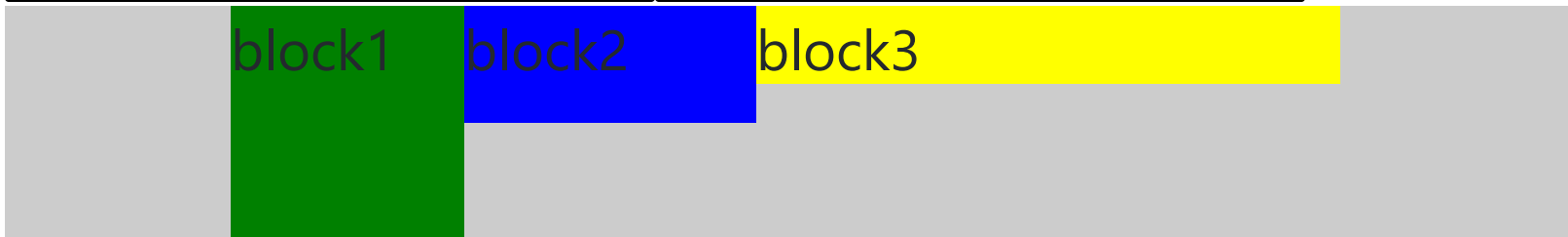
block3

Свойства отображения. Flexbox

Распределение элементов вдоль направления

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display:flex;  
  justify-content: center;  
}
```



Свойства отображения. Flexbox

Распределение элементов вдоль направления

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display:flex;  
  justify-content: justify;  
}
```

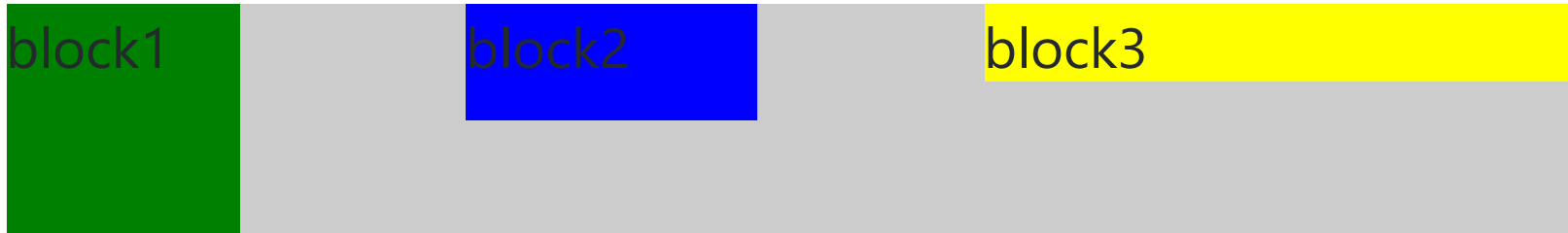


Свойства отображения. Flexbox

Распределение элементов вдоль направления

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display:flex;  
  justify-content: space-between;  
}
```



Свойства отображения. Flexbox

Распределение элементов вдоль направления

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display:flex;  
  justify-content: space-around;  
}
```

block1

block2

block3

Свойства отображения. Flexbox

Распределение элементов поперёк направления

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display:flex;  
  align-items: flex-start;  
}
```

block1

block2

block3

Свойства отображения. Flexbox

Распределение элементов поперёк направления

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display:flex;  
  align-items: flex-end;  
}
```



Свойства отображения. Flexbox

Распределение элементов поперёк направления

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display:flex;  
  align-items: center;  
}
```

block1

block2

block3

Свойства отображения. Flexbox

Распределение элементов поперёк направления

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  height:auto;  
}  
.container{  
  display:flex;  
  align-items: stretch;  
}
```

block1

block2

block3

Свойства отображения. Flexbox

Распределение элементов поперёк направления

```
.block1{  
  }  
.block2{  
  align-self:flex-end;  
}
```

```
.block3{  
  height:auto;  
  }  
.container{  
  display:flex;  
  align-items: stretch;  
}
```



Свойства отображения. Flexbox

Растягивание при наличии свободного места

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  flex-grow:1;  
}  
.container{  
  display:flex;  
}
```

block1

block2

block3

Свойства отображения. Flexbox

Растягивание при наличии свободного места

```
.block1{  
  }  
.block2{  
  flex-grow:1;  
}
```

```
.block3{  
  flex-grow:1;  
}  
.container{  
  display:flex;  
}
```

block1

block2

block3

Свойства отображения. Flexbox

Растягивание при наличии свободного места

```
.block1{  
  }  
.block2{  
  flex-grow:1;  
}
```

```
.block3{  
  flex-grow:2;  
}  
.container{  
  display:flex;  
}
```

block1

block2

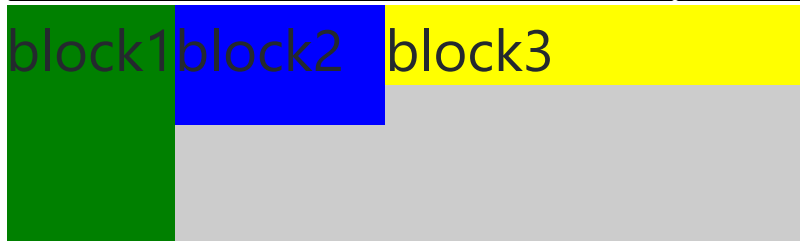
block3

Свойства отображения. Flexbox

Сжатие при нехватке свободного места

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display:flex;  
  width:400px;  
  }
```

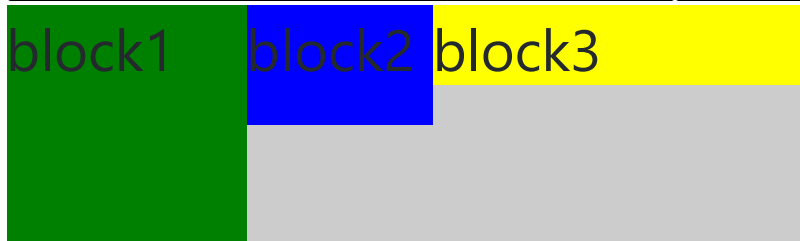


Свойства отображения. Flexbox

Сжатие при нехватке свободного места

```
.block1{  
  flex-shrink:0;  
}  
.block2{  
}
```

```
.block3{  
}  
.container{  
  display:flex;  
  width:400px;  
}
```

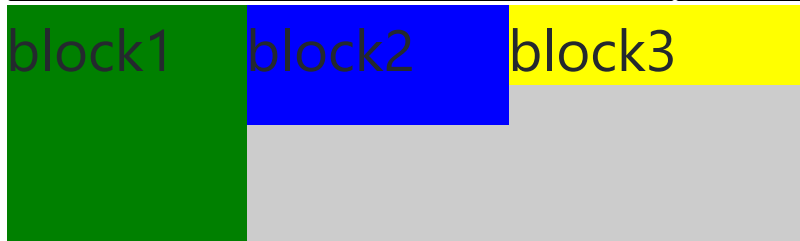


Свойства отображения. Flexbox

Сжатие при нехватке свободного места

```
.block1{  
  flex-shrink:0;  
}  
.block2{  
}
```

```
.block3{  
  flex-shrink:4}  
.container{  
  display:flex;  
  width:400px;  
}
```

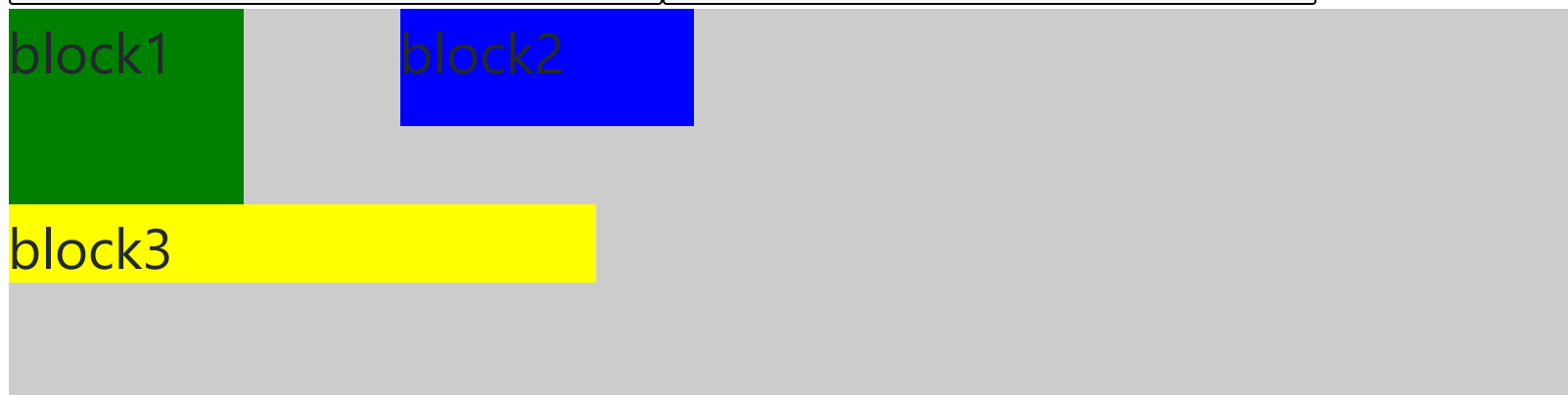


Свойства отображения. Grid

Распределение элементов по сетке

```
.block1{  
  }  
.block2{  
  }
```

```
.block3{  
  }  
.container{  
  display:grid;  
  grid-template-rows:100px 100px;  
  grid-template-columns: 200px 50%;  
}
```



Свойства отображения. Grid

Распределение элементов по сетке

```
.block1{  
  grid-area: block1}  
.block2{  
  grid-area: block2}
```

```
.block3{  
  grid-area: block3}  
.container{  
  display:grid;  
  grid-template-rows:100px 100px;  
  grid-template-columns: 200px 50%;  
  grid-template-areas: 'block1 block3' 'block2 block3'}
```

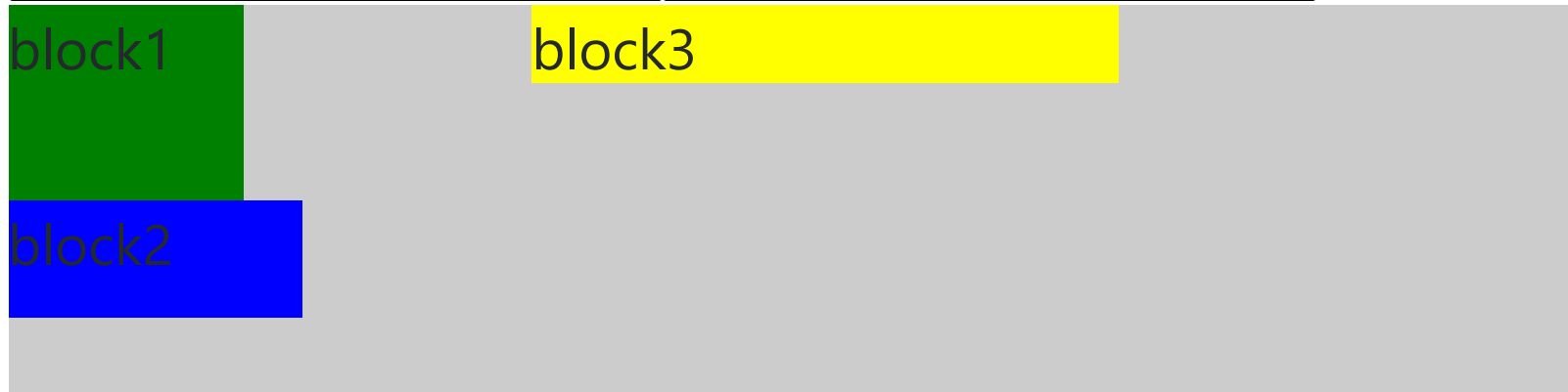


Свойства отображения. Grid

Распределение элементов по сетке

```
.block1{  
  grid-area: block1}  
.block2{  
  grid-area: block2}
```

```
.block3{  
  grid-area: block3}  
.container{  
  display:grid;  
  grid-template-rows:100px 100px;  
  grid-template-columns: 1fr 2fr;  
  grid-template-areas: 'block1 block3' 'block2 block3'}
```



Свойства отображения. Grid

Распределение элементов по сетке

```
.block1{  
  width:auto;  
  grid-area: block1}  
.block2{  
  width:auto;  
  grid-area: block2}
```

```
.block3{  
  width:auto;  
  grid-area: block3}  
.container{  
  display:grid;  
  grid-template-rows:100px 100px;  
  grid-template-columns: 200px 50%;  
  grid-template-areas: 'block1 block3' 'block2 block3'}
```

block1

block3

block2

Свойства отображения. Grid

Распределение элементов по сетке

```
.block1{  
  width:auto;  
  grid-area: block1}  
.block2{  
  width:auto;  
  grid-area: block2}
```

```
.block3{  
  width:auto;  
  grid-area: block3}  
.container{  
  display:grid;  
  grid-template-rows:100px 100px;  
  grid-template-columns: 200px 1fr;  
  grid-template-areas: 'block1 block3' 'block2 block3'}
```

block1

block3

block2

