

СЕТКИ

АЛЕКСАНДР БЕСПОЯСОВ / НЕТОЛОГИЯ



ТИПОВЫЕ ЗАДАЧИ

- Построение сетки
- Создание отступов между блоками сетки
- Горизонтальное выравнивание блока по центру
- N колонок фиксированной ширины или в процентах
- Вертикальное выравнивание блоков друг относительно друга
- Приклеенные блоки
- «Уши» сайта и выносные элементы



ЧТО ТАКОЕ СЕТКА

Модульная сетка (или просто сетка) — набор невидимых направляющих, вдоль которых располагаются элементы веб-страницы.

[Что такое модульная вёрстка \(в дизайне\)](#)

[Для чего нужна сетка](#)

[Модульные сетки в вебе](#)

ПРИМЕРЫ ПРОСТЫХ СЕТОК



Сетка с двумя...

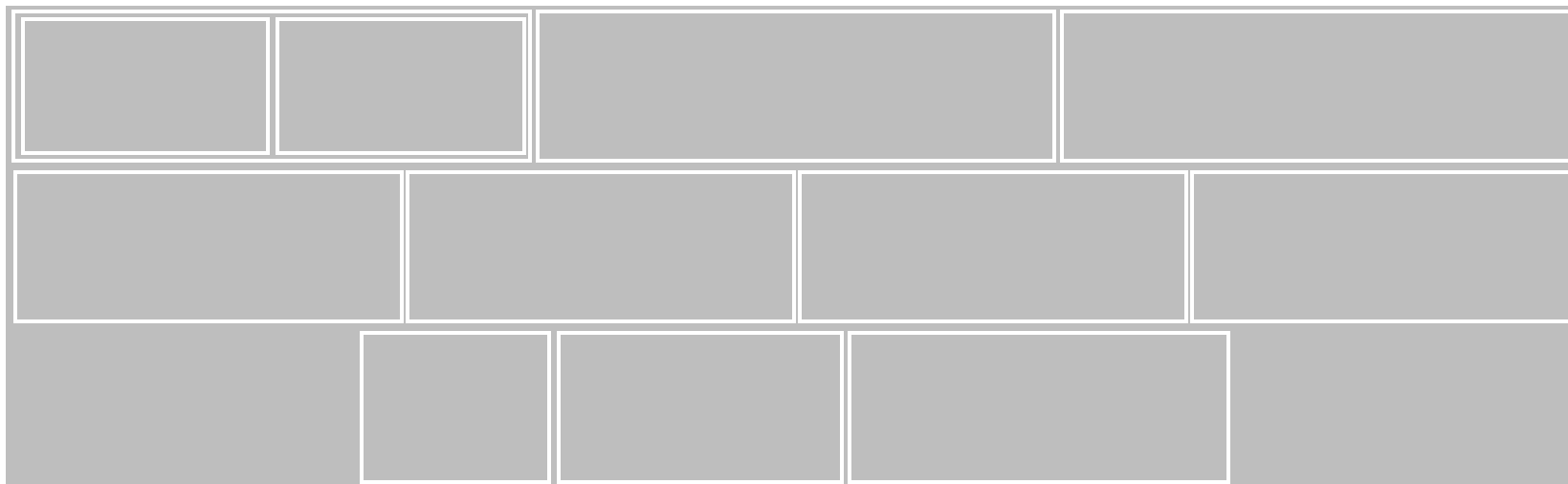


...три



...и четырьмя колонками

ПРИМЕР СЛОЖНОЙ СЕТКИ



Пример сложной сетки

ОРГАНИЗАЦИЯ СЕТОК В ВЕБЕ

Верстальщик выбирает способ создания сетки исходя из ограничений, основное из которых — поддержка выбранного способа браузерами.

- Таблица (table);
- Плавающие элементы (float);
- Строчные блоки (inline-block);
- Флексбокс (flexible box layout, flexbox);
- Грид-раскладка (grid layout).

ПЛАВАЮЩИЕ ЭЛЕМЕНТЫ

- + Кроссбраузерно.
- Не семантично;
- Вырывают блок из потока документа;
- Высота родительского блока не подстраивается под содержимое;
- Требуют `clear:both` после себя;
- Нет управления выравниванием.

[Демо](#)

ФЛОТЫ

Разметка для примеров:

```
<div class="grid">  
  <div class="grid-col"></div>  
  <div class="grid-col"></div>  
  <div class="grid-col"></div>  
</div>
```


ФЛОТЫ

Простая сетка, выравнивание по левому краю:



```
.grid-col {  
  height:50px;  
  width:100px;  
  float:left;  
  border:2px solid white;  
}
```

ФЛОТЫ

Простая сетка, блоки начинают заполнять контейнер справа:



```
.grid-col {  
  height:50px;  
  width:100px;  
  float:right;  
  border:2px solid white;  
}
```

ФЛОТЫ

Блоки с шириной в процентах:



```
.grid-col {  
  height:50px;  
  width:33%;  
  float:left;  
  border:2px solid white;  
}
```



СТРОЧНЫЕ БЛОКИ

- + Можно управлять выравниванием;
- + Контент не вырывается из потока;
- + Кроссбраузерно.
- Пробелы между элементами могут добавить ширины элементам;
- Трудно верстать сложные сетки.

[Демо](#)

СТРОЧНЫЕ БЛОКИ

Разметка для примеров:

```
<div class="grid">  
  <div class="grid-col"></div>  
  <div class="grid-col"></div>  
  <div class="grid-col"></div>  
</div>
```

СТРОЧНЫЕ БЛОКИ

Сетка с выравниванием по левому краю:



```
.grid-col {  
  height:50px;  
  width:100px;  
  display:inline-block;  
  vertical-align: top;  
  border:2px solid white;  
}
```

СТРОЧНЫЕ БЛОКИ

Сетка с выравниванием по правому краю:



```
.grid {  
    text-align: right;  
}  
.grid-col {  
    height: 50px;  
    width: 100px;  
    display: inline-block;  
    vertical-align: top;  
    border: 2px solid white;  
}
```

СТРОЧНЫЕ БЛОКИ

Блоки начинаются справа:



```
.grid {  
    direction: rtl;  
}  
.grid-col {  
    height: 50px;  
    width: 100px;  
    display: inline-block;  
    vertical-align: top;  
    border: 2px solid white;  
}
```


СТРОЧНЫЕ БЛОКИ

Блоки с шириной в процентах:



```
.grid-col {  
  height:50px;  
  width:33.33%;  
  display:inline-block;  
  vertical-align: top;  
  border:2px solid white;  
  margin-right:-2px;  
}
```



ФЛЕКСБОКС

- + [Можно управлять выравниванием по обеим осям;](#)
- + Можно управлять порядком элементов без изменения HTML-документа;
- + Гибко и наследуемо;
- + Есть готовая спецификация, стандарт.
- [Почти кроссбраузерно.](#)

[Полное руководство по флексбоксу](#)

[Демо](#)



ГРИДЫ

- + Можно вообще всё.
- Чуть сложнее, чем флексбокс;
- Не кроссбраузерно.

ПОЗИЦИОНИРОВАНИЕ

- `relative` — положение элемента устанавливается относительно его исходного места;
- `absolute` — положение задаётся абсолютно, элемент вырывается из потока;
- `fixed` — положение задаётся относительно окна браузера;
- `static` — по умолчанию, не влияет на `left`, `right` и т.д.

[Демо](#)

ПОЗИЦИОНИРОВАНИЕ

Разметка для примеров:

```
<div class="wrap">  
  <div class="pos pos--rel">  
    Я относительный  
    <div class="pos pos--abs">Я абсолютный</div>  
  </div>  
</div>
```

ПОЗИЦИОНИРОВАНИЕ

Абсолютное и относительное

Я относительный

Я абсолютный

```
.pos--rel {  
  position: relative;  
  background: #bebebe;  
  min-height: 100px;  
  padding: 10px 20px;  
}  
  
.pos--abs {  
  position: absolute;  
  top: 50px;  
}
```

ПОЗИЦИОНИРОВАНИЕ

Абсолютное и относительное

Я относительный

Я абсолютный

```
.pos--rel {  
  position: relative;  
  background: #bebebe;  
  min-height: 100px;  
  padding: 10px 20px;  
}  
  
.pos--abs {  
  position: absolute;  
  top: 50px;  
  right: 0;  
}
```

ПОЗИЦИОНИРОВАНИЕ

Фиксированное

Я относительный

```
.pos--rel {  
  position: relative;  
  background: #bebebe;  
  min-height: 100px;  
  padding: 10px 20px;  
}  
  
.pos--fixed {  
  position: fixed;  
  top: 10px;  
  right: 10px;  
}
```


ОТСТУПЫ

Свойство `margin` задаёт внешний отступ элементу. Отступы между элементами схлопываются.

Имеет форму записи: `margin: a b c d;`

Где `a` — отступ сверху, `b` — справа, `c` — снизу, `d` — слева.

Может задаваться в исчисляемых единицах: `px`, `%`, `em` и т. д.

и автоматически — `auto`.

[Демо](#)

ОТСТУПЫ МЕЖДУ БЛОКАМИ

В качестве примера рассмотрим ситуацию, когда нужно создать сетку с блоками и отступами между ними.



[Демо](#)

ОТСТУПЫ МЕЖДУ БЛОКАМИ

Разметка для примеров:

```
<div class="betw">
  <div class="betw-wrap">
    <div class="betw-col"></div>
    <div class="betw-col"></div>
    <div class="betw-col"></div>
  </div>
</div>
```

ОТСТУПЫ МЕЖДУ БЛОКАМИ

С помощью строчных блоков:



```
.betw {  
  position: relative;  
  text-align: justify;  
  box-sizing: border-box;  
}  
.betw-col {  
  display: inline-block;  
  width: 30%;  
}  
.betw:after {  
  content: ' ';  
  display: inline-block;  
  width: 100%;  
}
```

ОТСТУПЫ МЕЖДУ БЛОКАМИ

С помощью плавающих элементов:



```
.betw {  
  overflow: hidden;  
}
```

```
.betw-wrap {  
  margin: 0 -2.1%;  
}
```

ВНУТРЕННИЕ ПОЛЯ

Свойство `padding` задаёт внутренние поля.

Внутренние поля увеличивают ширину элемента. Чтобы этого не происходило, можно использовать `box-sizing: border-box`.

[Демо](#)

МАРКЕР СПИСКА


Стандартного `list-style: ...` часто не хватает для комфортной стилизации маркеров списков. Поэтому мы можем сделать это псевдоэлементом каждого из пунктов.

- Первый пункт
- Второй пункт
- Третий пункт

[Демо](#)

КНОПКА «ВНУТРИ» ПОЛЯ ПОИСКА

Важно помнить, что пользователь привык к обратной связи при работе с элементами форм. При фокусе поле должно реагировать появлением аутлайна, наведённая кнопка — состоянием при наведении и нажатии.



A horizontal search bar with a thin black border. The bar is divided into two sections: a large white input field on the left and a smaller red button on the right. The button contains the word 'Поиск' in white text.

[Демо](#)

КНОПКА «СНАРУЖИ» ПОЛЯ ПОИСКА

Кнопку «снаружи» поля можно сделать, поменяв пару свойств.

Это не единственный способ сделать подобную разметку.

A horizontal search bar with a white input field and a red button on the right. The button contains the word 'Поиск' in white text.

[Демо](#)



ВЕРТИКАЛЬНОЕ ВЫРАВНИВАНИЕ ПО ЦЕНТРУ

Вертикальное выравнивание — задача для строчных блоков или флексбокса.

Флексбокс облегчает работу с вертикальным выравниванием.

[Демо](#)

БЛОК С ФИКСИРОВАННОЙ ШИРИНОЙ ПО ЦЕНТРУ

Одна из типичных задач позиционирования — центрирование блока с фиксированной шириной. [Демо](#)

```
.block {  
    width: 300px;  
    margin: 0 auto;  
}
```

ФИКСИРОВАННАЯ ШАПКА

Создаётся с помощью позиционирования. [Демо](#)

```
.header {  
    position: fixed;  
    top: 0;  
    left: 0;  
    right: 0;  
}
```

ПОДВАЛ ВНИЗУ СТРАНИЦЫ

Решается сочетанием внутреннего поля и отрицательного отступа. [Демо](#)

```
.outerBlock { position: relative; }  
.innerBlock { padding-bottom: 100px; }  
.footer {  
    margin-top: -100px;  
    clear: both;  
    position: relative;  
}
```



«УШИ» САЙТА

Контент внутри блоков центрируется, а сами блоки верстаются во всю ширину.

Такая вёрстка располагает к разбивке макета по этажам. Удобно, когда каждый этаж независимый и может быть вставлен в любое место на странице.

[Демо](#)



ВЫНОСНЫЕ ЭЛЕМЕНТЫ

Чаще всего должны быть в основном потоке документа. Верстаются с применением позиционирования.

[Мастер-класс Сергея Чикуёнка](#)



ДОМАШНЕЕ ЗАДАНИЕ

Описание задания:

1. Необходимо повторить верстку предложенного [макета](#).
2. Верстка приблизительно должна соответствовать макету.
3. Сетки и центрирование блоков должны соответствовать макету.
4. Блок с пунктами меню должен «прилипнуть» к верхней части экрана.
5. Для придания сходства нужно использовать материалы, приложенные к PNG-макету.
6. Картинку фона для шапки сайта, иконку и фотографию. Тексты можно использовать любые.
7. Желтым цветом показаны вспомогательные линии и комментарии (даны для вашей информации, верстать их не нужно).

Дополнительные материалы:

- [Макет задания](#)
- [Исходные файлы к заданию](#)



Ваши вопросы?

АЛЕКСАНДР БЕСПОЯСОВ



bespoyasov@me.com



[webtwozero](https://www.soundcloud.com/webtwozero)



[facebook](https://www.facebook.com/webtwozero)