

## Центрирование в CSS: полное руководство

Центрирование элементов — самая популярная причина для жалоб на CSS. «Ну почему нельзя было сделать всё попроще?» — возмущаются все кому не лень. Думаю, проблема не в том, что это сложно сделать, а в том, что способов центрирования элементов такое множество, что бывает трудно выбрать подходящий.

Давайте построим древовидную схему для принятия решения и будем надеяться, что это всё немного упростит.

Мне нужно отцентрировать элемент...

### По горизонтали

#### Является ли элемент строчным или строчно-\* (как текст или ссылки)?

Строчные элементы внутри родительского блочного можно центрировать так:

```
.center-children {  
  text-align: center;  
}
```

Это сработает для элементов с типом отображения `inline`, `inline-block`, `inline-table`, `inline-flex` и т.д.

#### Является ли элемент блочным?

Блочный элемент можно центрировать, указав для `margin-left` и `margin-right` значение `auto` (а также прописав для него конкретный `width`, иначе он займёт всю ширину родительского контейнера и не будет нуждаться в центрировании). Для этого часто используют сокращённую запись:

```
.center-me {  
  margin: 0 auto;  
}
```

Способ работает независимо от ширины центрируемого блочного элемента и его родителя.

Обратите внимание, что заставить элемент плавать по центру с помощью `float` нельзя. [Хотя есть одна хитрость.](#)

## Вам нужно центрировать несколько блочных элементов?

Если вы имеете дело с двумя и больше блочными элементами, размещёнными *в ряд*, которые нужно центрировать по горизонтали, возможно, вам потребуется изменить тип `display`. Вот что произойдёт, если указать для них тип отображения `inline-block` или применить `flexbox`:

В том же случае, если у вас несколько блочных элементов размещены друг над другом, вполне сработает приём с автоматическими отступами:

## По вертикали

С вертикальным центрированием в CSS все немного сложнее:

### Это элемент строчного типа или строчно-\* (как текст или гиперссылки)?

#### Он помещается в одну строку?

Иногда строчные/текстовые элементы могут выглядеть отцентрированными по вертикали только потому, что у них одинаковые верхнее и нижнее поля (`padding`).

```
.link {
  padding-top: 30px;
  padding-bottom: 30px;
}
```

Если использование полей по какой-либо причине невозможно, а вы хотите отцентрировать текст, который точно не будет переноситься в следующую строку, можно использовать хитрость с установкой `line-height` равной высоте элемента, это отцентрирует текст:

```
.center-text-trick {
  height: 100px;
  line-height: 100px;
  white-space: nowrap;
}
```

## Элемент занимает несколько строк?

Центрирование нескольких строчек текста также можно сделать с помощью установки одинаковых верхнего и нижнего полей, однако, если этот вариант вам не подходит, можно сделать элемент, в который помещён текст, ячейкой таблицы — буквально или же просто её имитацией на CSS. За центрирование в этом случае отвечает свойство `vertical-align`, несмотря на то, что обычно он просто выравнивает по горизонтали элементы в ряду. ([Более подробно об этом.](#))

Если вы имеете дело с некоей имитацией таблицы, может быть, стоит использовать flexbox? Один дочерний flex-элемент можно довольно легко отцентрировать во flex-родителе.

```
.flex-center-vertically {
  display: flex;
  justify-content: center;
  flex-direction: column;
  height: 400px;
}
```

Помните, что этот способ подходит только если у родительского контейнера указана конкретная ширина (px, % и т.д.), поэтому здесь у контейнера есть высота.

Если оба этих способа не подходят, можно использовать приём «элемент-призрак», при котором в контейнер помещается псевдоэлемент с максимальной высотой, и текст выравнивается по горизонтали по этому элементу:

```
.ghost-center {
  position: relative;
}
.ghost-center::before {
  content: " ";
  display: inline-block;
  height: 100%;
  width: 1%;
  vertical-align: middle;
}
.ghost-center p {
  display: inline-block;
  vertical-align: middle;
}
```

## Это блочный элемент?

**Вам известна высота элемента?**

Обычно, когда речь идёт о веб-странице, высота может быть неизвестна по многим причинам: если изменяется ширина элемента, перераспределение текста может изменить высоту. Вариации в стилизации текста могут изменить высоту. Изменение количества текста может изменить высоту элемента. Элементы с фиксированным соотношением сторон, например, изображения, могут изменить высоту при масштабировании — и так далее.

Но, если вам известна высота, вертикальное центрирование можно сделать так:

```
.parent {
  position: relative;
}
.child {
  position: absolute;
  top: 50%;
  height: 100px;
  margin-top: -50px; /* учитывайте поля и границы, если не используете box-sizing: border-box;
*/
}
```

## Вам неизвестна высота элемента?

Его все-таки можно центрировать, подняв вверх на половину его высоты после того, как сдвинули его наполовину вниз:

```
.parent {
  position: relative;
}
.child {
  position: absolute;
  top: 50%;
  transform: translateY(-50%);
}
```

## Вы можете использовать flexbox?

Не удивительно, что всё это можно сделать значительно проще с помощью [flexbox](#).

```
.parent {
  display: flex;
  flex-direction: column;
  justify-content: center;
}
```

## И по горизонтали, и по вертикали

Описанные выше приёмы можно комбинировать как угодно, чтобы получить идеально отцентрованные элементы. Но, по моему опыту, обычно всё сводится к трём вариантам:

## У элемента фиксированная высота и ширина?

Использование отступов с отрицательными значениями, равными половине высоты и ширины элемента после того, как для него было прописано абсолютное позиционирование в точке 50%/50%, отцентрирует элемент внутри родителя. Кроме того, способ хорошо поддерживается браузерами:

```
.parent {
  position: relative;
}

.child {
  width: 300px;
  height: 100px;
  padding: 20px;

  position: absolute;
  top: 50%;
  left: 50%;

  margin: -70px 0 0 -170px;
}
```

## Высота и ширина элемента неизвестны?

Если вам не известны ширина или высота, для центрирования можно использовать свойство `transform` и отрицательное значение `translate` по 50% в обоих направлениях (оно вычисляется от текущей ширины/высоты элемента):

```
.parent {
  position: relative;
}

.child {
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
}
```

## Вы можете использовать flexbox?

Чтобы выполнить центрирование в обоих направлениях с помощью flexbox, нужно использовать два центрирующих свойства:

```
.parent {
  display: flex;
  justify-content: center;
  align-items: center;
}
```

## Заключение

С помощью CSS вы можете отцентрировать всё что угодно.

Оригинальная статья: [Centering in CSS: A Complete Guide](#) Статью вычитывали: [iamstarkov](#), [feonit](#), [NatalieF](#), [FMRobot](#), [yoksel](#)



**Chris Coyier**

<http://chriscoyier.net/>

Twitter: <http://twitter.com/chriscoyier>

GitHub: <http://github.com/chriscoyier>



**Наталья Фадеева**

Twitter: [https://twitter.com/very\\_busy\\_girl](https://twitter.com/very_busy_girl)

GitHub: <https://github.com/NatalieF>

© 2013 Frontender Magazine

Кроме материалов, опубликованных под лицензией [Creative Commons](#)