**Центрирование**

## Горизонтальное

инлайн-элементы в блоке:

text-align: center;

блок внутри родителя, у блока должна быть указана ширина:

margin: 0 auto;

## Вертикальное

### Если размер центрируемого элемента известен, а родителя – нет

**1. position:absolute + margin**

Родителю position:relative, потомку **position:absolute; top:50% и margin-top**:-<половина-высоты-потомка>. Аналогично можно отцентрировать и по горизонтали.

Если мы знаем, что это ровно одна строка, то её высота равна line-height. При стандартных настройках браузера высота строки line-height: 1.25, если поделить на два 1.25em / 2 = 0.625em. Конечно, высота может быть и другой, главное чтобы мы её знали заранее.

<style>

.outer {

**position: relative;**

height: 5em;

border: 1px solid blue;

}

.inner {

**position: absolute;**

**top: 50%;**

**margin-top: -0.625em;**

border: 1px solid red;

}

</style>

<div class="outer">

<div class="inner">Текст</div>

</div>

Минус данного способа — должна быть известна высота внутреннего блока.

2. **растягивание**

.outer {

**position: relative;**

height: 5em;

border: 1px solid blue;

}

.inner {

**position: absolute;**

**top: 0;**

**bottom: 0;**

**margin: auto 0;**

**height: 20px;**

border: 1px solid red;

}

Минус данного способа — должна быть известна высота внутреннего блока.

### Если нужно отцентрировать одну строку в блоке, высота которого известна

Поставить блоку **line-height: <высота блока>.** Нужны конкретные единицы высоты (px,em…). Значение line-height:100% не будет работать, т.к. проценты берутся не от высоты блока, а от текущей line-height.

### Высота родителя известна, а центрируемого элемента – нет.

Поставить **line-height** родителю во всю его высоту, а потомку поставить **display:inline-block**.

Для инлайновых элементов (display:inline/inline-block), включая картинки, свойство **vertical-align** центрирует сам инлайн-элемент в окружающем его тексте.

.outer {

**line-height: 120px;**

border: 1px solid blue;

}

.inner {

**display: inline-block;** /\* центрировать..\*/

**vertical-align: middle**; /\* ..по вертикали \*/

**line-height: 1.25;** /\* переопределить высоту строки на обычную \*/

border: 1px solid red;

}

### Высота обоих элементов неизвестна.

1. Сделать элемент-родитель ячейкой таблицы при помощи **display:table-cell(IE8)** или реальной таблицы, и поставить ему **vertical-align:middle**. Отлично работает, но мы имеем дело с таблицей вместо обычного блока.

Вместе с vertical-align родительский блок получает табличный алгоритм вычисления ширины и начинает подстраиваться под содержимое. Это не всегда желательно. Чтобы его растянуть, нужно указать width явно в пикселях. Если надо в %, то можно починить, завернув «псевдоячейку» в элемент с **display:table**, которому и поставим ширину **width: 100%;**

.outer {

**display: table-cell;**

**vertical-align: middle;**

border: 1px solid blue;

**height: 5em;**

}

.inner {

border: 1px solid red;

}

2. Решение **со вспомогательным элементом** outer:**before** и **инлайн-блоками**.

  <style>

    .outer {

      border: 1px solid blue;

      height: 5em;

    }

    .**outer:before** {

**content: '';**

**display: inline-block;**

**vertical-align: middle;**

**height: 100%;**

    }

    .inner {

**display: inline-block;**

**vertical-align: middle;**

      border: 1px solid red;

    }

  </style>

  <div class="outer">

    <span class="inner"> Центрированный<br>Элемент </span>

  </div>

Минус данного способа — он не может быть применен, если внутренний блок имеет абсолютное позиционирование.

3. Решение с использованием **flexbox**.

.outer {

**display: flex;**

**justify-content: center;** /\*Центрирование по горизонтали\*/

**align-items: center;** /\*Центрирование по вертикали \*/

border: 1px solid blue;

}

Минус данного способа − Flexbox поддерживается только современными браузерами.

4. Выравнивание с помощью **transform**

.outer {

**position: relative;**

**height: 5em;**

border: 1px solid blue;

}

.inner {

**position: absolute;**

**top: 50%;**

**transform: translateY(-50%);**

border: 1px solid red;

}

Минус данного способа в ограниченной поддержке свойства transform старыми версиями браузера IE.