

A decorative graphic on the left side of the slide consisting of two overlapping parallelograms. The front one is blue and the back one is a light mint green. They are positioned diagonally, with the blue one partially covering the green one.

Positioning

HTML & CSS



Positioning. Basics

Позиционирование - управление положением элементов на странице.

Зная особенности разных видов свойства `display` можно позиционировать элементы.

Например, для того, чтобы выстроить несколько блочных элементов в ряд - можно задать им свойство `display: inline-block`. Таким образом, они будут позиционированы в одну строку.

Один из вариантов, чтобы центрировать элемент по горизонтали - задать ему боковые `margin` как `auto`.



Position property

Для позиционирования элементов есть свойство `position` (по умолчанию его значение - `static`).

Когда мы задаем значение отличное от того, что по умолчанию - у элемента появляются новые свойства: `top`, `right`, `bottom`, `left` и `z-index`. С помощью характеристик каждого из значений `position` + вышеуказанных свойств мы можем более точно размещать элементы на странице.



Position property

Position: relative - если при таком position менять свойства top/right/bottom/left - элемент будет перемещаться в нужных направлениях относительно своего первоначального расположения. Место этого элемента остается за ним, т.е. другие элементы не будут пытаться занять пустое место

Position: absolute - элемент будет перемещаться в нужных направлениях относительно ближайшего позиционированного (кроме static) родителя, или относительно body. При этом элемент удаляется из “обычного” потока страницы, другие элементы займут его место.

Position: fixed - элемент располагается относительно viewport, следовательно, даже при скролле будет находится в одном и том же месте. Удаляется из обычного потока страницы.

Position: sticky: смесь position: relative + fixed. Ведет себя как relative до тех пор, пока не встретит указанный offset. после чего ведет себя как fixed



z-index

Z-index - свойство, которое появляется у элемента после того, как ему задан position отличный от static. Отвечает за порядок наложения элементов друг на друга. Чем больше значение - тем выше элемент будет находиться по отношению к другим элементам (в рамках своего контекста, т.е. в контексте страницы, или в новом - если есть группа элементов, которые находятся в позиционированном родителе со своим z-index.)

Посмотрите дома о контексте наложения z-index.



Overflow

Overflow - управление отображением содержимого элемента, если контент выходит за его пределы

Есть три подтипа свойства: `overflow`, `overflow-x`, `overflow-y`;

`overflow: visible` - отображает контент всегда

`overflow: hidden` - все, что выходит за элемент - обрезается

`overflow: scroll` - добавляется прокрутка всегда

`overflow: auto` - добавляется прокрутка если нужно



Float

Свойство float изначально придумано для обтекания изображений текстом. Однако в связи с тем, что в верстальщиках было недостаточно инструментов для позиционирования, его стали использовать в этих целях.

Элемент со свойством float (если не позиционирован как absolute, который игнорирует float):

- выпадает из нормального потока
- помещается вдоль указанного края контейнера, текст и другой контент обтекают элемент.
- в любом случае получает display: block
- Элементы ниже не видят зафлоченный элемент и будут пытаться закрыть его
- *Однако несколько зафлоченных элементов подряд становятся в одну строку*
- Если есть контейнер без указанных размеров, в котором только зафлоченные элементы - он схлопнется



Float

Чтобы другие элементы “увидели” зафлоченный элемент нужно либо задать им свойство `float`, либо же очистить флот - с помощью `clear: both`. Тогда элементы ниже войдут в нормальный поток. Стандартный подход выделить отдельный класс `.clearfix` только ради этого поведения.

Float vs inline-block:

`inline-block` элементы ведут себя по-другому, когда у всех разная высота. С помощью значения `vertical-align` мы можем более точно выровнять элементы в таком случае. + родитель `inline-block` элементов не схлопывается



margin collapsing

Схлопывание margin - явление, когда отступы у соседних по вертикали элементов не суммируются, а объединяются. Так сложилось исторически, чтобы отступы между абзацами рассчитывались правильно (у первого и последнего меньше, чем у всех остальных)

Правила, по которым считается объединение:

- Если два положительных отступа - берется значение того, который больше
- Положительный и отрицательный - их сумма
- Два отрицательных - наибольшее значение по модулю



margin collapsing

Схлопывание не срабатывает если у элементов:

- `display: inline` (думаю, понятно, почему);
- `position: absolute`;
- установлен `float`
- установлено свойство `padding`.
- на стороне схлопывания есть граница;
- + для HTML