## Позиционирование

|  |
| --- |
| [Поток документа](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/1) |
| [Относительное позиционирование](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/2) |
| [position:relative и top](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/3) |
| [position:relative и left](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/4) |
| [position:relative и bottom](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/5) |
| [position:relative и right](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/6) |
| [Относительное позиционирование на практике](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/7) |
| [Абсолютное позиционирование](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/8) |
| [Абсолютное позиционирование и строчные элементы](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/9) |
| [position:absolute и left](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/10) |
| [position:absolute и top](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/11) |
| [position:absolute и right](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/12) |
| [position:absolute и bottom](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/13) |
| [Точка отсчета координат](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/14) |
| [Тренируемся задавать координаты](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/15) |
| [Неявная точка отсчета](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/16) |
| [Абсолютное позиционирование на практике](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/17) |
| [Фиксированное позиционирование](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/18) |
| [z-index или кто кого перекроет](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/19) |
| [Испытание: собери слово «вечность»](https://npoed.htmlacademy.ru/htmlcss110/course/12/run/20) |

# Поток документа [1/20]

Порядок отображения элементов на странице называется потоком документа. Блочные элементы отображаются как прямоугольные области, идущие друг за другом сверху вниз, а строчные элементы располагаются сверху вниз и слева направо и при необходимости переносятся на новую строку.

Элементы можно вкладывать друг в друга. Чем раньше в коде расположен элемент, тем выше он расположен на странице.

Некоторые типы элементов могут вести себя в потоке более сложно. Например, они могут располагаться в несколько колонок, как блочно-строчные элементы. Эти типы проходились в разделе «Блочная модель документа».

Любой элемент, который находится в потоке, занимает на странице свою собственную площадь и, если надо, отталкивает от себя соседей с помощью маргинов.

В этом разделе мы будем изучать свойства, которые изменяют обычное поведение элементов в потоке.

# Относительное позиционирование [2/20]

CSS-свойство position задает режим позиционирования элементов. Значением по умолчанию является static, которое означает «обычное позиционирование».

Значение relative обозначает «относительное позиционирование». Относительно спозиционированный элемент можно перемещать относительно его исходного положения.

Относительно спозиционированный элемент остается в потоке документа, а при его перемещении наблюдаются интересные эффекты. Для начала просто зададим относительное позиционирование выбранному блоку.

# position:relative и top [3/20]

Пока что ничего не произошло, но мы еще и не перемещали элемент. Чтобы эффект перемещения был более наглядным, блок .block-relative обёрнут в блок .block-shadow, который служит «тенью» элемента.

Блок-тень имеет те же размеры, что и блок внутри него. Это позволит нам отследить, как изменяется поведение относительно спозиционированного элемента в потоке.

Свойство top позволяет сместить относительно спозиционированный элемент вверх или вниз относительно его исходного положения. Значение этого свойства обычно задается в пикселях, но можно использовать и другие единицы измерения CSS.

# position:relative и left [4/20]

Вы заметили, что когда мы сместили блок вниз, нижележащие блоки не сдвинулись?

Это происходит потому, что относительно спозиционированный элемент на самом деле никуда не смещается. Он остаётся в потоке ровно там же, где и был (это место обозначено блоком-тенью). А смещается иллюзорная копия блока.

Свойство left позволяет сместить блок влево или вправо относительно исходного положения.

В этом задании вы не только сместите блок по горизонтали, но и убедитесь в том, что он остался в потоке, увеличив его высоту. При этом нижний блок должен будет сместиться.

# position:relative и bottom [5/20]

Свойство bottom аналогично свойству top, только смещает элемент в другом направлении.

Положительное значения свойства bottom сместит элемент вверх, а отрицательное вниз.

Немного попрактикуемся.

# position:relative и right [6/20]

Свойство right аналогично свойству left и тоже смещает элемент в другом направлении.

Как вы уже убедились, относительно спозиционированный элемент можно смещать по горизонтали и по вертикали, комбинируя свойства top/bottom со свойствами left/right.

Ещё раз убедимся, что элемент на самом деле никуда не смещается, увеличив отступ снизу. Размер отступа будет расчитываться от нижнего края блока-тени, а не от края смещенного блока-иллюзии.

# Относительное позиционирование на практике [7/20]

В реальных проектах относительное позиционирование может использоваться для создания декоративных эффектов, хотя область его применения намного шире.

В этом задании вы создадите псевдотень для изображений похожую на тень в отзывах на главной странице HTML Academy. Все стили уже подготовлены, не хватает только тех, которые отвечают за позиционирование.

# Абсолютное позиционирование [8/20]

Значение absolute свойства position задаёт элементу абсолютное позиционирование.

Абсолютно спозиционированные элементы обладают следующими особенностями:

1. Выпадают из потока документа. Место, которое они занимали, становится как бы пустым и его занимают соседние элементы.
2. Ширина по умолчанию зависит от содержимого (а не растягивается на всю доступную ширину).
3. Остаются на том же месте, где были, если не заданы значения свойств top, left, right, bottom.

# Абсолютное позиционирование и строчные элементы [9/20]

Абсолютное позиционирование изменяет поведение не только блочных элементов (ширина по умолчанию), но и строчных.

Если строчный элемент спозиционирован абсолютно, то он ведет себя точно так же, как и абсолютно спозиционированный блочный. Например, такому строчному элементу можно задавать размеры с помощью width и height.

Можно сказать, что есть еще один тип элементов — абсолютно спозиционированные. Такими становятся элементы всех остальных типов (блочные, строчные, блочно-строчные и т.д.), если им задано свойство position:absolute;.

# position:absolute и left [10/20]

Свойства left, top, right и bottom по-разному работают для элементов с абсолютным и относительным позиционированием.

Если для «относительных» элементов эти свойства задают смещение относительно исходной позиции, то для «абсолютных» элементов они задают расположение относительно некой системы координат, как для точки на графике.

По умолчанию такой системой координат является окно браузера, а точки начала координат привязаны к его углам.

Свойство left задает отступ от *левого* края окна браузера до *левого* края абсолютно спозиционированного элемента. Если свойству задать отрицательное значение, то элемент сместится за левый край браузера (при этом полоса прокрутки не появится).

# position:absolute и top [11/20]

Свойство top задает отступ от *верхнего* края окна браузера до *верхнего* края абсолютно спозиционированного элемента.

Таким образом, с помощью свойств left и top можно позиционировать «абсолютный» элемент относительно левого верхнего угла окна браузера.

# position:absolute и right [12/20]

Свойство right задает отступ от *правого* края окна браузера до *правого* края «абсолютного» элемента.

Это свойство позволяет легко прижать элемент любой ширины к правому краю окна. Если задать отрицательное значение для right, то элемент сместится за край окна, но при этом появится полоса прокрутки.

C помощью top + right можно позиционировать элементы от правого верхнего угла.

# position:absolute и bottom [13/20]

Свойство bottom задает отступ от *нижнего* края окна браузера до *нижнего* края «абсолютного» элемента.

В связи с особенностями нашего мини-браузера к свойству bottom добавляется дополнительный отступ, которого в обычных браузерах нет.

Отрицательное значение bottom затолкнет элемент за нижний край окна браузера и при этом появится полоса прокрутки.

# Точка отсчета координат [14/20]

По умолчанию «абсолютные» элементы позиционируются в системе координат, привязанной к окну браузера. Однако, систему координат можно изменить.

Если какой-нибудь из родителей «абсолютного» элемента имеет относительное позиционирование, то элемент будет позиционироваться относительно этого родителя. Если среди родителей несколько относительно позиционированных элементов, то «абсолютный» элемент будет позиционироваться относительно ближайшего из них.

В этом задании вы увидите, как происходит смена системы координат (и соответственно точек отсчета). Интересно, что позиционирование по умолчанию отличается от позиционирования внутри тега body.

# Тренируемся задавать координаты [15/20]

В этом задании вы потренируетесь в позиционировании. Задача заключается в том, чтобы с помощью свойств left, top, right и bottom расположить красный квадрат в каждом из четырех углов синего прямоугольника.

Синие прямоугольники спозиционированы относительно, поэтому у каждого красного квадрата своя система координат.

# Неявная точка отсчета [16/20]

Неявная точка отсчета — это то место, где находился элемент до того, как ему задали абсолютное позиционирование. Если «абсолютному» элементу не задавать свойства left ... bottom, то он выпадет из потока, но останется на прежнем месте.

Такой элемент можно сместить из неявной точки отсчета с помощью свойства margin. Оно будет работать примерно так же, как и свойства left ... bottom для «относительного» элемента.

Кстати, когда значение для left не задано, оно равно auto. То же верно и для оставшихся трёх свойств. С помощью значения auto можно возвращать «абсолютные» элементы на исходное положение.

# Абсолютное позиционирование на практике [17/20]

На практике абсолютное позиционирование часто применяют в декоративных целях, когда нужно аккуратно разместить какие-либо иконки, управляющие элементы или другие «рюшечки».

В этом задании вы узнаете, как применяется абсолютное позиционирование в блоках достижений в HTML Academy. С помощью внутреннего отступа в левую часть блока добавляется пустое пространство, в которое с помощью абсолютного позиционирования выносится иконка достижения.

# Фиксированное позиционирование [18/20]

Значение fixed свойства position задаёт элементу фиксированное позиционирование.

Фиксированное позиционирование похоже на абсолютное, но есть и отличия:

1. Фиксированный элемент тоже выпадает из потока.
2. Фиксированный элемент привязывается к определенной точке в окне браузера и остается на ней всегда, даже при прокрутке страницы.
3. Фиксированный элемент можно позиционировать с помощью свойств top, left, right, bottom, но точка отсчета всегда привязана к окну браузера.

Можете представить себе фиксированный элемент, как стикер, который наклеен на монитор. Такие элементы часто применяют для создания навигационных панелей, привязанных к верхней или нижней части окна браузера.

# z-index или кто кого перекроет [19/20]

Если в одном месте страницы оказываются несколько «абсолютных» блоков, то они перекрывают друг друга. По умолчанию выше оказывается тот блок, который расположен дальше в коде страницы.

C помощью CSS-свойства z-index можно управлять тем, как перекрываются блоки. Значением этого свойства может быть целое число. Чем больше z-index, тем выше располагается блок.

Cвойство z-index работает для элементов, у которых position задано как absolute, fixed и relative. Таким образом, «относительный» элемент может перекрывать «абсолютный».

# Испытание: собери слово «вечность» [20/20]

В «Снежной королеве» Кай собирал из льдинок слово «Вечность». В этом испытании вам нужно собрать из 12 заготовок слово «HTML».

Используйте тот тип позиционирования и те приёмы позиционирования, которые вам больше понравились. Сетка на заднем фоне поможет вам правильно расчитать координаты. Напомним, что маленькая ячейка имеет размер 20 на 20 пикселей, а большая 100 на 100.