ЗАНЯТИЕ 7. АНИМАЦИИ

CSS3 анимация — вдохновение для веб-разработчиков. С ее помощью можно создавать эффекты, которые раньше воспроизводились с помощью скрипта и с использованием графических редакторов.

Анимация применяется к любым html-элементам, а также к псевдоэлементам:before и :after.

Анимация элементов с помощью свойства animation

Поддерживается следующими браузерами:  
IE: 10.0  
Firefox: 16.0, 5.0 -moz-  
Chrome: 43.0, 4.0 -webkit-  
Safari: 4.0 -webkit-  
Opera: 12.1, 12.0 -o-  
iOS Safari: 9, 7.1 -webkit-  
Opera Mini: —  
Android Browser: 44, 4.1 -webkit-  
Chrome for Android: 44

Первое, о чем пойдет речь – правило @keyframes.

**CSS3-анимация** состоит из двух компонентов: сначала идет объявление @keyframes, которое затем используется в свойстве animation элемента.

Правило @keyframes позволяет создавать анимацию с помощью **ключевых кадров** — состояний объекта в определенный момент времени.

Ключевые кадры анимации создаются с помощью ключевых слов from и to (эквивалентны значениям 0% и 100%) или с помощью процентных пунктов, которых можно задавать сколько угодно. Также можно комбинировать ключевые слова и процентные пункты.

Если 0% или 100% кадры не указаны, то браузер пользователя создает их, используя вычисляемые (первоначально заданные) значения анимируемого свойства.

Если у двух ключевых кадров будут одинаковые селекторы, то последующий отменит действие предыдущего.

Общий синтаксис

@keyframes <имя анимации>{

0% {

<cписок свойств>

}

30% {

<cписок свойств>

}

100% {

<cписок свойств>

}

}

Теперь, объявив @keyframes, мы можем ссылаться на него в свойстве animation:

Например

h1

{

color: darkmagenta;

-webkit-animation: shadow  2s infinite ease-in-out;

animation: shadow  2s infinite ease-in-out;

}

У анимации есть ряд ключевых параметров, о которых мы и поговорим:

* Название
* Длительность
* Временная функция
* Задержка
* Повтор
* Направление

Итак, название. Свойство animation-name задает имя анимации. Разумеется, оно должно быть смысловым и максимально отражать суть происходящего в ходе анимации. При этом можно использовать одно слово или несколько, связанных между собой при помощи пробела - или нижнего подчеркивания \_.

Синтаксис

div {

-webkit-animation-name: mymove;

animation-name: mymove;

}

Второе свойство – длительность анимации. Например: animation-duration: 1s;. Не наследуется. Значения задаются в секундах (s), либо миллисекундах (ms).

Синтаксис

div {

-webkit-animation-duration: 2s;

animation-duration: 2s;

}

Следующее свойство - временная функция.

Свойство определяет изменение скорости от начала до конца анимации с помощью временных функций. Задается при помощи ключевых слов или кривой Безье cubic-bezier(x1, y1, x2, y2). Не наследуется.

|  |  |
| --- | --- |
| **animation-timing-function** | |
| Значения: |  |
| ease | Функция по умолчанию, анимация начинается медленно, разгоняется быстро и замедляется в конце. Соответствует cubic-bezier(0.25,0.1,0.25,1). |
| linear | Анимация происходит равномерно на протяжении всего времени, без колебаний в скорости. Соответствует cubic-bezier(0,0,1,1). |
| ease-in | Анимация начинается медленно, а затем плавно ускоряется в конце. Соответствует cubic-bezier(0.42,0,1,1). |
| ease-out | Анимация начинается быстро и плавно замедляется в конце. Соответствует cubic-bezier(0,0,0.58,1). |
| ease-in-out | Анимация медленно начинается и медленно заканчивается. Соответствует cubic-bezier(0.42,0,0.58,1). |
| cubic-bezier(x1, y1, x2, y2) | Позволяет вручную установить значения от 0 до 1. [На этом сайте](http://html5book.ru/goto/http:/roblaplaca.com/examples/bezierBuilder/) есть возможность построить любую траекторию скорости изменения анимации. |
| inherit | Наследует это свойство от родительского элемента. |

Синтаксис

div {

-webkit-animation-timing-function: linear;

animation-timing-function: linear;

}

Далее – анимация с задержкой. Свойство игнорирует анимацию заданное количество времени, что дает возможность по отдельности запускать каждую анимацию. Отрицательная задержка начинает анимацию с определенного момента внутри её цикла, т.е. со времени, указанного в задержке. Это позволяет применять анимацию к нескольким элементам со сдвигом фазы, изменяя лишь время задержки. Если вы хотите, чтобы анимация началась с середины, задайте отрицательную задержку, равную половине времени, установленном в animation-duration. Не наследуется.

Задается в секундах и миллисекундах.

Синтаксис

div {

-webkit-animation-delay: 2s;

animation-delay: 2s;

}

Также возможно задать повтор анимации. Свойство позволяет запустить анимацию несколько раз. Значение 0 или любое отрицательное число удаляют анимацию из проигрывания, бесконечное количество повторов задается значением infinite. Не наследуется.

Синтаксис

div {

-webkit-animation-iteration-count: 3;

animation-iteration-count: 3;

}

Также может задать направление анимации. Свойство задает направление повтора анимации. Если анимация повторяется только один раз, то это свойство не имеет смысла. Не наследуется.

|  |  |
| --- | --- |
| **animation-direction** | |
| Значения: |  |
| alternate | Анимация проигрывается с начала до конца, затем в обратном направлении. |
| alternate-reverse | Анимация проигрывается с конца до начала, затем в обратном направлении. |
| normal | Значение по умолчанию, анимация проигрывается в обычном направлении, с начала и до конца. |
| reverse | Анимация проигрывается с конца. |
| inherit | Наследует это свойство от родительского элемента. |

Синтаксис

div {

-webkit-animation-direction: alternate;

animation-direction: alternate;

}

Стоит сказать, что все параметры воспроизведения анимации можно объединить в одном свойстве — animation, перечислив их через пробел:

animation: animation-name animation-duration animation-timing-function animation-delay animation-iteration-count animation-direction;