

CSS, часть 4



Web.Dev.
Courses

sp.courses.dp.ua

Псевдоклассы

Обтекание – свойство float

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>CSS Example</title>
5      <meta charset="utf-8" />
6      <style>
7
8      </style>
9  </head>
10 <body>
11     <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
12     varius, augue orci congue lacus.</p>
13     <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
14     varius, augue orci congue lacus.</p>
15     <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
16     varius, augue orci congue lacus.</p>
17     <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
18     varius, augue orci congue lacus.</p>
19     <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
20     varius, augue orci congue lacus.</p>
```



Скопируйте заготовку в

Notepad++

<http://files.courses.dp.ua/web/06/ex05.html>

Псевдоклассы

```
6  <style>
7
8  p:hover {
9      background-color: #f0f0f0;
10 }
11
12 </style>
```



Псевдоклассы определяют стили элемента для тех случаев когда состояние элемента меняется в зависимости от действий пользователя. Например псевдокласс `:hover`, который дописывается к селектору, позволяет задать стили который будут применяться только когда на элемент наведён курсор мыши.

Псевдоклассы

```
6  <style>
7  p:hover {
8      background-color: #f0f0f0;
9  }
10
11  p:active {
12      background-color: #909090;
13  }
14  </style>
```



Также будет полезен псевдокласс **:active** – задающий стили для элемента на который осуществляется нажатие.

Псевдоэлементы

Псевдоэлементы `::before`, `::after` и другие

Псевдоэлементы `::before` и `::after` позволяют чем-то дополнить содержимое тега (впереди и сзади соответственно), при помощи стилевого правила **`content`**, для добавляемого содержимого можно использовать и другие стилевые правила.

Псевдоклассы и псевдоэлементы определяются по разному (`:hover`, `::before`), но браузеры для совместимости позволяют записывать псевдоэлементы как и псевдоклассы `:before`.

<https://webref.ru/css/type/pseudoelement>

Псевдоэлементы

```
6 <style>
7 p:hover {
8     background-color: #f0f0f0;
9 }
10
11 p:active {
12     background-color: #909090;
13 }
14
15 p:first-line{
16     color: green;
17 }
18
19 p:first-letter {
20     font-size: 24pt;
21     color:red;
22 }
23 </style>
```



Псевдоэлементы позволяют работать с элементами которые никак не выделены тегами, но которые «концептуально» существуют.

Псевдоэлементы

```
6 <style>
7   p:hover {
8       background-color: #f0f0f0;
9   }
10
11   p:active {
12       background-color: #909090;
13   }
14
15   p::first-line{
16       color: green;
17   }
18
19   p::first-letter {
20       font-size: 24pt;
21       color:red;
22   }
23
24   p::selection {
25       color: red;
26       background: green;
27   }
28 </style>
```



Псевдоэлемент **::selection** позволяют работать с элементами которые никак не выделены тегами, но которые «концептуально» существуют.

Анимация средствами CSS

Анимация на веб-странице это, как правило, изменение во времени того или иного стилевого (css) свойства элемента во времени.

До появления CSS 3 анимация делалась исключительно средствами JavaScript, но ничто не стоит на месте.

*На самом деле **CSS** не позволяет сделать полноценную интерактивную анимацию, но **CSS** позволяет сделать анимированный переход между значениями стилевого правила. Другими словами: сделать плавный переход между двумя состояниями.*

Однако получается у него это великолепно:






<http://jsbin.com/eRiMeDE/2/edit?html,css,output>

Два подхода к анимации в CSS

1. Заранее определенный сценарий, который просто проигрывается и базируется на правиле **animation**.
2. Плавный переход между состояниями некоторых правил, базируется на правиле **transition**.

Анимация переходов

Transition – переходы между






Property					
transition	26.0 4.0 -webkit-	10.0	16.0 4.0 -moz-	6.1 3.1 -webkit-	12.1 10.5 -o-

```
3 <style>
4   div {
5     transition-property: width;
6     /* Свойство изменение которого мы хотим анимировать */
7
8     transition-duration: 2s;
9     /* Время за которое */
10
11    transition-timing-function: linear;
12    /* функция определяющая изменение скорости анимации */
13
14    transition-delay: 1s;
15    /* Задержка перед началом анимации */
16  }
17 </style>
```



```
3 <style>
4   div {
5     transition: width 2s linear 1s;
6   }
7 </style>
```

Transition – переходы между СОСТОЯНИЯМИ

Property					
transition	26.0 4.0 -webkit-	10.0	16.0 4.0 -moz-	6.1 3.1 -webkit-	12.1 10.5 -o-

```
3 <style>
4     div {
5         transition: 1s;
6         /* Будет выполнена анимация изменения всей
7            свойст которые поддерживают анимирование */
8     }
9 </style>
```

1. Анимировать можно только те свойства значения которых задаются в числовом виде, и браузер может рассчитать изменение этих значений во времени;
2. Чтобы анимация перехода работала необходимо указывать начальное значение свойства, т.к. значение с которого начнётся переход.

Transition – переходы между

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>CSS селекторы и не только.</title>
5   <meta charset="utf-8" />
6   <style>
7     div{
8       text-align: center;
9     }
10
11     h1{
12       font-family: Arial;
13       text-align: center;
14       color: blue;
15       font-size: 24pt;
16     }
17
18     img{
19       width: 50px;
20       margin: 10px;
21       padding: 5px;
22       border: 1px solid #dddddd;
23       border-radius: 5px;
24       vertical-align: top;
25     }
26   </style>
27 </head>
28 <body>
29   <h1>TRANSITION DEMO</h1>
30   <div>
31     
32     
33     
34     
35   </div>
36 </body>
37 </html>
```



Скопируйте заготовку в

Notepad++

<http://files.courses.dp.ua/web/05/ex01.html>

Transition – переходы между

```
div{
  text-align: center;
}

h1{
  font-family: Arial;
  text-align: center;
  color: blue;
  font-size: 24pt;
  transition: 1s;
}

h1:hover {
  font-size: 48pt;
  color: orange;
}

img{
  width: 50px;
  margin: 10px;
  padding: 5px;
  border: 1px solid #dddddd;
  border-radius: 5px;
  vertical-align: top;
  transition: 1s;
}
```

```
img:hover{
  width: 250px;
  border: 5px solid #20df43;
  box-shadow: 2px 2px 10px gray;
}
```

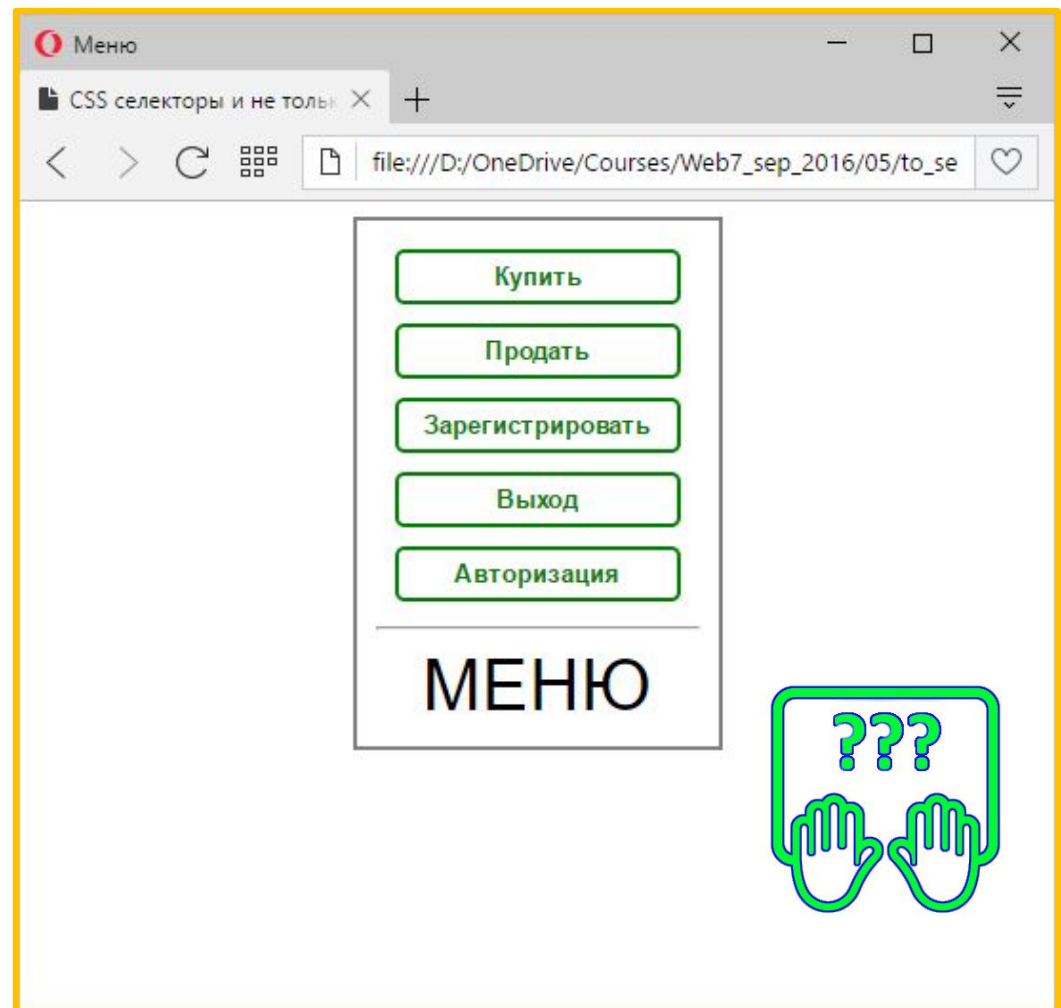
TRANSITION DEMO



*В случае если необходимо сделать плавный переход между состояниями того или иного CSS-правила (например при использовании псевдокласса **:hover**), применяют правило **transition**.*

Применение Transition – «выезжающее»

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>CSS селекторы и не только.</title>
5   <meta charset="utf-8" />
6   <style>
7     .menu {
8       width: 170px;
9       padding: 10px;
10      border: 2px solid gray;
11      text-align: center;
12      margin: 0 auto;
13    }
14    .menu div{
15      font-family: arial;
16      font-size: 28pt;
17    }
18    button {
19      padding: 5px;
20      margin: 5px;
21      background-color: white;
22      color: green;
23      border: 2px solid green;
24      border-radius: 5px;
25      font-weight: 700;
26      width: 150px;
27    }
28    button:active{
29      background-color: #ddffdd;
30    }
31  </style>
32 </head>
33 <body>
34   <div class="menu">
35     <button>Купить</button>
36     <button>Продать</button>
37     <button>Зарегистрировать</button>
38     <button>Выход</button>
39     <button>Авторизация</button>
40     <hr>
41     <div>МЕНЮ</div>
42   </div>
43 </body>
44 </html>
```



Скопируйте заготовку в

Notepad++

<http://files.courses.dp.ua/web/05/ex02.html>

```
.menu {  
  width: 170px;  
  padding: 10px;  
  border: 2px solid gray;  
  text-align: center;  
  margin: 0 auto;
```

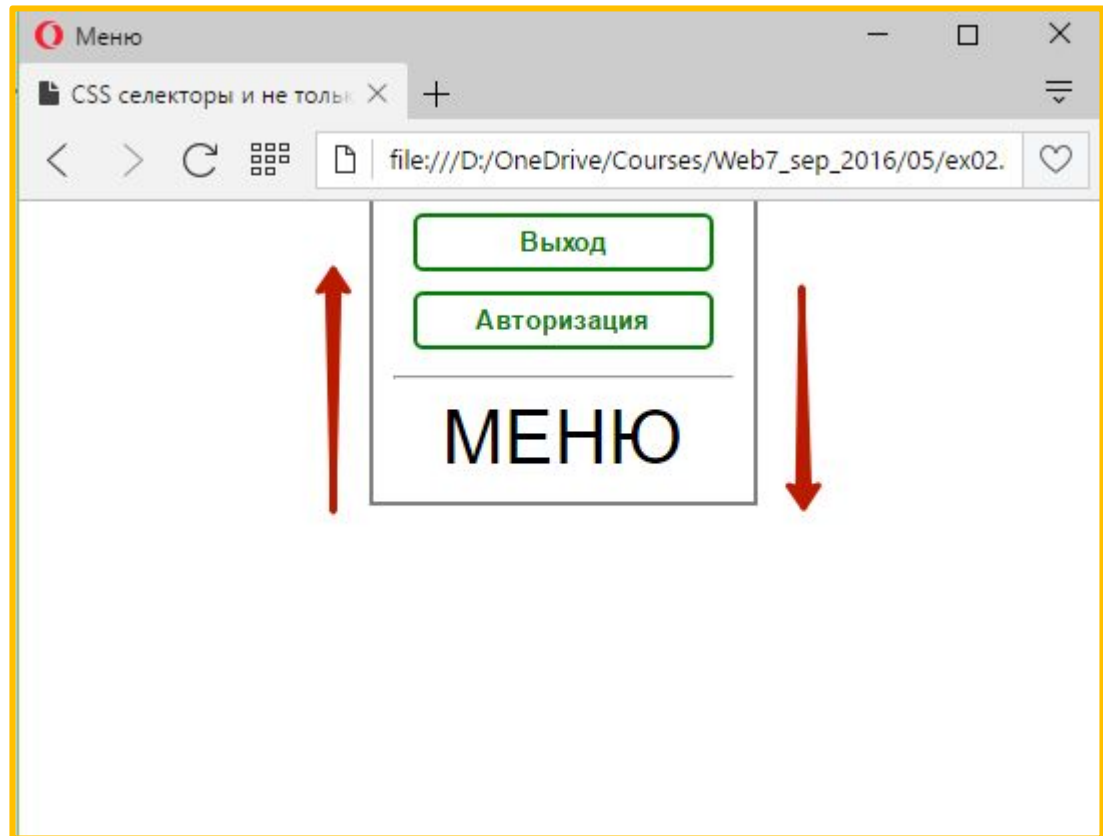
```
  position: relative;  
  top: -215px;  
  transition: 2s;
```

```
.menu:hover {  
  top: 0;  
}
```

```
.menu div {  
  font-family: arial;  
  font-size: 28pt;
```

```
button {  
  padding: 5px;  
  margin: 5px;  
  background-color: white;  
  color: green;  
  border: 2px solid green;  
  border-radius: 5px;  
  font-weight: 700;  
  width: 150px;
```

```
button:active {  
  background-color: #ddffdd;
```



*Разумеется **Transition** способен анимировать перемещение элемента по экрану, или за границы экрана.*

Применение Transition – «разворачивающиеся» элементы

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>CSS Example</title>
5   <meta charset="utf-8" />
6   <style>
7     .preview {
8       width: 500px;
9       height: 180px;
10
11       background: #ddd;
12       padding: 10px;
13       border-radius: 10px;
14       text-align: justify;
15     }
16   </style>
17 </head>
18 <body>
19   <div class="preview">
20     Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec tempus
21     faucibus velit, at bibendum felis faucibus at. Nullam ultrices, dui in
22     pulvinar bibendum, velit justo semper neque, vitae feugiat dui nibh at
23     lacus. Maecenas finibus ante lacus, et sodales magna cursus non. Quisque
24     a sapien ullamcorper, auctor lectus sit amet, convallis urna. Mauris
25     sagittis dignissim interdum. Ut et arcu id dui elementum blandit. Nam
26     ornare ipsum quam, nec finibus leo accumsan a. Fusce molestie mauris ac
27     nisl posuere aliquet. Vestibulum urna libero, elementum in nibh blandit,
28     bibendum posuere ipsum. Aenean ut condimentum urna. In accumsan arcu eget
29     turpis pretium efficitur. Sed ultricies eleifend lacinia. Aenean auctor
30     faucibus elit non molestie.
31   </div>
32 </body>
33 </html>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec tempus faucibus velit, at bibendum felis faucibus at. Nullam ultrices, dui in pulvinar bibendum, velit justo semper neque, vitae feugiat dui nibh at lacus. Maecenas finibus ante lacus, et sodales magna cursus non. Quisque a sapien ullamcorper, auctor lectus sit amet, convallis urna. Mauris sagittis dignissim interdum. Ut et arcu id dui elementum blandit. Nam ornare ipsum quam, nec finibus leo accumsan a. Fusce molestie mauris ac nisl posuere aliquet. Vestibulum urna libero, elementum in nibh blandit, bibendum posuere ipsum. Aenean ut condimentum urna. In accumsan arcu eget turpis pretium efficitur. Sed ultricies eleifend lacinia. Aenean auctor faucibus elit non molestie.



Скопируйте заготовку в

Notepad++

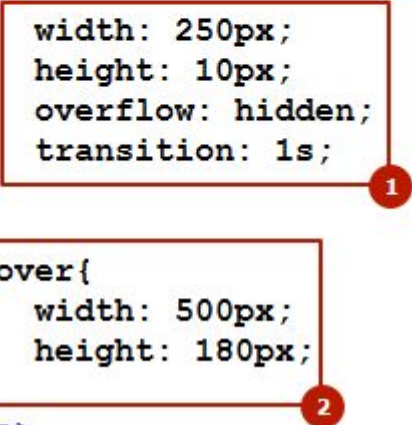
<http://files.courses.dp.ua/web/05/ex03.html>

Применение Transition – «разворачивающиеся» элементы

```
<style>
.preview {
    background: #ddd;
    padding: 10px;
    border-radius: 10px;
    text-align: justify;

    width: 250px;
    height: 10px;
    overflow: hidden;
    transition: 1s;
}

: hover{
    width: 500px;
    height: 180px;
}
</style>
```



Lorem ipsum dolor sit amet,

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec tempus faucibus velit, at bibendum felis faucibus at. Nullam ultrices, dui in pulvinar bibendum, velit justo semper neque, vitae feugiat dui nibh at lacus. Maecenas finibus ante lacus, et sodales magna cursus non. Quisque a sapien ullamcorper, auctor lectus sit amet, convallis urna. Mauris sagittis dignissim interdum. Ut et arcu id dui elementum blandit. Nam ornare ipsum quam, nec finibus leo accumsan a. Fusce molestie mauris ac nisl posuere aliquet. Vestibulum urna libero, elementum in nibh blandit, bibendum posuere ipsum. Aenean ut condimentum urna. In accumsan arcu eget turpis pretium efficitur. Sed ultricies eleifend lacinia. Aenean auctor faucibus elit non molestie.

Transition обеспечивает плавное изменение размеров элементов, даже в том случае если содержимое не помещается в его границы (но не без помощи свойства **overflow**).

Применение Transition – работа с ЦВЕТОМ

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>CSS Example</title>
5      <meta charset="utf-8" />
6      <style>
7          .preview {
8              text-align: center;
9          }
10
11          img {
12              text-align: center;
13              filter: grayscale(100%);
14
15              padding: 10px;
16              border: 2px solid gray;
17              border-radius: 20px;
18          }
19      </style>
20  </head>
21  <body>
22      <div class="preview">
23          
24      </div>
25  </body>
26  </html>
```



Скопируйте заготовку в

Notepad++

<http://files.courses.dp.ua/web/05/ex04.html>

Применение Transition – работа с цветом

```
<style>
.preview {
  text-align: center;
}

img {
  text-align: center;
  padding: 10px;
  border: 2px solid gray;
  border-radius: 20px;
  filter: grayscale(100%);
  transition: 1s;
}

img:hover {
  filter: grayscale(0%);
}
</style>
```

1

2



Transition позволяет выполнить плавный переход от одного цвета к другому, или плавное наложение того или иного спецэффекта (наложенных при помощи стилевого свойства **filter** http://www.w3schools.com/cssref/css3_pr_filter.asp).

Применение Transition – работа с состояниями

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>CSS Example</title>
5   <meta charset="utf-8" />
6   <style>
7
8     button {
9       display: block;
10      margin: 40vh auto;
11      background-color: green;
12      border: 2px solid gray;
13      padding: 20px;
14      color: white;
15      font-size: 16pt;
16      box-shadow: 4px 4px 4px #bbb;
17    }
18
19  </style>
20 </head>
21 <body>
22
23   <button>Button Example</button>
24
25 </body>
26 </html>
```

Button Example

Button Example

Button Example



Скопируйте заготовку в

Notepad++

<http://files.courses.dp.ua/web/05/ex13.html>

Применение Transition – работа с ЦВЕТОМ

```
6 <style>
7
8   button {
9       display: block;
10      margin: 40vh auto;
11      background-color: green;
12      border: 2px solid gray;
13      padding: 20px;
14      color: white;
15      font-size: 16pt;
16      box-shadow: 4px 4px 4px #bbb;
17      transition: 0.5s;
18   }
19
20   button:hover {
21       border-radius: 25px;
22       background-color: orange;
23   }
24
25   button:active{
26       border: 2px solid #555;
27       background-color: red;
28       box-shadow: 2px 2px 4px #bbb;
29       border-radius: 25px 75px;
30   }
31
32
33 </style>
```

Button Example

Button Example





Button Example

Transition позволяет выполнить плавный переход между состояниями, которые мы определяем при помощи псевдоклассов - `:hover`, `:active`).

Анимация «по сценарию»

Animation - анимация «по

```
3  @keyframes my-script-name {
4  /* my-script-name - имя нашего сценария */
5      0% {
6          /* Временная отметка */
7
8          rotate(0deg) ;
9          /* Перечень свойств и какие значения они должны
10             принять при достижении временной отметки */
11      }
12      25% {
13          rotate(30deg) ;
14      }
15
16      70% {
17          rotate(120deg) ;
18      }
19
20      100% {
21          rotate(360deg) ;
22      }
23  }
```

Property					
@keyframes	43.0 4.0 -webkit-	10.0	16.0 5.0 -moz-	9.0 4.0 -webkit-	30.0 15.0 -webkit- 12.0 -o-
animation	43.0 4.0 -webkit-	10.0	16.0 5.0 -moz-	9.0 4.0 -webkit-	30.0 15.0 -webkit- 12.0 -o-

Анимация «по сценарию» состоит из двух шагов.

Первый шаг: создается «сценарий» при помощи директивы **@keyframes** задаются «опорные кадры» т.е. состояния через которые наша анимация должна проходить.

Animation - анимация «по

```
3  div {
4      animation-name: my-script-name;
5      /* Имя сценария по которому выполняется анимация */
6
7      animation-duration: 5s;
8      /* Общее время выполнения анимация */
9
10     animation-timing-function: linear;
11     /* Функция определяющая изменение скорости анимации */
12
13     animation-delay: 2s;
14     /* Задержка перед стартом анимации */
15
16     animation-iteration-count: infinite;
17     /* Количество повторений анимации */
18
19     animation-direction: alternate;
20     /* Направление анимации forward/reverse/alternate */
21 }
```



```
4  <style>
5      div {
6          animation: my-script-name 5s linear 2s infinite alternate;
7      }
8  </style>
```

Анимация «по сценарию» состоит из двух шагов.

Второй шаг: при помощи правила **animation** «сценарий» крепиться к какому-либо элементу.

Анимация по

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>CSS Example</title>
5   <meta charset="utf-8" />
6   <style>
7     img {
8       width: 100%;
9     }
10    .preview {
11      text-align: justify;
12      width: 200px;
13      padding: 10px;
14      margin: 20px;
15      border: 3px solid #ddd;
16      border-radius: 20px;
17      font-size: 8pt;
18      font-family: sans-serif;
19      margin: 50px auto;
20    }
21  </style>
22 </head>
23 <body>
24   <div class="preview">
25     
26     Lorem ipsum dolor sit amet, consectetur adipiscing elit. In blandit commodo nibh, ac
27     semper quam finibus vel. Vestibulum sed vestibulum odio. Sed ac pretium orci. Fusce
28     ultrices, odio vitae lacinia ornare, dolor nibh elementum lectus, sed convallis massa
29     eros sed erat. Quisque et hendrerit libero, eget pretium augue.
30   </div>
31 </body>
32 </html>
```



Состоит из двух шагов.

Первый шаг: создается «сценарий» при помощи директивы **@keyframes** задаются «опорные кадры» т.е. состояния через которые наша анимация должна проходить.

Второй шаг: при помощи правила **animation** «сценарий» крепиться к какому-либо элементу.

<http://files.courses.dp.ua/web/05/ex05.html>

Анимация по «сценарию»



```
<style>
img {
  width: 100%;
}

.preview {
  text-align: justify;
  width: 200px;
  padding: 10px;
  margin: 20px;
  border: 3px solid #ddd;
  border-radius: 20px;
  font-size: 8pt;
  font-family: sans-serif;
  margin: 50px auto;
}

  position: relative;
  animation: my-scenario 10s linear infinite 2s;
}

@keyframes my-scenario {
  0% { background-color: white; left: 0; top: 0; width: 250px; }
  15% { background-color: red; left: 300px; top: 100px; width: 300px; transform: rotate(45deg); }
  50% { background-color: green; left: -300px; top: 100px; width: 150px; transform: rotate(-45deg); }
  75% { background-color: blue; left: 300px; top: 300px; width: 450px; transform: rotate(100deg); }
  90% { background-color: yellow; left: -300px; top: 300px; width: 200px; transform: rotate(-30deg); }
  100% { background-color: white; left: 0; top: 0; width: 250px; transform: rotate(0deg); }
}
</style>
```

Браузер проигрывает сценарий делая плавные переходы между состояниями для опорных кадров, сценарий может быть проигран однократно, или многократно, при многократном повторении необходимо позаботиться о «склейке» финального и начального кадра.

CSS Библиотеки

Font Awesome – иконки на базе CSS



<http://fontawesome.io/>

Как работает Font Awesome на примере

<https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css>

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>CSS Example</title>
5      <meta charset="utf-8" />
6      <link href="
7          https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-awes
8          ome.min.css" rel="stylesheet">
9  </head>
10 <body>
11     <p><span class="fa fa-file-o fa-3x"></span> Lorem ipsum
12     dolor sit amet, consectetur adipiscing elit. Duis libero
13     mauris, commodo ut nibh id, venenatis sollicitudin nibh.
14     Pellentesque pretium, mauris ut cursus varius, augue orci
15     congue lacus.</p>
16     <button><span class="fa fa-usd"></span> Купить</button>
17
18     <button><span class="fa fa-usd fa-spin"></span> Купить
19     </button>
20 </body>
21 </html>
```

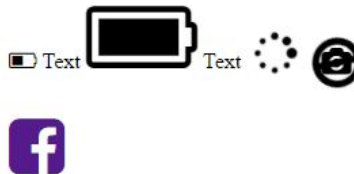


Font Awesome – иконки на базе CSS

```
10 <span class="fa fa-battery-half"></span> Text
11 <span class="fa fa-battery-full fa-4x"></span> Text
12 <span class="fa fa-spinner fa-pulse fa-2x fa-fw"></span>
13 <span class="fa-stack fa-lg">
14   <i class="fa fa-camera fa-stack-1x"></i>
15   <i class="fa fa-ban fa-stack-2x text-danger fa-spin"></i>
16 </span>
```



Font Awesome Demo



Font Awesome – CSS-библиотека содержащая в своём составе огромное количество векторных иконок, «на все случаи жизни».

Использовать иконки из состава библиотеки просто – достаточно указать для любого строчного элемента класс **fa** и класс конкретной иконки например **fa-facebook** в результате иконка будет вставлена в качестве **::before** для

Сокраще...

Сокращение текста

CSS предоставляет возможность сокращать текст который не помещается в пределы блока, и автоматически ставить троеточие. Работает только для блочных элементов.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>CSS Селекторы и не только.</title>
5   <meta charset="utf-8" />
6   <style>
7     .myclass{
8       white-space: nowrap;
9       overflow: hidden;
10      text-overflow: ellipsis;
11
12      width: 50vw;
13      background-color: yellow;
14    }
15
16    .myclass:hover{
17      white-space: normal;
18    }
19  </style>
20 </head>
21 <body>
22   <div class="myclass youclass" id="myid" style="">Lorem ipsum dolor sit amet, consectetur
23     adipiscing elit. Nullam vitae fringilla est. Praesent semper ullamcorper ipsum, sit amet
24     congue dui. Vivamus mollis ultricies ante aliquam dictum. Fusce tempus iaculis ornare.
25     Donec placerat convallis mi ut luctus. Ut scelerisque turpis at urna fermentum ultrices.
26   </div>
27 </body>
28 </html>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam vitae fr...

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam vitae fringilla est. Praesent semper ullamcorper ipsum, sit amet congue dui. Vivamus mollis ultricies ante aliquam dictum. Fusce tempus iaculis ornare. Donec placerat convallis mi ut luctus. Ut scelerisque turpis at urna fermentum ultrices.



Скопируйте код разметки в notepad++.

<http://files.courses.dp.ua/web/04/ex02.html>

Сокращение текста

CSS предоставляет возможность сокращать текст который не помещается в пределы блока, и автоматически ставить троеточие. Работает только для блочных элементов.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>CSS Селекторы и не только.</title>
5   <meta charset="utf-8" />
6   <style>
7     .myclass{
8       white-space: nowrap;
9       overflow: hidden;
10      text-overflow: ellipsis;
11
12      width: 50vw;
13      background-color: yellow;
14    }
15
16    .myclass:hover{
17      white-space: normal;
18    }
19  </style>
20 </head>
21 <body>
22   <div class="myclass youclass" id="myid" style="">Lorem ipsum dolor sit amet, consectetur
23     adipiscing elit. Nullam vitae fringilla est. Praesent semper ullamcorper ipsum, sit amet
24     congue dui. Vivamus mollis ultricies ante aliquam dictum. Fusce tempus iaculis ornare.
25     Donec placerat convallis mi ut luctus. Ut scelerisque turpis at urna fermentum ultrices.
26   </div>
27 </body>
28 </html>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam vitae fr...

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam vitae fringilla est. Praesent semper ullamcorper ipsum, sit amet congue dui. Vivamus mollis ultricies ante aliquam dictum. Fusce tempus iaculis ornare. Donec placerat convallis mi ut luctus. Ut scelerisque turpis at urna fermentum ultrices.



Скопируйте код разметки в notepad++.

<http://files.courses.dp.ua/web/04/ex02.html>

Для тренировки

Где взять задания для тренировки?

ЗАДАЧИ ПО HTML И CSS

HTML И CSS, ПРОГРАММИРОВАНИЕ

Задачи по HTML и CSS



HTML & CSS

Часть 1. Задачи по HTML без использования стилей

1. Задача: ["Только день не повторится..."](#)
2. Задача: ["Заповедь"](#)
3. Задача: ["Silence is golden"](#)
4. Задача: ["Проект"](#)
5. Задача: ["Статьи"](#)
6. Задача: ["Генераторы"](#)