

**CSS**

**<<Pseudo Classes and Pseudo Elements>>**

# Псевдоклассы

**Псевдоклассы** описывают характеристики элементов которые не отображаются в исходном документе и не принадлежат дереву документа **DOM**

В процессе взаимодействия пользователя со страницей **html-элементы** приобретают **динамические характеристики**. Например, переход по ссылке, **html-элемент** `<a>текст ссылки</a>` приобретает следующие свойства:

- когда наводите на ссылку, она получает **a:hover**
- когда перешли по ссылке — **a:visited**

Доступ к таким динамическим характеристикам можно получить только с помощью **псевдоклассов**

**Псевдоклассы** делятся на **статические**, например, **:last-of-type**, и **динамические**, например, **:hover**, т.е. появляющиеся у элемента в процессе взаимодействия пользователя с веб-страницей.

Имена псевдоклассов чувствительны к регистру.

Некоторые псевдоклассы взаимоисключающие, в то время как другие могут одновременно применяться к одному элементу

# Синтаксис и типы

```
selector : pseudo-class-name{  
    /* properties */  
}
```

Виды псевдоклассов:

- динамические псевдоклассы
- псевдоклассы пользовательского интерфейса
- структурные псевдоклассы
- целевой псевдокласс **:target**
- языковой псевдокласс **:lang()**
- псевдокласс отрицания **:not()**

# Динамические псевдоклассы

Динамические псевдоклассы применяются к элементам у которых меняются состояния, в основном это ссылки но и некоторые псевдоклассы можно применять и не только к ссылкам

**:link** — не посещенная ссылка

**:visited** — посещенная ссылка

**:focus** — ссылки, а также элементы форм, которые активированы посредством курсора мыши или на которые перешли с помощью клавиши **TAB**

**:hover** — ссылки, а также другие элементы, стили применяются при наведении пользователем на элемент

**:active** — выбирает элемент, активированный пользователем с помощью клика мышки. Обычно применяется для ссылок, но может отбирать и другие элементы на странице

```
a{
  border: 1px solid black;
  border-radius: 10px;
  padding: 10px 40px;
  text-transform: uppercase;
  text-decoration: none;
  color: black;
}

a:hover{
  border-color: blue;
  color: blue;
  background-color: yellow;
}
```

# Псевдоклассы пользовательского интерфейса

Относятся к элементам форм

**:disabled** — используется для выборки заблокированных для выбора и изменения элементов форм

**:enabled** — используется для выборки не заблокированных элементов формы

**:checked** — применяется для выбора и стилизации элементов `<input type="radio">`, `<input type="checkbox">`, а также элементов `<option></option>`, находящихся внутри элемента `<select></select>`

```
<input type="text" disabled>
<input type="text">
<input type="checkbox"><span>CSS</span>
<input type="checkbox"><span>HTML</span>
```

```
input:disabled{
    border:2px solid red;
}

input:checked + span{
    background-color: red;
}
```

# Структурные псевдоклассы

Позволяют отбирать элементы на основании информации, которая отражена в дереве документа и не может быть получена при помощи простых селекторов или их комбинации. Отсчёт начинается с элемента с индексом 1.

**:root** — элемент, являющийся корневым в документе

**:nth-child()** — элементы на основе их индекса (в порядке очереди) внутри их родительского контейнера. Значения `even`, `odd`, `position(3)`, `formula (3n+1)`

**:nth-last-child()** — дочерние элементы на основе их индекса внутри контейнера, при этом отсчёт идёт в обратном порядке, т.е. начиная с последнего элемента. Значениями аргумента могут быть положительные, отрицательные значения, а также ключевые слова `even` `odd`

**:nth-of-type()** — элементы одного типа на основе их индекса внутри контейнера

**:nth-last-of-type()** — элементы одного типа на основе их индекса внутри контейнера, начиная с последнего элемента к первому

# Структурные псевдоклассы

**:first-child** — элемент, который является первым дочерним элементом некоторого другого элемента

**:last-child** — последний дочерний элемент элемента-контейнера

**:first-of-type** — первый элемент данного типа среди дочерних элементов родительского элемента-контейнера

**:last-of-type** — последний элемент данного типа среди дочерних элементов родительского элемента-контейнера

**:only-child** — дочерний элемент, который является единственным дочерним элементом

**:only-of-type** — элемент, который является единственным элементом данного типа в родительском элементе;

**:empty** — элемент, который не содержит ни одного дочернего элемента.

```
<ul>
  <li>Java</li>
  <li>HTML</li>
  <li>CSS</li>
  <li>JavaScript</li>
  <li>SpringBoot</li>
  <li>Android</li>
  <li>iOS</li>
</ul>
```

```
ul li:nth-child(odd) {
  background-color: #333;
  color: white;
}
```

```
ul li:last-child{
  background-color: #333;
  color: white;
}
```

```
ul li:nth-child(3n+1) {
  background-color: #333;
  color: white;
}
```



# Целевой псевдокласс :target

С помощью данного псевдокласса можно стилизовать элемент, к которому перешли по якорной ссылке на странице (навигация по странице), а также элементы, которые расположены в одном блоке и при нажатии на кнопку/ссылку сменяются/перемещаются (например, css-слайдер)

```
<div>
  <a href="#one" class="current">One</a>
  <a href="#two">Two</a>
</div>
<div id="info">
  <p id="one">Lorem ...</p>
  <p id="two">Lorem...</p>
</div>
```

```
#info p {
  background-color: transparent;
}

#info p:target {
  background-color: red;
}
```

# Языковой псевдокласс :lang()

Используется когда в документе содержатся абзацы текста на разными языками. Чтобы браузер различал их, элементу с текстом добавляется атрибут **lang** с кодом языка

```
<p lang="he">טוב בוקר</p>  
<p lang="en">Hello world</p>
```

```
p:lang(he) {  
    text-align: right;  
    background-color: green;  
    color: white;  
}  
  
p:lang(en) {  
    background-color: blue;  
    color: white;  
}
```

# Псевдокласс отрицания :not()

Функциональный селектор псевдокласса, который принимает в качестве значения простой селектор, а затем отбирает элементы, которые не содержат указанное значение аргумента. Значениями аргументов могут быть только следующие:

селекторы элемента – **body :not(strong)**

селекторы класса и идентификатора – **p:not(.text)**

селекторы псевдоклассов – **ul:not(:first-child)**

селекторы атрибута – **input:not([type="checkbox"])**

# Комбинация псевдоклассов

При стилизации элементов возможна комбинация псевдоклассов

```
tr:nth-last-child(even) :hover {  
    background-color: #ccc;  
}
```

добавит стили при наведении каждой чётной строке таблицы (отсчёт в обратном порядке);

```
h2:not(:first-of-type) :not(:last-of-type) {  
    background-color: #ccc;  
}
```

добавит стили для всех элементов **<h2>**, кроме первого и последнего элемента данного типа.

# Псевдоэлементы

Позволяет стилизовать определенную часть выбранного элемента

Если псевдоклассы записываются с одним двоеточием, то псевдоэлементы – с двумя. Это было внедрено в **CSS3** для того, чтобы различать псевдоклассы и псевдоэлементы между собой.

**::after** – используется вместе со свойством **content** и позволяет вывести необходимые данные после содержимого элемента

**::before** – выполняет похожую функцию, что и предыдущий, только выводит данные перед содержимым элемента

**::selection** – этот псевдоэлемент распознается браузерами только при использовании двух двоеточий и позволяет установить цвет и фон для текста, который выделен пользователем

**::first-letter** – используется для изменения стиля первого символа в тексте элемента

**::first-line** – используется для изменения стиля первой строки текста элемента

```
<blockquote>Lorem ipsum dolor sit amet...</blockquote>
```

```
blockquote::before {  
  content: "«";  
}  
blockquote::after {  
  content: "»";  
}  
blockquote::selection {  
  color: #C8F7C5;  
  background-color: #1E824C;  
}
```

```
<div class="content">  
  <p>Lorem ipsum dolor sit amet...</p>  
  <p>Lorem ipsum dolor sit amet...</p>  
  <p>Lorem ipsum dolor sit amet...</p>  
</div>
```

```
.content p:first-letter {  
  font-size: large;  
  font-weight: bold;  
  color: red;  
}
```