

# ***Przydatne strony***

<http://michalbiniek.blogspot.com/2011/04/ps-eudoklasy-css3.html>

[http://www.w3schools.com/css/css\\_pseudo\\_classes.asp](http://www.w3schools.com/css/css_pseudo_classes.asp)

<http://poradnikwebmastera.blox.pl/2011/02/CSS3-pseudoklasa-target.html>

<http://www.sitepoint.com/current-generation-css3-selectors/>

<http://www.impressivewebs.com/dynamically-highlight-content-wikipedia-css3/>

# Co to są pseudoklasy?

```
selector:pseudo-class {  
    property:value;  
}
```

Style CSS są dodawane do elementów na podstawie ich pozycji w drzewie dokumentu. Taki model nie zawsze jednak jest wystarczający. Pseudoklasy klasyfikują elementy inaczej niż po ich nazwie, atrybutach czy zawartości, tzn. nie są ustalane na podstawie drzewa dokumentu. Mogą być dynamiczne, to znaczy element "nabywa" lub "traci" pseudoklasę podczas interakcji z użytkownikiem. Przykładem jest podświetlenie elementu po wskazaniu go myszką przez użytkownika.

# Pseudoklasy akcji

<b>Hover</b>	<b>Selektor nadaje styl elementowi, gdy urządzenie wskazujące, np. myszka, znajduje się nad elementem. Styl może być “nadpisany”, jeśli po :hover znajduje się np. styl :link. Example : a:hover {color: rgb(100%,0%,100%) }</b>
<b>Active</b>	<b>Styl jest nadawany wtedy, gdy element staje się aktywny, np. w czasie między naciśnięciem przycisku i zwolnieniem go. Przykład : a:active {color: rgb(100%,0%,100%) }</b>
<b>Focus</b>	<b>Odnosi się do aktywnego elementu (zgodnie ze specyfikacją - akceptującego zdarzenia z klawiatury lub myszy) - z reguły są to pola formularza.</b>
<b>Link</b>	<b>Element nieodwiedzony</b>
<b>Visited</b>	<b>Element odwiedzony</b>

# Pseudoklasy struktury- odwołania do dzieci

`dziecko:first-child { cecha: wartość }`

*Dowolny rodzic:*

`rodzic > dziecko:first-child { cecha: wartość }`

*Określony rodzic*

`p > b:first-child { color: red }` *(uwaga- jeżeli pierwszym dzieckiem nie będzie b, to styl nie zostanie nadany)*

*odpowiednie atrybuty otrzyma pogrubienie (znacznik b), będący pierwszym dzieckiem elementu p.*

`p:first-child b { color: red }`

*Odpowiednie cechy otrzyma znacznik b, będący potomkiem elementu p, który z kolei jest czyimś pierwszym dzieckiem (nie ma znaczenia czyim)*

`dziecko:last-child { cecha: wartość }`

*Dowolny rodzic:*

`rodzic > dziecko:last-child { cecha: wartość }`

*Określony rodzic*

`dziecko: nth-child(an+b){ cecha: wartość }`

```
#tab3 tr:nth-child(3n){  
background-color:#85EA6C;}  
#tab3 tr:nth-child(3n+1){  
background-color:#4E46D7;}
```

ta tabela występuje jako druga w DOM id="tab1" #tab1  
 tr:first-child{background-color:rgb(227,152,206);} #tab1  
 tr:last-child{background-color:#6D4461;}

abc	dert
ala	ola
xcv	ssss
wsx	rfv

ta tabela występuje jako pierwsza w DOM id="tab4"  
 table:first-child{ background-color:#85EA6C;}

Month	Savings
January	\$100
February	\$80
Sum	\$180

id="tab2" table th:first-child{ color:red; }

First Name	Last Name	Points
Jill	Smith	50
Eve	Jackson	94
John	Doe	80

#tab3 tr:nth-child(3n){ background-color:#85EA6C; } #tab3  
 tr:nth-child(3n+1){ background-color:#4E46D7; } #tab3  
 tr:nth-child(3n+2){ background-color:#FAEF21; }

Jill	Smith	50
Eve	Jackson	94
Jill	Smith	50
Eve	Jackson	94
Jill	Smith	50
Eve	Jackson	94
Jill	Smith	50
Eve	Jackson	94

```

td, th {
border-width:2px;
border-color: black;
border-style:dotted;
padding:5px;
}
table th:first-child{
color:red;
}
table{
margin-top:50px;
margin-left:20px;
border-style:solid;
border-color: green;
}
body{
color:rgb(3%,1%,60%);
font-size:25px;
font-weight:bold;
margin:0px;
padding:0px;
}
#tab4{position:absolute;
top:40px;
right:50px;
width:40%;
}
table:first-child{
background-color:#85EA6C;
}
#tab1 tr:first-child{
background-color:rgb(227,152,206);
}
#tab1 tr:last-child{
background-color:#6D4461;
}
#tab3 tr:nth-child(3n){
background-color:#85EA6C;
}
#tab3 tr:nth-child(3n+1){
background-color:#4E46D7;
}
#tab3 tr:nth-child(3n+2){
background-color:#FAEF21;
}

```

# Pseudoklasa: target

Służy do wyróżniania celów wewnętrznych linków w dokumencie. Przykładowo, gdy w długim dokumencie zawierającym śródtytuły tworzymy spis treści z odnośnikami do poszczególnych śródtytułów, kliknięcie odnośnika spowoduje nie tylko skok do odpowiadającego mu śródtytułu, ale i wyróżnienie tego śródtytułu odmiennym formatowaniem.

```
<body>
<h1>spis treści</h1>
<ul>
<li><a href="#jeden">rozdział 1</a></li>
<li><a href="#dwa">rozdział 2</a></li>
<li><a href="#trzy">rozdział 3</a></li>
<li><a href="#cztery">rozdział 4</a></li>
</ul>
<h1 id="jeden" >rozdział 1</h1>
Pellentesque habitant morbi tristique senectus et
eget, tempor sit amet, ante. Donec eu libero sit
Pellentesque habitant morbi tristique senectus et
eget, tempor sit amet, ante. Donec eu libero sit
Pellentesque habitant morbi tristique senectus et
eget, tempor sit amet, ante. Donec eu libero sit
Pellentesque habitant morbi tristique senectus et
eget, tempor sit amet, ante. Donec eu libero sit
<h1 id="dwa" >rozdział 2</h1>
Pellentesque habitant morbi tristique senectus et
eget, tempor sit amet, ante. Donec eu libero sit
```

```
:target{
color:red;
font-style:italic;
}
```

## spis treści

- [rozdział 1](#)
- [rozdział 2](#)
- [rozdział 3](#)
- [rozdział 4](#)

## *rozdział 1*


Pellentesque habitant morbi tristique senectus et malesuada fames ac turpis egestas. Vestibulum quam sapien-  
tiam ut utriusque ullamcorper magna consequat. Pellentesque  
mi vitae est. Mauris placerat eleifend leo. Ut magna dolor  
ultricies eget, tempor sit amet, ante. Donec eu libero sit  
tristique senectus et netus et malesuada fames ac turpis  
egestas semper. Aenean ultricies mi vitae.

## rozdział 2

Pellentesque habitant morbi tristique senectus et

# Pseudoelementy w CSS3

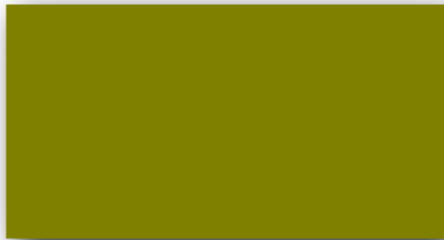
```
selector::pseudo-element { property:value; }
```

<b><i>::first-line</i></b> (tylko dla selektorów typu block)	<b><i>p::first-line</i></b> { <b><i>color: #ff0000;</i></b> <b><i>font-variant: small-caps;</i></b> }	<b>YOU CAN USE THE ::FIRST-LINE PSEUDO-ELEMENT TO ADD A</b> special effect to the first line of a text. Some more text. And even more, and more, and more, and more, and more, and more, and more, and more, and more, and more, and more.
<b><i>::first-letter</i></b> (tylko dla selektorów typu block)	<b><i>p::first-letter</i></b> { <b><i>color: #ff0000;</i></b> <b><i>font-size: xx-large;</i></b> }	<b>Y</b> ou can use the ::first-letter pseudo-element to add a special effect to the first character of a text!
<b><i>::before</i></b>	<b><i>h1::before</i></b> { <b><i>content: url(smiley.gif);</i></b> }	 <b>This is a heading</b>  The ::before pseudo-element inserts content before an element.
<b><i>::after</i></b>	<b><i>h1::after</i></b> { <b><i>content: url(smiley.gif);</i></b> }	<b>This is a heading</b>   The ::after pseudo-element inserts content after an element.

# box-shadow

***box-shadow : h-shadow v-shadow blur color inset***

```
<div style="width: 200px; height: 100px; background-color:olive; box-shadow: 5px 5px 20px #182019;">  
</div>
```



```
<div style="width: 200px; height: 100px; background-color:olive; box-shadow: inset -5px -5px 15px #182019;">  
</div>
```





# Border-radius

1. Zaokrąglenie górnego-lewego narożnika:

```
selektor { border-top-left-radius: zaokrąglenie }
```

2. Zaokrąglenie górnego-prawego narożnika:

```
selektor { border-top-right-radius: zaokrąglenie }
```

3. Zaokrąglenie dolnego-prawego narożnika:

```
selektor { border-bottom-right-radius: zaokrąglenie }
```

4. Zaokrąglenie dolnego-lewego narożnika:

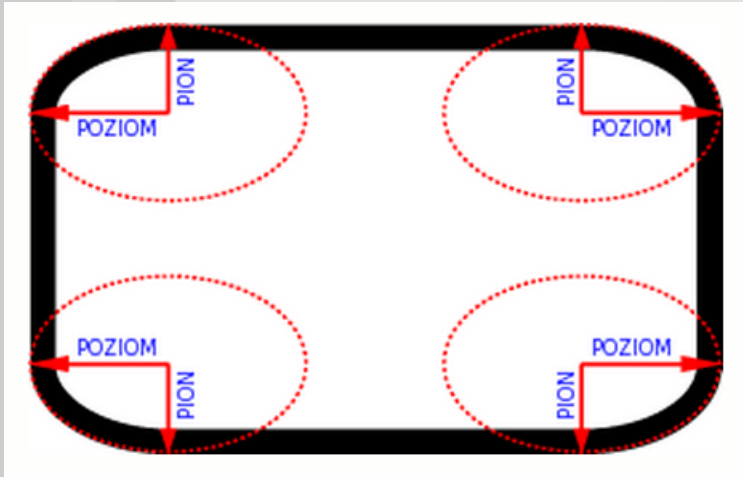
```
selektor { border-bottom-left-radius: zaokrąglenie }
```

```
border-radius:2em;
```

is equivalent to:

```
border-top-left-radius:2em;  
border-top-right-radius:2em;  
border-bottom-right-radius:2em;  
border-bottom-left-radius:2em;
```

# Border-radius



```
border-top-left-radius: 2em 0.5em;  
border-top-right-radius: 1em 3em;  
border-bottom-right-radius: 4em 0.5em;  
border-bottom-left-radius: 1em 3em;
```

Zaokrąglone narożniki to bardzo ciekawy efekt wizualny, który może zwiększyć estetykę strony. Co ciekawe, element wcale nie musi mieć obramowania, aby miał zaokrąglone narożniki. Wystarczy jeśli będzie miał ustalone jakieś tło, aby efekt był widoczny na ekranie. Ale uwaga: tekst wewnątrz elementu nie zostanie zawinięty do zaokrąglonych rogów. Jeśli promień zaokrąglenia będzie zbyt duży, tekst wyjdzie na zewnątrz.

# *multimedia*

```
<p>video src="italian.mp4" controls</p>  
<video src="italian.mp4" controls  
width="600" height="400"></video>
```

# transition

```
div {  
  width: 100px;  
  height: 100px;  
  background: red;  
  -webkit-transition-property: width; /* Safari */  
  -webkit-transition-duration: 5s; /* Safari */  
  transition-property: width;  
  transition-duration: 5s;  
}  
  
div:hover {  
  width: 300px;  
}
```



Hover over the div element above, to see the transition effect.



Hover over the div element above, to see the transition effect.

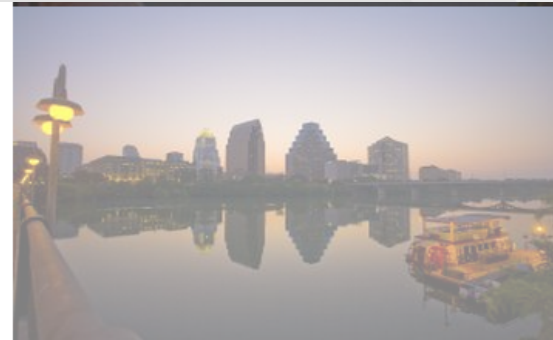
# opacity

**Polecenie opacity służy do ustalania stopnia nieprzezroczystości (zaczernienia) obiektu. Im wyższa wartość w przedziale [0-1], tym bardziej nieprzezroczysty jest obiekt, im niższa - tym bardziej przezroczysty (transparentny).**

**Przykład:**

```

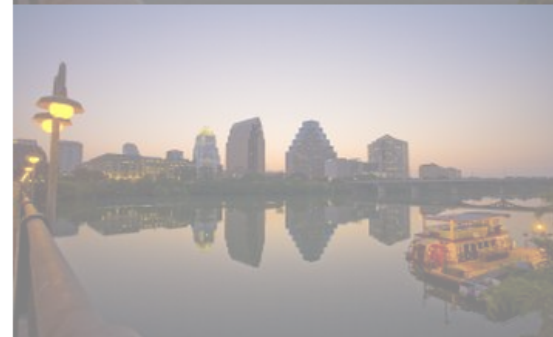
```



opacity:0.5;



opacity:0.8;



opacity: 0.1;

# opacity

```
#im1{  
opacity: 1;  
transition-property: opacity;  
transition-duration: 6s;  
}  
#im1:hover{  
opacity:0;  
}
```



połączenie kilku własności: `transition-property: opacity;`  
`transition-duration: 6s;` `#im1:hover{ opacity:0;}` Proszę  
przesunąć kursor na obrazek

