

## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

**Tematy:** [Pozycjonowanie elementów](#) | [Warstwy](#) | [Pływanie i tamowanie elementów](#)

Kolejnym zagadnieniem związanym z kaskadowymi arkuszami stylów jest pozycjonowanie elementów. Dzięki tej funkcji można tworzyć strony o wyszukanych kształtach, a wszystkie elementy dowolnie rozmieszczać w oknie przeglądarki przy wykorzystaniu warstw. Dla projektanta z wyobraźnią, który zna tę część specyfikacji stylów, tworzenie najwymyślniejszych stron nie powinno stanowić problemu.

Z pozycjonowaniem nierozdzielnie wiąże się możliwość nakładania na siebie elementów oraz określanie pozycji w stosie. **Pozycjonowanie elementów**

position

Dostępne wartości:

static | relative | absolute | fixed | inherit

Wartość domyślna:

static

Przykład:

```
p {position: absolute;}
```

Dziedziczenie:

Nie

Obsługa:

Opera, Mozilla Firefox, MS Internet Explorer (bez fixed), Chrome, Safari

Specyfikacja CSS rozróżnia następujące metody pozycjonowania:

-

domyślne — statyczne,

-

względne,

-

bezwzględne,

-

ustalone.

### Pozycjonowanie statyczne

Każdy element umieszczony na stronie ma nadane pozycjonowanie statyczne; jest to jego domyślne położenie. Dlatego też nie ma większego sensu nadawanie takiego formatowania elementom na stronie. Definicja pozycjonowania statycznego ma następującą postać:

P, H1

```
{  
  position:static;  
}
```

<<przykład>>

```
<div class="przyklad">
```

```
<div style="width:500px;height:500px;background-color:silver;">
```

```
<h1 style="position:static;">Nagłówek w pozycji domyślnej</h1>
```

```
<p style="position:static;">Przykładowy blok tekstu w pozycji domyślnej. Przykładowy blok tekstu w pozycji domyślnej. Przykładowy blok tekstu w pozycji domyślnej. Przykładowy blok tekstu w pozycji domyślnej. </p>
```

```
</div>
```

```
</div>
```

<</przykład>> **Pozycjonowanie bezwzględne**

Pozycjonowanie bezwzględne polega na określaniu dokładnego położenia elementu względem lewego górnego narożnika okna przeglądarki. Punkt ten przyjmuje współrzędne (0;0), a ustawianie elementów polega na wyznaczeniu w wirtualnym układzie współrzędnych pozycji lewego górnego rogu pozycjonowanego elementu.

Poniżej znajduje się przykład definiujący lokalizację bloku tekstu jako oddaloną o 80 pikseli od lewego górnego narożnika. Dodatkowo pozycjonowany jest nagłówek stopnia pierwszego, którego pozycja została ustawiona na 40 pikseli od górnej krawędzi okna przeglądarki.

P

```
{  
  position:absolute;  
  left:80px;  
  top:80px;  
}
```

H1

## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

```
{  
  position:absolute;  
  top:40px;  
}  
<<przykład>>  
<div class="przyklad">  
<div style="width:500px;height:500px;background-color:silver;">  
<h1 style="position:absolute;top:40px;">Nagłówek w pozycji domyślnej</h1>  
<p style="position:absolute;left:80px;top:80px;">Przykładowy blok tekstu w pozycji domyślnej.</p>  
</div>  
</div>  
<</przykład>>
```

### Pozycjonowanie względne

Punktem odniesienia do określania położenia względnego elementu jest jego domyślna lokalizacja na stronie. Pozycjonowanie względne jest powiązane bezpośrednio z konkretnym znacznikiem lub może zostać rozciągnięte za pomocą div.

W pierwszej kolejności ustawiłem pozycję absolutną dla nagłówka oraz przykładowego akapitu. Dla obu elementów pozycja jest identyczna, dlatego uprościłem sobie przypisanie stylu, wykorzystując znacznik div.

DIV

```
{  
  position:absolute;  
  top:1cm;  
  left:1cm;  
}  
<<przykład>>  
<div class="przyklad">  
<div style="width:500px;height:500px;background-color:silver;">  
<div style="position:absolute;top:1cm;left:1cm;border:solid 1px red;width:400px;height:400px">  
<h1>Nagłówek w pozycji domyślnej</h1>
```

## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

```
<p>Przykładowy blok tekstu w pozycji domyślnej.</p>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<</przykład>>
```

Teraz pora określić nową względną pozycję dla samego akapitu.

```
P
```

```
{
```

```
position:relative;
```

```
top:1cm;
```

```
left:1cm;
```

```
}
```

```
<<przykład>>
```

```
<div class="przykład">
```

```
<div style="width:500px;height:500px;background-color:silver;">
```

```
<div style="position:absolute;top:1cm;left:1cm;border:solid 1px red;width:400px;height:400px">
```

```
<h1>Nagłówek w pozycji domyślnej</h1>
```

```
<p style="position:relative;left:1cm;top:1cm;">Przykładowy blok tekstu w pozycji domyślnej.</p>
```

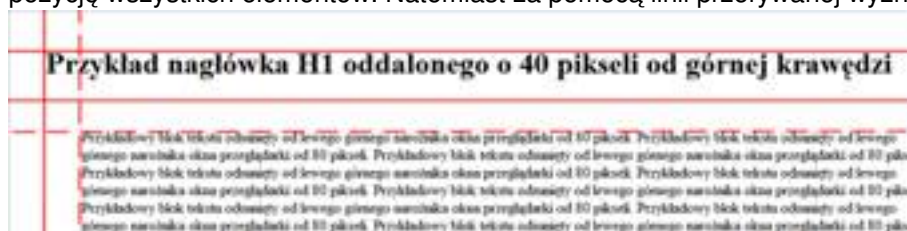
```
</div>
```

```
</div>
```

```
</div>
```

```
<</przykład>>
```

Na rysunku 10.1 widać różnicę pomiędzy pozycją bezwzględną i względną. Linie ciągłe wyznaczają bezwzględną pozycję wszystkich elementów. Natomiast za pomocą linii przerywanej wyznaczono względną pozycję akapitu.



## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

Rysunek 10.1. Schemat pozycji bezwzględnej i względnej

Należy zwrócić uwagę na to, że pozycja akapitu nie jest liczona od brzegu okna przeglądarki, lecz od domyślnego położenia tego elementu na stronie. W zaprezentowanym przykładzie akapit tekstu jest oddalony od krawędzi przeglądarki o sumę odległości przewidzianych dla definicji względnej i bezwzględnej. W tym konkretnym przykładzie jest to odległość dwóch centymetrów. **Pozycjonowanie stałe**

Pozycjonowanie stałe jest odmianą pozycjonowania bezwzględnego, jednak element, którego dotyczy, zawsze pozostaje w widocznej części okna przeglądarki. W praktyce wygląda to tak, jakby na stronie została zastosowana ramka. Poniżej zamieszczam przykład arkusza, który pozwala na zasymulowanie strony wykorzystującej ramki.

DIV.lewa

```
{  
position:fixed;  
top:10px;  
width:180px;  
height:100%;  
background-color:#C0FFC0;  
}
```

DIV.prawa

```
{  
position:absolute;  
top:10px;  
left:200px;  
width:500px;  
}
```

<<przykład>>

<html>

<head>

<meta http-equiv="Content-type" content="text/html;charset=ISO-8859-2">

<title>Tytuł Strony WWW</title>

<style type="text/css">

## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

DIV.lewa

```
{  
position:fixed;  
top:10px;  
width:180px;  
height:100%;  
background-color:#C0FFC0;  
}
```

DIV.prawa

```
{  
position:absolute;  
top:10px;  
left:200px;  
width:500px;  
}
```

</style>

</head>

<body>

<div class="lewa">

<ul>

<li><a href="http://onet.pl">Onet.pl</a></li>

<li><a href="http://republika.pl">Republika.pl</a></li>

</ul>

<p>

Lewa kolumna zawiera menu nawigacyjne i jest stale widoczna w oknie przeglądarki.

</p>

</div>

<div class="prawa">

## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

<h1>Przykład nagłówka H1</h1>

<p>Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli. Pozycja tekstu została ustalona w sposób bezwzględny. Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli.

</p>

<p>Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli. Pozycja tekstu została ustalona w sposób bezwzględny. Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli.

</p>

<p>Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli. Pozycja tekstu została ustalona w sposób bezwzględny. Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli.

</p>

<p>Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli. Pozycja tekstu została ustalona w sposób bezwzględny. Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli.

</p>

<p>Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli. Pozycja tekstu została ustalona w sposób bezwzględny. Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli.

</p>

<p>Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli. Pozycja tekstu została ustalona w sposób bezwzględny. Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli.

</p>

<p>Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli. Pozycja tekstu została ustalona w sposób bezwzględny. Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli.

</p>

<p>Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli. Pozycja tekstu została ustalona w sposób bezwzględny. Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli.

</p>

<p>Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli. Pozycja tekstu została ustalona w sposób bezwzględny. Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli.

</p>

## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

<p>Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli. Pozycja tekstu została ustalona w sposób bezwzględny. Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli.

</p>

<p>Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli. Pozycja tekstu została ustalona w sposób bezwzględny. Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli.

</p>

<p>Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli. Pozycja tekstu została ustalona w sposób bezwzględny. Przykładowy blok tekstu odsunięty od lewej krawędzi okna przeglądarki o 200 pikseli.

</p>

</div>

</body>

</html>

<</przykład>>

Pomimo wielu zalet pozycjonowanie stałe nie jest obsługiwane przez przeglądarkę MS Internet Explorer, która traktuje tę metodę tak samo jak pozycjonowanie bezwzględne. **Określanie pozycji**



## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

Opera, Mozilla Firefox, MS Internet Explorer, Chrome, Safari

Niezależnie od metody pozycjonowania elementów na stronie ich pozycja jest określana za pomocą poleceń left, right, top oraz bottom uzupełnionych o wartość liczbową lub procentową.

DIV

```
{  
  
position:absolute;  
  
left:5cm;  
  
top:10cm;  
  
}
```

<<przykład>>

```
<div class="przyklad">
```

```
<div style="width:500px;height:500px;background-color:silver;">
```

```
<div style="position:absolute;top:1cm;left:1cm;border:solid 1px red;width:400px;height:400px;">
```

```
<a href="http://helion.pl" style="position:absolute;left:5cm;top:5cm;">helion.pl</a>
```

```
</div>
```

```
</div>
```

```
</div>
```

<</przykład>> **Warstwy**

Odpowiednio użyte pozycjonowanie pozwala na nakładanie na siebie różnych elementów. W chwili gdy kilka elementów zostanie ustawionych na sobie, pojawia się pojęcie warstw. Rozwiązanie to pozwala na uzyskanie bardzo ciekawych efektów graficznych, np. ozdobnych cieniowanych nagłówek.

Na początek prosty przykład, w którym dwa identyczne wyrazy ustawiłem niemal dokładnie na sobie.

DIV

```
{  
  
font-size:60pt;  
  
font-weight:bold;  
  
font-family:verdana;  
  
}
```

## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

DIV.cien

```
{  
color:silver;  
  
position:absolute;  
  
top:30px;  
left:200px;  
}
```

DIV.tekst

```
{  
color:black;  
  
position:absolute;  
  
top:34px;  
left:204px;  
}
```

<<przykład>>

```
<div class="przyklad">
```

```
<div style="font-size:60pt;font-weight:bold;font-family:verdana;">
```

```
<div style="color:black;position:absolute;top:34px;left:204px;">helion.pl</div>
```

```
<div style="color:silver;position:absolute;top:30px;left:200px;">helion.pl</div>
```

```
</div>
```

```
</div>
```

<</przykład>>

Nakładać na siebie możemy dowolne elementy strony, np. nagłówki, akapity, tabele czy obrazki. **Kolejność nakładanych elementów**

z-index

Dostępne wartości:

auto | wartość liczbowa | inherit

## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

Wartość domyślna:

auto

Przykład:

```
p {z-index:2;}
```

Dziedziczenie:

Nie

Obsługa:

Opera, Mozilla Firefox, MS Internet Explorer, Chrome, Safari

Przy nakładaniu elementów przeglądarka domyślnie określa ich kolejność. Jest ona ustalana na podstawie występowania elementów w kodzie strony. Jeżeli zachowamy idealną kolejność, to nałożone elementy będą wyglądały tak, jak na jednym z poprzednich przykładów. Niestety, bardzo często bywa tak, iż elementy, które chcemy nałożyć na siebie, nie są umieszczone w kodzie XHTML/HTML w odpowiedniej kolejności i efekt końcowy bardzo różni się od zamierzonego. W takich sytuacjach musimy sięgnąć po kolejny element przewidziany w specyfikacji CSS, pozwalający sterować kolejnością wyświetlanych elementów.

Kontrola kolejności poszczególnych warstw jest możliwa za pomocą polecenia z-index uzupełnionego o parametr określający kolejność warstw w stosie. Zmodyfikowany kod arkusza dla prezentowanego przykładu powinien wyglądać następująco:

DIV

```
{  
font-size:60pt;  
font-weight:bold;  
font-family:verdana;  
}
```

DIV.cien

```
{  
color:silver;  
position:absolute;  
top:30px;  
left:200px;
```

## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

```
z-index:1;
```

```
}
```

DIV.tekst

```
{
```

```
color:black;
```

```
position:absolute;
```

```
top:34px;
```

```
left:204px;
```

```
z-index:2;
```

```
}
```

<<przykład>>

```
<div class="przyklad">
```

```
<div style="font-size:60pt;font-weight:bold;font-family:verdana;">
```

```
<div style="color:black;position:absolute;top:34px;left:204px;z-index:2;">helion.pl</div>
```

```
<div style="color:silver;position:absolute;top:30px;left:200px;z-index:1;">helion.pl</div>
```

```
</div>
```

```
</div>
```

<</przykład>> **Pływanie i tamowanie elementów**

### **Pływanie elementów**

float

Dostępne wartości:

left | right | none | inherit

Wartość domyślna:

none

Przykład:

```
img {float:right;}
```

## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

Dziedziczenie:

Nie

Obsługa:

Opera,  Mozilla Firefox, MS Internet Explorer, Chrome, Safari

Kaskadowe arkusze stylów pozwalają kontrolować oblewanie pozostałych elementów przez tekst. Ciekawostką jest to, że oblewać możemy praktycznie każdy element.

Do kontrolowania oblewania używamy polecenia float, któremu możemy przypisać jedną z trzech wartości: left, right lub none.

```

```

```

```

```

```

<<przykład>>

```
<div class="przyklad">
```

```
<div style="width:500px;">
```

```
<h3>Przykład: float:left</h3>
```

```

```

```
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean laoreet pharetra felis, quis aliquet odio auctor condimentum. Integer eleifend elit nec est dignissim nec gravida est varius. Morbi porttitor justo sit amet enim lacinia id accumsan ligula feugiat. Nam bibendum felis eu purus rutrum eleifend. Curabitur posuere nisl at tortor laoreet lobortis. Nulla facilisi. Aliquam erat volutpat. Curabitur eget lacus nunc. Integer tristique ullamcorper elit ac malesuada. Duis molestie magna fringilla lorem dictum vitae suscipit nibh aliquam. </p>
```

```
<h3>Przykład: float:right</h3>
```

```

```

```
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean laoreet pharetra felis, quis aliquet odio auctor condimentum. Integer eleifend elit nec est dignissim nec gravida est varius. Morbi porttitor justo sit amet enim lacinia id accumsan ligula feugiat. Nam bibendum felis eu purus rutrum eleifend. Curabitur posuere nisl at tortor laoreet lobortis. Nulla facilisi. Aliquam erat volutpat. Curabitur eget lacus nunc. Integer tristique ullamcorper elit ac malesuada. Duis molestie magna fringilla lorem dictum vitae suscipit nibh aliquam. </p>
```

```
<h3>Przykład: float:none</h3>
```

## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

```

```

```
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean laoreet pharetra felis, quis aliquet odio auctor condimentum. Integer eleifend elit nec est dignissim nec gravida est varius. Morbi porttitor justo sit amet enim lacinia id accumsan ligula feugiat. Nam bibendum felis eu purus rutrum eleifend. Curabitur posuere nisl at tortor laoreet lobortis. Nulla facilisi. Aliquam erat volutpat. Curabitur eget lacus nunc. Integer tristique ullamcorper elit ac malesuada. Duis molestie magna fringilla lorem dictum vitae suscipit nibh aliquam. </p>
```

```
</div>
```

```
</div>
```

```
<</przykład>>
```

Jeszcze jeden przykład pokazujący, że za pomocą funkcji float można oblewać również tekst.

P SPAN

```
{  
font-size:80pt;  
font-weight:bold;  
float:left;  
color:black;  
margin-top:-25px;  
margin-bottom:-25px;  
margin-right:10px;  
}
```

```
<<przykład>>
```

```
<div class="przykład">
```

```
<div style="width:500px;">
```

```
<p><span  
style="font-size:80pt;font-weight:bold;float:left;color:black;margin-top:-25px;margin-bottom:-25px;margin-right:10px  
;">L</span>orem ipsum dolor sit amet, consectetur adipiscing elit. Aenean laoreet pharetra felis, quis aliquet odio  
auctor condimentum. Integer eleifend elit nec est dignissim nec gravida est varius. Morbi porttitor justo sit amet  
enim lacinia id accumsan ligula feugiat. Nam bibendum felis eu purus rutrum eleifend. Curabitur posuere nisl at  
tortor laoreet lobortis. Nulla facilisi. Aliquam erat volutpat. Curabitur eget lacus nunc. Integer tristique ullamcorper  
elit ac malesuada. Duis molestie magna fringilla lorem dictum vitae suscipit nibh aliquam. </p>
```

```
</div>
```

```
</div>
```

```
<</przykład>> Tamowanie elementów
```

## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

clear

Dostępne wartości:

left | right | none | both | inherit

Wartość domyślna:

none

Przykład:

```
img {clear:right;}
```

Dziedziczenie:

Nie

Obsługa:

Opera, Mozilla Firefox, MS Internet Explorer, Chrome, Safari

Podczas tworzenia stron WWW czasem chcemy, by jakiś element nie był obłany. Wtedy z pomocą przychodzą nam polecenie clear i wartości left, right, both oraz none.

```
<h1 style="clear:left;">Lorem ipsum dolor sit amet, consectetur adipiscing elit</h1>
```

```
<h1 style="clear:right;">Lorem ipsum dolor sit amet, consectetur adipiscing elit</h1>
```

```
<h1 style="clear:both;">Lorem ipsum dolor sit amet, consectetur adipiscing elit</h1>
```

```
<h1 style="clear:none;">Lorem ipsum dolor sit amet, consectetur adipiscing elit</h1>
```

Dalsze rozważania na temat tamowania elementów przeprowadzę na poniższym przykładzie, w którym tabelę obłano blokiem tekstu i uzupełniono o nagłówek stopnia pierwszego.

<<przykład>>

```
<div class="przyklad">
```

```
<div style="width:500px;">
```

```
<table style="width:200px;height:400px;float:left;">
```

```
<tr>
```

## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

```
<td style="border:solid 1px black;">1</td><td style="border:solid 1px black;">2</td>
```

```
</tr>
```

```
<tr>
```

```
<td style="border:solid 1px black;">3</td><td style="border:solid 1px black;">4</td>
```

```
</tr>
```

```
</table>
```

```
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed blandit viverra massa sed tristique. Praesent feugiat vulputate odio, vitae faucibus nisl lacinia sit amet. Nulla cursus suscipit urna, sit amet luctus dui laoreet nec. Donec in eros lacus. Etiam blandit dictum massa a vulputate. Mauris tempus, felis nec gravida dictum, urna erat commodo purus, et aliquam risus velit id nibh. Nunc diam elit, pulvinar blandit pellentesque ac, convallis id sapien.</p>
```

```
<h1>Nagłówek H1, który ma znaleźć się pod tabelą</h1>
```

```
</div>
```

```
</div>
```

```
<</przykład>>
```

Korzystając z funkcji clear, można wymusić, aby nagłówek stopnia pierwszego znalazł się bezpośrednio pod tabelą i nie oblewał jej, jak to widać na przykładzie. Odpowiedni wpis do arkusza stylów ma postać:

```
H1
```

```
{
```

```
clear:left;
```

```
}
```

```
<<przykład>>
```

```
<div class="przyklad">
```

```
<div style="width:500px;">
```

```
<table style="width:200px;height:400px;float:left;">
```

```
<tr>
```

```
<td style="border:solid 1px black;">1</td><td style="border:solid 1px black;">2</td>
```

```
</tr>
```

```
<tr>
```



## Pozycjonowanie elementów

Dodał Administrator  
czwartek, 11 marzec 2010 05:36

---

```
<td style="border:solid 1px black;">3</td><td style="border:solid 1px black;">4</td>
```

```
</tr>
```

```
</table>
```

```
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed blandit viverra massa sed tristique. Praesent feugiat vulputate odio, vitae faucibus nisl lacinia sit amet. Nulla cursus suscipit urna, sit amet luctus dui laoreet nec. Donec in eros lacus. Etiam blandit dictum massa a vulputate. Mauris tempus, felis nec gravida dictum, urna erat commodo purus, et aliquam risus velit id nibh. Nunc diam elit, pulvinar blandit pellentesque ac, convallis id sapien.</p>
```

```
<h1 style="clear:left;">Nagłówek H1, który ma znaleźć się pod tabelą</h1>
```

```
</div>
```

```
</div>
```

```
<</przykład>>
```

Oczywiście w zależności od użytej wartości możemy tamować pływanie elementów z prawej, lewej lub z obu stron jednocześnie.