

Young Programmer

www.inceptio.org.pl



Dr inż. Małgorzata Janik, Zajęcia #2

Organizator:



Partnerzy:



Dofinansowanie:



MIĘDZYNARODOWY
FESTIWAL
WARSZAWA



Ramowy program warsztatów

Zajęcia 1: Zajęcia wprowadzające, HTML

Zajęcia 2: Style CSS

(tabele i kaskadowe arkusze stylów)

Zajęcia 3: Podstawy języka PHP

Zajęcia 4: Język PHP cd.

Zajęcia 5: Formularze HTML

Zajęcia 6: Synteza kursu

<http://www.if.pw.edu.pl/~majanik/wiki/index.php/HTML%2BPHP>

HTML+PHP #2

Tabele i Style CSS

Tabele

Tabele – podstawy

```
<table>
  <tr>
    <td>Pierwsza komórka</td>
    <td>Druga komórka</td>
  </tr>
  <tr>
    <td>Trzecia komórka</td>
    <td>Czwarta komórka</td>
  </tr>
</table>
```

<tr> - wiersze tabeli

<td> - komórki w wierszu „kolumny”

Pierwsza komórka Druga komórka

Trzecia komórka Czwarta komórka

Tabele – nagłówek

```
<table>
  <tr>
    <th>Kolumna #1</th>
    <th>Kolumna #2</th>
  </tr>

  <tr>
    <td>Pierwsza komórka</td>
    <td>Druga komórka</td>
  </tr>
  <tr>
    <td>Trzecia komórka</td>
    <td>Czwarta komórka</td>
  </tr>
</table>
```

Kolumna #1	Kolumna #2
Pierwsza komórka	Druga komórka
Trzecia komórka	Czwarta komórka

<th> - komórka nagłówka kolumny (pogrubiona i wyśrodkowana)

Tabele niesymetryczne

```
<table>
  <tr>
    <th>Kolumna #1</th>
    <th>Kolumna #2</th>
  </tr>

  <tr>
    <td colspan="2">Pierwsza komórka połączona</td>
    <!-- nie ma drugiej komórki -->
  </tr>
  <tr>
    <td>Trzecia komórka</td>
    <td>Czwarta komórka</td>
  </tr>
</table>
```

Kolumna #1	Kolumna #2
Pierwsza komórka połączona	
Trzecia komórka	Czwarta komórka

Analogicznie można łączyć komórki w kolumnach używając atrybutu *rowspan*.

Wygląd strony na podstawie tabeli



<http://www.if.pw.edu.pl/~majanik/data/test2.html>

Wygląd strony na podstawie tabeli

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
```

```
<html>  
<head>  
<meta charset="iso-8859-2" />  
<title>WEB PAGE TITLE GOES HERE</title>
```

```
</head>
```

```
<body style="margin: 0px; padding: 0px; font-family: 'Trebuchet MS',ve
```

```
<table width="100%" style="height: 100%;" cellpadding="10" cellspacing="0"  
border="0">
```

```
<tr>
```

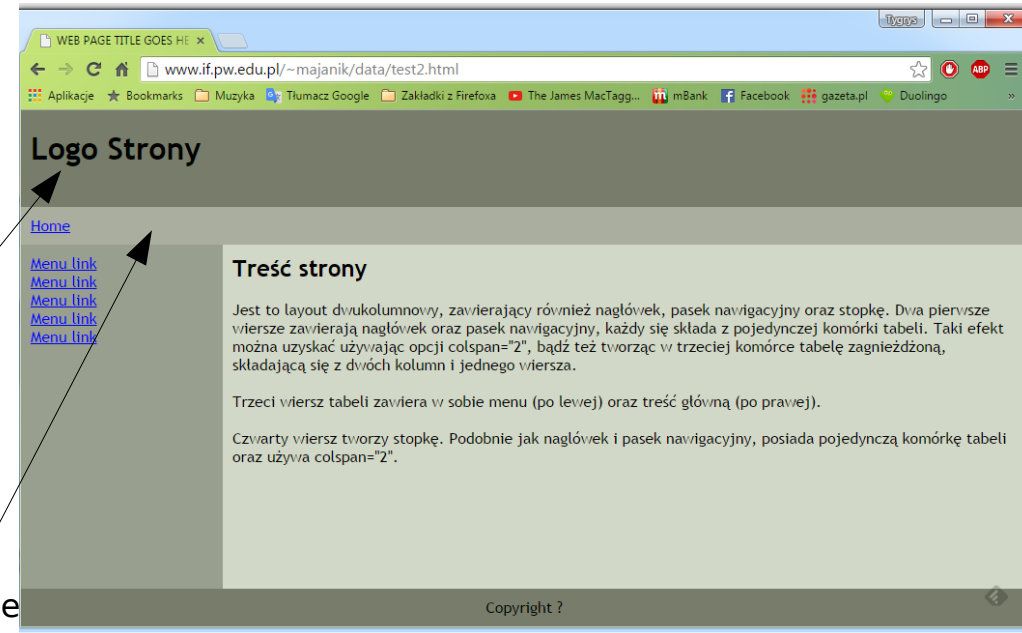
```
<!-- ===== NAGŁÓWEK ===== -->
```

```
<td colspan="2" style="height: 100px;" bgcolor="#777d6a"><h1>Logo  
Strony</h1></td></tr>
```

```
<!-- ===== PASEK NAWIGACYJNY ===== -->
```

```
<tr><td colspan="2" valign="middle" height="30" bgcolor="#a9ae9f"><a  
href="#">Home</a></td></tr>
```

```
<tr>
```



Wygląd strony na podstawie tabeli

```
<!-- ===== LEWA KOLUMNA (MENU) ===== -->  
<td width="20%" valign="top" bgcolor="#999f8e">  
<a href="#">Menu link</a><br>  
<a href="#">Menu link</a><br>  
<a href="#">Menu link</a><br>  
<a href="#">Menu link</a><br>  
<a href="#">Menu link</a>  
</td>
```

```
<!-- ===== PRAWA KOLUMNA (TREŚĆ) ===== -->  
<td width="80%" valign="top" bgcolor="#d2d8c7">  
  
<h2>Treść strony</h2>
```

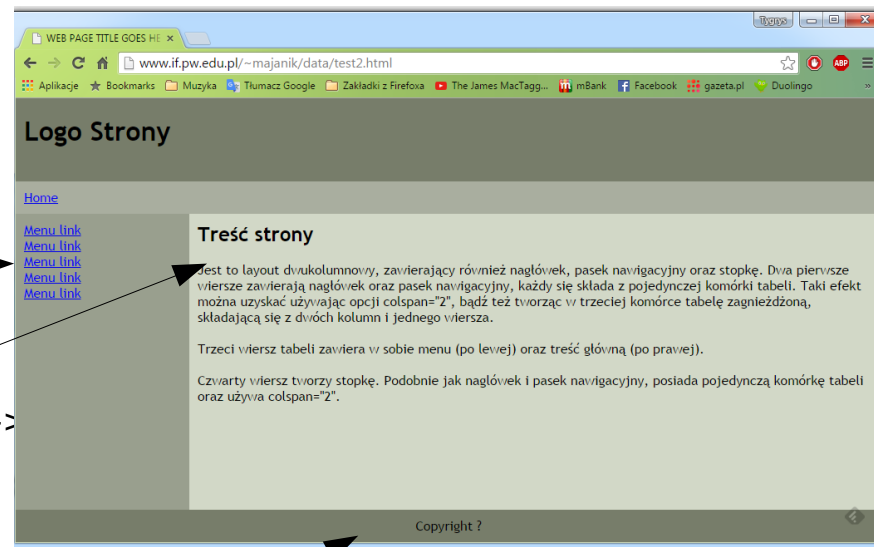
Jest to layout dwukolumnowy, zawierający również nagłówek, pasek nawigacyjny oraz stopkę. Dwa pierwsze wiersze zawierają nagłówek oraz pasek nawigacyjny, każdy się składa z pojedynczej komórki tabeli. Taki efekt można uzyskać używając opcji `colspan="2"`, bądź też tworząc w trzeciej komórce tabelę zagnieżdżoną, składającą się z dwóch kolumn i jednego wiersza.

```
<br>  
<br>
```

Trzeci wiersz tabeli zawiera w sobie menu (po lewej) oraz treść główną (po prawej).

Czwarty wiersz tworzy stopkę. Podobnie jak nagłówek i pasek nawigacyjny, posiada pojedynczą komórkę tabeli oraz używa `colspan="2"`.</td></tr>

```
<!-- ===== STOPKA ===== -->  
<tr><td colspan="2" align="center" height="20" bgcolor="#777d6a">Copyright ?</td></tr>  
</table>  
</body>
```



Znaki specjalne

** ** – znak odstępu
< – znak mniejszości „<”
> – znak większości „>”
© – znak praw autorskich ©
& – znak ampersandu (&)

HTML+PHP #2

Tabele i Style CSS

CSS - Kaskadowe Arkusze Stylów

Lokalne osadzanie stylów

(1) Osadzanie lokalne: doraźnie, dla pojedynczych elementów
(z poprzedniego wykładu):

```
<body style="background-color: green;">
<p>Witaj Czesławie!</p>
<p style="color: red; font-size: x-large;">Jak się masz?</p>
</body>
```

Osadzanie lokalne jest przydatne, gdy chcemy doraźnie ustawić styl pojedynczego elementu.

Korzystajmy z tej formy osadzania rzadko.



Przykładową paletę wartości kolorów można znaleźć tutaj:
http://www.w3schools.com/html/html_colors.asp

Zaś lista nazw kolorów obsługiwana przez CSS jest tutaj:
http://www.w3schools.com/cssref/css_colornames.asp

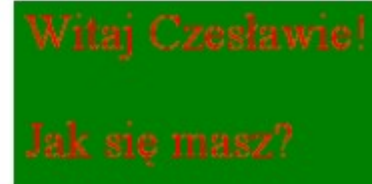
Formatowanie tekstu: style II

(2) Style zagnieżdżone: definiowane dla wszystkich znaczników w dokumencie.

```
<head>
  <!-- inne nagłówki sekcji head -->
  <style type="text/css">
    body {
      background-color: green;
    }
    p {
      color: red;
      font-size: x-large;
    }
  </style>
</head>
```

dla wszystkich znaczników <p> kolor jest czerwony a czcionka ma rozmiar x-large

```
<body>
<p>Witaj Czesławie!</p>
<p>Jak się masz?</p>
</body>
```



Witaj Czesławie!
Jak się masz?

UWAGA: możemy zawsze zastosować specjalny znacznik , który sam nic nie robi, ale możliwe jest przyporządkowanie mu konkretnego stylu.

Definicje

Poprawne nazewnictwo:

```
selektor{  
    właściwość1: wartość1;  
    właściwość2: wartość2;  
    ....  
}
```

```
selektor1, selektor2, selektor3{  
    właściwość1: wartość1;  
    właściwość2: wartość2;  
    ....  
}
```

Ciekawostki

Kilka selektorów:

```
selektor, selektor2, selektor3 {  
    właściwość1: wartość1;  
    właściwość2: wartość2;  
    ....  
}
```

```
h1, h2, h3 {  
    color: red;  
}
```

Selektor uniwersalny:

```
* {  
    color: red;  
}
```

Jeśli znaczniki mają wystąpić w konkretnej kolejności:

```
tr td ol {  
    color: red;  
}
```

Lista uporządkowana umieszczona w komórce w wierszu tabeli

```
tr * ol {  
    color: red;  
}
```


Zewnętrzne

Tylko po co na każdej podstronie pisać to samo...

```
<head>
  <!-- inne nagłówki sekcji head -->
  <link rel="stylesheet" href="plik.css" type="text/css">
</head>

<body>
<p>Witaj Czesławie!</p>
<p>Jak się masz?</p>
</body>
```

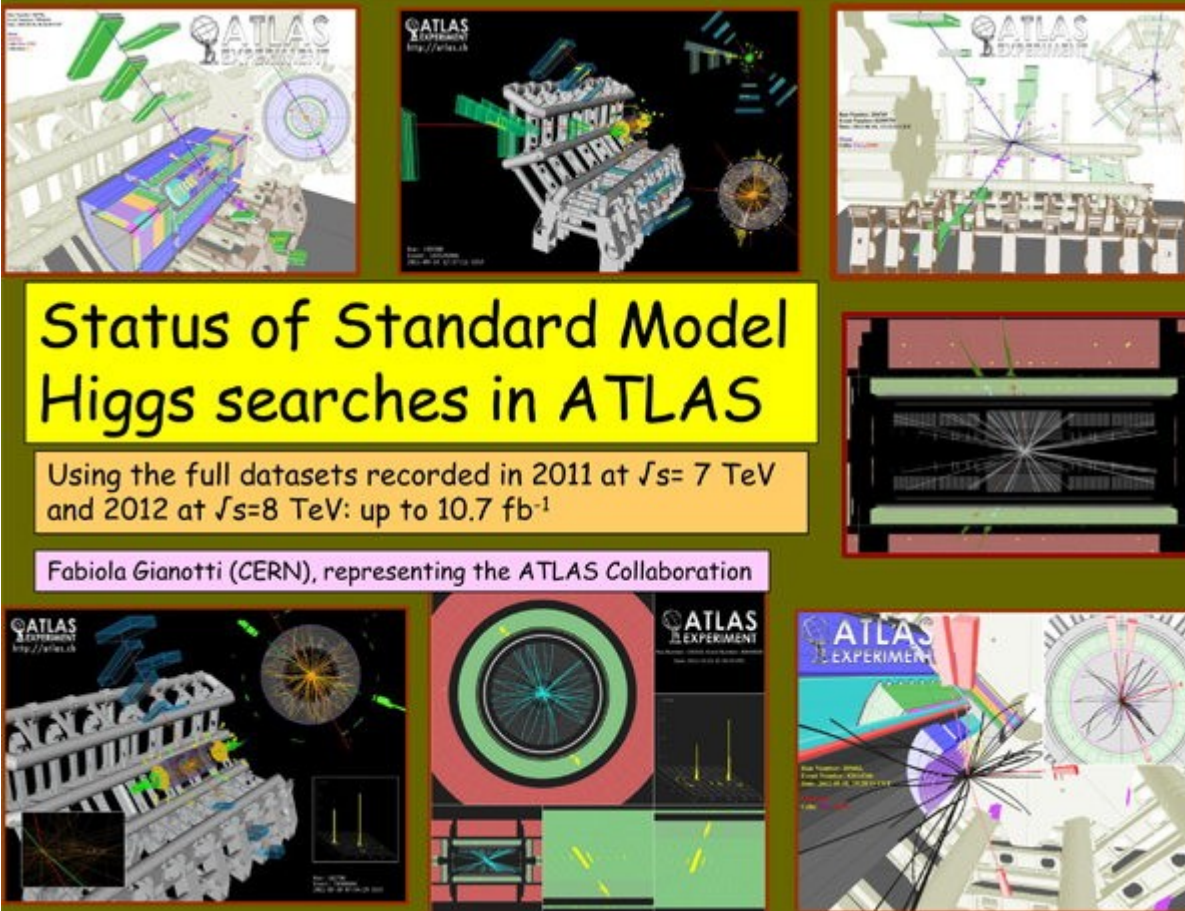
Witaj Czesławie!

Jak się masz?

```
plik.css
body {
  background-color: green;
}
p {
  color: red;
  font-size: x-large;
}
```

Osobny plik ze stylami pozwala linkować go z różnych podstron naszej witryny!

Anegdota – zmiana czcionki na całym portalu

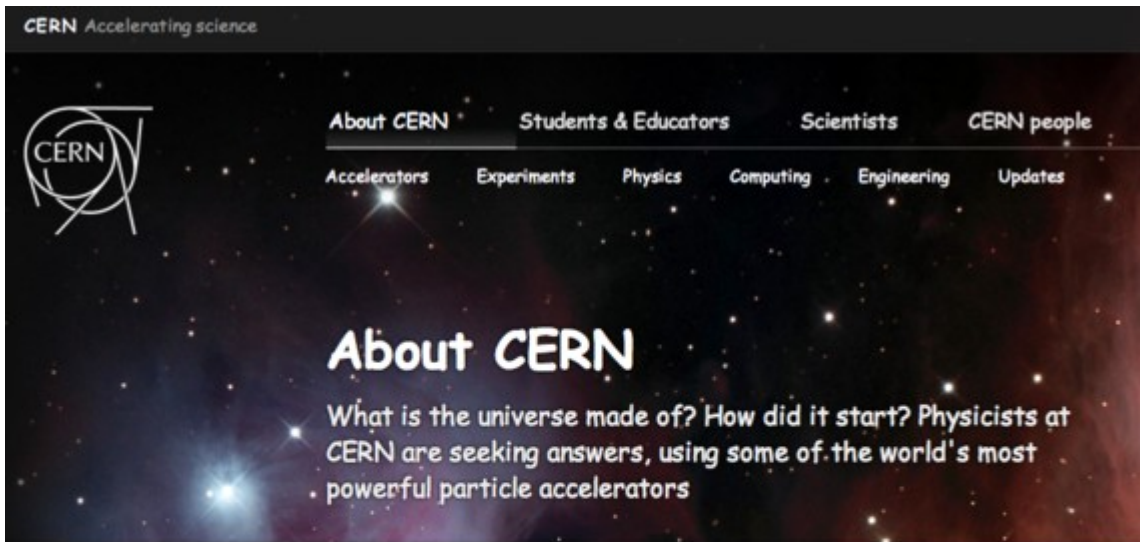


Status of Standard Model Higgs searches in ATLAS

Using the full datasets recorded in 2011 at $\sqrt{s}=7$ TeV and 2012 at $\sqrt{s}=8$ TeV: up to 10.7 fb^{-1}

Fabiola Gianotti (CERN), representing the ATLAS Collaboration

Anegdota – zmiana czcionki na całym portalu



At CERN, the European Organization for Nuclear Research, physicists and engineers are probing the fundamental structure of the universe. They use the world's largest and most complex scientific instruments to study the basic constituents of matter - the fundamental particles. The particles are made to collide together at close to the speed of light. The process gives the physicists clues about how the particles interact, and provides insights into the fundamental laws of nature.



Cian O'Luanaigh



Former ATLAS spokesperson Fabiola Gianotti announces the change to Comic Sans (Video: CERN)

From today, all of CERN's official communication channels are switching to exclusive use of the font Comic Sans. The move comes after weeks of deliberation by CERN management and top web designers about how best to update the image of the

Klasy

```
<head>
  <!-- inne nagłówki sekcji head -->
  <link rel="stylesheet" href="plik.css" type="text/css">
</head>

<body>
<p>Witaj Czesławie!</p>
<p class="duzy">Jak się masz?</p>
</body>
```

Witaj Czesławie!

Jak się masz?

plik.css

```
body {
  background-color: green;
}
p {
  color: red;
}
p.duzy {
  font-size: x-large;
}
```

Użycie *klas* pozwala na wyróżnienie pewnych klas obiektów, które będą miały identyczny styl.

Klasy

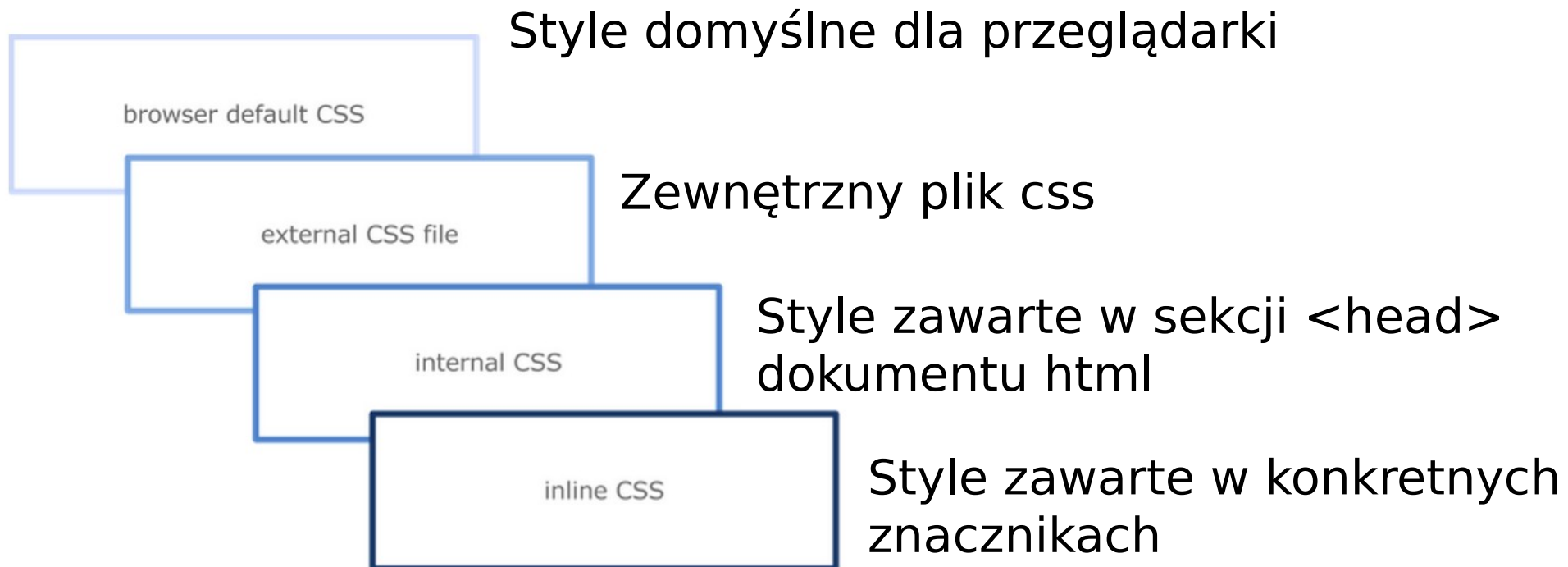
Można też zdefiniować klasę bez powiązania z konkretnym elementem:

```
.duzy {  
    font-size: x-large;  
}
```

A potem możemy stosować tą klasę dla różnych znaczników:

```
<p class="duzy"> Witaj Czesławie! </p>  
<a href="www.wp.pl" class="duzy">link </a>
```


Kaskada



Kaskada

Plik css:

```
p{  
    color: red;  
}  
  
p.duzy{  
    font-size: x-large;  
}
```

Sekcja <head>:

```
p{  
    color: green;  
    font-style: italic;  
}
```

Treść (<body>):

```
<p class="duzy" style="color: blue;">Napis</p>
```

Jaki będzie napis?

Dziedziczenie

Dziedziczenie:

Przejmowanie atrybutów od przodków. W CSS właściwości dotyczące *pierwszego planu* (bez właściwości tła) są dziedziczone.

Np. Wszystkie znaczniki w dokumencie są potomkami znacznika `<body>`, więc jeśli coś ustawimy dla `body` to wszystkie również przejmą te własności. Chyba, że dla niektórych **jawnie** zdefiniujemy coś inaczej.

Podobnie może działać znacznik `<div> </div>` wyszczególniający konkretny fragment dokumentu.

Ale tak samo jak zdefiniujemy tabelę przez `<table>` i ustawimy jej konkretny styl, to wszystkie elementy w środku (m.in. `td`, `tr`) również przejmą styl danej tabeli.

Ale może być jeszcze bardziej złożone...

Dziedziczenie

Plik css:

```
p{  
    border: thin solid black;    Cienka pełna kreska  
}
```

```
p.redbottom{  
    border-bottom: thin dotted red;    Cienka kropkowana  
                                        czerwona kreska  
}
```

Treść (<body>):

```
<p>Napis 1</p>  
<p class="redbottom">Napis 2</p>
```

Jak będą wyglądać napisy?

Dziedziczenie

Plik css:

```
p{  
    border: thin solid black;  
}
```

```
p.redbottom{  
    border-bottom: thin dotted red;  
}
```

Treść (<body>):

```
<p>Napis 1</p>  
<p class="redbottom">Napis 2</p>
```

Jak będą wyglądać napisy?

Napis 1

Napis 2

HTML+PHP #2

Tabele i Style CSS

Najważniejsze style

Style tekstu

Styl:	<i>color</i> – kolor tekstu
Wybrane wartości:	nazwa koloru (np. <i>red</i> , <i>green</i> , <i>navy</i>) zapis szesnastkowy (np. #80ff80) <i>rgb (100%,50%,25%)</i> – składowe koloru

<http://www.colorspire.com/rgb-color-wheel/>

Styl:	<i>font-size</i> – rozmiar czcionki
Wybrane wartości:	rozmiar w punktach (np. <i>16pt</i>) rozmiar w procentach (np. <i>150%</i>) <i>small</i> , <i>medium</i> , <i>large</i> , <i>x-large</i> , <i>xx-large</i>

Styl:	<i>text-align</i> – wyrównanie tekstu
Wybrane wartości:	<i>left</i> , <i>right</i> <i>center</i> <i>justify</i> – wyjustowanie

Style tekstu cd.

Styl:	<i>font-family</i> – krój czcionki
Wybrane wartości:	nazwa czcionki (np. Times, Arial, Georgia, Tahoma) <i>serif, sans-serif</i> – szeryfowa / bezszeryfowa <i>monospace</i> – o stałej szerokości
Styl:	<i>text-indent</i> – wcięcie pierwszej linijki akapitu
Wybrane wartości:	rozmiar w punktach (np. <i>16pt</i>) rozmiar w procentach (np. <i>5%</i>)
Styl:	<i>vertical-align</i> – wyrównanie tekstu w pionie
Wybrane wartości:	<i>baseline</i> – na dole <i>middle</i> – po środku <i>top</i> – na górze

Style tekstu cd.

Styl:	<i>font-weight</i> – grubość czcionki
Wybrane wartości:	<i>normal</i> <i>bold, bolder</i> – gruba, grubsza <i>lighter</i> – cieńsza
Styl:	<i>font-style</i> – styl czcionki
Wybrane wartości:	<i>italic</i> – kursywa <i>oblique</i> – chyba też pochylona
Styl:	<i>text-decoration</i> – dodatkowa dekoracja tekstu
Wybrane wartości:	<i>underline</i> – podkreślona <i>overline</i> – nadkreślona <i>line-through</i> – przekreślona

Tło

Styl:	<i>background</i> – kolor tła
Wybrane wartości:	<i>nazwa koloru (np. red, green, navy)</i> <i>zapis szesnastkowy (np. #80ff80)</i> <i>rgb (100%,50%,25%)</i> – składowe koloru
Styl:	<i>background-image</i> – obrazek tła
Wybrane wartości:	<i>url ("adres_obrazka.jpg")</i>
Styl:	<i>background-position</i> – położenie obrazka
Wybrane wartości:	<i>left/right/center top/center/bottom</i> np. <i>left center</i>

Tło cd.

Styl:	<i>background-repeat</i> – powtarzanie obrazka
Wybrane wartości:	<i>repeat</i> – powielaj w obu kierunkach (X i Y) <i>repeat-x, repeat-y</i> – powielaj tylko w kierunku X lub Y <i>no-repeat</i> – nie powielaj
Styl:	<i>background-attachment</i> – „przypięcie” obrazka
Wybrane wartości:	<i>scroll</i> – przewijaj obrazek razem z treściami <i>fixed</i> – przypnij obrazem na stałe na ekranie <i>local</i>

Marginesy i odstępy

Styl:	<i>margin</i> – wszystkie marginesy <i>margin-left, margin-right, margin-top, margin-bottom</i>
Wybrane wartości:	rozmiar w punktach (np. 16pt) rozmiar bezwzględny (np. 15mm, 1cm) rozmiar w procentach (np. 5%)

Styl:	<i>padding</i> – wszystkie odstępy (np. od brzegu komórki) <i>padding-left, padding-right, padding-top, padding-bottom</i>
Wybrane wartości:	rozmiar w punktach (np. 16pt) rozmiar bezwzględny (np. 15mm, 1cm) rozmiar w procentach (np. 5%)

Marginesy dobrze jest ustawić dla całego dokumentu w sekcji *body*.
Odstępy najczęściej przydają się w określaniu stylu komórki *td*.

Obramowanie

Styl:	<i>border-style</i> – styl obramowania ze wszystkich stron <i>border-left-style, border-right-style, border-top-style, border-bottom-style</i>
Wybrane wartości:	<i>none, solid</i> – brak / ciągła linia <i>dotted, dashed</i> – kropkowana / przerywane linie <i>dot-dash, wave, groove</i> – inne, bardziej udziwnione
Styl:	<i>border-width</i> – grubość obramowania <i>border-left-width, border-right-width, border-top-width, itd.</i>
Wybrane wartości:	<i>none, thin, medium, thick</i> – brak / cienka / średnia / gruba rozmiar bezwzględny (np. 1mm) rozmiar w pikselach (np. 10px)
Styl:	<i>border-color</i> – kolor obramowania <i>border-left-color, border-right-color, border-top-color, itd.</i>
Wybrane wartości:	nazwa lub wartość koloru

Rozmiar

Styl:	<i>width</i> – szerokość <i>height</i> – wysokość
Wybrane wartości:	rozmiar w punktach (np. 16pt) rozmiar bezwzględny (np. 15mm, 1cm) rozmiar w procentach (np. 5%)

Style tekstu – przykład

```
<style type="text/css">
  div.ramka
  {
    border-style: solid;
    border-color: red;
    border-width: 3px;
    height: 100px;
    width: 100px;
    background-color: gold;
    text-align: center;
    font-size: 95px;
    font-family: monospace;
    font-weight: bold;
  }
</style>

<div class="ramka">5</div>
```



Tabele – przykład

```
<style type="text/css">
  table {
    border-color: blue;
    border-width: 2px;
    border-style: solid;
  }
  td {
    border-color: red;
    border-width: 1px;
    border-style: solid;
    padding: 5px;
  }
  td:hover {
    background-color: gold;
  }
  th {
    border-color: red;
    border-width: 1px;
    border-style: solid;
    background-color: lime;
    padding: 5px;
  }
  tr.parzysty {
    background-color: #c0c0c0;
  }
  tr.nieparzysty {
    background-color: #e0e0e0;
  }
</style>
```

```
<table style="width: 500px">
  <tr>
    <th>Kolumna #1</th>
    <th>Kolumna #2</th>
  </tr>
  <tr class="parzysty">
    <td>abc</td>
    <td>def</td>
  </tr>
  <tr class="nieparzysty">
    <td>ghi</td>
    <td>jkl</td>
  </tr>
  <tr class="parzysty">
    <td>mno</td>
    <td>pqr</td>
  </tr>
</table>
```

Kolumna #1	Kolumna #2
abc	def
ghi	jkl
mno	pqr

Pseudoklasy – przykład #1

Pseudoklasy pozwalają na precyzyjne określenie wyglądu elementów w zależności od ich „stanu”, np.:

- `a:link` – nieodwiedzony link
- `a:visited` – odwiedzony link
- `a:hover` – link, na który został najechany kursor
- `input:hover` – pole formularza, na które został najechany kursor

```
a {
  color: blue;
  text-decoration: none;
}

a:hover {
  color: green;
  text-decoration: underline;
}
```

w3schools

Wyczerpujący kurs CSS z licznymi przykładami (po angielsku) można znaleźć tutaj:
<http://www.w3schools.com/css/>

Aby dowiedzieć się szczegółów o interesującym nas stylu, często wystarczy wpisać w Google: *css nazwa_stylu*

Young Programmer: HTML+PHP



Powodzenia!

Treść zadania do zrealizowania na pierwszych zajęciach znajduje się na stronie:

<http://www.if.pw.edu.pl/~majanik/wiki/index.php/HTML%2BPHP/Zadanie2>

Dr inż. Małgorzata Janik