



Technologie Internetu

Kaskadowe Arkusze Stylów (CSS)

Aleksander Denisiuk

`denisjuk@pja.edu.pl`

Polsko-Japońska Akademia Technik Komputerowych

Wydział Informatyki w Gdańsku

ul. Brzezi 55

80-045 Gdańsk

Kaskadowe Arkusze Stylów (CSS)

Najnowsza wersja tego dokumentu dostępna jest pod adresem <http://users.pja.edu.pl/~denisjuk/>

Warstwowa struktura dokumentu



- ⑥ Warstwa danych
 - △ Elementy logiczne
 - △ (X)HTML
- ⑥ Warstwa prezentacji
 - △ Sposób wizualizacji poszczególnych elementów
 - △ CSS
- ⑥ Warstwa zachowania
 - △ Interakcja z użytkownikiem
 - △ JavaScript
- ⑥ Zasada oddzielenia warstw

Kaskadowe arkusze stylów

- ⑥ CSS = Cascading Style Sheets
- ⑥ Oddzielenie warstwy prezentacji
- ⑥ CSS demo
- ⑥ W3C: *All browsers support CSS today*
- ⑥ Test Acid 2

Krótki kurs historii CSS

1996 CSS 1


1998 CSS 2.0

2000 CSS 2.1 — obecny standard

od 1999 CSS 3 — wersja rozwojowa

- większość przeglądarek obsługuje większość właściwości CSS, sprawdzać na [W3C](#)

Składnia CSS



```
h1, h2, h3 {text-decoration: underline;}
```

selektor *właściwość* *wartość*

deklaracja

- ⑥ selektor określa, do których elementów dokumentu stosuje się reguła
- ⑥ właściwość oznacza cechę składu dokumentu
- ⑥ wartość jest wielkością związaną z daną cechą

Składnia CSS

- ⑥ wielkość liter nie ma znaczenia
- ⑥ białe znaki są ignorowane
- ⑥ komentarze

```
h1 { text-decoration: underline; }
```

```
h2 {  
    text-transform: uppercase;  
    /* text-color: navy; */  
}
```

@-reguły

⑥ @charset

definiuje kodowanie znaków użyte w pliku ze stylem;
musi wystąpić na jego początku

⑥ @import

pozwała na dołączenie do stylu innego stylu; muszą
wystąpić na początku stylu

⑥ @media

pozwała na stworzenie różnych stylów dla różnych
typów mediów

⑥ @page

jest niezbyt dobrze obsługiwana przez przeglądarki

@-reguły. Przykłady

```
@charset "ISO-8859-2"; /* domyslnie UTF-8 */
@import url("/css/main.css");
@import "common.css";
@import url(/css/screen.css) screen,
    projection;
/* styl do wyswietlania */
@media screen, projection {
    html { background: #ffffef0; color: #300; }
}
/* styl do druku */
@media print {
    html { background: #fff; color: #000; }
}
```

Umieszczanie stylów w dokumencie

6 Styl zewnętrzny

```
<link rel="stylesheet" href="arkusz.css"
      type="text/css" />
<link rel="stylesheet" href="arkusz.css"
      media="print" />
```

6 Styl wewnętrzny

```
<style type="text/css">
  h1 { font-style: italic; }
</style>
```

6 Styl elementu

```
<h1 style="text-weight:bold;color:red">
  Czerwony </h1>
```

6 Decyduje kolejność stylów

Selektor znacznika

```
znacznik {prop: value;}
```

```
h1 {  
  text-align: center;  
}
```

```
<h1>Wycentrowany</h1>
```

Selektor identyfikatora

```
element#identyfikator {prop: value;}
```

```
h1#top {  
  border-bottom: dashed;  
}
```

```
<h1 id='top'>  
  <!-- ... -->  
</h1>
```

Selektor klasy

```
element.klasa {prop: value;}
```

```
span.potega {  
    vertical-align: super;  
    font-size: 80%;  
    font-style: italic;  
}
```

```
<!-- 210 -->  
2<span class='potega'>10</span>
```

Selektor ogólny



*

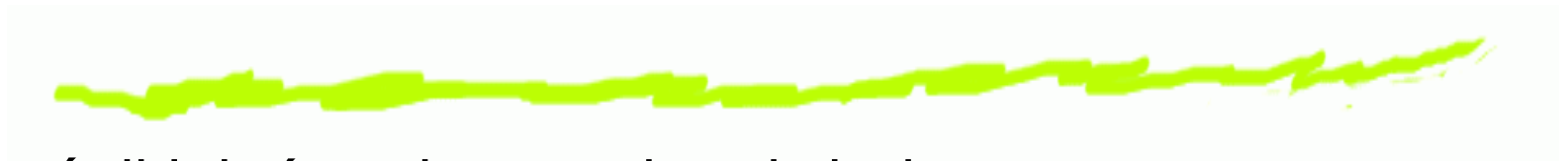
- 6 Wybiera wszystkie elementy.
- 6 Jest domyślnie dołączany do selektorów, które nie określają elementu.

* {margin: 2px}

* .error {color: red; }

.error {color: red; }

Pseudoklasy



link Odnośniki, które nie zostały odwiedzone.

visited Odnośniki, które zostały odwiedzone.

hover Pseudoklasa nadawana elementom, gdy znajduje się nad nimi wskaźnik myszy.

focus Elementy, które są zogniskowane

active Elementy, które są aktywne

first-child Element, który jest pierwszym dzieckiem rodzica.

nth-child(n) est dzieckiem numer n .

lang Określa specjalne zasady dla różnych języków.

not Określa elementy, które nie będą przetwarzane.

... inne

Selektor pseudoklasy

```
element:pseudoklasa {prop: value;}
```

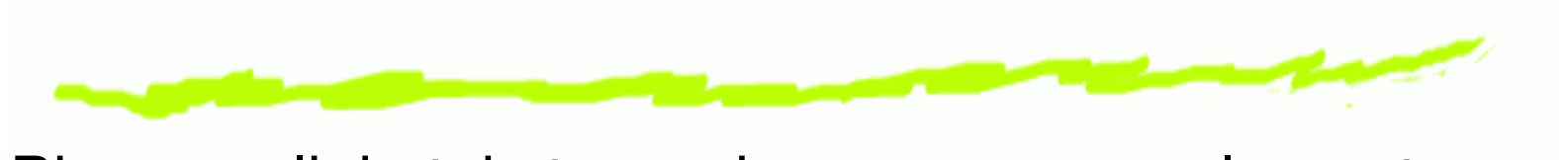
```
a:visited {  
    text-decoration:line-through;  
}
```

```
*:lang(pl) {  
    text-indent: 2em;  
}
```

```
:not(p) {  
    color: red;  
}
```

Kolejność deklaracji pseudoklas: link, visited, hover, focus, active

Pseudoelementy



first-line Pierwsza linia tekstu zawieranego przez element.

first-letter Pierwsza litera tekstu zawieranego przez element.

selection Tekst, zaznaczony kursorem

before Treść, poprzedzająca zawartość danego elementu.
Domyślnie jest pusta, ale może zostać uzupełniona za pomocą właściwości **content**

after Treść na końcu danego elementu.

Selektor pseudoelementów

```
element::pseudoelement {prop: value;}
```

```
p::selection{  
    background: khaki;  
    color: olive;  
}
```

```
span.issn::before {  
    content: "ISSN";  
}
```

```
span.pdf::after{  
    content: url(/ikona_pdf.png) ;  
}
```

Selektory atrybutu

- ⑥ element zawiera podany atrybut

```
element [atrybut] {prop: value; }
```

- ⑥ atrybut ma podaną wartość

```
element [atrybut = wartość] {prop: value; }
```

- ⑥ atrybut zawiera wartość na liście rozdzielonej spacjami

```
element [atrybut ~= wartość] {prop: value; }
```

- ⑥ atrybut zawiera wartość na liście rozdzielonej łącznikami

```
element [atrybut |= wartość] {prop: value; }
```

Selektory atrybutu

- 6 wartość atrybutu zaczyna się od `początek`
`element[attribut ^= początek]{prop: value;}`
- 6 wartość atrybutu kończy się na `koniec`
`element[attribut $= koniec]{prop: value;}`
- 6 wartość atrybutu zawiera `tekst`
`element[attribut *= tekst]{prop: value;}`

Selektory atrybutu. Przykłady

```
p[title] { font-style:italic; }  
img[src="logo.png"] { border-style:none; }  
p[class~=sub] { vertical-align:sub; }  
[lang|=en] { display:none; }
```

```
<p class="math sub">
```

```
<p lang="en-US">
```

Kombinator potomka

```
selector1 selector2 {prop: value;}
```

Zaznacza elementy pasujące do `selector2`, które są potomnymi (w sensie drzewa HTML) elementu, pasującego do `selector1`

```
#menu li:hover {background-color: #BB0; }
```

```
<div id="menu">  
  <ul>  
    <li><a href='..'>Back</a></li>  
  </ul>  
</div>
```

Kombinator dziecka

```
selector1 > selector2 {prop: value;}
```

Zaznacza elementy pasujące do `selector2`, które są dziećmi (w sensie drzewa HTML) elementu, pasującego do `selector1`

```
#menu > img {border-style:none; }
```

```
<div id="menu">  
  <ul><li>  
    <a href='..'>  
      <img src='buttons/back.png' alt='back' />  
    </a>  
  </li></ul>  
  <img src='obrazki/logo.png' alt='logo' />  
</div>
```

Kombinator siostrzanego elementu

```
selector1 + selector2 {prop: value;}
```

Zaznacza elementy pasujące do `selector2`, które są następnymi siostrzanymi (w sensie drzewa HTML) elementu, pasującego do `selector1`, nie licząc tekstu oraz komentarzy

```
h1 + h2 { margin-top: -5mm }
```

```
<h1>Technologie Internetu</h1>
```

```
<h2>Podstawy HTML</h2>
```


Kombinator siostrzanego elementu

```
selector1 ~ selector2 {prop: value;}
```

Zaznacza elementy pasujące do `selector2`, które są poprzedzonymi siostrzanymi (w sensie drzewa HTML) elementu, pasującego do `selector1`

```
h2 ~ h1 { margin-bottom: -5px }
```

```
<h1>Technologie Internetu</h1>
```

```
<h2>Podstawy HTML</h2>
```

Kilka kombinatorów

- ⑥ Z prawej w lewo
`div ol>li p`

Łączenie i grupowanie selektorów

- ⑥ Jedna reguła może mieć wiele selektorów

```
h1, h2, h3, h4 { font-weight: bold; }
```

- ⑥ Proste selektory można ze sobą łączyć

```
element.klasa1.klasa2#id:hover::first-letter
```

- ⑥ W poprzednim przykładzie nie można użyć spacji, ani złamać linii

Właściwości

- 6 Nazwy właściwości są jednym słowem lub kilkoma słowami połączonymi łącznikiem.
- 6 Wiele właściwości ma skrócone wersje


```
{border-color: red green blue yellow; }  
/* Wersja skrócona */
```

```
{border-top-color: red; }  
{border-right-color: green; }  
{border-bottom-color: blue; }  
{border-left-color: yellow; }
```

Rozszerzenia

- ⑥ Przeglądarki dodają własne rozszerzenia do właściwości css
- ⑥ Nowe rozszerzenia mają prefiks
 - ms- Microsoft
 - mso- Microsoft Office
 - moz- Mozilla Foundation (Gecko-based browsers)
 - o- Opera Software
 - atsc- Advanced Television Standards Committee
 - wap- The WAP Forum
 - webkit- Safari (WebKit-based browsers)
 - khtml- Konqueror

Rozszerzenia. Przykład



```
.animated {  
  -webkit-transition: background-color 2s;  
  -ms-transition: background-color 2s;  
  -o-transition: background-color 2s;  
  -moz-transition: background-color 2s;  
  transition: background-color 2s;  
  
  border: 1px solid black;  
}
```

Wartości. Jednostki metryczne



⑥ cm

⑥ mm

⑥ in

⑥ pt

⑥ pc

△ $1in = 2,54cm = 72,27pt$

△ $1pc = 12pt$

1.2em 33.33% 120px 4ex 10pt

Wartości. Jednostki relatywne

- ⑥ piksele `px`
- ⑥ procenty `%`
- ⑥ `em` równe szerokości litery M (`font-size`)
- ⑥ `rem` 1em elementu korzeniowego
- ⑥ `ex` równe wysokości litery X
- ⑥ `ch` szerokość cyfry 0 (zero)
- ⑥ `vw` 1% szerokości okna
- ⑥ `vh` 1% wysokości okna
- ⑥ `vmin` 1% mniejszego wymiaru okna
- ⑥ `vmax` 1% większego wymiaru okna

Wartości. Kolory



⑥ nazwy

△ red, orange, yellow, olive, purple, lime, green, blue, teal, black, gray i inne (razem 147 nazw)

⑥ postać szesnastkowa

△ #RrGgBb, #RGB(=#RRGGBB)

⑥ postać rgb(r, g, b)

⑥ postać rgba(r, g, b, a) (IE9+).

Wartości. Kolory, cd

- 6 postać `hsl(h, s, l)` (IE9+)
- 6 postać `hsla(h, s, l, a)` (IE9+)
- 6 `transparent`

Kolory. Przykłady



```
#0f0;
```

```
#00FF00;
```

```
rgb(0, 255, 0);
```

```
rgb(0, 100%, 0);
```

```
rgba(0, 100%, 0, 1);
```

```
/* Jaskrawy zielony zapisany na  
różne sposoby*/
```

```
transparent;
```

```
rgba(0, 0, 0, 0);
```

```
/* Brak wypełnienia*/
```

Właściwości. URL



```
url (obrazek.png)  
url ("http://example.pl/obrazek.png")  
url ('../arkusz.css')
```

Inherit

inherit

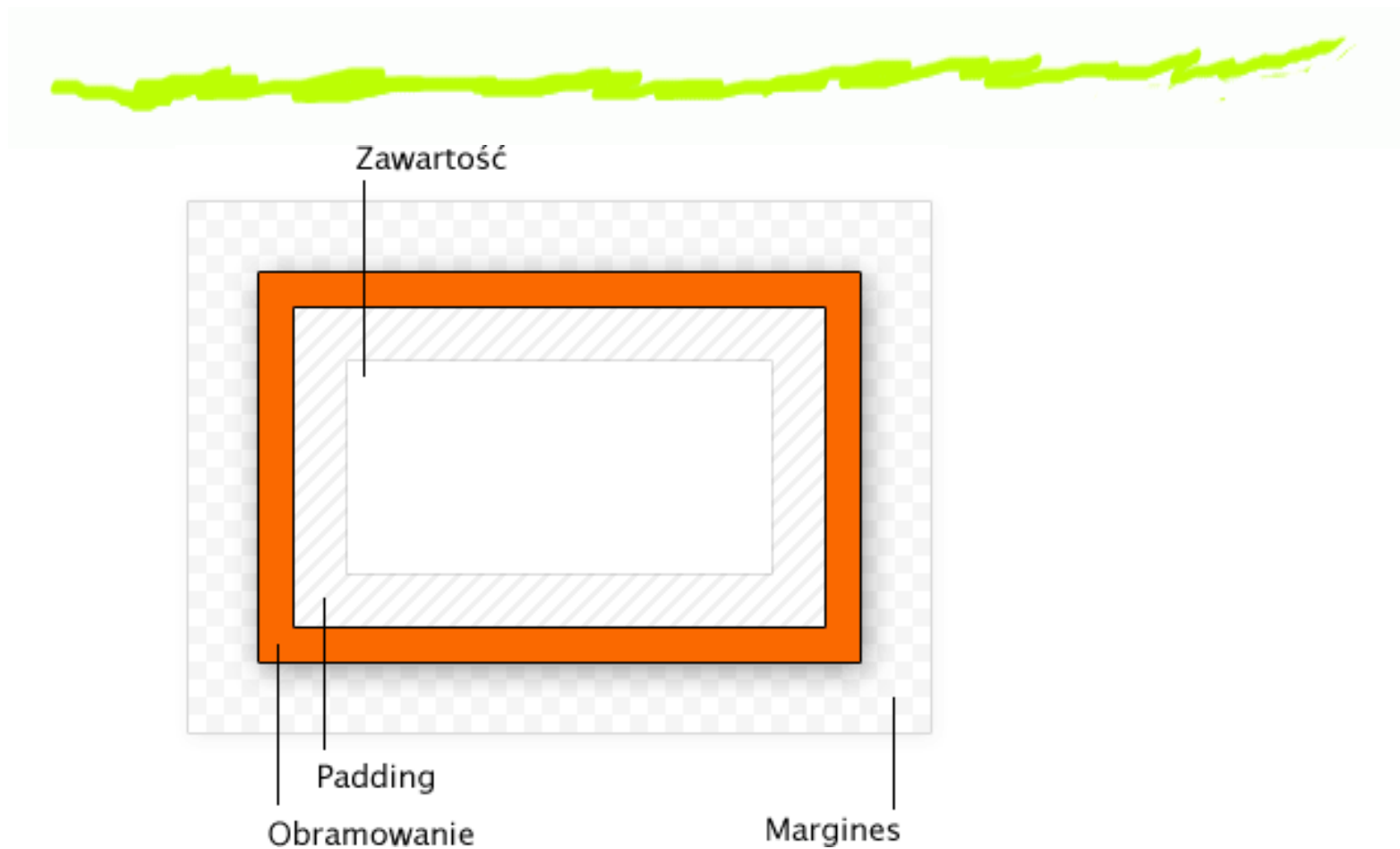
Kaskada

- ⑥ Jeśli kilka reguł pasuje do danego elementu w dokumencie, to nałożą się wg ich ważności.
- ⑥ Właściwości z najważniejszych reguł przykryją właściwości z mniej ważnych.
- ⑥ Ważność:
 1. Reguły oznaczone jako `!important` mają pierwszeństwo.
 2. Im bardziej precyzyjny selektor, tym ważniejszy.
 3. Później dołączone arkusze są ważniejsze
 4. Późniejsze reguły nadpisują wcześniejsze

Dziedziczenie

- ⑥ Elementy otrzymują swoje właściwości przez dziedziczenie, czyli przekazanie właściwości z elementu-rodzica do elementu-dziecka.
- ⑥ Nie wszystkie właściwości są domyślnie dziedziczone. Np. krój i wielkość pisma jest, ale tło i obramowanie — nie.
- ⑥ Wymusić dziedziczenie można podając `inherit` jako wartość.

Box model



Typy boksów

- ⑥ blokowe
- ⑥ liniowe
- ⑥ tabele

Wymiary boków



- ⑥ width, height
 - △ wartości absolutne, procentowe
 - △ auto
- ⑥ wymiary nie uwzględniają padding, border, margin

Minimalne i maksymalne wymiary



- ⑥ `min-width`, `max-width`, `min-height` i `max-height`
- ⑥ `min-width/min-height` ma pierwszeństwo nad `max-width/max-height`
- ⑥ przykład:
`width: 50%;`
`max-width: 600px;`
`min-width: 20ex;`

Przepełnienie boksów



- ⑥ overflow, overflow-x, overflow-y
 - △ visible
 - △ hidden
 - △ scroll
 - △ auto

Odstępy



- ⑥ padding, margin
 - △ rozmiar
 - △ `margin:auto` (centrowanie)

Obramowanie



6 border

△ rozmiar, styl, kolor, obrazek, `border-radius`



- ⑥ background
 - △ rozmiar, kolor, położenie, obrazek

Właściwość `white-space`



⑥ `white-space: normal` — zwykły sposób

⑥ **`white-space: pre`**

zachowanie białych znaków oraz łamanie linii

⑥ **`white-space: nowrap`**

zachowanie białych znaków, nie łamie linii

⑥ **`white-space: pre-wrap`**

jak `pre`, łamie linię, jeżeli tekst nie mieści się w kontenerze

⑥ **`white-space: pre-line`**

łamie linię, jeżeli tekst nie mieści się w kontenerze, spacje w trybie zwykłym

Liczniki

- ⑥ Dwa działania:
 - △ inicjalizacja `counter-reset`
 - △ incrementacja `counter-increment`
- ⑥ wyświetlenie za pomocą `:before` i `:after`

Liczniki. Przykład

```
li {list-style:none;}  
ol { counter-reset: list 12; }  
li:before {  
    counter-increment: list -2;  
    content: counter(list) ". ";  
}
```

- ⑥ Od dziesięć w dół
- ⑥ [Zobacz](#)

Style tekstu. Kolor



- ⑥ **color**

- ⑥ przykład:

```
h1 { color: red }
```

- ⑥ ustala także domyślny kolor dla podkreślenia i obramowania elementu

Style tekstu. Krój pisma



⑥ **font-family**

⑥ przykład:

```
font-family: "DejaVu Sans", Verdana,  
            Helvetica, sans-serif;
```

⑥ na końcu typ kroju:

```
sans-serif, serif, monospace,  
fantasy, cursive
```

Style tekstu. Wielkość pisma



- ⑥ **font-size**
- ⑥ jednostki absolutne, względne, `small`, `normal`, `medium`, etc
- ⑥ przykład:
font-size: 200%;
- ⑥ dobra praktyka: ustalać rozmiar w elemencie **html**,
a w pozostałych elementach ustalać względny rozmiar

Style tekstu. Grubość pisma



⑥ **font-weight**

⑥ jednostki: `normal, bold, bolder, lighter, 100, ..., 400, ..., 700, ..., 900`

⑥ przykład:

`font-weight: bold;`

Style tekstu. Pochylenie



- ⑥ `font-style`
- ⑥ normal, *italic*, *oblique*
- ⑥ przykład:
`font-style: italic;`

Transformacja tekstu



⑥ **text-transform**

⑥ normal, uppercase, lowercase, capitalize

⑥ przykład:

```
h1{text-transform: capitalize}
```


Odstęp między literami



⑥ letter-spacing

⑥ przykład:

```
.waski { letter-spacing: -0.3em }  
.szerszy { letter-spacing: 0.3em }  
.normalny { letter-spacing: -0em }
```

Odstęp między słowami



- ⑥ **word-spacing**

- ⑥ dozwolone są wartości ujemne

- ⑥ przykład:

```
.waski {word-spacing: 75%}
```

```
.szerszy {word-spacing: 120%}
```

```
.normalny {word-spacing: 100%}
```

Wysokość linii



⑥ line-spacing

⑥ przykład:

```
.waski{line-spacing: 8pt}  
.szerszy{line-spacing: 1.25}  
.normalny{line-spacing: 100%}
```

Skrócony zapis



- ⑥ font: styl variant grubość
wielkość/wysokość-linii krój-pisma;

Dekoracja tekstu



⑥ **text-decoration**

- △ none
- △ ~~line-through~~
- △ overline
- △ underline

Położenie tekstu wewnątrz elementu



⑥ `text-align`

- △ `left`, `right`, `center`, `justify`
 - `justify` lepiej nie używać bez przenoszenia wyrazów

Położenie w linii tekstu



⑥ **vertical-align**

- △ długość (w tym procent, m. b. ujemna), `baseline`, `sub`, `super`, `middle`, `top`, `bottom`, `text-top`, `text-bottom`

Wcięcie akapitowe



⑥ **text-indent**

- △ długość (w tym procent, m. b. ujemna)

Cień



⑥ text-shadow

- △ przesunięcie poziome, pionowe, stopień rozmycia (opcjonalne), kolor
- △ przykład:

```
text-shadow: 0 0 4px white,  
            0 -5px 4px #ff3,  
            2px -10px 6px #fd3,  
            -2px -15px 11px #f80;
```

⑥ IE9+



- ⑥ `list-style`
 - △ typ, położenie, obrazek

Właściwości słuchowe. Przykład

```
h1, h2, h3, h4 {  
  voice-family: male;  
  richness: 80;  
  cue-before: url("beep.au")  
}
```