



Szybki start ▶

Jason Cranford Teague

CSS3

**Bądź na bieżąco
ze światowymi trendami!**

- Co to są kaskadowe arkusze stylów?
- Jakie nowości kryją CSS3 i HTML5?
- Jak selektywnie formatować elementy?

Wydanie V

<http://helion.pl/kaszytypodreczniki-css3-wic-meyer-od-tworzenia-nowoczesnych-ukladow- stron-www-smashing-magazine-eric-a-meyer-flocca.htm>
<http://helion.pl/kaszytypodreczniki-css3-wic-meyer-od-tworzenia-nowoczesnych-ukladow- stron-www-smashing-magazine-eric-a-meyer-flocca.htm>
<http://helion.pl/kaszytypodreczniki-css3-wic-meyer-od-tworzenia-nowoczesnych-ukladow- stron-www-smashing-magazine-eric-a-meyer-flocca.htm>



Helion

» Idź do

- Spis treści
- Przykładowy rozdział
- Skorowidz

» Katalog książek

- Katalog online
- Zamów drukowany katalog

» Twój koszyk

- Dodaj do koszyka

» Cennik i informacje

- Zamów informacje o nowościach
- Zamów cennik

» Czytelnia

- Fragmenty książek online

» Kontakt

Helion SA
ul. Kościuszki 1c
44-100 Gliwice
tel. 32 230 98 63
e-mail: helion@helion.pl
© Helion 1991–2011

CSS3. Szybki start. Wydanie V

Autor: Jason Cranford Teague
Tłumaczenie: Łukasz Piwko, Maria Chaniewska
ISBN: 978-83-246-3386-9
Tytuł oryginału: [Visual QuickStart Guide \(5th Edition\)](#)
Format: 170×230, stron: 460



Bądź na bieżąco ze światowymi trendami!

- Co to są kaskadowe arkusze stylów?
- Jakie nowości kryją CSS3 i HTML5?
- Jak selektywnie formatować elementy?

Gdy webmasterzy wspominają czasy przed powstaniem kaskadowych arkuszy stylów, dostają gęsiej skórki. Struktura dokumentu przemieszana z opisem wyglądu oraz dziesiątki niestandardowych znaczników inaczej obsługiwanych przez każdą przeglądarkę to koszmar każdego ówczesnego projektanta stron internetowych. Idea stylów CSS okazała się wybawieniem – okazało się nagle, że proste rozwiązania większości problemów związanych z opisem wyglądu strony WWW są na wyciągnięcie ręki!

Najnowsza wersja CSS wprowadza wiele udogodnień, ułatwień i nowych możliwości. Wszystkie potrzebne informacje o wersji CSS3 znajdziesz właśnie w tej książce. W trakcie lektury poznasz historię kaskadowych arkuszy stylów, typy reguł oraz zasady ich tworzenia. Ponadto odkryjesz nowe możliwości języka HTML5 i przekonasz się, że ich wykorzystanie nie musi być trudne! Autor szczegółowo omawia sposoby selektywnej stylizacji, formatowania tekstów, wybierania kolorów oraz dostosowywania wyglądu tabel. Twoją szczególną ciekawość na pewno wzbudzi rozdział przedstawiający trzydzieści dwie najlepsze praktyki CSS. Dzięki nim Twoje arkusze będą jeszcze lepsze. Ta książka musi znaleźć się na półce każdego webmastera chcącego podążać za światowymi trendami!

- Historia CSS i budowa reguł CSS
- Nowości w CSS3 i format SVG
- Nowości w HTML5
- Selektory i importowanie arkusza stylów
- Definiowanie klas i selektywna stylizacja
- Formatowanie potomków
- Przygotowywanie stylów dla różnych mediów
- Typografia w sieci
- Bezpieczne czcionki dla stron WWW
- Własności tekstu i definiowanie kolorów
- Formatowanie list i tabel
- Marginesy, zaokrąglone rogi, elementy pływające
- Ustawianie przezroczystości elementów
- Przekształcenia i efekty przejścia
- Zerowanie ustawień CSS i zarządzanie arkuszami
- Najlepsze praktyki

Poznaj i wykorzystaj możliwości nowych standardów CSS3!

Spis treści

	Wprowadzenie	9
Rozdział 1.	Wprowadzenie do CSS 3	17
	Co to jest styl	18
	Co to są kaskadowe arkusze stylów (CSS)	19
	Rozwój CSS	22
	CSS i HTML	24
	Typy reguł CSS	25
	Budowa reguły CSS	27
	Nowości w CSS 3	29
Rozdział 2.	Wprowadzenie do HTML 5	31
	Co to jest HTML	32
	Typy elementów HTML	35
	Rozwój HTML 5	38
	Co nowego w HTML 5?	41
	Jak działa struktura HTML 5	42
	Używanie struktury HTML 5 obecnie	43
	Podsumowanie	48
Rozdział 3.	Podstawy CSS	49
	Podstawowe selektory	50
	Style śródliniowe, czyli formatowanie pojedynczych elementów	51
	Style osadzone, czyli formatowanie całych stron internetowych	54
	Style zewnętrzne, czyli formatowanie całych witryn internetowych	57
	Redefiniowanie elementów HTML	64
	Definiowanie klas	67
	Definiowanie identyfikatorów	71
	Definiowanie stylów uniwersalnych	75
	Grupowanie, czyli formatowanie wielu elementów za pomocą tych samych reguł CSS	78
	Komentarze w CSS	82
	Podsumowanie	84
Rozdział 4.	Selektywna stylizacja	85
	Drzewo rodzinne elementów	86
	Definiowanie stylów zależnie od kontekstu	87
	Pseudoklasy	96

	Pseudoelementy	108
	Definiowanie stylów dla elementów w zależności od atrybutów	112
	Nowość w CSS 3: zapytania o media ★	116
	Dziedziczenie	125
	Podnoszenie rangi deklaracji	127
	Kaskada	129
	Podsumowanie	132
Rozdział 5.	Własności pisma	133
	Typografia w sieci WWW	135
	Stos fontów	140
	Szukanie fontów	142
	Ustawianie rozmiaru pisma	149
	Nowość w CSS 3: dostosowywanie rozmiaru fontów alternatywnych ★	152
	Kursywa	155
	Ustawianie grubości pisma	158
	Kapitaliki	160
	Zbiorcze definiowanie własności pisma	162
	Podsumowanie	166
Rozdział 6.	Własności tekstu	167
	Ustawienia odstępów	169
	Wielkie i małe litery	174
	Nowość w CSS 3: cienie tekstu ★	176
	Wyrównywanie tekstu w poziomie	178
	Wyrównywanie tekstu w pionie	180
	Wcięcia akapitów	183
	Kontrolowanie odstępów	185
	Zdobienie tekstu	188
	Wkrótce!	191
	Podsumowanie	193
Rozdział 7.	Kolory i tła	195
	Definiowanie kolorów	197
	Nowość w CSS 3: gradienty w tle ★	203
	Wybór palety kolorów	207
	Ustawianie koloru tekstu	212
	Ustawianie koloru tła	214
	Ustawianie obrazu tła	216
	Zbiorcze definiowanie własności tła	224
	Podsumowanie	228

Rozdział 8.	Własności list i tabel	229
	Ustawianie punktatorów	232
	Tworzenie własnych punktatorów	233
	Korygowanie położenia punktatorów	234
	Zbiorcze definiowanie własności list	235
	Własności układu tabeli	236
	Ustawianie odstępów między komórkami	238
	Scalanie obramowań między komórkami	239
	Postępowanie z pustymi komórkami	241
	Ustawianie położenia podpisu tabeli	242
	Podsumowanie	243
Rozdział 9.	Własności interfejsu użytkownika i treści generowanej	245
	Zmiana wyglądu kursora	248
	Dodawanie treści za pomocą CSS	250
	Uczenie przeglądarki liczenia	252
	Określanie rodzaju cudzysłówów	254
	Podsumowanie	256
Rozdział 10.	Własności pól elementów	257
	Pole elementu	261
	Wyświetlanie elementów	264
	Ustawianie szerokości i wysokości elementów	267
	Postępowanie w przypadku przepełnienia elementów	270
	Elementy pływające	273
	Ustawianie marginesów	276
	Ustawianie obrysu	279
	Ustawianie obramowania	281
	Nowość w CSS 3: zaokrąglanie rogów ★	284
	Nowość w CSS 3: obrazy w tle obramowania elementów ★	287
	Ustawianie dopełnienia	290
	Wkrótce!	292
	Podsumowanie	293
Rozdział 11.	Własności formatowania wizualnego	295
	Różnica między oknem a dokumentem	299
	Wybór rodzaju pozycjonowania	301
	Pozycjonowanie elementów	306
	Układanie elementów na osi pionowej	308
	Ustawianie widoczności elementów	310
	Obcinanie widocznego obszaru elementu	312
	Nowość w CSS 3: ustawianie przezroczystości elementów ★	314
	Nowość w CSS 3: tworzenie cieni elementów ★	316
	Podsumowanie	318

Rozdział 12.	Przekształcenia i efekty przejścia	319
	Nowość w CSS 3: przekształcenia elementów ★	323
	Nowość w CSS 3: efekty przejścia ★	332
	Podsumowanie	337
Rozdział 13.	Poprawki i ulepszenia	339
	Dostosowywanie CSS do Internet Explorera	340
	Rozwiązywanie problemów z modelem polowym w IE	349
	Zerowanie ustawień CSS	351
	Naprawa elementów pływających	356
Rozdział 14.	Podstawowe techniki CSS	359
	Tworzenie układów wielokolumnowych przy użyciu elementów pływających	362
	Formatowanie odnośników a nawigacja	366
	Sprite'y CSS	370
	Tworzenie menu rozwijanego przy użyciu CSS	373
Rozdział 15.	Zarządzanie arkuszami stylów	377
	Techniki pisania czytelnego kodu CSS	378
	Biblioteki i ramy projektowe CSS	383
	Strategie dołączania arkuszy stylów do stron	384
	Usuwanie usterek	388
	Znajdowanie błędów za pomocą narzędzi Firebug i Inspektor www	392
	Sprawdzanie poprawności kodu	397
	Minimalizacja kodu	398
	32 najlepsze praktyki CSS	401
Dodatek A	Przewodnik po CSS	409
Dodatek B	Kody znaków HTML i UTF	423
	Skorowidz	429

Tekst można znaleźć wszędzie. Służy zarówno do tak przyziemnych celów jak wypisanie składników płatków śniadaniowych na opakowaniu, jak również do pisania poezji najwyższych lotów. Jest to najlepszy system do kodowania skomplikowanych informacji, jaki kiedykolwiek wynaleziono. Dla wielu osób tekst to po prostu zapis słów, ale typografia jest sztuką, która wzbogaca słowa o dodatkowe brzmienie. Sztuka składania tekstu nie polega tylko na określeniu rozmiaru i kształtu liter, lecz również na zdefiniowaniu odstępów między literami, słowami, wierszami oraz akapitami.

Niestety monitor komputerowy to niedoskonałe medium, przez co możliwości typograficzne w internecie są mocno ograniczone. Najważniejsze zadania stojące przed projektantami to sprawienie, aby tekst był jak najbardziej czytelny, oraz przykucie uwagi odbiorcy do wybranych fragmentów. Nie jest to łatwe, ale w tym rozdziale poznasz narzędzia, które będą Ci w tym pomocne.

W rozdziale:

Ustawienia odstępów	169
Wielkie i małe litery	174
Nowość w CSS 3: cienie tekstu ★	176
Wyrównywanie tekstu w poziomie	178
Wyrównywanie tekstu w pionie	180
Wcięcia akapitów	183
Kontrolowanie odstępów	185
Zdobienie tekstu	188
Wkrótce!	191
Podsumowanie	193

Listing 6.1. Kod HTML 5, na podstawie którego utworzono przykłady do tego rozdziału. Znajduje się w nim połączenie z ostateczną wersją pliku CSS z rozdziału 5. oraz z nowym plikiem o nazwie `text-properties.css`, którego budowa zostanie przedstawiona w tym rozdziale

```

Listing
<!-- HTML5 -->
<!DOCTYPE html>
<html lang="pl">
<head>
<meta charset="utf-8">
<title>Alicja w Krainie Czarów | Rozdział II</title>
<link href="../../css/font-properties.css" type="text/css" rel="stylesheet">
<link href="../../css/text-properties.css" type="text/css" rel="stylesheet">
<!--[if IE]-->
  <style>@import url(_css/ie.css);</style>
  <script src="script/HTML5forIE.js" type="text/javascript"></script>
<![endif]-->
</head>
<body>
<header>
<h1>Alicja w Krainie Czarów</h1>
<p class="byline"><span class="author">Lewis Carroll</span></p>
<header>
<article><!-- Article -->
<header>
<h2><strong>Rozdział II.</strong> Siedzawka z łez</h2>
</header>
<p>Ach jak zdumiewająco! Coraz zdumiewającej! – krzyknęła Alicja. <span class="strike">Była tak
zdumiona, że aż zapomniała o poprawnym wyrażaniu się.</span> – Rozciągam się teraz jak
↳ największy teleskop na świecie. Do widzenia, nogi! – Spoglądając w dół, Alicja zauważyła, że jej
↳ nogi wydłużyły się coraz bardziej i ginęły w oddali. – O moje biedne nóżki, ktoś wam teraz
↳ będzie wkładał skarpetki i buciki? Bo <em>ja</em> z pewnością nie dam sobie z tym rady, będąc od
↳ was tak daleko. Musicie sobie teraz radzić same.
<p>„Powinnam jednak być dla nich uprzejma – pomyślała Alicja – bo mogą nie pójść tam, gdzie ja
↳ będę chciała. Zaraz, zaraz... Wiem. Będę im dawała po nowej parze bucików na każde Boże
↳ Narodzenie.“</p>
<p>Tu Alicja zaczęła zastanawiać się, w jaki sposób doręczy im prezenty. „Chyba przez posłańca -
↳ pomyślała. – Ale jakie to będzie śmieszne posyłać podarunki swoim własnym nogom. A jak zabawnie
↳ będzie wyglądał adres...</p>
<div class="asis">
  Wielmożna Pani Prawa Noga Alicji, Dywanik przed Kominkiem, tuż obok
  paleniska, z serdecznym pozdrowieniem od Alicji.
</div>
</article>
<footer>
  <ul>
    <li><a href="" target="_self">Spis treści</a></li>
    <li><a href="" target="_self">0 autorze</a></li>
    <li><a href="" target="_self">0 książkach</a></li>
    <li><a href="" target="_self">0 witrynie</a></li>
  </ul>
</footer>
</body>
</html>

```


Tabela 6.1. Wartości własności *letter-spacing*

Wartość	Zgodność
normal	IE4, FF1, S1, C1, O3.5, CSS1
<długość>	IE4, FF1, S1, C1, O3.5, CSS1

Listing 6.2. *text-properties.css* — w nagłówku *h1* (tytuł książki) odstępy między literami zostały zmniejszone, czego efektem jest ściśnięty tekst **A**. Tytuł rozdziału został natomiast rozstrzelony, aczkolwiek we fragmencie z numerem rozdziału (Rozdział II) styl ten został anulowany, tak że ma on normalny wygląd. Ponadto zmniejszono odstęp po pierwszej literze akapitu, która została sformatowana jako inicjał

```
Listing
/** Rozdział 6. | text-properties.css */

h1 {
  letter-spacing: -.05em; }

h2 {
  letter-spacing: 2px; }

h2 strong {
  letter-spacing: 0; }

header + p:first-letter {
  letter-spacing: -.05em; }
```



A Kod z listingu 6.2 zastosowany do strony z listingu 6.1. Dzięki zmniejszeniu odstępów między literami tytuł wygląda bardziej jak logo i wyróżnia się na tle reszty tekstu. To samo dotyczy tytułu rozdziału, lecz w tym przypadku zastosowano zwiększenie odstępów między literami

Ustawienia odstępów

W kaskadowych arkuszach stylów ustawianie takich parametrów tekstu jak odstępy między literami (ang. *tracking*), słowami i liniami w akapicie (ang. *leading*) jest bardzo łatwe. Oczywiście podobny efekt można by uzyskać, korzystając wyłącznie z języka HTML. Wystarczyłoby w tym celu tylko odpowiednio użyć spacji niełamliwych i elementu złamania wiersza. Lecz taką bezładną gmatwaninę znaczników niełatwo zaimplementować, kontrolować i modyfikować. W CSS wszystkie te właściwości tekstu kontroluje się o wiele łatwiej.

Ustawianie odstępów między literami

Do ustawiania odstępów między literami (ang. *tracking*) w języku CSS służy własność o nazwie *letter-spacing* (tabela 6.1). Zazwyczaj zwiększenie odległości między poszczególnymi literami powoduje, że tekst staje się bardziej czytelny, ale nie można przesadzać, gdyż po przekroczeniu pewnego progu kolejne wyrazy zaczynają się zlewać i występuje efekt odwrotny.

Aby określić odstęp między literami:

1. Dodaj do reguły w arkuszu stylów własność **letter-spacing**. Wpisz na liście deklaracji własność *letter-spacing* i postaw dwukropek (listing 6.2).

```
letter-spacing:
```

2. Określ odległość między literami.

```
-.05em;
```

Własności *letter-spacing* można przypisać jeden z dwóch rodzajów wartości:

- ▲ Dodatnią lub ujemną liczbę z jednostką długości, np. `-.05em`, która ustala odległość między literami w tekście. Więcej informacji na temat jednostek i wartości znajduje się w części wstępnej książki.
- ▲ Słowo kluczowe `normal`, które anuluje odziedziczone ustawienia.

Wskazówki

- Kerning i tracking to dwa terminy typograficzne, które są często mylone. Oba odnoszą się do metod regulacji światła międzyliterowego, ale na różne sposoby (ramka „Tracking czy kerning”).
- Najlepszą jednostką do ustawiania odstępów między literami jest em, ponieważ jej wartość zależy od rozmiaru pisma. Kiedy rozmiar ten się zmieni, odstęp między literami automatycznie się do niego dostosują.
- Dodatnie wartości własności letter-spacing powodują zwiększenie światła międzyliterowego, a ujemne — jego zmniejszenie. Wartość 0 go nie zwiększa ani nie zmniejsza, ale uniemożliwia justowanie (podrozdział „Wyrównywanie tekstu w poziomie”).

Tracking czy kerning

Termin **tracking** określa regulację odstępów międzyliterowych w całym słowie, natomiast **kerning** oznacza ustawianie odstępów między konkretnymi parami znaków w fontach proporcjonalnych. Czy to nie masło maślane? Nie, między tymi pojęciami jest ważna różnica.

Tracking to ustawianie równego odstępów między wszystkimi parami liter w słowie, natomiast kerning polega na definiowaniu światła między poszczególnymi parami liter w celu nadania tekstowi jednolitego wyglądu. Jednakże zastosowanie trackingu może spowodować powiększenie odstępów między niektórymi parami liter w celu zwiększenia czytelności tekstu. Oczywiście można ustawić światło między dowolną parą liter za pomocą CSS, ale nie zalecam robienia tego. Profesjonalny kerning polega na użyciu specjalnych programów do składania tekstu, gdyż trudno jest wszystko ustawić na oko.

Listing 6.3. *text-properties.css* — po zastosowaniu tego kodu do strony z listingu 6.1 słowa w nagłówku h1 prawie na siebie nachodzą, ale to nie przeszkadza, gdyż każde z nich zaczyna się od wielkiej litery, dzięki czemu łatwo je odróżnić **B**. Natomiast w tytule rozdziału słowa są rozstrzelone, co sprawia, że strona dzięki dodatkowej przestrzeni wydaje się lepsza

```
Listing
/** Rozdział 6. | text-properties.css */

h1 {
  letter-spacing: -.05em;
  word-spacing: -.1em; }

h2 {
  letter-spacing: 2px;
  word-spacing: 3px; }

h2 strong {
  letter-spacing: 0; }

p {
  word-spacing: .075em; }

header + p:first-letter {
  letter-spacing: -.05em; }

.byline {
  word-spacing: -.3em; }

.byline .author {
  word-spacing: 0; }
```



B Kod z listingu 6.3 zastosowany do strony z listingu 6.1. W tytule książki odstęp między wyrazami został zmniejszony, dzięki czemu uzyskano ciekawy i czytelny efekt, natomiast w tytule rozdziału wyrazy są rozstrzelone, co daje wrażenie lekkości tekstu

Ustawianie odstępów między wyrazami

Podobnie jak w przypadku odstępów między literami, regulując światło międzywyrazowe, można równie dobrze polepszyć, jak i pogorszyć czytelność tekstu. Nieznaczące zwiększenie odstępów między wyrazami za pomocą własności `word-spacing` (tabela 6.2) może sprawić, że będzie się go czytało łatwiej, lecz zbyt duże odstępów utrudniają czytelnikowi przenoszenie wzroku na kolejne słowa, co bardzo przeszkadza w czytaniu.

Aby określić odstęp między wyrazami:

1. Dodaj do reguły w arkuszu stylów własność `word-spacing`. Wpisz na liście deklaracji własność `word-spacing` i postaw dwukropki (listing 6.3).
2. Określ odległość między wyrazami.

`.1em;`

Własności `word-spacing` można przypisać jeden z dwóch rodzajów wartości:

- ▲ Dodatnią lub ujemną liczbę z jednostką długości, np. `-.1em`, która ustala odległość między wyrazami w tekście. Więcej informacji na temat jednostek i wartości znajduje się w części wstępnej książki.
- ▲ Słowo kluczowe `normal`, które anuluje odziedziczone ustawienia.

Wskazówka

- Przestrzenie między wyrazami, podobnie jak między literami, należy zmieniać tylko w wyjątkowych przypadkach. Standardowe ustawienia są już najczęściej idealnie dostosowane do wygodnego czytania i ich zmiana często przynosi więcej złego niż dobrego.

Tabela 6.2. Wartości własności `word-spacing`

Wartość	Zgodność
<code>normal</code>	IE4, FF1, S1, C1, O3.5, CSS1
<code><długość></code>	IE4, FF1, S1, C1, O3.5, CSS1

Ustawianie odstępów między wierszami tekstu

Każdy, kto pisał na studiach pracę na zaliczenie, wie, że obowiązuje podwójny odstęp międzywierszowy, aby sprawdzającemu łatwiej czytało się tekst oraz aby miał miejsce na wpisywanie uwag. Odstęp między wierszami (ang. *leading*) zwiększa się też w celu uzyskania bardziej dramatycznego efektu. Własność `line-height` (tabela 6.3) reguluje odległość między **bazowymi liniami pisma** (znajdującymi się na wysokości spodniej strony większości liter) wierszy tekstu.

Aby określić odstęp między wierszami tekstu:

1. Dodaj do reguły w arkuszu stylów własność `line-height`. Wpisz na liście deklaracji własność `line-height` i postaw dwukropek (listing 6.4).

```
line-height:
```

2. Określ odległość między wierszami tekstu.

```
.9;
```

Własności `line-height` można przypisać jeden z czterech rodzajów wartości:

- ▲ Liczbę do pomnożenia przez rozmiar pisma — np. 2.0 oznacza podwojenie przestrzeni. Można też wpisać samą dwójkę, ale bez kropki dziesiętnej kod nie przejdzie pomyślnie walidacji.

Tabela 6.3. Wartości własności `line-height`

Wartość	Zgodność
<code>normal</code>	IE3, FF1, S1, C1, O3.5, CSS1
<code><liczba></code>	IE4, FF1, S1, C1, O3.5, CSS1
<code><długość></code>	IE3, FF1, S1, C1, O3.5, CSS1
<code><procent></code>	IE3, FF1, S1, C1, O3.5, CSS1

Listing 6.4. `text-properties.css` — dla elementu `body` został ustalony `leading` o wartości 1.5 spacji, natomiast w akapitach zmieniono go na 24 piksele. Jeśli rozmiar pisma wynosi 12 pikseli, to odstęp międzywierszowy będzie podwójny. W tytule książki odstęp został nieco zmniejszony ©

```
Listing
/** Rozdział 6. | text-properties.css */

body {
  line-height: 1.5; }

h1 {
  letter-spacing: -.05em;
  word-spacing: -.1em;
  line-height: .9; }

h2 {
  letter-spacing: 2px;
  word-spacing: 3px; }

h2 strong {
  letter-spacing: 0; }

p {
  word-spacing: .075em;
  line-height: 24px; }

header + p:first-letter {
  letter-spacing: -.05em;
  line-height: 24px; }

.byline {
  word-spacing: -.3em; }

.byline .author {
  word-spacing: 0; }
```

Wskazówki

- Zwiększenie odstępu międzywierszowego powoduje, że tekst staje się bardziej czytelny, zwłaszcza gdy jest go dużo. Zazwyczaj najlepszy efekt uzyskuje się, stosując wartości z przedziału 1.5 – 2 rozmiarów pisma.
- Aby otrzymać podwójny odstęp międzywierszowy, nadaj własności line-height wartość 2 lub 200%. Analogicznie w celu potrojenia go zastosuj wartości 3 lub 300% itd.
- Aby zmniejszyć odstęp między wierszami, należy zastosować wartość mniejszą niż 100% lub liczbę mniejszą od 1. Efekt może być dość ciekawy wizualnie, ale raczej nie przysporzy Ci czytelników, jeśli tekstu nie będzie się dało czytać.
- Odstęp międzywierszowy można ustawić jednocześnie z rozmiarem pisma przy użyciu własności zbiorczej font (podrozdział „Zbiorcze definiowanie własności pisma” w rozdziale 5.).
- ▲ Liczbę z jednostką długości, np. 24px. Wówczas odległość między wierszami zostanie ustawiona na podaną wartość i będzie niezależna od rozmiaru pisma. Jeśli więc rozmiar pisma zostanie ustawiony na 12px, a odstęp międzywierszowy na 24px, to tekst będzie miał podwójny odstęp między wierszami. Więcej informacji na temat jednostek i wartości znajduje się w części wstępnej książki.
- ▲ Wartość procentową, która reguluje odstęp międzywierszowy względem rozmiaru pisma.
- ▲ Słowo kluczowe normal, które anuluje odziedziczone ustawienia.



© Kod z listingu 6.4 zastosowany do strony z listingu 6.1. Odstęp między wierszami w akapitach został zwiększony, aby ułatwić czytanie i przeglądanie tekstu

Wielkie i małe litery

Kiedy treść prezentowana na stronach jest wytwarzana na bieżąco, np. pobierana z bazy danych, nigdy nie można być pewnym, czy tekst zostanie napisany małymi, wielkimi, czy mieszanymi literami. Za pomocą własności `text-transform` (tabela 6.4) można zapanować nad wielkością liter bez względu na to, w jakim stanie przyjdą ze źródła.

Aby określić wielkość liter w tekście:

1. Dodaj do reguły w arkuszu stylów własność `text-transform`. Wpisz na liście deklaracji własność `text-transform` i postaw dwukropki (listing 6.5).

text-transform:

2. Określ wielkość liter.


uppercase;

Własności `text-transform` można jako wartość przypisać jedno z czterech słów kluczowych:

- ▲ `capitalize` — pierwsza litera każdego wyrazu będzie wielka;
- ▲ `uppercase` — wszystkie litery będą wielkie;
- ▲ `lowercase` — wszystkie litery będą małe;
- ▲ `none` — anuluje odziedziczone ustawienia i pozostawia tekst w pierwotnej postaci.

Tabela 6.4. Wartości własności `text-transform`

Wartość	Zgodność
<code>capitalize</code>	IE4, FF1, S1, C1, O3.5, CSS1
<code>uppercase</code>	IE4, FF1, S1, C1, O3.5, CSS1
<code>lowercase</code>	IE4, FF1, S1, C1, O3.5, CSS1
<code>none</code>	IE4, FF1, S1, C1, O3.5, CSS1

Listing 6.5. `text-properties.css` — własność `text-transform` służy do ustawiania wielkości liter. W tym przypadku nazwisko i imię autora oraz treść elementów należących do klasy `asis` zostaną napisane wielkimi literami 

```
Listing
/** Rozdział 6. | text-properties.css */

body {
    line-height: 1.5; }

h1 {
    letter-spacing: -.05em;
    word-spacing: -.1em;
    line-height: .9em; }

h2 {
    letter-spacing: 2px;
    word-spacing: 3px; }

h2 strong {
    letter-spacing: 0; }

p {
    word-spacing: .075em;
    line-height: 24px; }

header + p:first-letter {
    letter-spacing: -.05em;
    line-height: 24px; }

.byline {
    word-spacing: -.3em; }

.byline .author {
    word-spacing: 0;
    text-transform: uppercase; }

.asis {
    text-transform: uppercase; }
```

Wskazówki

- Własność `text-transform` najlepiej nadaje się do formatowania treści generowanej dynamicznie. Jeśli na przykład w bazie danych wszystkie nazwy są zapisane w całości wielkimi literami, to za pomocą tej własności można je dostosować do czytania.
- Pamiętaj, że wartość `capitalize` zamieni na wielkie pierwsze litery wszystkich wyrazów, a nawet pojedyncze litery, np. „i” albo „a”. W praktyce wartość ta jest przydatna tylko do formatowania imion i nazwisk.
- **NIE PISZ TEKSTU SAMYMI WIELKIMI LITERAMI.** Nad tekstem zapisanym w dokumencie HTML w taki sposób trudniej jest zapanować za pomocą arkuszy stylów. Co więcej, osoby posługujące się czytnikami ekranu usłyszą to jako krzyk. O wiele lepiej jest pisać małymi literami i zwiększać je w razie potrzeby za pomocą wartości `uppercase`.

ALICJA W KRAINIE CZARÓW

LEWIS CARROLL

Rozdział II. **SADZAWKA Z ŁEZ**

Ach jak zdumiewająco! Coraz zdumiewającej! - krzyknęła Alicja. Była tak zdumiona, że aż zapomniała o poprawnym wyrażaniu się. - Rozciągam się teraz jak największy teleskop na świecie. Do widzenia, nogi! - Spoglądając w dół, Alicja zauważyła, że jej nogi wydłużyły się coraz bardziej i ginęły w oddali. - O moje biedne nóżki, któż wam teraz będzie wkładał skarpetki i buciki? Bo ja z pewnością nie dam sobie z tym rady, będąc od was tak daleko. Musicie sobie teraz radzić same.

„Powinam jednak być dla nich uprzejma - pomyślała Alicja - bo mogą nie pójść tam, gdzie ja będę chciała. Zaraz, zaraz... Wiem. Będę im dawała po nowej parze bucików na każde Boże Narodzenie”.

A Kod z listingu 6.5 zastosowany do strony z listingu 6.1. Imię i nazwisko autora są napisane wielkimi literami (zobacz też opis własności `small-caps` w rozdziale 5.)

Nowość w CSS 3: cienie tekstu ★

Dodatek cienia to tradycyjna metoda nadawania tekstowi wrażenia głębi i niezwykłego wyglądu. Większość przeglądarek obsługuje własność CSS 3 o nazwie `text-shadow` (tabela 6.5), która umożliwia zdefiniowanie takich parametrów cienia tekstu jak kolor, przesunięcie względem tekstu (w poziomie i w pionie), rozpiętość oraz zamazanie. W przeglądarkach, w których nie jest ona obsługiwana, po prostu nie zostanie wyświetlony cień.

Aby utworzyć cień tekstu:

1. Dodaj do reguły w arkuszu stylów własność `text-shadow`. Wpisz na liście deklaracji własność `text-shadow` i postaw dwukropek (listing 6.6).

`text-shadow:`

2. Określ przesunięcie w poziomie i w pionie. Zrób spację, a za nią wstaw dwie dodatnie lub ujemne wartości z jednostkami długości, które również oddziel spacjami.

`2px 2px;`

Pierwsza wartość określa przesunięcie cienia względem tekstu w pionie (wartość dodatnia przesuwa go do góry, a ujemna w dół), a druga w poziomie (dodatnia przesuwa go w prawo, a ujemna w lewo).

3. Określ poziom rozmazania. Zrób spację, a za nią wpisz dodatnią wartość z jednostką długości. Wartości ujemne są traktowane jako 0.

`2px`

Tabela 6.5. Wartości własności `text-shadow`

Wartość	Zgodność
<code><kolor></code>	FF1.9, S1.1, C2, O10, CSS3
<code><przesunięcie ↳-w-poziomie></code>	FF1.9, S1.1, C2, O10, CSS3
<code><przesunięcie ↳-w-pionie></code>	FF1.9, S1.1, C2, O10, CSS3
<code><zamazanie></code>	FF1.9, S1.1, C2, O10, CSS3
<code>none</code>	FF1.9, S1.1, C2, O10, CSS3



Ⓐ Kod z listingu 6.6 zastosowany do strony z listingu 6.1. Aby zauważyć różnicę, może być konieczne porównanie tego zrzutu ekranu z poprzednim. Tytuł książki sprawia wrażenie, jakby był wytłoczony na stronie, natomiast o tytule rozdziału można powiedzieć, że wręcz wyskakuje ze swojego miejsca

Listing 6.6. *text-properties.css* — własność *text-shadow* umożliwia ustawienie położenia cienia w poziomie i w pionie oraz zdefiniowanie promienia rozmazania. Tytuł książki ma podwójny cień, co sprawia, że wygląda, jakby był wytłoczony **A**, natomiast tytuł rozdziału ma ciemny cień przesunięty tylko nieznacznie, przez co słowa wydają się unosić nad stronę

```

Listing
/** Rozdział 6. | text-properties.css */

body {
  line-height: 1.5; }

h1 {
  letter-spacing: -.05em;
  word-spacing: -.1em;
  line-height: .9em;
  text-shadow: rgba(51,51,51,.9) -1px -1px
  ↳3px, rgba(203,203,203,.9) 1px 1px 3px; }

h2 {
  letter-spacing: 2px;
  word-spacing: 3px;
  line-height: 1em;
  text-shadow: rgba(0,0,0,.5) 2px 2px 2px; }

h2 strong {
  letter-spacing: 0;
  text-shadow: none; }

p {
  word-spacing: .075em;
  line-height: 24px; }

p strong {
  text-transform: uppercase; }

header + p:first-letter {
  letter-spacing: -.05em;
  line-height: 24px;
  text-shadow: rgba(51,51,51,.9) -1px -1px
  ↳3px, rgba(203,203,203,.9) 1px 1px 3px; }

.byline {
  word-spacing: -.3em; }

.byline .author {
  word-spacing: 0;
  text-transform: uppercase; }

.asis {
  text-transform: uppercase; }

```

4. Określ kolor. Zrób spację, a za nią wpisz wartość koloru cienia.

```
rgba(0,0,0,.5)
```

Więcej informacji na temat definiowania kolorów znajdziesz w podrozdziale „Definiowanie kolorów” rozdziału 7.

5. Dodaj kolejne cienie. Jeśli chcesz, możesz zdefiniować więcej cieni (dowolną ich liczbę). W tym celu dodaj nową deklarację własności *text-shadow* albo wpisz nową definicję po przecinku.

```

text-shadow: rgba(51,51,51,.9)
↳-1px -1px 3px, rgba(203,203,203,.9)
↳1px 1px 3px;

```

Wskazówki

- Aby anulować cień zdefiniowany wcześniej w arkuszu stylów, można użyć wartości *none*.
- Do definiowania kolorów cieni najlepiej nadaje się notacja RGBA. Dzięki niej wszystko, co znajduje się pod cieniem, będzie przez niego prześwytawać, tworząc realistyczny efekt.
- Mimo iż tworzony jest cień, można go pokolorować na dowolny kolor. Jeśli więc tekst znajduje się na czarnym tle, to można użyć jasnego koloru, aby utworzyć poświatę.
- W rozdziale 11. dowiesz się, jak tworzyć cienie elementów blokowych, które stosuje się właśnie do nich zamiast do tekstu.
- Cień nie ma wpływu na położenie tekstu, do którego jest zastosowany.
- Za pomocą cieni użytych w połączeniu z pseudoklasą *:hover* można sprawić, że odnośniki będą „podskakiwały”, kiedy najedzie się na nie kursorem.

Wyrównywanie tekstu w poziomie

Zwykle tekst wyrównuje się do lewego marginesu albo justuje (wyrównuje zarówno do prawego, jak i lewego marginesu), co czasami nazywa się **stylem gazetowym**. Od czasu do czasu w celu wyróżnienia fragmentu albo uzyskania jakiegoś specjalnego efektu tekst może zostać wyśrodkowany, a nawet wyrównany do marginesu prawego. Do ustawiania tych parametrów w CSS służy własność `text-align` (tabela 6.6).

Aby określić wyrównanie tekstu:

1. Dodaj do reguły w arkuszu stylów własność `text-align`. Wpisz na liście deklaracji własność `text-align` i postaw dwukropek (listing 6.7).

`text-align:`

2. Określ wyrównanie tekstu w poziomie.

`center;`

Własności `text-align` można jako wartość przypisać jedno z następujących słów kluczowych:

- ▲ `left` — wyrównanie tekstu do lewego marginesu;
- ▲ `right` — wyrównanie tekstu do prawego marginesu;
- ▲ `center` — wyśrodkowanie tekstu;
- ▲ `justify` — wyjustowanie tekstu;
- ▲ `inherit` — wyrównanie takie samo jak w elemencie nadrzędnym;
- ▲ `auto` — domyślne wyrównanie, a więc najczęściej do lewego marginesu.

Tabela 6.6. Wartości własności `text-align`

Wartość	Zgodność
<code>left</code>	IE3, FF1, S1, C1, O3.5, CSS1
<code>right</code>	IE3, FF1, S1, C1, O3.5, CSS1
<code>center</code>	IE3, FF1, S1, C1, O3.5, CSS1
<code>justify</code>	IE4, FF1, S1, C1, O3.5, CSS1
<code>inherit</code>	IE4, FF1, S1, C1, O3.5, CSS1
<code>auto</code>	IE4, FF1, S1, C1, O3.5, CSS1



▲ Kod z listingu 6.7 zastosowany do strony z listingu 6.1. Na razie po zdefiniowaniu różnych rodzajów wyrównania do różnych fragmentów strona wydaje się jakby nieco „rozbiegana”, ale formatowanie niektórych elementów jeszcze zmienimy, kiedy będzie mowa o pozycjonowaniu

Listing 6.7. *text-properties.css* — klasa *byline* zostaje wyrównana do prawego marginesu **A**, natomiast tytuł rozdziału i zawartość nagłówka *h2* zostają wyśrodkowane, a treść książki wyjustowana

```
Listing
/** Rozdział 6. | text-properties.css */
body {
  line-height: 1.5; }

h1 {
  letter-spacing: -.05em;
  word-spacing: -.1em;
  line-height: .9em;
  text-shadow: rgba(51,51,51,.9) -1px -1px
↳3px, rgba(203,203,203,.9) 1px 1px 3px; }

h2 {
  letter-spacing: 2px;
  word-spacing: 3px;
  line-height: 1em;
  text-shadow: rgba(0,0,0,.5) 2px 2px 2px;
text-align: center; }

h2 strong {
  letter-spacing: 0;
  text-shadow: none; }

p {
  word-spacing: .075em;
  line-height: 24px;
text-align: justify; }

p strong {
  text-transform: uppercase; }

header + p:first-letter {
  letter-spacing: -.05em;
  line-height: 24px;
  text-shadow: rgba(51,51,51,.9) -1px -1px
↳3px, rgba(203,203,203,.9) 1px 1px 3px; }

.byline {
  word-spacing: -.3em;
text-align: right; }

.byline .author {
  word-spacing: 0;
  text-transform: uppercase; }

.asis {
  text-transform: uppercase; }
```

Wskazówki

- Tekst jest domyślnie wyrównywany do lewego marginesu.
- Ponieważ justowanie wymaga, aby wszystkie wiersze tekstu miały taką samą długość, czasami konieczna jest zmiana odstępów między wyrazami, która może powodować, że tekst na ekranie wygląda dość dziwnie. Ponadto opinie na temat tego, czy justowanie pomaga, czy przeszkadza w czytaniu, są podzielone.

Wyrównywanie tekstu w pionie

Za pomocą własności `vertical-align` można określić pionowe ułożenie elementu śródliniowego względem elementów znajdujących się nad i pod nim **A**. Oznacza to, że własności tej (tabela 6.7) można używać tylko do formatowania elementów śródliniowych i tabel, tzn. elementów nietworzących przed ani za sobą złamania wiersza, np. kotwic (`a`), obrazów (`img`), uwypuklonego tekstu (`em`), silnie wyróżnionego tekstu (`strong`) oraz danych tabeli (`td`).

Aby określić wyrównanie elementu w pionie:

1. Dodaj do reguły w arkuszu stylów własność `vertical-align`. Wpisz na liście deklaracji własność `vertical-align` i postaw dwukropek (listing 6.8).

```
vertical-align:
```

Szalony	Kapelusznik	Linia bazowa
Szalony	Kapelusznik	Indeks górny
Szalony	Kapelusznik	Indeks dolny
Szalony	Kapelusznik	Linia górna
Szalony	Kapelusznik	Linia środkowa
Szalony	Kapelusznik	Linia dolna
Szalony	Kapelusznik	Wierzch tekstu
Szalony	Kapelusznik	Spód tekstu

A Różne rodzaje wyrównania. Linie służą tylko jako pomoc

Tabela 6.7. Wartości własności `vertical-align`

Wartość	Zgodność
super	IE4, FF1, S1, C1, O3.5, CSS1
sub	IE4, FF1, S1, C1, O3.5, CSS1
baseline	IE4, FF1, S1, C1, O3.5, CSS1
<wartość- →względna>	IE5.5, FF1, S1, C1, O3.5, CSS1
<długość>	IE5.5, FF1, S1, C1, O3.5, CSS1
<procent>	IE7, FF1, S1, C1, O3.5, CSS1



B Kod z listingu 6.8 zastosowany do strony z listingu 6.1. Numer rozdziału został przesunięty do góry względem tekstu, do którego się odnosi

Listing 6.8. *text-properties.css* — własności *vertical-align* można przypisywać zarówno wartości względne, jak i bezwzględne **B**

```

Listing
/** Rozdział 6. | text-properties.css */

body {
  line-height: 1.5; }

h1 {
  letter-spacing: -.05em;
  word-spacing: -.1em;
  line-height: .9em;
  text-shadow: rgba(51,51,51,.9) -1px -1px 3px, rgba(203,203,203,.9) 1px 1px 3px; }

h2 {
  letter-spacing: 2px;
  word-spacing: 3px;
  line-height: 1em;
  text-shadow: rgba(0,0,0,.5) 2px 2px 2px;
  text-align: center; }

h2 strong {
  letter-spacing: 0;
  text-shadow: none;
  vertical-align: super; }

p {
  word-spacing: .075em;
  line-height: 24px;
  text-align: justify; }

p strong {
  text-transform: uppercase; }

header + p:first-letter {
  letter-spacing: -.05em;
  line-height: 24px;
  text-shadow: rgba(51,51,51,.9) -1px -1px 3px, rgba(203,203,203,.9) 1px 1px 3px; }

.byline {
  text-align: right;
  word-spacing: -.3em; }

.byline .author {
  word-spacing: 0;
  text-transform: uppercase;
  vertical-align: -.6em; }

.asis {
  text-transform: uppercase; }

```

2. Określ wyrównanie elementu w pionie.

super;

Własności `vertical-align` można jako wartość przypisać jedno z poniższych słów kluczowych.

- ▲ **super** — przenosi tekst do indeksu górnego nad linią bazową.
- ▲ **sub** — przenosi tekst do indeksu dolnego pod linią bazową.
- ▲ **baseline** — umieszcza tekst na linii bazowej.
- ▲ Jedna z wartości względnych wymienionych w tabeli 6.8 — ustawia element w odniesieniu do ustawienia elementu nadrzędnego. Aby na przykład wyrównać wierzchnią część tekstu elementu z wierzchnią częścią tekstu jego rodzica, należy napisać `text-top`.
- ▲ Wartość procentowa — podnosi lub obniża linię bazową elementu proporcjonalnie do rozmiaru pisma elementu nadrzędnego (np. o 25%).

Wskazówki

- W indeksie górnym można umieszczać odwołania w postaci łączy do przypisów, które znajdują się na dole bieżącej lub innej strony.
- Indeks górny i dolny można również uzyskać za pomocą elementów `sup` i `sub`, ale nie należy ich używać jako elementów projektowych (jak w tym przypadku), lecz do tworzenia przypisów i w zapisie działań matematycznych.
- Przy tworzeniu wielokolumnowego układu strony elementy wyrównywane w pionie mogą nie być wyświetlane tam, gdzie powinny. Dobrym rozwiązaniem tego problemu jest użycie pozycjonowania względnego, którego opis znajduje się w rozdziale 10.

Tabela 6.8. Ustawienia położenia elementu względem rodzica

Ustawienie	Opis
<code>top</code>	Wyrównuje wierzchołek elementu z górną linią najwyższej położonego elementu w linii
<code>middle</code>	Wyrównuje linię środkową elementu z linią środkową jego rodzica
<code>bottom</code>	Wyrównuje linię dolną elementu z linią dolną najniższej położonego elementu w linii
<code>text-top</code>	Wyrównuje wierzch tekstu elementu z wierzchem tekstu jego rodzica
<code>text-bottom</code>	Wyrównuje spód tekstu elementu ze spodem tekstu jego rodzica

Matematyka i nauka

Indeksy górne i dolne są używane w zapisie działań matematycznych i w notacji naukowej. Na przykład aby zapisać twierdzenie Pitagorasa, trzeba użyć indeksu górnego:

$$a^2 + b^2 = c^2$$

Zapis wzoru chemicznego cząsteczki wody wymaga natomiast zastosowania indeksu dolnego:



Pamiętaj jednak, że użycie indeksu nie powoduje zmniejszenia rozmiaru pisma, a więc w celu uzyskania prawdziwej notacji naukowej może być potrzebne użycie dodatkowo własności `font-size` (podrozdział „Ustawianie rozmiaru pisma” w rozdziale 5.).

Tabela 6.9. Wartości własności *text-indent*

Wartość	Zgodność
<długość>	IE3, FF1, S1, C1, O3.5, CSS1
<procent>	IE3, FF1, S1, C1, O3.5, CSS1
inherit	IE3, FF1, S1, C1, O3.5, CSS1



A Kod z listingu 6.9 zastosowany do strony z listingu 6.1. Wszystkie akapity z wyjątkiem pierwszego mają wcięty pierwszy wiersz na szerokość dwóch liter. Szerokość wcięcia będzie się automatycznie dostosowywać do rozmiaru pisma

Wcięcia akapitów

Wcięcie pierwszego wiersza (najczęściej na głębokość pięciu spacji) to tradycyjna metoda rozpoczynania akapitów.

W sieci to się nie przyjęło, ponieważ większość przeglądarek redukuje każdą liczbę spacji do jednej. Dlatego zrezygnowano z tradycyjnej metody i zaczęto stosować dodatkowe złamanie wiersza.

Jeśli jednak chcesz wrócić do tradycyjnych metod i zaczynać każdy akapit od wcięcia, możesz to zrobić, korzystając z własności *text-indent* (tabela 6.9).

Aby zastosować wcięcie pierwszego wiersza tekstu w elemencie:

1. Dodaj do reguły w arkuszu stylów własność **text-indent**. Wpisz na liście deklaracji własność *text-indent* i postaw dwukropek (listing 6.9).

text-indent:

2. Określ głębokość wcięcia.

2em;

Własności *text-indent* można przypisać omówione poniżej wartości.

- ▲ Odpowiedni efekt można uzyskać, stosując liczbę z jednostką długości, np. 2em (więcej informacji na temat jednostek i wartości znajduje się w części wstępnej książki).
- ▲ Wartość procentowa, np. 10%, tworzy wcięcie o szerokości proporcjonalnej do szerokości elementu nadrzędnego (akapitu).

Wskazówki

- Jako że wcięcia akapitów są znacznie bardziej popularne w druku niż na stronach internetowych, warto rozważyć możliwość zastosowania tego efektu tylko w arkuszach stylów przeznaczonych dla druku.

- Jeśli do oznaczania początku akapitów stosujesz wcięcia, to wyzeruj marginesy i dopełnienie, aby usunąć dodatkowy odstęp standardowo tworzony przez element p. Więcej o marginesach i dopełnieniu dowiesz się w rozdziale 8.

Listing 6.9. *text-properties.css* — mimo iż wcięcie można zastosować w każdym bloku tekstu, tradycyjnie kojarzy się ono z akapitami i dlatego ustawiłem wcięcie dla wszystkich akapitów na 2em **A**. Jednakże pierwszego akapitu w sekcji z reguły się nie wcina, więc dla wszystkich pierwszych akapitów znajdujących się bezpośrednio za nagłówkiem wcięcie ustawiono na 0. W tym przykładzie jestem trochę nieuczciwy, ponieważ zastosowałem marginesy i dopełnienie, których opis znajduje się dopiero w rozdziale 8.

```

Listing
/** Rozdział 6. | text-properties.css */

body {
  line-height: 1.5; }

h1 {
  letter-spacing: -.05em;
  word-spacing: -.1em;
  line-height: .9em;
  text-shadow: rgba(51,51,51,.9) -1px -1px 3px, rgba(203,203,203,.9) 1px 1px 3px; }

h2 {
  letter-spacing: 2px;
  word-spacing: 3px;
  line-height: 1em;
  text-shadow: rgba(0,0,0,.5) 2px 2px 2px;
  text-align: center; }

h2 strong {
  letter-spacing: 0;
  text-shadow: none;
  vertical-align: super; }

p {
  word-spacing: .075em;
  line-height: 24px;
  text-align: justify;
  text-indent: 2em;
  margin: 0;
  padding: 0; }

p strong {
  text-transform: uppercase;
  vertical-align: middle; }

header + p {
  text-indent: 0; }

header + p:first-letter {
  letter-spacing: -.05em;
  line-height: 24px;
  text-shadow: rgba(51,51,51,.9) -1px -1px 3px, rgba(203,203,203,.9) 1px 1px 3px; }

.byline {
  word-spacing: -.3em;
  text-align: right; }

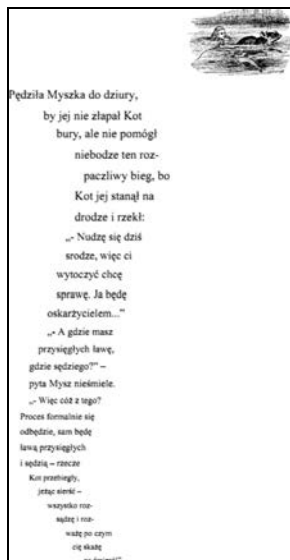
.byline .author {
  word-spacing: 0;
  text-transform: uppercase;
  vertical-align: -.6em; }

.asis {
  text-transform: uppercase; }

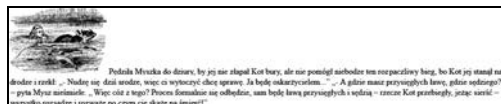
```


Tabela 6.10. Wartości własności *white-space*

Wartość	Zgodność
normal	IE5.5, FF1, S1, C1, O4, CSS1
pre	IE5.5, FF1, S1, C1, O4, CSS1
nowrap	IE5.5, FF1, S1, C1, O4, CSS1



A Wartość *pre* sprawia, że wszystkie spacje zostają zachowane, dzięki czemu wiersz ma kształt ogona myszy



B Bez wartości *pre* ciągi spacji zostają zredukowane do pojedynczych znaków



C Wartość *nowrap* sprawia, że cały wiersz znajduje się w jednej linii, przez co aby go przeczytać, konieczne jest skorzystanie z poziomego paska przewijania

Kontrolowanie odstępów

Jak już wiemy, przeglądarki dowolnie długie ciągi spacji zawsze interpretują jako pojedynczy znak. Jedynym wyjątkiem od tej reguły jest zawartość elementu `pre`. Za pomocą CSS można wyłączyć tę funkcję oraz zdecydować, czy tekst może być zawijany na spacjach (działanie podobne do działania elementu `nowrap`). Służy do tego własność `white-space` (tabela 6.10).

Doskonałym przykładem sytuacji, w której można użyć tej własności, jest formatowanie opowieści Myszy z trzeciego rozdziału *Alicji w Krainie Czarów*. Wiersz ten jest zapisany tak, że wygląda jak wygięty ogon myszy **A**. Jeśli dla klasy poem zostanie zdefiniowana własność `white-space` o wartości `pre`, to wszystkie ciągi spacji zostaną zachowane **B**, natomiast stosując wartość `nowrap`, sprawisz, że spacje i złamania wiersza będą ignorowane, wskutek czego całość tekstu zostanie wyświetlona w jednym bardzo długim wierszu **C**.

Aby określić sposób postępowania ze spacjami:

1. Dodaj do reguły w arkuszu stylów własność **white-space**. Wpisz na liście deklaracji własność **white-space** i postaw dwukropek (listing 6.10).

white-space:

2. Określ sposób postępowania ze spacjami.

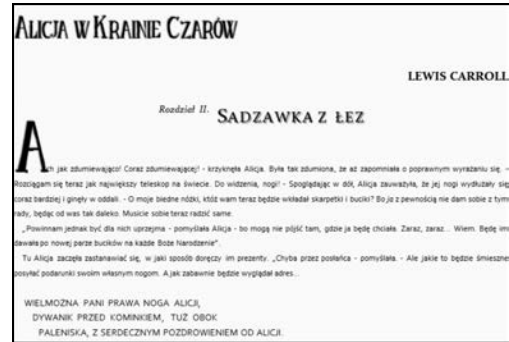
pre;

Własności **white-space** można przypisać następujące wartości:


- ▲ **pre** — zachowuje wszystkie spacje.
- ▲ **nowrap** — wyłącza zawijanie wierszy w miejscach, gdzie nie ma elementu **br**.
- ▲ **normal** — decyduje, jak potraktować spacje, pozostawia przeglądarce. Zazwyczaj oznacza to redukcję ciągów spacji do pojedynczych znaków.

Wskazówki

- Nie należy mylić elementów **HTML nobr** i **pre** z wartościami **nowrap** i **pre** własności **white-space**. Mimo iż ich zastosowanie jest w zasadzie takie samo, element **nobr** jest wycofywany i nie należy go już używać.
- Treść elementu, dla którego zostanie zdefiniowana wartość **nowrap**, będzie wyświetlona w jednym bardzo długim wierszu bez względu na szerokość okna przeglądarki. Aby przeczytać cały tekst, użytkownik może być zmuszony do użycia poziomego paska przewijania, a więc lepiej tego unikać.
- Wartość **nowrap** doskonale nadaje się do zapobiegania zawijaniu tekstu w komórkach tabel.



Ⓛ Kod z listingu 6.10 zastosowany do strony z listingu 6.1. Adres został sformatowany poprzez dodanie spacji. Gdyby nie wartość **pre** własności **white-space**, spacje te zostałyby zredukowane do pojedynczych znaków

Listing 6.10. *text-properties.css* — zawartość elementów przypisanych do klasy *asis* będzie wyświetlana na ekranie w taki sam sposób, jak jest zapisana w kodzie źródłowym strony. Jest to możliwe dzięki użyciu własności *white-space* ustawionej na wartość *pre*. Dzięki temu można sformatować adres do „nóżek Alicji” zgodnie z intencją autora 

```

Listing
/** Rozdział 6. | text-properties.css */

body {
  line-height: 1.5; }

h1 {
  letter-spacing: -.05em;
  word-spacing: -.1em;
  line-height: .9em;
  text-shadow: rgba(51,51,51,.9) -1px -1px 3px, rgba(203,203,203,.9) 1px 1px 3px; }

h2 {
  letter-spacing: 2px;
  word-spacing: 3px;
  line-height: 1em;
  text-shadow: rgba(0,0,0,.5) 2px 2px 2px;
  text-align: center; }

h2 strong {
  letter-spacing: 0;
  text-shadow: none;
  vertical-align: super; }

p {
  line-height: 24px;
  text-align: justify;
  text-indent: 1em;
  margin: 0;
  padding: 0; }

header + p {
  text-indent: 0; }

header + p:first-letter {
  letter-spacing: -.05em;
  line-height: 24px;
  text-shadow: rgba(51,51,51,.9) -1px -1px 3px, rgba(203,203,203,.9) 1px 1px 3px; }

p strong {
  text-transform: uppercase;
  vertical-align: middle; }

.byline {
  word-spacing: -.3em;
  text-align: right; }

.byline .author {
  word-spacing: 0;
  text-transform: uppercase;
  vertical-align: -.6em; }

.asis {
  text-transform: uppercase;
  white-space: pre; }

```

Zdobienie tekstu

Za pomocą własności `text-decoration` (tabela 6.11) można ozdobić tekst na trzy sposoby: dodając podkreślenie, linię nad tekstem oraz przekreślenie. Dodatków tych używa się w celu podkreślenia wagi fragmentów tekstu oraz przyciągnięcia uwagi czytelnika.

Aby dodać lub usunąć dekorację tekstu:

1. Dodaj do reguły w arkuszu stylów własność `text-decoration`. Wpisz na liście deklaracji własność `text-decoration` i postaw dwukropek (listing 6.11).

`text-decoration:`

2. Określ, jakie dekoracje tekstu chcesz zastosować.

`none;`

Własności `text-decoration` można przypisać następujące wartości:

- ▲ `underline` — rysuje linię pod tekstem;
- ▲ `overline` — rysuje linię nad tekstem;
- ▲ `line-through` — rysuje linię przez tekst;
- ▲ `none` — anuluje wcześniejsze ustawienia dekoracji.

Tabela 6.11. Wartości własności `text-decoration`

Wartość	Zgodność
<code>none</code>	IE3, FF1, S1, C1, O3.5, CSS1
<code>underline</code>	IE3, FF1, S1, C1, O3.5, CSS1
<code>overline</code>	IE4, FF1, S1, C1, O3.5, CSS1
<code>line-through</code>	IE3, FF1, S1, C1, O3.5, CSS1



A Kod z listingu 6.11 zastosowany do strony z listingu 6.1. Jeden wiersz tekstu jest skreślony



B Kod z listingu 6.11 zastosowany do strony z listingu 6.1. Usunięto podkreślenie odnośników w stopce

Listing 6.11. *text-properties.css* — przy użyciu własności *text-decoration* utworzyłem klasę o nazwie *strike* do skreślania niechcianego tekstu **A** oraz usuwam podkreślenie hipertączy **B**

```

Listing
/** Rozdział 6. | text-properties.css */

body {
  line-height: 1.5; }

h1 {
  letter-spacing: -.05em;
  word-spacing: -.1em;
  line-height: .9em;
  text-shadow: rgba(51,51,51,.9) -1px -1px 3px, rgba(203,203,203,.9) 1px 1px 3px; }

h2 {
  letter-spacing: 2px;
  word-spacing: 3px;
  line-height: 1em;
  text-shadow: rgba(0,0,0,.5) 2px 2px 2px;
  text-align: center; }

h2 strong {
  letter-spacing: 0;
  text-shadow: none;
  vertical-align: super; }

p {
  line-height: 24px;
  text-align: justify;
  text-indent: 1em;
  margin: 0;
  padding: 0; }

p:first-of-type {
  text-indent: 0; }

article p:first-of-type:first-letter {
  letter-spacing: -.05em;
  line-height: 24px;
  text-shadow: rgba(51,51,51,.9) -1px -1px 3px, rgba(203,203,203,.9) 1px 1px 3px; }

p strong {
  text-transform: uppercase;
  vertical-align: middle; }

a {
  text-decoration: none; }

.byline {
  word-spacing: -.3em;
  text-align: right; }

.byline .author {
  word-spacing: 0;
  text-transform: uppercase;
  vertical-align: -.6em; }

.asis {
  text-transform: uppercase;
  white-space: pre; }

.strike {
  text-decoration: line-through; }

```

Wskazówki

- Jeśli chcesz, możesz po spacji wpisać jeszcze jedną wartość (tabela 6.11). O ile pierwszą wartością nie jest `none`, możesz zdefiniować wszystkie pozostałe wartości naraz, oddzielając je spacjami:


```
overline underline line-through
```
- Koniec migających oczopląsów. W CSS 3 zrezygnowano z wartości `blink` własności `text-decoration`. Natomiast element HTML `blink` został usunięty już dawno. Teraz aby podenerwować użytkowników migającym tekstem, trzeba użyć Flasha.
- Ozdoby tekstu są stosowane do całości tekstu, a nie do pojedynczych liter. Dlatego w elemencie potomnym nie można zmienić tej własności, jeśli została ustawiona dla rodzica.
- Mimo iż element potomny nie może zmienić dekoracji zastosowanej do rodzica, można do niego dodać nową dekorację. Zwróć uwagę na to, że w przedstawionym przykładzie element `em` ma przekreślenie, które zostało dodane do podkreślenia zdefiniowanego dla akapitu.
- Przekreślanie to dobra metoda zaznaczania usuniętego tekstu. Ja na przykład stosowałem ją w katalogach internetowych zawierających ceny sprzedaży. Oryginalną cenę przekreślałem, a obok pokazywałem aktualną. Mimo to lepszy do tego celu jest element `del`, który dodaje informację semantyczną.
- Tak naprawdę własności `text-decoration` używam wyłącznie do wyłączania podkreślenia odnośników. Podkreślenie to jest odporne w formatowaniu — to po prostu zwykła kreska w kolorze tekstu. Trwają pewne prace nad umożliwieniem sprawowania nad nim lepszej kontroli, czyli dodaniem możliwości określania odstępu, koloru i stylu.

Czy podkreślać odnośniki

Deklaracja `text-decoration: none` dla elementu a oznacza brak podkreślenia odnośników bez względu na ustawienia przeglądarki. Z moich doświadczeń wynika, że wielu użytkowników oczekuje, iż odnośniki będą podkreślone. Mimo iż osobiście nie lubię podkreślać hiperłączy — na stronie powstaje bałagan, a poza tym w CSS są lepsze metody — jak tylko z tego zrezygnuję, od razu przychodzą do mnie e-maile z pretensjami.

Jednym z rozwiązań zastępczych dla zwykłego podkreślenia jest zastosowanie do elementów hiperłączy dolnej części obramowania. Własność `border-bottom` daje znacznie większe możliwości kontroli wyglądu podkreślenia — umożliwia nawet zmianę koloru. Więcej informacji znajduje się w rozdziale 14.

Wkrótce!

Choć na razie dodano do CSS 3 tylko kilka nowych własności do formatowania tekstu, trwają prace nad kilkoma kolejnymi.

Obrys tekstu — własność `text-outline` będzie umożliwiać poprowadzenie wokół glifów linii o dowolnie wybranym kolorze i grubości. W przeglądarce Safari zaimplementowano już podobną własność jako rozszerzenie o nazwie `-webkit-text-stroke` i niewykluczone, że tak będzie wyglądać ostateczna wersja, którą przyjmie W3C.

Zawijanie tekstu — aktualnie tekst na stronach internetowych można zawijać tylko na spacjach. Nie ma żadnych własności pozwalających kontrolować zastosowanie łączników i złamań wiersza. Lukę tę zapełni własność `text-wrap`.

Justowanie tekstu — mimo iż tekst można wyjustować, ustawiając własność `text-align` na `justify`, nie można tego wyrównania w żaden sposób kontrolować. Z tego powodu w tekście mogą wystąpić nieestetycznie wyglądające różnice odstępów między wyrazami. Własność `text-justify` będzie oferowała kilka opcji justowania, a własność `text-align-last` pozwoli na określenie wyrównania ostatniego wiersza justowanego tekstu.

Kerning — przypominam, że jest to niestandardowy odstęp między literami. Nowa własność `punctuation-trim` będzie umożliwiać kontrolowanie podstawowych parametrów kerningu.

Znaki przestankowe — jednym z najczęstszych problemów w typografii jest sytuacja, gdy na początku bloku tekstu znajduje się znak inny niż alfanumeryczny, np. cudzysłów. Własność `hanging-punctuation` będzie umożliwiała określenie, jak należy postępować z tymi znakami, czy zaliczać je do bloku tekstu, czy nie.

Ozdoby tekstu — do opisanej wcześniej własności `text-decoration` zostaną dodane nowe własności uzupełniające takie jak `text-line-color`, `text-line-style`, `text-line-skip` oraz `text-underline-position`.

Jeśli chcesz wiedzieć, jak idą prace nad tymi własnościami, zagładaj na stronę modułu tekstowego CSS 3 pod adresem www.w3.org/TR/css3-text.

Podsumowanie

- 1. Zdefiniuj odstępy między literami, słowami i wierszami tekstu w nagłówkach.** Wypróbuj różne opcje. Ściśnij tekst, a potem go rozluźnij.
- 2. Zdefiniuj odstęp między wierszami tekstu w treści głównej dokumentu.** Określ odstęp między wierszami w akapitach. Sprawdź, jaki wpływ ma zmniejszanie i zwiększanie tego parametru na czytelność tekstu. Od jakiego momentu zwiększanie odstępu międzywierszowego zaczyna pogarszać czytelność?
- 3. Ustaw wyświetlanie treści nagłówka wielkimi literami.** Możesz też spróbować wyróżnić nagłówki, modyfikując rozmiar pisma.
- 4. Dodaj do nagłówków cień.** Postaraj się, aby cienie były dyskretne i nie rozpraszały uwagi.
- 5. Wyjustuj treść dokumentu.** Zastosuj własność justowania dla akapitów. Czy tekst wygląda lepiej, czy gorzej niż w przypadku wyrównania do lewego marginesu, kiedy to linie tekstu są nierówne z prawej strony?
- 6. Zastosuj wcięcie akapitów.** Wypróbuj kilka różnych ustawień wcięcia, aby zobaczyć, które jest najlepsze.

A

absolute, 301, 304
 active, 96, 98, 101, 102
 after, 110, 111
 agregacja, 386
 align, 242
 alpha(), 314
 anchor, 33
 animacje, 292
 arkusz stylów
 dla określonych języków, 106
 dołączanie do dokumentu HTML, 60
 stron, 384
 importowanie, 62
 kaskadowy, 17
 organizacja
 alfabetyczna, 381
 oparta na strukturze stron, 381
 według typów selektorów, 381
 zgodnie z przeznaczeniem, 381
 śródliniowy, 277
 usterki, 388
 warunkowy dla Internet Explorera, 344
 article, 42, 43, 45
 artykuł, 43, 209
 aside, 41, 42, 43, 45
 atrybut, 33, 34
 attr(), 251
 auto, 126, 178, 237, 272

B

backface-visibility, 329
 background, 224
 background-attachment, 220, 226
 background-clip, 222
 background-color, 214
 background-origin, 223
 background-position, 220

background-repeat, 219, 225
 barwa, 199
 baseline, 182
 baza wiedzy, 26
 bazowe linie pisma, 172
 before, 110, 111
 bezpieczna paleta kolorów, 13
 bezpieczny font, 144
 bezszeryfowe fonty, 137
 biblioteka CSS, 383
 block, 264
 block-level, 36
 blokowy element, 36, 264
 Blueprint, 383
 body, 33, 44, 66, 250
 bold, 158
 bolder, 158
 border, 200, 222, 223, 279, 281, 282
 border-bottom, 190
 border-box, 222, 223
 border-collapse, 239
 border-image, 287
 border-radius, 284, 286
 border-spacing, 238
 bottom, 242, 304
 box-shadow, 316
 box-shadows, 332
 box-sizing, 292, 350
 br, 41, 274

C

canvas, 41
 capitalize, 174
 caption, 165, 242
 caption-side, 242
 Cascading Style Sheets, *Patrz* CSS
 center, 178
 cień
 class, 34, 70
 clear, 274
 elementu, 316

- cień
 - inset, 317
 - kolor, 317
 - przesunięcie, 316
 - rozmazanie, 316
 - rozproszenie, 317
 - tekstu, 176
 - clearfix, 357
 - close-quote, 251
 - collapse, 239
 - color, 212
 - dziedziczenie, 212
 - Color
 - Jack, 211
 - Palette Generator, 211
 - content, 110, 222, 223, 250, 252
 - w Internet Explorerze, 251
 - content-box, 222, 223
 - counter, 252
 - counter(), 251
 - counter-increment, 252
 - counter-reset, 252
 - CSS, 9, 17, 19, 24, 57
 - biblioteka, 383
 - deklaracja, 51
 - interpretacja przez przeglądarki, 21
 - ramy projektowe, 383
 - reguły, 25, 27
 - reset, 75, 351, 355
 - Erica Meyera, 354
 - prosty, 352
 - YUI2, 353
 - składnia reguły, 27
 - sprite, 370
 - Validator, 397
 - w Internet Explorerze
 - dostosowanie, 340–50
 - model pola, 349
 - sztuczka ze znakiem podkreślenia, 342
 - warunkowy arkusz stylów, 344
 - własności, 20
 - zasada działania, 20
 - zerowanie ustawień, 351
 - CSS 1, 23
 - CSS 2, 23
 - CSS 3, 17, 23, 29
 - CSS-reset, 351
 - cudzysłów, 254
 - currentcolor, 198
 - cursive, 137
 - cursor, 248
 - czcionki
 - internetowe, 30
 - rodzina, *Patrz* fonty rodzina
- ## D
- deklaracja, 27
 - konflikt, 129
 - ranga, 127
 - stylów, 66
 - typu dokumentu, 33
 - wymuszenie, 127
 - dialog, 42
 - dingbat, 138
 - display, 264, 266, 311
 - div, 70
 - dl, 42
 - DOCTYPE, 33, 43, 46
 - dodawanie treści
 - przed i za elementem, 110
 - za pomocą CSS, 250
 - dokument, 299
 - HTML, 43
 - tworzenie, 48
 - typ, 43, 46
 - dołączanie
 - arkusza stylów, 384
 - zewnętrznego pliku CSS, 60
 - dopełnienie, 262, 290
 - dynamiczna pseudoklasa, *Patrz* pseudoklasa dynamiczna
 - działania matematyczne, 182
 - dziecko, 86, 261
 - definicja stylu, 105
 - za pomocą pseudoklas, 104
 - dziedziczenie
 - auto, *Patrz* auto
 - inherit, *Patrz* inherit
 - none, *Patrz* none
 - normal, *Patrz* normal
 - stylów, 125
 - wymuszenie, 126
 - z przeglądarki, 34
- ## E
- edycja dokumentu, 41
 - efekt
 - przejścia, 332, 334
 - rollover, 370

- element, 32
 - blokowy, 36, 264
 - selektor, 36
 - dodanie treści przed i za, 110
 - dziecko, *Patrz* dziecko pływający, 273
 - naprawa, *Patrz* pływający element naprawa tworzenie układu wielokolumnowego, 362
 - wylączenie zawijania treści, 274
 - plywania uniemożliwienie, 275
 - potomny, *Patrz* potomek
 - przezroczystość, 314
 - rodzic, *Patrz* rodzic
 - siostrzany, 86, 92
 - poprzedzający, *Patrz* poprzedzający element siostrzany
 - przystający, *Patrz* przystający element siostrzany
 - sposób wyświetlania na stronie, 264
 - szerokość, 267
 - śródliniowy, 35, 180, 264
 - selektor, 35
 - widoczność, 310
 - wysokość, 268
 - zamiana na pływający, 273
 - em, 12, 34, 170
 - Emastic, 383
 - Embedded OpenType, 143
 - empty-cells, 241
 - encja znakowa, 139
 - EOT, 143
- F**
- fantasy, 138
 - figure, 42, 45
 - filter, 203, 314
 - Firebug, 392
 - first-child, 104
 - first-letter, 108
 - first-line, 108
 - first-of-type, 104
 - fixed, 220, 226, 237, 301, 304
 - float, 273, 274, 362
 - focus, 101, 102, 103
 - font, 162
 - Font Squirrel, 147
 - font-family, 140
 - font-size, 149
 - font-size-adjust, 152
 - font-style, 155
 - font-variant, 160
 - font-weight, 158
 - fonty, 162, 141
 - alternatywne, 141
 - dostosowanie rozmiaru tekstu, 152
 - bezpieczne, 142
 - definiowanie, 144
 - format pliku, 143
 - konwersja pliku przy użyciu Font Squirrel, 147
 - pobieranie z sieci, 143
 - rodzina, 135, 136, 140, 141
 - bezszerzefowe, 137
 - definiowanie, 140
 - fantasy, 138
 - kroje nieproporcjonalne, 137
 - kursywa, 137
 - o stałej szerokości znaków, 137
 - obrazkowe, 138
 - ozdoby drukarskie, 138
 - symboliczne, 138
 - szeryfowe, 136
 - sieciowe, 144
 - definiowanie, 144
 - stos, 140
 - technika Webfonts, 139, 143
 - typu
 - OpenType, 143
 - TrueType, 143
 - Webdings, 138
 - własności, 162
 - z sieci, 148
 - serwisy oferujące, 148
 - zastępcze, 141
- footer, 42, 43, 45
 - formatowanie
 - potomków, 87
 - stylu
 - całej witryny internetowej, 58
 - elementów w całym dokumencie, 54
 - pojedynczego elementu HTML, 51
 - wielu elementów, 78
 - formularz, 209
 - www, 41
 - frame, 41
 - frameset, 41
- G**
- glif, 135
 - gradient, 200
 - liniowy, 203
 - promienisty, 203
 - w Internet Explorerze, 203

gradient
 w Mozilli, 204
 w tle, 203
 w Webkit, 205
 grafika wektorowa, 34
 grubość pisma, 158
 bold, 158
 bolder, 158
 lighter, 158
 normal, 158
 numer, 158
 grupowanie, 78
 selektorów, 78

H

hack, 341
 hanging indent, 234
 hanging-punctuation, 192
 head, 33, 44
 header, 42, 43, 44
 height, 269
 hidden, 271, 310
 hide, 241
 hiperłącze, 96, 104, 190
 hover, 96, 98, 101, 102, 103
 href, 33, 98
 HSL, 199
 html, 44
 HTML, 5, 9, 17, 24, 31, 32, 38
 w Internet Explorerze, 46
 HTML5 Shiv, 46
 http, 13
 HyperText Markup Language, *Patrz* HTTP

I

icon, 165
 id, 34
 identyfikator, 26, 71
 deklaracje, 72
 reguły, 26
 zależny, 72
 if, 344
 iframe, 41, 299
 ilustracja, 45
 img, 41
 import, 59, 62, 384
 important, 127, 128
 importowanie arkuszy stylów, 62

indeks
 dolny, 182
 górny, 182
 inherit, 126, 178, 198, 238, 251, 265, 290, 304,
 305, 310
 inline, 35, 264
 style, 51
 inline-block, 265
 input, 103
 inset, 317
 inside, 234
 Inspektor, 392
 interfejs użytkownika, 245
 internet
 dla każdego, 26
 na wszystkim, 26
 Internet Explorer
 dostosowanie CSS, 340–50
 Irish Paul, 146
 italic, 155

J

jakość wydruku stron internetowych, 124
 jasność, 199
 JavaScript, 46
 łąka, 48
 jednostka
 bezwzględna, 12
 długości, 323
 względna, 12
 język
 arkuszy stylów, 19
 znaczników, 17, 19, 33
 justify, 178
 justowanie, 178, 191

K

kapitaliki, 160
 normal, 160
 small-caps, 160
 kaskadowość, 55, 129
 kaskadowy arkusz stylów, *Patrz* CSS
 kąty, 323
 kerning, 170, 192
 klasa, 25, 44
 niezależna, 68
 selektor, 67
 zależna, 67, 68

kod
 minimalizacja, 398
 sprawdzanie poprawności, 397
 kody znaków
 HTML, 423
 UTF, 423
 kolejność pionowa, 301
 kolor, 29, 118, 195
 asocjacje, 207
 bezpieczna paleta, 13
 cienia, 200, 317
 HSL, 199
 obramowania elementu, 200
 obrysu wokół elementu, 200
 pierwszego planu, 212
 RGB, 198
 schemat kolorystyczny, *Patrz* schemat kolorystyczny
 słowa kluczowe, 197
 tekstu, 212
 tła, 214
 wartość, 13
 alfa, 199
 koło barw, 210
 selektor kolorów, 211
 kombinacyjny selektor, 87
 komentarz, 82
 do arkusza stylów, 82
 konflikt deklaracji, 129
 kontekstowy selektor, 86
 kotwica, 96
 krawędź elementu, 300
 kroje nieproporcjonalne, 137
 Kuler, 211
 kursor zmiana wyglądu, 248
 kursywa, 41, 137, 155, 157

L

lang, 105
 lang(), 105
 large, 149
 larger, 149
 last-child(), 104
 last-of-type(), 104
 leading, 169, 172
 left, 178, 274, 304
 letter-spacing, 169
 li, 232
 liczby, 323
 licznik, 252

lighter, 158
 line-height, 172, 173
 line-through, 188
 linia nad tekstem, 188
 link, 60, 96, 98, 384
 lista, 209
 formatowanie, 229
 zbiorcze definiowanie własności, 235
 list-item, 232, 265
 list-style, 235
 list-style-image, 233
 list-style-position, 234
 list-style-type, 232, 235
 local, 220, 226
 lowercase, 174
 ltr, 44

Ł

łańcuch tekstu, 251
 łałka JavaScript, 48
 łącze, 59
 aktywny, *Patrz* active
 kursor nad elementem, *Patrz* hover
 nadanie stylu, 98
 nawigacyjne, 366
 normalny, *Patrz* link
 odwiedzony, *Patrz* visited

M

mapa bitowa, 41
 margin, 238, 276
 margin-bottom, 277
 margines, 263, 276, 278
 scalanie, 278
 ujemny, 276
 margin-left, 277
 margin-right, 277
 margin-top, 277
 matrix(), 326
 matrix3d(), 331
 max-height, 269
 max-width, 269
 mechanizm renderowania, 20
 media, 30, 116, 121
 medium, 149
 menu, 165
 rozwijane, 373
 message-box, 165
 meta, 135

Meyer Eric, 354
 min-height, 269
 minimalizacja kodu, 398
 min-width, 269
 monospace, 137

N

nagłówek, 33, 42, 43, 44, 209
 sekcji, 380
 narzędzia kolorystyczne, 211
 nasycenie, 199
 nav, 42, 43, 44
 nawigacja, 43, 209, 366
 no-close-quote, 251
 noframes, 41
 none, 126, 152, 174, 188, 265
 no-open-quote, 251
 no-repeat, 219, 226
 normal, 126, 155, 158, 160, 186, 251
 not, 107
 nowrap, 185, 186
 nth-child(), 104
 nth-last-of-type(), 104
 nth-of-type(), 104

O

obcinanie
 tekstu, 272
 widocznego obszaru elementu, 312
 oblique, 155
 obramowanie, 29, 209, 262, 279, 281, 283
 obraz w tle, 287
 obraz
 tła, 216–22
 ustawianie, 223
 w tle obramowania elementu, 287
 obrazkowe fonty, 138
 obrys, 263
 elementu, 279
 tekstu, 191
 obszar
 renderowania, 300
 rzeczywisty, 300
 widoku, 299
 odnośnik, 209, 366
 odstęp
 kontrolowanie, 185
 międzyliterowy, 169
 tracking, 170

 międzywierszowy, 169, 172, 173
 między słowami, 169, 171
 kerning, 170
 odtwarzanie czasowe multimediiów, 41
 ograniczniki, 209
 okno, 299
 przeglądarki, 299
 ol, 252
 one, 251
 opacity, 199, 314
 open-quote, 251
 OpenType, 143
 opływanie, 274
 opóźnienie, 335
 orientacja, 118
 origin(), 327
 OTF, 143
 outline, 200, 279, 280
 outside, 234
 overflow, 269, 270, 292, 358
 overflow-x, 270
 overflow-y, 270
 overline, 188
 ozdoba
 tekstu, 192
 drukarska, 138

P

padding, 222, 223, 290
 padding-box, 222, 223
 paleta kolorów z obrazów i zdjęć, 211
 pasek boczny, 45
 perspective, 328, 329
 perspective(), 329
 perspective-origin, 328
 pewność, 26
 picture-font, 138
 pierwsza litera formatowanie, 108
 pierwszy wiersz formatowanie, 108
 pismo pochylone, 155
 plik
 arkuszy stylów, 57
 fontów, 143
 konwersja z Font Squirrel, 147
 zewnętrzny, 57
 pływający element, 273
 naprawa, 356
 br, 357
 overflow, 358
 sztuczka ze złamaniem wiersza, 356
 tworzenie układu wielokolumnowego, 362

- plywanie elementu uniemożliwienie, 275
 - podkreślenie, 188
 - pogrubienie, 41
 - pole, 29, 257, 262
 - dopełnienie, 262
 - element potomny, 262
 - margines, 263
 - obramowanie, 262
 - obrys, 263
 - przepełnienie, 263
 - szerokość, 262
 - tło, 262
 - treść, 262
 - wysokość, 262
 - położenie elementu
 - na osi pionowej, 308
 - określenie techniką pozycjonowania, 306
 - poprawność kodu, 397
 - poprzedzający element siostrzany, 86
 - position, 301, 305
 - potomek, 86, 262
 - definicja stylu, 88
 - selektor uniwersalny, 89
 - formatowanie, 87
 - poufność, 26
 - powielenie, 287
 - pozycjonowanie, 295, 301
 - bezwzględne, 301, 302
 - dziedziczenie ustawień, 305
 - kolejność pionowa, 301
 - określenie położenia elementu, 306
 - położenie elementu na osi pionowej, 308
 - stałe, 301, 303
 - statyczne, 301
 - ucinięcie, 301
 - widoczność, 301
 - wyбір rodzaju, 303
 - względne, 301, 302
 - pre, 185, 186
 - precyzja, 131
 - przeciągnij i upuść, 41
 - przejścia efekt, 332
 - przejście, 29
 - uniwersalne, 336
 - przekreślenie, 188
 - przekształcenia, 323
 - dwuwymiarowe, 324, 325
 - jednostki długości, 323
 - kąty, 323
 - liczby, 323
 - trójwymiarowe, 327, 328
 - przepełnienie elementu, 263, 270
 - przesunięcie, 316
 - przezroczystość, 30, 198, 314, 315
 - przystający element siostrzany, 86
 - pseudoelement, 104, 108, 111
 - after, *Patrz* after
 - before, *Patrz* before
 - first-letter, *Patrz* first-letter
 - first-line, *Patrz* first-line
 - pseudoklasa, 96
 - dynamiczna, 96
 - definiowanie, 102
 - first-of-type, *Patrz* first-of-type
 - focus, *Patrz* focus
 - hover, *Patrz* hover
 - kolejność definicji, 103
 - lang(), *Patrz* lang()
 - link, *Patrz* link
 - łącze, 96
 - strukturalna, 96
 - visited, *Patrz* visited
 - punctuation-trim, 192
 - punktor
 - korygowanie położenia, 234
 - styl, 232
 - tworzenie własnego, 233
 - ustawianie, 232
- ## Q
- quirks mode, 46
 - quotes, 254
- ## R
- ramy projektowe CSS, 383
 - ranga deklaracji, 127
 - rect, 312
 - redefiniowanie elementu HTML, 64, 66
 - reguła, 27, 50
 - CSS, 25
 - identyfikatora, 26
 - uzupełniająca, 79
 - relative, 301, 304
 - renderowanie, 20
 - repeat, 219, 225, 289
 - repeat-x, 219, 225
 - repeat-y, 219, 226
 - reset CSS, 75, 351, 355
 - Erica Meyera, 354
 - prosty, 352
 - YUI2, 353

- resize, 292
 - RGB wartości
 - dziesiętne, 198
 - procentowe, 199
 - szesnastkowe, 198
 - right, 178, 274, 304
 - rodzic, 86, 261
 - rodzina
 - czcionek, *Patrz* fonty rodzina
 - fontów, *Patrz* fonty rodzina
 - rogi
 - w kształcie elipsy, 285
 - zaokrąglone, 284
 - rollover, 370
 - rotate(), 326
 - rotate3D, 329
 - rotate3d(), 330
 - rotateX(), 326
 - rotateY(), 326
 - rotateZ, 329
 - rotateZ(), 330
 - round, 289
 - rozciągnięcie, 287
 - rozdzielczość, 118
 - rozmazanie, 316
 - rozmiar tekstu, 149, 151
 - bezwzględny, 149
 - dostosowanie fontów alternatywnych, 152
 - none, 152
 - wartość procentowa, 151
 - względny, 149
 - rozproszenie, 317
 - rozszerzenie projektowe, 30
 - run-in, 265
- S**
- sans-serif, 137
 - Scalable Vector Graphics, 144, *Patrz* SVG
 - scale(), 326
 - scale3d(), 331
 - scaleX(), 326
 - scaleY(), 326
 - scaleZ(), 331
 - schemat kolorystyczny
 - analogiczny, 211
 - czterokolorowy, 210
 - komplementarny, 210
 - monochromatyczny, 210
 - trójkolorowy, 210
 - scroll, 220, 226, 272
 - section, 42, 45
 - sekcja, 45
 - selektor, 25, 27, 50
 - dziecka, 90
 - elementu
 - blokowego, 36
 - śródliniowego, 35
 - siostrzanego, 92, 94
 - identyfikatora, 71
 - klasy, 67
 - kolorów z koła barw, 211
 - kombinacyjny, 87
 - kontekstowy, 86
 - negacji, 107
 - typu, 64
 - uniwersalny, 75, 94
 - definiowanie stylu potomka, 89
 - wyłączający stylizację wybranych elementów, 107
 - selektywna stylizacja, 85
 - separate, 239
 - serif, 136
 - SGML, 33
 - show, 241
 - siostrzany element, 86
 - poprzedzający, *Patrz* poprzedzający
 - element siostrzany
 - przystający, *Patrz* przystający element siostrzany
 - skew(), 326
 - skewX(), 326
 - skewY(), 326
 - small, 149
 - small-caps, 160
 - small-caption, 165
 - smaller, 149
 - spacja, 186
 - span, 70
 - sprite'y CSS, 370
 - src, 28
 - stan łącza, 96
 - Standard Generalized Markup Language, *Patrz* SGML
 - static, 301, 304
 - status-bar, 165
 - stopka, 42, 43, 45
 - stos fontów, 140, 144
 - stretch, 289
 - strona, 299
 - dołączanie arkusza stylów, 384
 - sposób wyświetlania elementu, 264
 - tło, 208
 - układ wielokolumnowy, 182

strong, 64
 struktura dokumentu, 32
 styl, 18
 definiowanie
 z wykorzystaniem reguły @media, 122
 przy użyciu zapytań o media, 119
 w zależności od atrybutów, 112–15
 dla druku, 124
 dziedziczenie, 125
 z przeglądarki, 34
 element w określonym języku definiowanie, 106
 formatowanie pojedynczego elementu, 51
 gazetowy, 178
 kaskadowy arkusz, *Patrz* CSS
 kontekstowy, 87
 osadzony, 54
 pisma użytkownika imitowanie, 165
 punktora, 232
 śródliniowy, 51, 53
 tekstu, 155
 italic, 155
 normal, 155
 oblique, 155
 uniwersalny, 75
 w określonym języku, 105
 wyłączenie dla wybranego elementu, 107
 zasady stosowania, 129
 zewnątrzny, 54, 57, 62
 style, 34, 51, 54, 67
 sub, 182
 sup, 182
 super, 182
 SVG, 34, 144
 symbol font, 138
 symboliczne fonty, 138
 szerokość, 118, 262
 elementu, 267
 maksymalna, 269
 minimalna, 269
 szeryf, 136
 sztuczka ze znakiem podkreślenia, 342

Ś

śródliniowy element, 35, 180, 264
 światło
 między dowolną parą liter, 170
 międzyliterowe, 170
 międzywyrazowe, 171

T

tabela, 209
 auto, 236
 fixed, 236
 formatowanie, 229
 odstęp między komórkami, 238
 scalanie obramowań komórek, 239
 układ, 236
 automatyczny, 236
 stały, 236
 ukrywanie pustych komórek, 241
 ustawienie położenia podpisu, 242
 table, 265
 table-layout, 236
 td, 239
 tekst, 29
 cienie, 176
 dodawanie za pomocą CSS, 250
 grubość, 158
 justowanie, 191
 kerning, 192
 kolor, 212
 kursywa, 155
 łańcuch, 251
 obcinanie, 272
 obrys, 191
 odstępy między
 liniami, 169, 172
 literami, 169
 słowami, 169, 171
 ozdoba, 192
 pochylony, 155
 rozmiar, 149
 wyrównanie
 w pionie, 180
 w poziomie, 178
 zawijanie, 191
 zbiórcze formatowanie, 162
 zdobienie, 188
 linia nad, 188
 podkreślenie, 188
 przekreślenie, 188
 zmiana wielkości liter, 174
 znak przestankowy, 192
 text/css, 54
 text-align, 178
 text-align-last, 191
 text-decoration, 188, 190, 192
 text-indent, 37, 183
 text-justify, 191

text-line-color, 192
 text-line-skip, 192
 text-line-style, 192
 text-outline, 191
 text-overflow, 292
 text-shadow, 176, 200
 text-top, 182
 text-transform, 174
 text-underline-position, 192
 text-wrap, 191
 tło, 29, 262

- kolor, 214
- obramowania elementu, 287
 - wstawienie obrazu, 287
- obraz, 216–22
 - ustawianie, 223
- strony, 208
- treści, 209
- zbiorcze definiowanie własności, 227

 top, 242
 tracking, 169, 170
 transform, 323
 transformacja, 29
 transform-origin, 328
 transition, 334
 transitional, 350
 transition-delay, 335
 transition-duration, 335
 transition-property, 334
 transition-timing-function, 335
 translate(), 326
 translate3d(), 331
 translateX(), 326
 translateY(), 326
 translateZ(), 331
 transparent, 198
 treść, 30, 262

- generowana, 250
- strony, 33

 TrueType, 143
 tryb

- osobliwości, 46, 340
 - sztuczka ze znakiem podkreślenia, 342
- prawie standardowy, 340
- ścisły, 340

 TTF, 143
 tworzenie palety, 211
 typ dokumentu, 43, 46

- deklaracja, 33

 typografia, 133
 Typogridphy, 383

U

ucinanie, 301
 układ

- normalny, 300
- wielokolumnowy, 292, 362

 underline, 188
 Unicode Transformation Format 8 bit, *Patrz* UTF
 Uniform Resource Locator, *Patrz* URL
 uppercase, 174
 URL, 13
 url(), 251
 UTF, 423
 UTF-8, 44, 136
 uwaga na marginesie, 43, 45

V

vertical-align, 180
 viewport, 299
 visibility, 310, 311
 visible, 271, 310
 visited, 96, 98, 101

W

W3C, 17
 wartość, 27

- alfa, 199
- koloru, 13

 wcięcie, 183
 Web Hypertext Application Technology
 Working Group, *Patrz* WHATWG, 39
 Web Open Font Format, *Patrz* WOFF
 Webdings, 138
 Webfonts, 133, 139, 143
 webkit-text-stroke, 191
 WHATWG, 39
 white-space, 185, 186
 widoczność elementu, 301, 310
 width, 269
 wielkość liter, 174
 wielokolumnowy układ, 362
 własność, 27
 WOFF, 144
 word-spacing, 171
 World Wide Web, *Patrz* www
 Consortium, *Patrz* W3C
 współczynnik kształtu obrazu, 116
 www, 26

wyrównanie tekstu
 w pionie, 180
 w poziomie, 178
 wyróżnienie
 pierwszego wiersza, 108
 pierwszej litery, 108
 wysokość, 118, 262
 maksymalna, 269
 minimalna, 269
 wysokość elementu, 268
 wysunięcie, 234
 wyświetlanie elementu na stronie, 264

X

XHTML, 38
 x-large, 149
 x-small, 149
 xx-large, 149
 xx-small, 149

Y

YUI Grids, 383

Z

zagnieżdżony element, 86
 zaokrąglenie rogów, 284
 zapytania o media, 116
 definicja stylów, 119
 zawijanie tekstu, 191, 234
 zdarzenie interaktywne
 stylizacja, 102
 zerowanie ustawień CSS, 351
 zewnętrzny
 arkusz stylów, *Patrz* arkusz stylów
 plik, 57
 z-index, 308
 złamanie wiersza, 185
 sztuczka, 356
 znacznik, 25, 32, 257
 znak przestankowy, 192

Szybki start ▶ CSS3

Poznaj i wykorzystaj możliwości nowych standardów CSS3!

Gdy webmasterzy wspominają czasy przed powstaniem kaskadowych arkuszy stylów, dostają gęsiej skórki. Struktura dokumentu przemieszana z opisem wyglądu oraz dziesiątki niestandardowych znaczników inaczej obsługiwanych przez każdą przeglądarkę to koszmar każdego ówczesnego projektanta stron internetowych. Idea stylów CSS stała się wybawieniem – okazało się nagle, że proste rozwiązania większości problemów związanych z opisem wyglądu strony WWW są na wyciągnięcie ręki!

Najnowsza wersja CSS wprowadza wiele udogodnień, ułatwień i nowych możliwości. Wszystkie potrzebne informacje o wersji CSS3 znajdziesz właśnie w tej książce. W trakcie lektury poznasz historię kaskadowych arkuszy stylów, typy reguł oraz zasady ich tworzenia. Ponadto odkryjesz nowe możliwości języka HTML5 i przekonasz się, że korzystanie z nich nie musi być trudne! Autor szczegółowo omawia sposoby selektywnej stylizacji, formatowania tekstów, wybierania kolorów oraz dostosowywania wyglądu tabel. Twoją szczególną ciekawość na pewno wzbudzi rozdział przedstawiający trzydzieści dwie najlepsze praktyki CSS. Dzięki nim Twoje arkusze będą jeszcze lepsze. Ta książka musi znaleźć się na półce każdego webmastera chcącego podążać za światowymi trendami!

- Historia CSS i budowa reguł CSS
- Nowości w CSS3 i format SVG
- Nowości w HTML5
- Selektory i importowanie arkusza stylów
- Definiowanie klas i selektywna stylizacja
- Formatowanie potomków
- Przygotowywanie stylów dla różnych mediów
- Typografia w sieci
- Bezpieczne czcionki dla stron WWW
- Własności tekstu i definiowanie kolorów
- Formatowanie list i tabel
- Marginesy, zaokrąglone rogi, elementy pływające
- Ustawianie przezroczystości elementów
- Przekształcenia i efekty przejścia
- Zerowanie ustawień CSS i zarządzanie arkuszami
- Najlepsze praktyki

helion.pl
księgarnia
internetowa

Nr katalogowy: 7224

Księgarnia internetowa:
<http://helion.pl>

Zamówienia telefoniczne:
0 801 339900
0 601 339900



Helion

Sprawdź najnowsze promocje:

📍 <http://helion.pl/promocje>

📖 Książki najchętniej czytane:

📍 <http://helion.pl/bestsellery>

📖 Zamów informacje o nowościach:

📍 <http://helion.pl/novosci>

Helion SA

ul. Kosciuszki 1c, 44-100 Gliwice

tel.: 32 230 98 83

e-mail: helion@helion.pl

<http://helion.pl>

sięgnij po WIĘCEJ



KOD KORZYŚCI

ISBN 978-83-246-3386-9



Cena 69,00 zł

Informatyka w najlepszym wydaniu