

Przydatne strony

<https://www.w3.org/TR/2017/NOTE-css-2017-20170131/#w3c-process>

<https://developer.mozilla.org/pl/docs/Web/CSS>

<http://www.htmldog.com/examples>

<http://littlewebhut.com/>

Schemat konstrukcji stylu

```
Selektor {  
    własność1: wartość1;  
    własność2: wartość2;  
}
```

Zamiast słowa *selektor* podstawiamy znacznik z języka HTML, np.: znacznik akapitu <p>.

```
p {  
    color: green;  
    font-size: 14px;  
}
```

Oczywiście można zapisać również tak:

```
p {color: green; font-size: 14px;}
```

Miary absolutne

- in - cal, (1in = 2.54cm)
- cm - centymetr, 1cm
- mm - milimetr, 1mm
- pt - punkt, 1pt = 1/72in

CSS-miary

Relacja między bezwzględными jednostkami wygląda następująco:

$$1 \text{ cal} = 2,54 \text{ cm} = 25,4 \text{ mm} = 72 \text{ pt} = 6 \text{ pc}$$

Miary względne

Jednostka **px** nie jest zdefiniowana jako stała długość, ale zależy od typu urządzenia oraz jego zastosowania.

- em - rozmiar czcionki
- ex - x-height - wysokość litery x
- px - piksele, w odniesieniu do rozdzielczości tła
- % - procent szerokości/wysokości elementu ograniczającego

	Zalecane	Sporadyczne użycie	Niezalecane
Ekran	em, px, %	ex	pt, cm, mm, in, pc
Druk	em, cm, mm, in, pt, pc, %	px, ex	

JESZCZE WIĘCEJ JEDNOSTEK

Aby ułatwić pisanie stylów zależnych wyłącznie od domyślnego rozmiaru czcionki standard CSS w 2013 roku wprowadził nową jednostkę: **rem**. Jest ona zdefiniowana jako wielkość tzw. bazowej czcionki dokumentu (stąd nazwa, bazowy **em** od angielskiego *root em*). W przeciwieństwie do **em**, który może być inny dla każdego elementu, **rem** jest niezmienny w całym dokumencie. Na przykład, aby nadać elementom P i H1 taki sam lewy margines w stylach przed 2013 rokiem należało napisać:

Nowe jednostki **vw i **vh** pozwalają zapisać reguły zależne od rozmiaru okna użytkownika. Pierwsza stanowi jedną setną szerokości okna, natomiast druga jedną setną wysokości. Dodano także jednostkę **vmin**, odpowiadającą mniejszej z wartości **vw** i **vh** oraz **vmax** odpowiadającą większej z wartości **vw** i **vh****

CSS-miary

Wielkość bieżącego kroju jest dostępna poprzez jednostki **em** oraz **ex**.

em jest szerokością litery m w danym kroju, zaś **ex** – wysokością litery

x. Wykorzystując te jednostki możesz pisać:

```
font-size : 1.2em;  
font-size : 0.8ex;
```

Jednostki **em** oraz **ex** pozwalają na definiowanie pudełek, których

wymiary zależą od obszaru zajmowanego przez tekst:

```
width : 8em;  
height : 4ex;
```

Właściwości CSS- czcionka

Cecha	Opis	Możliwe wartości
font-family	deklaracja kroju czcionki	główne kroje czcionek: Arial, Verdana, Georgia, Times New Roman, Courier, New Helvetica, Tahoma. Główne rodziny czcionek: serif, Sans-serif
font-size	deklaracja rozmiaru czcionki	wysokość podawana w wybranej jednostce (np. 1 cm, 10 pt, 15 px, 70%) lub jako predefiniowana wartość: xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger
font-style	deklaracja stylu czcionki	normal, italic, oblique
font-variant	deklaracja typu czcionki	normal, small-caps
font-weight	deklaracja grubości czcionki	lighter, normal, bold, bolder, 100, 200, 300, 400,.....
font	Skrócona forma deklarowania wszystkich własności czcionki w jednym miejscu	Oddzielone spacjami wartości w następującej kolejności: style, variant, weight, size, family

Default fonts

CSS defines five generic names for fonts: serif, sans-serif, monospace, cursive and fantasy.

Term	Definition	Example
<code>serif</code>	Fonts that have serifs (the flourishes and other small details you see at the ends of the strokes in some typefaces)	My big red elephant
<code>sans-serif</code>	Fonts that don't have serifs.	My big red elephant
<code>monospace</code>	Fonts where every character has the same width, typically used in code listings.	My big red elephant
<code>cursive</code>	Fonts that are intended to emulate handwriting, with flowing, connected strokes.	My big red elephant
<code>fantasy</code>	Fonts that are intended to be decorative.	My big red elephant

Przykład- czcionka

**Wprowadź następujący styl dla akapitu p:
krój czcionki – Courier New, styl –
pochylenie, pogrubiona , rozmiar – 20
pixeli**

Właściwości CSS- tekst

Cecha	Opis	Możliwe wartości
color	deklaracja koloru tekstu	nazwy kolorów RGB lub ich oznaczenia szesnastkowe
line-height	deklaracja odstępu między liniami	wysokość podawana w wybranej jednostce (np. 1 cm, 10 pt, 15 px, 70%)
letter-spacing	deklaracja odstępu między znakami tekstu	odstęp podawany w wybranej jednostce
text-align	deklaracja poziomego wyrównania	left, right, center, justify
text-decoration	deklaracja wyróżniania lub deklaracja czcionki	wave, underline, overline, line-through, blink
text-indent	deklaracja wcięcia 1. wiersza tekstu	długość podawana w wybranej jednostce
text-transform	deklaracja zmiany wielkości liter w tekście	none, capitalize, uppercase, lowercase
word-spacing	deklaracja odległości między wyrazami w tekście	długość podawana w wybranej jednostce

Przykład- tekst

Wprowadź następujący styl dla nagłówka h1: dekoracja tekstu – nadkreślenie, odstępy między wyrazami – 10 px, wyśrodkowanie, kolor tekstu – zielony

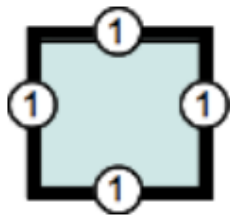
Właściwości CSS- tło

Cecha	Opis	Możliwe wartości
background-attachment	definiuje, czy obrazek tła przewija się razem ze stroną, czy jest nieruchomy	fixed, scroll
background-color	deklaracja koloru tła	nazwa koloru RGB lub zapis szesnastkowy
background-image	deklaracja obrazka jako tła strony	adres URL
background-position	deklaracja pozycji obrazka tła	współrzędne podawane w wybranych jednostkach (np. 10px 30px), procentach lub wartości predefiniowane: topleft, right, bottom, center
background-repeat	deklaracja sposobu powtarzania obrazka w tle strony	repeat, repeat-x, repeat-y, no-repeat
background	skrótowa forma deklarowania wszystkich właściwości tła w jednym miejscu	oddzielone spacjami wartości w następującej kolejności: color, image, repeat, attachment, position

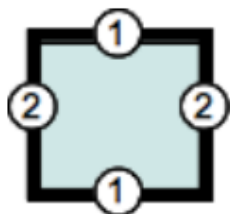
Właściwości CSS- obramowania

Cecha	Opis	Możliwe wartości
border-color border-top-color border-left-color border-right-color border-bottom-color	deklaracja koloru wszystkich krawędzi lub wybranej krawędzi	nazwa koloru RGB lub zapis szesnastkowy
border-style	deklaracja stylu wszystkich krawędzi lub wybranej krawędzi	none, hidden, dotted, dashed, solid, double, groove, ridge, inset, outset
border-width	deklaracja grubości wszystkich/wybranych krawędzi	wartość określona w pikselach lub wartość predefiniowana: thin, medium, thick
border	skrótowa forma deklarowania wszystkich właściwości obramowania w jednym miejscu lub wybranej krawędzi	oddzielone spacjami wartości w następującej kolejności: width, style, color

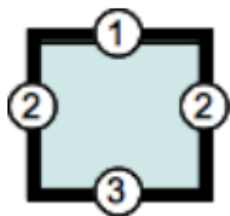
Shorthands handling properties related to edges of a box, like `border-style`, `margin` or `padding`, always use a consistent 1-to-4-value syntax representing those edges:



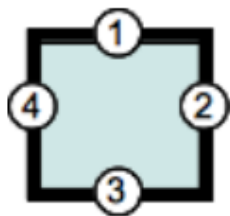
The 1-value syntax: `border-width: 1em` — The unique value represents all edges



The 2-value syntax: `border-width: 1em 2em` — The first value represents the vertical, that is top and bottom, edges, the second the horizontal ones, that is the left and right ones.

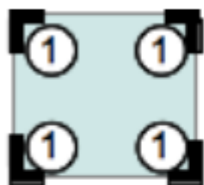


The 3-value syntax: `border-width: 1em 2em 3em` — The first value represents the top edge, the second, the horizontal, that is left and right, ones, and the third value the bottom edge

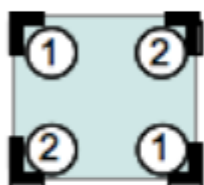


The 4-value syntax: `border-width: 1em 2em 3em 4em` — The four values represent the top, right, bottom and left edges respectively, always in that order, that is clock-wise starting at the top (The initial letter of Top-Right-Bottom-Left matches the order of the consonant of the word *trouble*: TRBL)

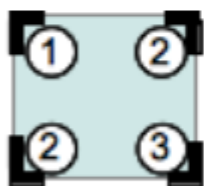
Similarly, shorthands handling properties related to corners of a box, like `border-radius`, always use a consistent 1-to-4-value syntax representing those corners:



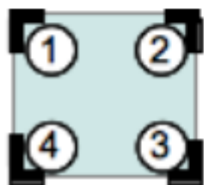
The 1-value syntax: `border-radius: 1em` — The unique value represents all corners



The 2-value syntax: `border-radius: 1em 2em` — The first value represents the top left and bottom right corner, the second the top right and bottom left ones.

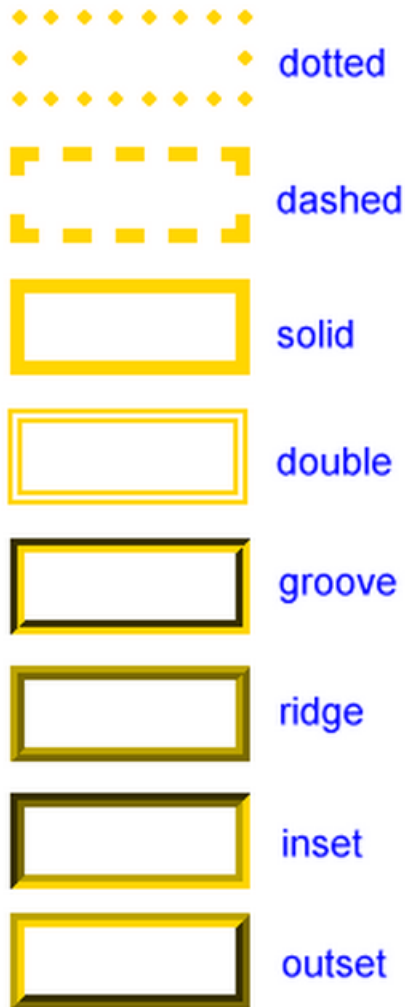


The 3-value syntax: `border-radius: 1em 2em 3em` — The first value represents the top left corner, the second the top right and bottom left ones, and the third value the bottom right corner



The 4-value syntax: `border-radius: 1em 2em 3em 4em` — The four values represent the top left, top right, bottom right and bottom left corners respectively, always in that order, that is clock-wise starting at the top left.

Przykład- obramowanie



**Wprowadź następujący styl
obramowania dla listy
wypunktowanej: styl
obramowania – kropkowane, kolor
– czerwony, grubość – średnia.**

Właściwości CSS- rozmiary

Cecha	Opis	Możliwe wartości
height	deklaracja wysokości elementu	wartość podawana w wybranej jednostce (np. 1 cm, 10 pt, 15 px, 70%)
width	deklaracja szerokości elementu	

Właściwości CSS- marginesy

Cecha	Opis	Możliwe wartości
margin-top margin-left margin-right margin-bottom	deklaracja górnego, lewego, prawego i dolnego marginesu	odległość podawana w wybranej jednostce (np. 1 cm, 10 pt, 15 px, 70%)
margin	skrótowa forma deklarowania wszystkich marginesów jednocześnie	wartości oddzielone spacjami w następującej kolejności: top, right, bottom, left; podanie 1 wartości spowoduje zmianę wszystkich marginesów; podanie 2 ustawi odpowiednio top i bottom, a potem right i left; podanie 3 ustawi marginesy top, right, bottom

Właściwości CSS- linki(odsyłacze)

Selektor:pseudoklasa	Opis
a:link	Deklaracja odsyłacza na stronie
a:visited	Deklaracja linku odwiedzonego
a:hover	Deklaracja linku po najechaniu myszką
a:active	Deklaracja linku przy kliknięciu

Przykład- linki

Wprowadź następujący styl dla odsyłaczy:

**Odsyłacz zwykły i odwiedzony:
kolor – czarny, dekoracja – brak,
tło – szare, obramowanie – linia
ciągła. Odsyłacz aktywny i przy
najechaniu myszką: kolor –
srebrny, dekoracja – brak, tło –
czarne, obramowanie – linia
kreskowana, zmiana wyglądu
kursora na krzyżyk.**

Właściwości CSS- listy

Cecha	Opis	Możliwe wartości
list-style-type	Rodzaj markera w listach wyliniowanych	disc, circle, square, decimal, lower-roman, upper-roman, lower-alpha, upper-alpha, lower-greek, lower-latin, upper-latin, hebrew, armenian, georgian, cjk-ideographic, hiragana, ...
list-style-position	Pozycja znacznika listy wypunktowanej	inside, outside
list-style-image	Deklaracja obrazka zastępującego marker w listach	Adres URL
list-style	Skrócona forma deklarowania wszystkich właściwości list	Oddzielone spacjami wartości w kolejności: type, position, image

Właściwości CSS- kursor

Cecha	Opis	Możliwe wartości
cursor	Typ kursora	Adres URL pliku z kursorem lub właściwość <u>predefiniowana</u> : <u>auto</u> , <u>crosshair</u> , <u>default</u> , <u>pointer</u> , <u>move</u> , <u>e-resize</u> , <u>ne-resize</u> ...

Właściwości CSS- marginesy

Cecha	Opis	Możliwe wartości
margin-top margin-left margin-right margin-bottom	górny lewy prawy dolny margines	Odległość od krawędzi strony podawana w wybranej jednostce: cm, pt, px, %
margin	Forma skrócona deklarowania wszystkich marginesów jednocześnie	Wartości oddzielone spacjami Podanie jednej wartości spowoduje zmianę dla wszystkich marginesów; Podanie dwóch wartości spowoduje odpowiednio zmianę dla górnego i dolnego marginesu, a następnie dla lewego i prawego; <u>Trzy wartości to zmiana dla marginesu górnego, prawego i lewego, dolnego (trzecia wartość)</u>

Zasady ogólne

1. **Twórz uniwersalne arkusze stylów — tak aby można było z nich korzystać w różnych dokumentach.**
2. **Pamiętaj o skalowalności dokumentów — ponieważ dokumenty będą wyświetlane w dynamicznych środowiskach i na różnym sprzęcie, szczególnie istotne jest wybranie odpowiednich jednostek miary — jednostki względne powinny mieć przewagę nad absolutnymi (dotyczy to na przykład szerokości marginesów, wielkości czcionki itp.)**
3. **Licz się z czytelnikiem — niektórzy czytelnicy będą korzystali z własnych ustawień. Twoje arkusze stylów powinny to brać pod uwagę. W jaki sposób? Odpowiednie definicje stylów powinny być umieszczane we właściwych miejscach. Na przykład te, które są stosowane do całego dokumentu, umieszczaj w sekcji <body> — i tylko tam. W ten sposób czytelnik bez trudu wprowadzi odpowiednie zmiany.**
4. **Testuj arkusze stylów — testuj je w kilku przeglądarkach, aby nie sprawić swoim czytelnikom niemiłej niespodzianki. Jeśli korzystasz z właściwości CSS, która nie jest obsługiwana powszechnie, upewnij się, że strona wygląda względnie poprawnie nawet w mniej zaawansowanych przeglądarkach.**

Zasady ogólne

5. *Uwzględniaj struktury dokumentu* — arkusze stylów umożliwiają oddzielenie wyglądu strony od jej struktury logicznej. Zawsze zaczynaj od zapisania kodu strony standardowymi znacznikami nagłówek, akapitów tekstu i tabel, a dopiero później nakładaj na gotowy dokument style modyfikujące jego wygląd.
6. *Dostarczaj nazwy rodzin czcionek* — pamiętaj, że dokument będzie wyświetlany na różnych komputerach, z różnym „wyposażeniem” w czcionki. Podanie nazwy ich rodziny pozwala na zachowanie charakteru strony. Unikaj też „egzotycznych” krojów pisma, jest bowiem wysoce prawdopodobne, że może ich brakować. Najlepiej ograniczać się do czcionek dostępnych standardowo w systemach operacyjnych Windows i Linux.
7. *Zachowaj umiar w stosowaniu ujemnych marginesów* — ujemne marginesy pozwalają uzyskać ciekawe efekty, ale nie wszystkie przeglądarki radzą sobie z nimi. Jeśli stosujesz takie marginesy, testuj stronę w różnych przeglądarkach.
8. *Upraszczaj formy* — konstruując arkusz stylów, zachowaj umiar. Możesz użyć wielu różnych krojów pisma i wypełnić stronę kolorami, lecz nie wpłynie to pozytywnie na przekaz informacji.