

IDŹ DO

PRZYKŁADOWY ROZDZIAŁ



SPIS TREŚCI

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

CSS. Leksykon kieszonkowy

Autor: Eric A. Meyer

ISBN: 978-83-246-1090-7

Tytuł oryginału: [CSS Pocket Reference](#)

Format: B6, stron: 120



Podręczne źródło informacji o najnowszych specyfikacjach kaskadowych arkuszy stylów – CSS2 i CSS2.1

- Chcesz nauczyć się dołączać arkusze stylów do witryn internetowych?
- Chcesz dowiedzieć się, jak nadać styl różnym elementom stron internetowych?
- Chcesz mieć szybki dostęp do opisu wszystkich selektorów i właściwości stylów?

Pierwszą rzeczą, jaką widzą osoby otwierające strony internetowe, jest ich wygląd. Wiele witryn ma niespójny i nieelegancki styl, co odstrasza użytkowników, a przecież utworzenie atrakcyjnych stron jest całkiem proste – wystarczy przygotować odpowiedni plik CSS. Za pomocą kaskadowych arkuszy stylów można wygodnie zmieniać sposób wyświetlania wszystkich elementów kodu HTML i sterować wyglądem całej witryny. Technologia CSS jest obsługiwana przez wszystkie najważniejsze przeglądarki, a ponadto nie wymaga przesyłania dużych ilości danych, dlatego jest doskonałym narzędziem do szybkiego i efektywnego formatowania witryn internetowych.

Książka „CSS. Leksykon kieszonkowy” to zwięzły przewodnik po możliwościach kaskadowych arkuszy stylów. Dowiesz się z niego, jak stosować style w dokumentach HTML i XHTML, poznasz zasady funkcjonowania stylów oraz reguły rządzące pozycjonowaniem elementów. Przede wszystkim jednak błyskawicznie znajdziesz opis wszystkich selektorów i właściwości stylów CSS, co pomoże Ci sprawnie tworzyć przejrzyste i atrakcyjne witryny internetowe.

- Dołączanie stylów do stron internetowych
- Funkcjonowanie stylów
- Zasady pozycjonowania elementów
- Wartości stosowane w stylach
- Przegląd selektorów, pseudoklas i pseudoelementów
- Opis wszystkich właściwości

**Naucz się stosować kaskadowe arkusze stylów
i twórz eleganckie witryny internetowe**



Spis treści

Konwencje zastosowane w książce	5
Dodawanie stylów do dokumentów HTML oraz XHTML	6
Struktura reguł	9
Pierwszeństwo stylów	10
Klasyfikacja elementów	13
Sposoby wyświetlania elementów	15
Podstawowy układ wizualny	16
Reguły ułożenia elementów pływających	19
Reguły pozycjonowania	21
Układ tabel	29
Wartości	37
Selektory	43
Pseudoklasy i pseudoelementy	49
Spis właściwości	56

Tabele	132
Media stronicowane	136
Właściwości usunięte z CSS2.1	140
Style wizualne	140
Media stronicowane	143
Style dźwiękowe	145
Skorowidz	157

Reguły pozycjonowania

Pozycjonowaniem elementów rządzi kilka reguł specjalnych. Określają one nie tylko położenie bloku-pojemnika, ale także położenie zawartych w nim elementów.

Rodzaje pozycjonowania

Pozycjonowanie statyczne

Pudełko elementu tworzone jak zwykle. Elementy blokowe generują prostokątne pudełko stanowiące część układu dokumentu. Elementy wierszowe tworzą jedno lub więcej pudełek obejmujących linie, stanowiących część układu elementu rodzica.

Pozycjonowanie względne

Pudełko elementu zostaje przesunięte o określoną odległość. Zawierający je blok uznawany jest za obszar, który zajmowałby element, gdyby nie był pozycjonowany. Element zachowuje swój kształt. Zachowywany jest także obszar, który element normalnie zajmowałby w dokumencie.

Pozycjonowanie bezwzględne

Pudełko elementu jest całkowicie usuwane z układu dokumentu i pozycjonowane w odniesieniu do bloku-pojemnika, którym może być inny element dokumentu lub pierwotny blok-pojemnik (opisany w kolejnym podrozdziale). Obszar, który element zajmowałby w normalnym układzie dokumentu, nie jest już rezerwowany, tak jakby element już nie istniał. Pozycjonowany element generuje własne pudełko blokowe, bez względu na rodzaj pudełka, jaki generowałby w normalnym układzie.

Pozycjonowanie ustalone

Pudełko elementu zachowuje się tak jak w przypadku pozycjonowania bezwzględnego, ale z blokiem-pojemnikiem jako punktem odniesienia.

Blok-pojemnik

Blok pojemnik dla pozycjonowanego elementu wyznaczany jest w myśl następujących reguł:

1. Blok-pojemnik dla elementu źródłowego (nazywany także *pierwotnym blokiem-pojemnikiem*) wskazywany jest przez przeglądarkę. W HTML-u elementem źródłowym jest `html`, chociaż niektóre przeglądarki korzystają z elementu `body`.
2. Dla elementów nieźródłowych, których wartość atrybutu `position` to `relative` lub `static`, blok-pojemnik jest wyznaczany przez krawędź zawartości najbliższego pudełka nadrzędnego elementu blokowego, tabeli, komórki tabeli lub bloku wierszowego. Mimo tej reguły elementy pozycjonowane względnie będą przesunięte, a nie rozmieszczone w odniesieniu do bloku pojemnika.
3. Dla elementów nieźródłowych, których wartość właściwości `position` to `absolute`, blokiem-pojemnikiem jest najbliższy element nadrzędny (dowolnego rodzaju), mający wartość właściwości `position` inną niż `static`. Wyznaczanie przebiega w następujący sposób:
 - a. Jeżeli element nadrzędny jest elementem blokowym, blokiem pojemnikiem jest krawędź dopełnienia tego elementu; innymi słowy, jest to obszar wyznaczony przez obramowanie.

- b. Jeżeli element nadrzędny to element wierszowy, blokiem-pojemnikiem jest krawędź zawartości bloku nadrzędnego. W językach zapisywanych od lewej do prawej górna i lewa krawędź bloku-pojemnika są krawędziami górną i lewą pierwszego pudełka w obiekcie nadrzędnym, a dolna i prawa krawędź są krawędziami dolną i górną zawartości ostatniego pudełka. W językach zapisywanych od prawej do lewej, prawa krawędź bloku-pojemnika odpowiada prawej krawędzi zawartości pierwszego pudełka, a lewa jest krawędzią ostatniego pudełka. Krawędzie górna i dolna wyznaczone są tak jak poprzednio.
- c. Jeżeli nie ma elementów nadrzędnych, tak jak opisano to w punktach a i b, blok-pojemnik elementu wyznaczany jest przez początkowy blok-pojemnik.

Układ elementów pozycjonowanych bezwzględnie

W poniższych podrozdziałach pojawiają się następujące pojęcia:

Dopasowanie przez zmniejszenie

Zbliżone do obliczania szerokości komórki tabeli za pomocą algorytmu automatycznego tworzenia układu tabeli. Ogólnie rzecz biorąc, przeglądarka próbuje wyznaczyć minimalną szerokość elementu, która będzie w stanie pomieścić zawartość. Zawijanie do kolejnych wierszy będzie przeprowadzone tylko w sytuacji, gdy będzie nie do uniknięcia.

Pozycja ustalona

Miejsce, w którym znalazłaby się krawędź elementu, gdyby wartością jego właściwości `position` było `static`.

Układ poziomy elementów niezastępowanych pozycjonowanych bezwzględnie

Równanie opisujące układ tego rodzaju elementów na postać:

$$\text{left} + \text{margin-left} + \text{border-left-width} + \text{padding-left} + \text{width} + \text{padding-right} + \text{border-right-width} + \text{margin-right} + \text{right} = \text{szerokość (width) bloku-pojemnika}$$

Kolejne kroki ustalania rozmieszczenia są następujące:

1. Jeżeli wszystkie właściwości `left`, `width` oraz `right` mają wartości `auto`, wartości `auto` dla `margin-left` oraz `margin-right` resetowane są do 0. Następnie, jeżeli kierunek zapisywania tekstu jest od lewej do prawej, właściwość `left` przyjmuje wartość pozycji ustalonej i stosowana jest trzecia reguła opisana w punkcie 3. W innym przypadku właściwość `right` przyjmuje wartość pozycji ustalonej i stosowana jest pierwsza reguła opisana w punkcie 3.
2. Jeżeli żadna spośród właściwości `left`, `width` oraz `right` nie ma wartości `auto`, wybierana jest odpowiednia reguła z poniższej listy:
 - a. Jeżeli zarówno `margin-left`, jak i `margin-right` mają wartość `auto`, równanie rozwiązywane jest z założeniem, iż oba marginesy będą miały równe wartości.
 - b. Jeżeli tylko `margin-left` lub `margin-right` ma wartość `auto`, równanie jest rozwiązywane dla tej wartości.
 - c. Jeżeli wartości są nadmiernie ograniczone (czyli żadna z nich nie ma wartości `auto`), wartość `left` jest ignorowana (jeżeli kierunek tekstu jest od prawej do lewej) lub wartość `right` jest ignorowana (jeżeli kierunek tekstu jest od lewej do prawej). Równanie rozwiązywane jest dla tej wartości.

3. Jeżeli niektóre z wielkości `left`, `width` oraz `right` mają wartość `auto`, ale inne nie, wartości `auto` dla `margin-left` i `margin-right` są resetowane do 0, wybierana jest odpowiednia reguła z poniższej listy:
- a. Jeżeli `left` ma wartość `auto`, ale `right` już nie, wartość szerokości (`width`) dopasowywana jest przez zmniejszenie. Równanie rozwiązywane jest dla wartości `left`.
 - b. Jeżeli `left` i `right` mają wartości `auto`, ale `width` nie, to dla kierunku tekstu od lewej do prawej `left` przyjmuje pozycję ustaloną (dla kierunku tekstu od prawej do lewej jest `right` (dla kierunku tekstu od prawej do lewej) lub `right` (`right`)). Równanie jest rozwiązywane odpowiednio dla (dla kierunku tekstu od lewej do prawej).
 - c. Jeżeli `width` oraz `right` mają wartości `auto`, ale `left` nie, szerokość (`width`) jest dopasowywana przez zmniejszanie. Równanie rozwiązywane jest dla wartości `right`.
 - d. Jeżeli `left` ma wartość `auto`, ale `width` oraz `right` nie, równanie rozwiązywane jest dla wartości `left`.
 - e. Jeżeli `width` ma wartość `auto`, ale `left` oraz `right` nie, równanie rozwiązywane jest dla wartości `width`.
 - f. Jeżeli `right` ma wartości `auto`, ale `left` oraz `width` nie, równanie jest rozwiązywane dla wartości `right`.

Układ pionowy elementów niezastępowanych pozycjonowanych bezwzględnie

Równanie opisujące układ rodzaju elementów ma postać:

$$\begin{aligned} & \text{top} + \text{margin-top} + \text{border-top-width} + \\ & \text{padding-top} + \text{height} + \text{padding-bottom} + \\ & \text{border-bottom-width} + \text{margin-bottom} + \text{bottom} = \\ & \text{wysokość (height) bloku-pojemnika} \end{aligned}$$

Kolejne kroki ustalania rozmieszczenia są następujące:

1. Jeżeli wszystkie `top`, `height` oraz `bottom` mają wartość `auto`, `top` przyjmuje pozycję ustaloną i stosowana jest trzecia reguła opisana w punkcie 3.
2. Jeżeli żadna spośród wielkości `top`, `height` oraz `bottom` nie ma wartości `auto`, wybierana jest odpowiednia reguła spośród poniższych:
 - a. Jeżeli zarówno `margin-top`, jak i `margin-bottom` mają wartości `auto`, równanie rozwiązywane jest z ograniczeniem, iż wielkości marginesów będą równe.
 - b. Jeżeli tylko `margin-top` lub `margin-bottom` ma wartość `auto`, równanie rozwiązywane jest dla tej wartości.
 - c. Jeżeli wartości są nadmiernie ograniczone (żadna z nich nie ma wartości `auto`), wartość `bottom` jest ignorowana, a równanie rozwiązywane jest dla tej wartości.
3. Jeżeli niektóre spośród `top`, `height` oraz `bottom` mają wartość `auto`, ale inne nie, wybierana jest odpowiednia reguła z poniższej listy:
 - a. Jeżeli `top` oraz `height` mają wartość `auto`, ale `bottom` nie, wysokość (`height`) obliczona jest na podstawie zawartości elementu (tak jak w układzie ustalonym). Wszystkie wartości `auto` dla `margin-top` oraz `margin-bottom` resetowane są do 0, a równanie rozwiązywane jest dla `top`.
 - b. Jeżeli `top` oraz `bottom` mają wartości `auto`, ale `height` nie, `top` przyjmuje ustalone położenie. Wszystkie wartości `auto` dla `margin-top` oraz `margin-bottom` resetowane są do 0, a równanie rozwiązywane dla `bottom`.
 - c. Jeżeli `height` oraz `bottom` mają wartości `auto`, ale `top` nie, wysokość (`height`) obliczona jest na podstawie zawartości elementu (tak jak w układzie ustalonym). Wszystkie warto-

ści auto dla margin-top oraz margin-bottom resetowane są do 0, a równanie rozwiązywane dla bottom.

- d. Jeżeli top ma wartość auto, ale height oraz bottom nie, wszystkie wartości auto dla margin-top oraz margin-bottom resetowane są do 0, a równanie rozwiązywane dla top.
- e. Jeżeli height ma wartość auto, ale top oraz bottom nie, wszystkie wartości auto dla margin-top oraz margin-bottom resetowane są do 0, a równanie rozwiązywane dla height.
- f. Jeżeli bottom ma wartość auto, ale top i height nie, wszystkie wartości auto dla margin-top oraz margin-bottom resetowane są do 0, a równanie rozwiązywane dla height.

Układ poziomy elementów zastępowanych pozycjonowanych bezwzględnie

Zasady mające zastosowanie przy określaniu położenia i wielkości elementów zastępowanych najłatwiej wyrazić w postaci wykonywanych kolejno reguł:

1. Jeżeli width ma wartość auto, obliczona wartość width określana jest przez wewnętrzną szerokość zawartości elementu. Tak więc szerokość obliczona na podstawie obrazu szerokiego na 50 pikseli to 50px. Jeżeli szerokość jest wyraźnie zadeklarowana (np. jako 100px lub 50%), width przyjmuje taką wartość.
2. Jeżeli left ma wartość auto w języku zapisywanym od lewej do prawej, auto zastępowane jest położeniem ustalonym. W przypadku języków zapisywanych od prawej do lewej wartość auto dla right zastępowana jest położeniem ustalonym.

3. Jeżeli `left` lub `right` ma ciągle wartość `auto` (czyli nie została zastąpiona w poprzednim punkcie), wszystkie wartości `auto` dla `margin-left` oraz `margin-right` zastępowane są wartością 0.
4. Jeżeli wartości `margin-left` i `margin-right` to ciągle `auto`, przyjmują równe wartości, a więc element zostaje wyśrodkowany w bloku-pojemniku.
5. Jeżeli po wprowadzeniu powyższych reguł pozostaje tylko jedna wartość `auto`, jest tak zmieniana, aby była równa pozostałej części równania.

Pionowy układ elementów zastępowanych pozycjonowanych bezwzględnie

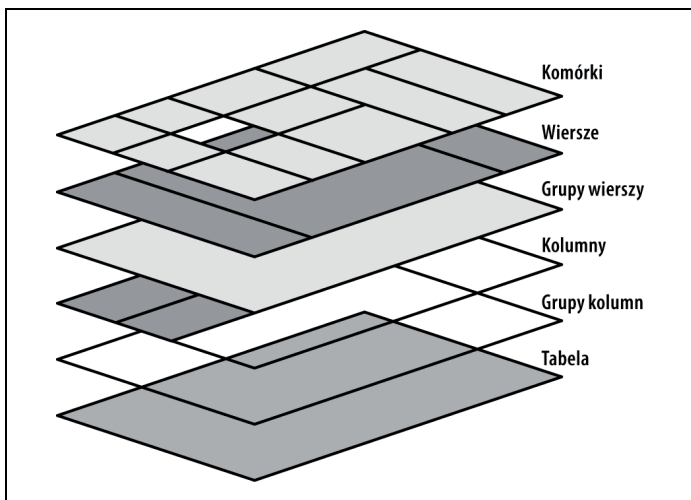
Zasady mające zastosowanie przy określaniu położenia i wielkości elementów zastępowanych najłatwiej wyrazić w postaci wykonywanych kolejno reguł:

1. Jeżeli `height` ma wartość `auto`, obliczona wartość `height` określana jest przez wewnętrzną wysokość zawartości elementu. Tak więc wysokość obliczona na podstawie obrazu wysokiego na 50 pikseli to 50px. Jeżeli wysokość jest wyraźnie zadeklarowana (np. jako 100px lub 50%), `height` przyjmuje taką wartość.
2. Jeżeli `top` ma wartość `auto`, zastępowane jest położeniem ustalonym elementu.
3. Jeżeli `bottom` ma wartość `auto`, wszystkie wartości `auto` dla `margin-top` i `margin-bottom` zastępowane są wartością 0.
4. Jeżeli wartości `margin-top` i `margin-bottom` to ciągle `auto`, przyjmują równe wartości, a więc element zostaje wyśrodkowany w bloku-pojemniku.

5. Jeżeli po wprowadzeniu powyższych reguł pozostaje tylko jedna wartość `auto`, jest tak zmieniana, aby była równa pozostałej części równania.

Układ tabel

Układ tabel może być dość skomplikowany, w szczególności kiedy weźmie się pod uwagę fakt, iż CSS definiuje dwa różne sposoby obliczania szerokości tabeli i jej komórek oraz dwa różne sposoby obsługi krawędzi tabeli i elementów wewnątrz niej. Rysunek 4. przedstawia komponenty tabeli.



Rysunek 4. Komponenty układu tabeli