

Kaskadowe arkusze styli - CSS

By Szumski Szymon

The bottom of the slide features a decorative graphic consisting of several overlapping, wavy lines in shades of beige and light brown, creating a modern, flowing effect.

Plan:

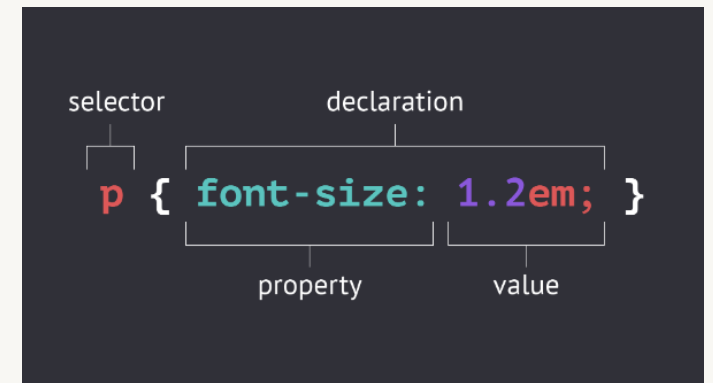
1. Wprowadzenie do CSS
2. Selektory
3. Pseudoklasy
4. Kaskada stylów
5. Podstawowe właściwości
6. Niestandardowe czcionki
7. Model blokowy
8. Flexbox
9. Tło elementu
10. Pozycjonowanie elementu
11. Responsywność

Wprowadzenie do CSS

CSS (Cascading Style Sheets, Kaskadowe arkusze stylów) - jest językiem do opisu i zmiany wyglądu elementów. HTML służy do definiowania struktury i semantyki treści dokumentu, a CSS jest używany do tworzenia wyglądu naszej strony i jej pozycjonowania.

Składnia

Blok kodu CSS nazywany jest regułą, która składa się z **selektora** i **bloku deklaracji** w nawiasach klamrowych. Selektor informuje przeglądarkę, do których elementów zastosować style z tej reguły. Każda deklaracja składa się z pary właściwość: wartość;, gdzie właściwość i jej wartość są oddzielone dwukropkiem ze spacją i koniecznie kończą się średnikiem. One opisują, co dokładnie należy zrobić ze stylami elementu.



Podstawowe właściwości

font-family: Określa rodzaj czcionki używanej dla tekstu. Można tu podać konkretne nazwy czcionek (np. "Arial", "Times New Roman") lub rodziny czcionek (np. sans-serif, serif, monospace), co pozwala na zapewnienie alternatywnych wyborów czcionek dla różnych systemów.

font-size: Ustala rozmiar czcionki. Można używać różnych jednostek, takich jak piksele (px), em, procenty (%) lub rem (relatywna wielkość bazująca na domyślnym rozmiarze czcionki).

font-weight: Określa wagę (grubość) tekstu. Wartości mogą obejmować normal, bold, bolder, lighter, lub liczbowe wartości (od 100 do 900).

font-style: Pozwala na ustawienie stylu tekstu, takiego jak kursywa. Wartości to normal, italic, oblique lub inherit.

text-align: Ustala położenie tekstu wewnątrz jego kontenera. Pozwala na wybór wyrównania tekstu do lewej, do prawej, do środka lub wyjustowanego (justified).

Lista wszystkich właściwości dostępna na stronie: https://www.w3schools.com/cssref/css3_pr_all.php

Wprowadzenie do CSS – rodzaje styli

Dodawanie styli

Istnieją trzy sposoby dodawania stylów do dokumentu HTML: style wbudowane (inline styles), wbudowany arkusz stylów (embedded stylesheet) i zewnętrzny arkusz stylów (external stylesheet). Każdy sposób ma swoje zalety, wady, ograniczenia i zakres użycia.

1. Style wbudowane
2. Wbudowany arkusz styli
3. Zewnętrzny arkusz styli

Style wbudowane

Style wbudowane (inline) są ustawiane dla znacznika bezpośrednio w atrybucie **style**. Nie można ich skalować, są trudne do zastąpienia i ponownego użycia, więc są rzadko używane. Używane głównie w przypadku stylów które nadajemy za pomocą Javascript dynamicznie.

```
<p style="color:blue; font-size:24px;">Ten tekst będzie niebieski.</p>
```

Style wbudowane - zadanie

- Utworz sekcje z nagłówkiem (h2) - "Style wbudowane"
- Dodaj styl wbudowany do tego nagłówka (rozmiar czcionki-font-size)
- Utwórz w tej sekcji akapit wraz ze stylem wbudowanym (np. Kolor)

```
<head>
  <style type="text/css">
    p {
      color: blue;
      font-size: 24px;
    }
  </style>
</head>
```

Wbudowany arkusz styli

Przy takim podejściu reguły CSS dodają się wewnątrz znacznika `<style>` w nagłówku dokumentu. Nie można ich ponownie wykorzystać na innych stronach i są trudne do skalowania i konserwacji.

Wbudowany arkusz styli - zadanie

- Utwórz w sekcji `<head>` arkusz styli
- Nadaj w nim styl dla `h1` – `color: red` oraz `font-size: 32px`
- Nadaj odległości dla sekcji – `padding: 10px`

Zewnętrzny arkusz styli

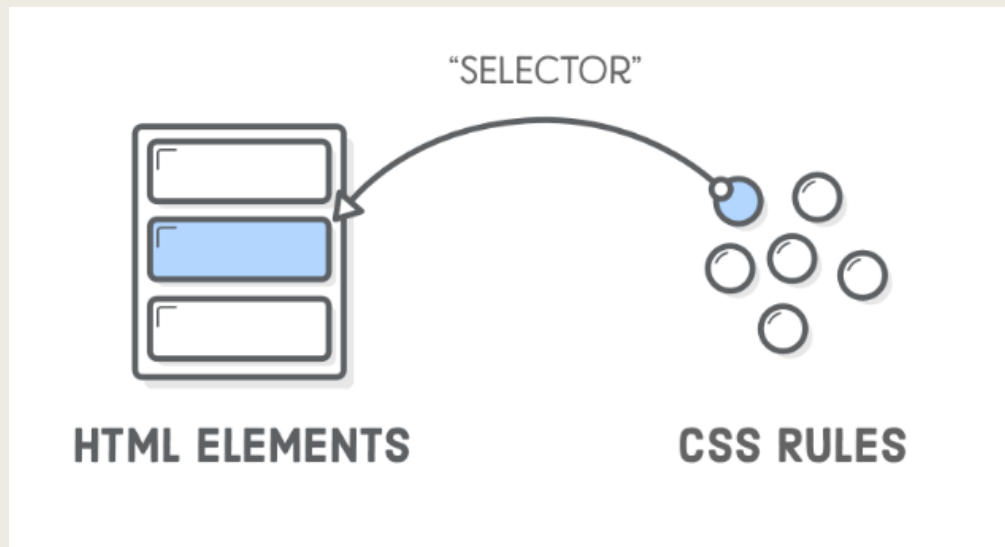
Zewnętrzny kod CSS jest łatwy do skalowania, utrzymania i ponownego wykorzystania na innych stronach, więc jest to standard łączenia stylów. Projekt tworzy oddzielny plik arkusza stylów z rozszerzeniem **.CSS** i podłącza się do dokumentu HTML.

- Na tym samym poziomie co `index.html` tworzony jest folder `css`, a wewnątrz niego arkusz stylów `styles.css`.
- W znaczniku `<head>`, za pomocą znacznika `<link>`, podłączany jest wcześniej utworzony plik.
- Atrybut `href` określa ścieżkę do arkusza stylów względem dokumentu HTML.
- Atrybut `rel` określa typ dołączonego dokumentu - `stylesheet` (arkusz stylów).

```
<head>
  <link rel="stylesheet" href="./css/styles.css" />
</head>
```

Zewnętrzny arkusz styli - zadanie

- Utwórz plik styles.css w folderze repozytorium
- Utwórz sekcję z nagłówkiem (h2) zewnętrzny arkusz styli
- Dzięki nazwie selektora nadaj inny kolor (np. Red, green itp.) dla nagłówka



Selektory CSS

Selektor CSS to fragment kodu używany w arkuszach stylów (CSS), który identyfikuje elementy na stronie internetowej, do których ma zostać zastosowany określony styl lub zestaw reguł.

Selektor elementu (znacznika)

```
/* Zastosowane do wszystkich akapitów na stronie */  
p {  
  font-size: 24px;  
}  
  
/* Zastosowane do wszystkich linków na stronie */  
a {  
  text-decoration: none;  
}
```

Opisuje do jakich elementów html będzie stosowana reguła CSS. Selektor elementu nie jest wystarczająco specyficzny, aby nadać styl poszczególnym elementom, dlatego jest najczęściej używany do stylizacji wszystkich znaczników tego samego typu na stronie. Na przykład, jeśli trzeba usunąć podkreślenie ze wszystkich linków w dokumencie.

Selektor znacznika - zadanie

W utworzonym pliku `styles.css` dodaj style za pomocą znacznika (kolor, rozmiar czcionki wystarczy)

```
<h1 class="title">Cześć, jestem Mango.</h1>
<p class="text">
  Witam na mojej osobistej stronie. Tutaj możesz zobaczyć
  <a class="link" href="">projekty</a>.
</p>
```

```
/* Będzie dotyczyć wszystkich znaczników z klasą title */
.title {
  font-weight: 500;
}

/* Będzie dotyczyć wszystkich znaczników z klasą text */
.text {
  color: brown;
  font-size: 18px;
}
```

Selektor klasy

Najczęściej używany selektor. Używany w połączeniu z atrybutem globalnym `class`. Nazwa klasy ustawia się tylko w języku angielskim i jest to konieczne. Używany w połączeniu z atrybutem globalnym `class`. rzeczownik opisujący, jakiego rodzaju jest ten element.

Selektora klasy używamy do nadania stylu jednemu lub większej liczbie elementów o tej samej wartości atrybutu `class` (nazwa klasy). W selektorze przed nazwą klasy umieszczana jest kropka `.`.

Selektor klasy - zadanie

Nadaj kilka klas w pliku html, następnie za pomocą selektora klasy nadaj style dla tych elementów dla których zostały dopisane klasy (2,3 max)


```
<p class="alert success">Zakończono doładowanie konta</p>
<p class="alert warning">Uwaga, taryfy ulegną zmianie</p>
<p class="alert error">Błąd transakcji</p>
```

```
/* Wspólne style dla wszystkich typów alertów */
.alert {
  font-size: 24px;
  font-weight: 500;
}

/* Specyficzne style dla każdego typu */
.success {
  color: green;
}

.warning {
  color: orange;
}

.error {
  color: red;
}
```

Kompozycja klas

Element może mieć kilka klas, wtedy są one oddzielone spacją. Można to wykorzystać do łączenia i ponownego wykorzystywania stylów. Utwórzmy znaczniki i style dla komponentu alertów.

W stylach opisujemy kilka reguł, które są wspólne dla wszystkich alertów i specyficzne dla każdego typu.

Selektor identyfikatora

Podobnie jak w przypadku klasy, znaczniki mogą mieć globalny atrybut `id`. Różnica polega na tym, że wartość `id` musi być unikalna na stronie, to znaczy nie mogą istnieć dwa elementy o tej samej wartości identyfikatora.

W selektorze CSS przed nazwą identyfikatora umieszczany jest symbol krzyżyka `#`.

```
<h1 id="title">Nagłówek strony</h1>
```

```
#title {  
  font-weight: 500;  
  color: orange;  
}
```

Pseudoklasy stanowe

Selektory stanowe służą do użycia stylów dla elementów interaktywnych na określone zdarzenie, takie jak najechanie kursorem na link lub fokus z klawiatury.

Pseudoklasa definiuje określony stan elementu i jest dołączona do selektora znacznika, klasy itp. Pomiędzy selektorem a pseudoklasą nie ma spacji wskazującej, że są one ze sobą powiązane. Jeśli dodać spację, to pseudoklasa będzie dotyczyć wszystkich elementów w dokumencie.

```
selektor:pseudoklasa {  
    /* Właściwości */  
}
```

Pseudoklasy stanowe

1. **:hover** - Zostaje uruchamiana, gdy kursor myszy znajduje się w granicach elementu, na przykład gdy wskaźnik myszy znajduje się nad linkiem lub innym elementem.
2. **:focus** - Jest aktywowana, gdy element interaktywny (link, przycisk, pole formularza) otrzymuje fokus podczas nawigacji po stronie za pomocą klawiatury.
3. **:active** - Występuje, gdy element jest aktywowany. Na przykład link staje się aktywny, jeśli najechać na niego kursorem i kliknąć myszą. Pomimo faktu, że prawie każdy element może stać się aktywny, pseudoklasa `:active` jest używana głównie dla linków i przycisków.

Pseudoklasy - zadanie

- Utwórz sekcje w której utworzysz przycisk, input oraz link (href="#")
- Nadaj style dzięki pseudoklasie:
- Przycisk – hover
- Input – focus
- Link - active

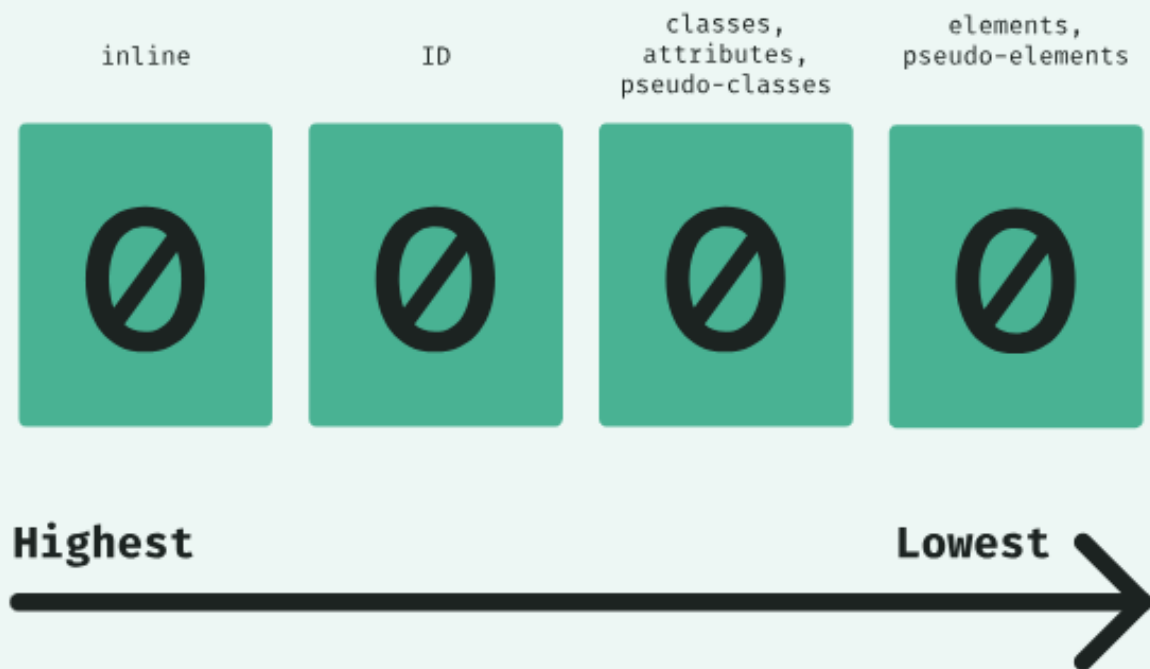
Kaskada styli

Kaskadowość - to mechanizm, który zarządza końcowymi wartościami właściwości elementu, gdy stosuje się do niego wiele reguł CSS.

- Jeśli do elementu stosuje się wiele reguł, ich właściwości są łączone
- Jeśli reguły mają te same właściwości z różnymi wartościami, powoduje to konflikt

Aby zebrać ostateczne style elementu i rozwiązać konflikty wartości właściwości, przeglądarka wykorzystuje mechanizm: **specyficzność**.

```
/* Jakiego koloru będzie tekst akapitów? */  
p {  
  color: blue;  
  background-color: orange;  
}  
  
p {  
  color: teal;  
}
```



Specyficzność selektorów

Dla każdej reguły CSS przeglądarka oblicza specyficzność (wagę) selektora. Jeśli właściwości z różnych reguł muszą być zastosowane do elementu i istnieją sprzeczne (identyczne) właściwości, używa się wartości właściwości z reguły z największą specyficznością selektora.

Wartość specyficzności składa się z czterech rang, których znaczenie rośnie od prawej strony do lewej.

Specyficzność - zadanie

Utwórz sekcje z nagłówkiem (h2) oraz akapitem

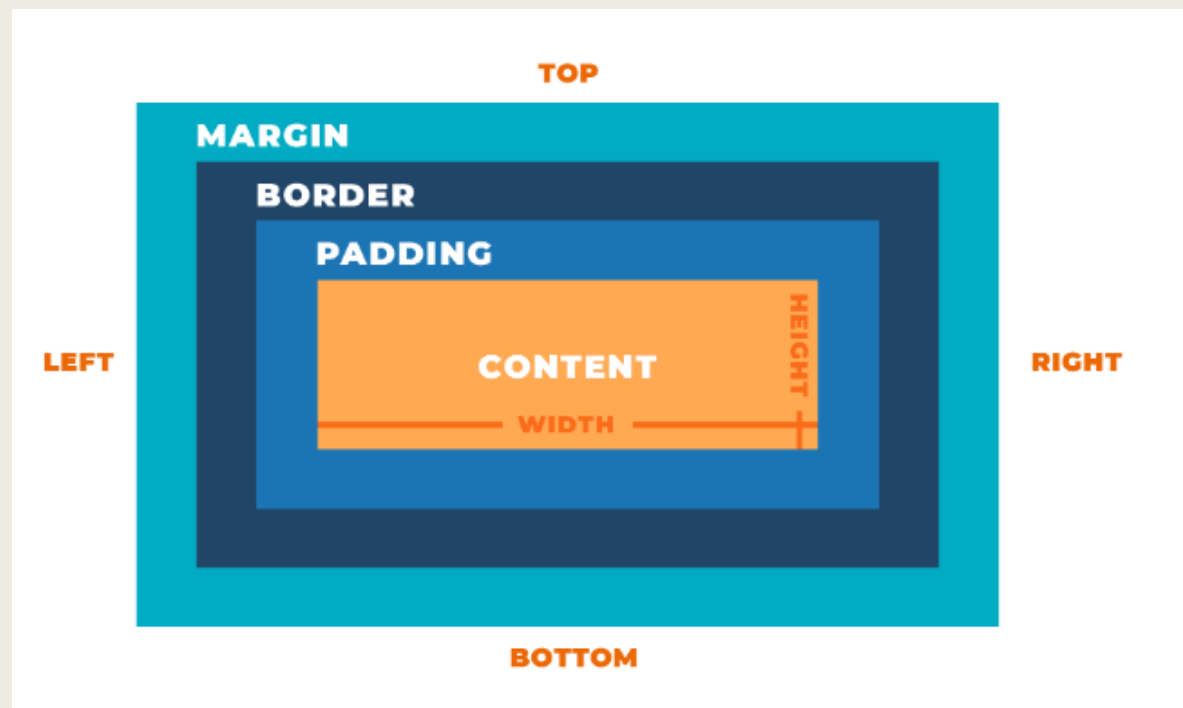
- Dla akapitu nadaj id oraz klasę
- W pliku styles.css zapisz 3 style z 3 różnymi kolorami
- Za pomocą nazwy selektora
- Za pomocą klasy
- Za pomocą identyfikatora

Zobacz który z nich ma priorytet

Model blokowy

W modelu blokowym każdy element jest prostokątnym kontenerem, niezależnie od tego, jak jest wizualnie przedstawiony na stronie. Element ma obszar zawartości i opcjonalne marginesy, dopełnienie i obramowanie.

- **Obszar zawartości (content)** - jest to zawartość elementu, takiego jak tekst, obraz, wideo lub elementy zagnieżdżone. Domyślna wysokość obszaru zawartości jest określana przez kontent, szerokość - przez typ elementu (blok lub inline).
- **Odstęp wewnętrzny (padding)** - jest to odległość między zawartością elementu a jego obramowaniem (border). Używany do efektów dekoracyjnych. Nie można określić wartości ujemnych.
- **Obramowanie (border)** - granica elementu, domyślnie jej szerokość wynosi zero. Jeśli kolor obramowania nie zostanie określony, przybiera kolor głównej treści, takiej jak tekst.
- **Margines (margin)** - dodaje dopełnienie na zewnątrz elementu, od zewnętrznej krawędzi (obramowania) do sąsiednich elementów, oddzielając je w ten sposób na stronie. Może przyjmować wartości ujemne. Tło elementu nigdy nie rozciąga się do marginesu.



Geometria elementu

Geometria elementu ma 4 boki: górny (top), prawy (right), dolny (bottom) i lewy (left). W tej kolejności, zaczynając od góry zgodnie z ruchem wskazówek zegara, ustawiane są wartości właściwości padding, margin i border.



Geometria - zadanie

Utworz element `<div>`, nadaj mu klase "block"

- Dla podanej klasy w pliku styles.css nadaj style (background-color: red, width: 100px, height: 100px)
- Nadaj mu 4 rozne paddingi: góra, prawo, dół, lewo

Dodatkowe dla chętnych

Flexbox - jest to moduł CSS, który określa zestaw właściwości do pozycjonowania, wyrównywania i rozdzielania przestrzeni między elementami w kontenerze, nawet jeśli ich rozmiar jest nieznany lub dynamiczny.

<https://flexboxfroggy.com/#pl>