

Podstawy informatyki

HTML5 / CSS3

Laboratorium 2

Podstawy CSS: składnia, selektory i dołączanie stylów

Czas trwania: 2 × 45 min**Tryb:** studia stacjonarne**Jednostka:** Katedra Elektrotechniki i Technologii Nadprzewodnikowych**Prowadzący:** dr inż. Michał Łanczont

Cel ćwiczenia

Celem jest poznanie składni CSS, trzech sposobów dołączania stylów, selektorów typu, klasy i identyfikatora, podstaw kaskady oraz stylizacji tekstu, list i elementu `iframe`.

Wstęp

CSS (Cascading Style Sheets) opisuje prezentację dokumentu HTML. Reguła CSS składa się z selektora oraz bloku deklaracji. Każda deklaracja zawiera nazwę właściwości i wartość.

```
h2 {  
  color: blue;  
  text-align: center;  
}
```

Selektor elementu, klasy i identyfikatora

Atrybuty HTML `class` i `id` nie są same w sobie “selektorami atrybutów”. Są atrybutami, które mogą być wskazane odpowiednio selektorem klasy `.nazwa` oraz selektorem identyfikatora `#nazwa`.

```
<p class="wazny-tekst">Ten akapit jest wazny.</p>  
<h2 class="wazny-tekst">Ten naglowek tez jest wazny.</h2>  
<div id="stopka-glowna">Informacja koncowa</div>
```

```
.wazny-tekst {  
  color: red;  
  font-weight: bold;  
}  
  
#stopka-glowna {  
  font-size: 0.8rem;  
  text-align: center;  
}
```

Jedna klasa może wystąpić na wielu elementach, a jeden element może mieć wiele klas. Wartość `id` powinna być unikalna w obrębie dokumentu. W praktyce do stylowania najczęściej preferuje się klasy, ponieważ łatwiej je ponownie wykorzystać i mają niższą specyficzność.

Trzy sposoby dołączania CSS

Styl liniowy jest zapisany w atrybucie `style`:

```
<h1 style="color: red; font-size: 32px;">Czerwony naglowek</h1>
```

Styl wewnętrzny jest zapisany w `style` w sekcji `head`:

```
<style>
  p { color: green; }
</style>
```

Styl zewnętrzny znajduje się w osobnym pliku i jest rozwiązaniem zalecanym dla projektu zawierającego więcej niż prosty eksperyment:

```
<link rel="stylesheet" href="style.css">
```

```
/* style.css */
body {
  font-family: Verdana, Arial, sans-serif;
  background-color: #f0f0f0;
}
```

Kaskada – ważna korekta

Nie należy mówić, że plik zewnętrzny ma z definicji “najniższy priorytet”. Styl wewnętrzny i zewnętrzny należą do tego samego pochodzenia autora, a wynik zależy m.in. od ważności deklaracji, warstw kaskady, specyficzności selektora i kolejności reguł. Styl liniowy ma bardzo wysoką specyficzność w ramach stylów autora, ale również może zostać pokonany przez odpowiednio ważne reguły.

Specyficzność i dziedziczenie

W uproszczonym modelu selektor ID jest bardziej specyficzny niż selektor klasy, a selektor klasy bardziej specyficzny niż selektor elementu. Nie każda właściwość jest dziedziczona: np. `color` i `font-family` zwykle są, natomiast `border`, `margin` czy `padding` nie.

```
#komunikat { color: red; }
.komunikat { color: blue; }
p { color: green; }
```

Dla elementu `<p id="komunikat" class="komunikat">` wygra zwykła deklaracja ID, jeżeli wszystkie reguły mają tę samą ważność i pochodzenie.

Typografia i tekst

Właściwość	Znaczenie	Przykład
<code>color</code>	kolor tekstu	<code>color: #333;</code>
<code>font-size</code>	rozmiar pisma	<code>font-size: 1.1rem;</code>
<code>font-family</code>	rodzina i lista zastępcza	<code>font-family: Arial, sans-serif;</code>
<code>text-align</code>	wyrównanie tekstu	<code>text-align: justify;</code>
<code>line-height</code>	interlinia	<code>line-height: 1.6;</code>
<code>background-color</code>	kolor tła elementu	<code>background-color: #ffffcc;</code>

Listy w CSS

Wygląd markerów list kontrolują m.in. `list-style-type`, `list-style-position` i skrót `list-style`.

```
ul {
  list-style-type: circle;
}
```

```
ol {  
  list-style-type: upper-roman;  
  list-style-position: inside;  
}
```

Stylizacja iframe

CSS może stylizować pudełko `iframe`, ale nie daje swobodnego dostępu do wewnętrznego dokumentu pochodzącego z innej domeny.

```
iframe {  
  border: 3px solid #3498db;  
  border-radius: 10px;  
  box-shadow: 0 5px 15px rgb(0 0 0 / 0.2);  
  max-width: 100%;  
}
```

Właściwość `opacity` działa na cały element wraz z osadzoną treścią, więc należy używać jej świadomie.

Przykład kompletny – trzy warstwy kaskady

```
/* style.css */  
body { color: #333; font-family: Arial, sans-serif; }  
p { color: #1f4e79; }  
.wazny { color: #b00020; }  
#komunikat { color: #146b2e; }
```

```
<p>Zwykly akapit.</p>  
<p class="wazny">Akapit z klasa.</p>  
<p class="wazny" id="komunikat">Akapit z klasa i ID.</p>
```

W tym zestawie ostatni akapit otrzyma kolor wynikający z selektora ID, o ile nie zadziała reguła o wyższej ważności. Dodanie stylu liniowego jest kolejnym eksperymentem, ale nie powinno być docelowym sposobem organizacji dużej witryny.

Typowe błędy CSS

- brak średnika między deklaracjami lub brak zamykającego nawiasu klamrowego,
- zapis `class=".wazny"` w HTML – kropkę stosuje się w selektorze CSS, nie w wartości atrybutu,
- powtarzanie tego samego `id` na wielu elementach,
- zakładanie, że kolejność plików wewnętrzny/zewnętrzny automatycznie rozstrzyga każdą kolizję bez uwzględnienia specyficzności,
- brak rodziny zastępczej w `font-family`.

Przebieg ćwiczenia

Zadanie 1. Style liniowe i wewnętrzne

Do wykonania

Otwórz `dokumentacja.html` z L1. Ustaw kolor `h1` przez `style="color: blue"`. Następnie w `head` dodaj tymczasowy blok `style`, który ustawi `body { color: #333; }`. Sprawdź, które elementy dziedziczą kolor.

Zadanie 2. Zewnętrzny arkusz stylów

Do wykonania

Utwórz `style.css`, usuń blok `style` z HTML i połącz plik przez `link`. Ustaw `font-family: Verdana, Arial, sans-serif` dla `body`. Wyjaśnij, dlaczego lista czcionek kończy się rodziną ogólną.

Zadanie 3. Selektory elementu, klasy i ID

Do wykonania

Dla wszystkich akapitów ustaw `text-align: justify` i `font-size: 1.1em`. Nagłówkom `h2` nadaj rodzinę `Georgia, serif`. Akapitowi wstępu dodaj klasę `wstep-artykulu`; stopce albo innej unikalnej sekcji nadaj `id`. Zdefiniuj osobne reguły klasy i ID.

Zadanie 4. Stylizacja list

Do wykonania

Dla `ol` ustaw `list-style-type: upper-roman` i sprawdź różnicę między `inside` oraz `outside`. Dla `ul` ustaw `circle`. Następnie utwórz klasę listy bez markerów i zastosuj ją tylko do jednego wybranego elementu `ul`.

Zadanie 5. Stylizacja iframe

Do wykonania

Do ramki z L1 zastosuj `border-radius`, `box-shadow` oraz `max-width: 100%`. Utwórz klasę `ramka-wyrozniona` z czerwonym obramowaniem przerywanym i `opacity: 0.7`. Sprawdź, że przezroczystość dotyczy również treści ramki.

Zadanie 6. Doświadczenie z kaskadą

Do wykonania

W zewnętrznym CSS ustaw `body { color: #333; }`. Następnie dodaj do `body` styl liniowy `color: red`. Zapisz wynik. Potem usuń styl liniowy i przygotuj dwie reguły tego samego selektora w arkuszu, z różnymi kolorami. Sprawdź znaczenie kolejności reguł.

Zadanie końcowe

Integracja

Doprowadź stronę z L1 do spójnego wyglądu: typografia, nagłówki, akapit wstępu, listy i `iframe` mają być stylowane wyłącznie przez zewnętrzny arkusz. Użyj co najmniej selektora elementu, dwóch klas i jednego ID, a następnie wyjaśnij ich różną specyficzność.

Kryteria zaliczenia

- poprawne dołączenie zewnętrznego CSS,
- poprawne rozróżnienie selektora klasy, ID i atrybutu,

- umiejętność wyjaśnienia kaskady i dziedziczenia na prostym przykładzie,
- spójna typografia i listy,
- brak zbędnych stylów liniowych w rozwiązaniu końcowym.

Sprawozdanie

Do odbioru należy przygotować pliki źródłowe utworzone podczas ćwiczenia. Jeżeli prowadzący wymaga sprawozdania, powinno ono zawierać: krótki opis rozwiązania, najważniejsze fragmenty kodu, zrzut ekranu wyniku oraz wnioski dotyczące napotkanych problemów. Pliki HTML i CSS należy zachować w strukturze katalogów zgodnej z instrukcją.

Materiały pomocnicze

- HTML Living Standard: <https://html.spec.whatwg.org/>
- MDN Web Docs – HTML: <https://developer.mozilla.org/docs/Web/HTML>
- MDN Web Docs – CSS: <https://developer.mozilla.org/docs/Web/CSS>
- Walidator HTML: <https://validator.w3.org/>