

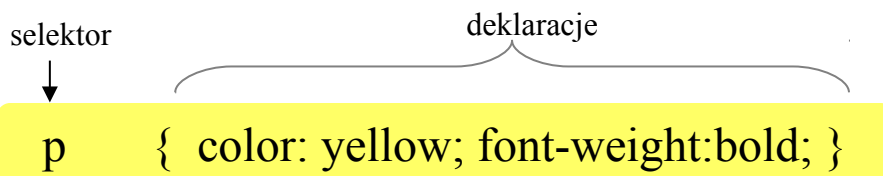
## Kaskadowe Arkusze Styli - CSS

### Wstęp

Cascading Style Sheets (Kaskadowe Arkusze Styli, w skrócie CSS) jest językiem, który opisuje sposób w jaki będzie wyświetlana strona internetowa. Dzięki temu możemy zmieniać kolory czcionek, ustawiać tło, pozycjonować różne elementy, a także wykonywać wiele innych operacji. CSS znacznie rozszerza możliwości HTML'a odnośnie sterowania wyglądem strony, dodatkowo umożliwiając łatwą zmianę wyglądu bez ingerencji w sam kod strony oraz znacznie skracając sam kod poprzez wydzielenie kodu odnoszącego się do wyglądu do jednego, wspólnego pliku.

### Składnia języka CSS

Język CSS ma bardzo prostą strukturę – składa się z selektorów oraz deklaracji:



Powyższy przykład ustawia kolor żółty oraz pogrubioną czcionkę dla akapitów (znacznik p). Selektor najczęściej jest nazwą elementu, dla którego chcemy ustawić formatowanie za pomocą deklaracji znajdujących się pomiędzy nawiasami klamrowymi. Deklaracje składają się z atrybutu CSS (w naszym przykładzie *color* oraz *font-weight*) oraz wartości (*yellow* oraz *bold*), powinny też kończyć się średnikiem. W celu zwiększenia czytelności kodu CSS bardzo często deklaracje umieszcza się w oddzielnych liniach, przykładowo powyższy kod może zostać również zapisany jako:

```
p{  
    color: yellow;  
    font-weight: bold;  
}
```

Oprócz selektorów i definicji możemy również stosować komentarze, które pomogą w późniejszym czasie zrozumieć kod, który napisaliśmy. Komentarz w języku CSS zaczyna się znakami */\** i kończy *\*/*. Przykładowo:

```
/* zmiana wyglądu p */  
p{  
    /* kolor żółty */  
    color: yellow;  
    /* pogrubienie */
```

## Projektowanie statycznych stron www

---

```
    font-weight: bold;
}
```

Kod CSS możemy dodać do strony na dwa sposoby. Pierwszym z nich jest umieszczenie go bezpośrednio w kodzie HTML:

- w znaczniku, który ma zostać zmieniony (styl jest ustawiany tylko dla tego jednego egzemplarza znacznika, pozostałe będą posiadały nadal styl domyślny), np.:

```
<p style="color: yellow; font-weight: bold;" > ... </p>
```

- lub w sekcji head wewnątrz znacznika style:

```
<head>
...
    <style>
        p{
            color: yellow;
            font-weight: bold;
        }
    </style>
</head>
```

Drugim sposobem (zalecanym) jest zapisanie kodu CSS do oddzielnego pliku (np. style.css) i dołączenie go w sekcji head do strony:

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

Dzięki temu mamy możliwość używania tych samych stylów na wielu stronach html bez niepotrzebnego powielania tego samego kodu.

### Ćwiczenie 1

Utwórz prostą stronę internetową, która będzie zawierała elementy takie jak nagłówek H1 oraz dwa akapity (p). Następnie utwórz plik style.css oraz dołącz go do strony html (z użyciem znacznika link – drugi sposób).

1. Zmiana koloru tła strony

Zmiana koloru tła strony z użyciem HTML wyglądałaby następująco:

```
<body bgcolor="black" >
```

Z użyciem CSS wystarczy w pliku style.css dodać linię:

```
body { background-color: black; }
```

Po zmianie koloru tła strony na czarny, tekst przestaje być widoczny, dlatego należy nadać mu kolor biały. Możemy zrobić dodając atrybut color z wartością white:

```
body { background-color: black; color: white; }
```

Dzięki temu, że strona html ma ustaloną hierarchię, nawet tekst wewnątrz znaczników h1 oraz p będzie posiadał biały kolor.

2. Zmiana wyglądu akapitów

W naszym ćwiczeniu chcemy, aby tło pierwszego akapitu było koloru jasnoszarego (#DDDD7) natomiast drugiego białe. Jak na razie, aby to wykonać musimy w pliku html dodać do odpowiednich znaczników atrybut style:

```
<p style="background-color: #DDDD7;"> ... </p>
```

```
<p style="background-color: white;"> ... </p>
```

Ponownie tekst przestaje być widoczny – musimy na nowo ustawić kolor tekstu.

## Projektowanie statycznych stron www

---

```
<p style="background-color: #DDDD7; color: black"> ... </p>
<p style="background-color: white; color: black;"> ... </p>
```

Dodatkowo chcemy, aby każdy akapit był wyjustowany (wyrównany do lewej i prawej strony). Do wyrównania służy atrybut `text-align`, który może mieć następujące wartości:

Wartość	Opis
left	Wyrównanie do lewej strony
right	Wyrównanie do prawej strony
center	Wyśrodkowanie
justify	Wyrównanie do lewej i prawej strony
inherit	Dziedziczenie po znaczniku znajdującym się wyżej w hierarchii

W naszym przypadku wykorzystamy wartość `justify`:

```
p { text-align: justify; }
```

## Identyfikatory i klasy

W poprzednim ćwiczeniu można było zauważyć, że w przypadku gdy chcieliśmy zastosować różne style dla różnych użyć tego samego znacznika trudno było umieścić całość w jednym pliku CSS. Nadal kod strony zawierałby niepotrzebnie powielany kod. Dlatego do języka CSS został dodany mechanizm identyfikatorów i klas, które służą do ustawiania stylów dla konkretnych, wybranych elementów strony.

### Identyfikatory

Identyfikatory służą do oznaczenia jednego, konkretnego użycia znacznika, dlatego ich nazwa musi być unikalna – na stronie nie może pojawić się żaden inny element o tym samym identyfikatorze.

W celu użycia identyfikatora należy dodać do znacznika atrybut `id`, np.:

```
<p id="pierwszyAkapit"> ... </p>
```

Następnie można ustawić dla niego styl, poprzedzając nazwę identyfikatora znakiem `#`:

```
#pierwszyAkapit {background-color: #DDDD7; color: black;}
```

### Klasy

Jeśli chcemy zdefiniować styl, który może być używany przez wiele różnych elementów, niekoniecznie tego samego typu możemy skorzystać z klas. Klasa wymaga zastosowania atrybutu `class` w znaczniku, dla którego ma być zastosowana:

```
<p id="pierwszyAkapit" class="czarnyTekst"> ... </p>
```

W kodzie CSS nazwę klasy należy poprzedzić kropką:

```
.czarnyTekst {color: black;}
```

Możemy również zdefiniować różne style w przypadku zastosowania innych znaczników dla tej samej klasy, w tym celu przed nazwą klasy należy podać nazwę znacznika, np.:

```
p.czarnyTekst {color: black;}
```

### Ćwiczenie 2

Zmodyfikuj stronę z ćwiczenia 1 tak, aby korzystało z identyfikatorów i klas.

## Najczęściej używane style CSS

### Formatowanie tła

Do tej pory ustawialiśmy jedynie kolor tła strony. Jednak podobnie jak HTML, język CSS pozwala również na ustawienie jako tła obrazu. Służy do tego atrybut *background-image*, który jako wartość przyjmuje ścieżkę dostępu do pliku na serwerze, np. ustawienie jako tła obrazka o nazwie tło.jpg znajdującego się w folderze grafika może wyglądać następująco:

```
body {background-image:url('grafika/tło.jpg');}
```

Oprócz atrybutu *background-image* zostały zdefiniowane dodatkowe style pozwalające na powtarzanie wyświetlanie obrazka w tle za pomocą atrybutu *background-repeat*, który może przyjąć następujące wartości:

- repeat-x - powtarzanie obrazka w poziomie
- repeat-y - powtarzanie obrazka w pionie
- repeat - powtarzanie obrazka w pionie i poziomie
- no-repeat - brak powtarzania obrazka

Atrybuty te możemy wykorzystać np. do tworzenia tła w postaci gradientu.

Czasami chcemy, aby obrazek w tle był nieruchomy, nawet przy przewijaniu strony. Służy do tego atrybut *background-attachment*, który jako wartości przyjmuje:

- scroll – obrazek będzie przewijał się razem z resztą strony
- fixed – obrazek pozostanie nieruchomy przy przewijaniu

Oprócz tego, możemy oznaczyć dokładnie w którym miejscu strony ma zostać wyświetlany obrazek tła za pomocą atrybutu *background-position*. Przyjmuje on jako wartość dwie współrzędne punktu, które mogą być liczbami lub jednymi z angielskich określeń: left, right, center dla pierwszej współrzędnej oraz top, bottom, center dla drugiej, np.:

```
background-position: right center
```

ustawi nam obrazek tła wyrównany do prawej strony i wyśrodkowany w poziomie.

Język CSS pozwala również na zgrupowanie wszystkich powyższych właściwości tła w jednym atrybucie o nazwie *background*. Jako jego wartość należy podawać kolejno po spacji wartości atrybutów *background-color*, *background-image*, *background-repeat*, *background-attachment*, *background-position* (można pominąć wartości dla niektórych atrybutów).

### Formatowanie tekstu

Formatowanie tekstu w CSS zostało zgrupowane głównie w atrybutach z rodziny font oraz text. Jako pierwsze omówimy atrybuty font.

Do ustawiania rodzaju czcionki, która ma zostać wykorzystana do wyświetlenia tekstu używamy atrybutu *font-family*. Jako jego wartość podajemy nazwę czcionki (np. „Times New Roman”) lub jej rodzaj (np. serif, sans-serif). Możemy podać całą listę czcionek oddzielając je

## Projektowanie statycznych stron www

---

przecinkami, dzięki czemu jeśli dana czcionka wybrana przez nas nie jest zainstalowana w systemie, zostanie wybrana kolejna z listy.

Styl czcionki (atrybut *font-style*) określa sposób w jaki ma być ona wyróżniana. Jako wartości możemy podać:

- normal – brak wyróżnienia
- italic – kursywa
- oblique - tekst pochylony (najczęściej zbliżony lub identyczny do kursywy)

Kolejnym elementem, który możemy zmienić jest wariant czcionki (atrybut *font-variant*), który pozwala na użycie kapitalików (zamiast normalnych małych liter używane są duże litery o mniejszym rozmiarze). W celu użycia kapitalików jako wartość podajemy *small-caps*, w przeciwnym wypadku, gdy chcemy zwykłą czcionkę – *normal*.

Bardzo często zdarza się, że część tekstu powinna się wyróżniać od reszty. W tym celu możemy użyć tekstu pochylonego (przedstawionego wyżej) lub pogrubionego. Pogrubienie ustawiamy za pomocą atrybutu *font-weight*. Podobnie jak w większości atrybutów tego typu, wartość *normal* oznacza brak pogrubienia, natomiast *bold* – pogrubienie.

Ostatnim atrybutem z rodziny font jest *font-size*, który pozwala na ustawienie rozmiaru czcionki. Jako jego wartość należy podać liczbę oraz jednostkę wielkości (np. pixele – px, punkty – pt, procenty - %).

Podobnie jak dla ustawień tła, możemy zgrupować wszystkie ustawienia czcionki w jednym atrybucie (o nazwie *font*), dla którego wartości należy podawać w takiej kolejności, w jakiej omówiliśmy poszczególne atrybuty.

W przypadku akapitów najczęściej stosuje się wcięcie w pierwszej linii, które pozwala odróżnić je od siebie. W celu osiągnięcia tego efektu używamy atrybutu *text-indent*, dla którego jako wartość podajemy wielkość wcięcia w pikselach, punktach lub procentach.

Sterowanie wyrównaniem tekstu odbywa się za pomocą atrybutu *text-align*, który przyjmuje jako wartości:

- left – wyrównanie do lewej
- right – wyrównanie do prawej
- center – wyśrodkowanie
- justify – wyjustowanie – wyrównanie tekstu do lewej i prawej strony

Czasami spotykamy się z tekstem, który jest podkreślony lub przekreślony. Taki wygląd tekstu możemy osiągnąć poprzez użycie atrybutu *text-decoration*:

- underline – podkreślenie
- overline – linia nad tekstem
- line-through – przekreślenie

## Pseudoklasy

Pseudoklasy są specjalnym rozszerzeniem selektorów, które mogą być używane do dodawania specjalnych właściwości. Nazwę pseudoklasy podajemy po dwukropku po nazwie selektora.

Najczęściej używanymi pseudoklasami są pseudoklasy formatujące znacznik *a*, ponieważ pozwalają na ustawianie różnych stylów w zależności od tego, czy np. link był już odwiedzany bądź znajduje się nad nim kursor myszy. Ich nazwy to:

- link – definiuje styl dla linka, który jeszcze nie był odwiedzany
- visited – styl dla odwiedzonego linka

## Projektowanie statycznych stron www

---

- `hover` – styl linka, nad którym znajduje się kursor myszy
- `active` – styl linka, który jest aktualnie wybrany

np. kod:

```
a:hover {color:#00FF00; }
```

spowoduje podświetlenie linka na zielono po najechniu na niego myszą.

Inną, często używaną pseudoklasą jest *first-letter*, która służy do nadaniu innego stylu dla pierwszej litery np. akapitu.

### Zadanie

Stwórz stronę prostą HTML, a następnie przetestuj na niej wszystkie omówione w instrukcji style.