

# I. HTML i CSS



## Szablon dokumentu HTML

```
<!doctype html>
<html>
  <head>
    <meta charset="UTF-8" />
    <title>Tytuł strony...</title>
  </head>
  <body>
  </body>
</html>
```

1. Pierwszy wiersz tworzy tak zwana deklaracja typu dokumentu, reprezentowana przez znacznik **<!DOCTYPE>**. Deklaracja typu dokumentu informuje przeglądarkę WWW, jak ściśle powinna się ona trzymać standardu języka HTML w czasie interpretowania kodu strony. Powyższa deklaracja również podkreśla, że strona WWW zawiera kod zapisany w zgodzie ze standardami HTML
2. Pomędzy znacznikami **<html>** i **</html>** umieszczmy wszystkie elementy składające się na stronę WWW. Za znacznikiem **</html>** nie może znajdować się już nic.
3. Kolejne dwa wiersze zawierają znaczniki otwierający i zamykający elementu **head**, deklarującego obszar nagłówka strony. Nagłówek strony nie jest nigdzie wyświetlany.
4. Elementy wyświetlane na stronie umieszczamy pomiędzy znacznikami **<body>** **</body>**.

**Znaczniki meta** umieszczone w sekcji nagłówka dokumentu HTML służą do wskazywania zakodowanych informacji i określania **metadanych** (nie pojawiają się na stronie internetowej).

```
<meta charset="UTF-8" />
```

```
<meta name="description" content="Opis strony"/>
```

```
<meta name="keywords" content="słowo_kluczowe_1, słowo_kluczowe_2"/>
```

```
<meta name="author" content="Autor" />
```

```
<meta name="copyright" content="właściciel praw autorskich" />
```

```
<meta name="robots" content="follow"/>
```

```
<meta name="robots" content="noindex"/>
```



## Styl wpisany

Używamy ich najczęściej gdy chcemy zmodyfikować styl pojedynczego elementu.

**<znacznik style="własność: wartość">**

**...**

**</znacznik>**

Przykład:

```
<body style= "background-color: #c4c59c;  
              min-width: 800px;  
              max-width: 800px;  
              margin: 0 auto;" >  
</body>
```



## Styl osadzony

Ogólna postać stylu osadzonego wygląda następująco:

```
<style type=„text/css”>
  <!--
    selektor{własność: wartość; własność: wartość...}
  -->
</style>
```

# Rodzaje arkuszy stylu: styl osadzony



**Przykład:**

```
<style type="text/css">
  <!--
    body
    {
      background-color: black;
      min-width: 800px;
      max-width: 800px;
      margin: 0 auto
    }
    h1 {color: blue; text-align: center}
    h2 {color: blue; text-align: left}
    p
    {
      color: white;
      font-family: arial, sans-serif;
      font-size: small
    }
  -->
</style>
```

## Rodzaje arkuszy stylu: zewnętrzne arkusze stylu

### Jak przygotować plik CSS?

Zewnętrzny arkusz stylów możemy stworzyć w dowolnym edytorze. Plik nie powinien zawierać znaczników HTML.

### Przykład zawartości pliku:

```
hr {color: red}
h1 {text-align: center}
p {
    text-align: center;
    text-size: large
}
```

Tak przygotowany plik zapisujemy na dysku - nazwa dowolna, rozszerzenie **.css** np.”styl.scc”



## Rodzaje arkuszy stylu: zewnętrzne arkusze stylu

---

Jak dołączyć plik CSS do dokumentu HTML?

Polecenie link umieszczamy w sekcji HEAD dokumentu:

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

gdzie: style.css – nazwa arkusza stylu (nie musi być właśnie taka)

KLASY, czyli jak nadać różne style dla jednego typu elementu.

**element.nazwa\_klasy {właściwość: wartość...}**

**Przykład:**

```
h1.zielony {color: green; text-align:center}
```

```
h1.niebieski {color: blue; text-align:center}
```

```
p.arial {font-family: arial, sans-serif}
```

```
p.courier_new {font-family: "courier new", serif}
```

Zastosowanie przykładu z poprzedniej strony w dokumencie HTML:

```
<body>
    ....
    <h1 class="zielony">Ten nagłówek będzie zielony</h1>
    <h1 class="niebieski">Ten nagłówek będzie niebieski</h1>
    <p class="arial">Ten akapit będzie pisany czcionką ARIAL</p>
    <p class="courier_new">Ten akapit będzie pisany czcionką COURIER NEW</p>
    ....
</body>
```

# KLASY – niezwiązane z konkretnym znacznikiem



Klasy niezwiązane z konkretnym obiektem.

**.nazwa\_klasy {właściwość: wartość...}**

**Przykład:**

```
.zielony {color: green; text-align: center}
.niebieski {color: blue; text-align: center}
.arial {font-family: arial, sans-serif}
.courier_new {font-family: "courier new", serif}
```

# KLASY – niezwiązane z konkretnym znacznikiem

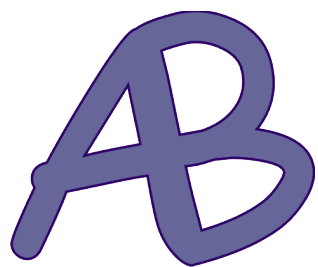


Zastosowanie przykładu z poprzedniej strony w dokumencie HTML:

**Zastosowanie:**

```
<body>
    ....
    <h1 class="zielony">Ten nagłówek będzie zielony</h1>
    <p class="zielony">Teraz kolorem zielonym pisany jest akapit</h1>
    <p class="arial">Ten akapit będzie pisany czcionką ARIAL</p>
    <h1 class="courier_new">Ten nagłówek będzie pisany czcionką COURIER NEW</h1>
    ....
</body>
```

ciąg dalszy nastąpi...



## **II. Formatowanie tekstu**



**Elementy liniowe** – mogą być wmontowane w blok tekstu (np. w akapit)

- ✓ Formatujemy kolor, podkreślenie itp.
- ✓ Nie można im nadać marginesów (wewnętrznych i zewnętrznych), interlinii itp.

To jest przykładowy tekst. To jest przykładowy tekst. **To jest przykładowy tekst.** To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst.

- ✓ Do elementów liniowych należą:  
<a>, <sup>, <sub>, <pre>...
- ✓ Do elementów liniowych odnoszą się właściwości z rodziny „font” (np. font-family)

**Elementy blokowe** – wyznaczają bloki tekstu. Blok jest wpisany w prostokąt.

- ✓ Można formatować marginesy, interlinię, obramowania, kolor tła (dla całego bloku), wcięcia...
- ✓ Blokom można nadawać wszystkie aspekty formatu elementów liniowych.

- ✓ Do elementów liniowych należą: `<p>`, `<li>`, `<ul>`, `<h1>`...
- ✓ Do bloków odnoszą się własności z rodziny „text” (np. text-align)

To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst.

**To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst.**

To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst. To jest przykładowy tekst.





**Kontenery** – znaczniki, które same w sobie niczemu nie służą. Grupują element, którym możemy nadać wspólne formatowanie

- ✓ kontener na elementy liniowe (to nie do końca książkowe określenie)

`<span>...</span>`

- ✓ kontener na elementy blokowe

`<div>....</div>`



Jedną z najważniejszych decyzji, które należy podjąć przed rozpoczęciem pisania reguł stylistycznych, jest rodzaj użytych jednostek:

- ✓ bezwzględne (punkty, cale itd.)
- ✓ czy względne (procenty, em).

Stara metoda polegała na stosowaniu pikseli, obecnie nastąpił zwrot ku jednostkom względnym. Oto zalety i wady tej metody:



### Zalety:

1. Rozmiar tekstu zmienia się proporcjonalnie w każdym miejscu po użyciu opcji zmiany rozmiaru tekstu w menu Widok przeglądarki. Jest to bardzo przyjazne podejście do użytkownika.
2. Dopracowując szczegóły projektu można **proporcjonalnie zmienić rozmiar całego tekstu na stronie, modyfikując tylko wartość dla elementu body**. Zmieniany jest tylko rozmiar podstawowy, a wszystkie pozostałe rozmiary dostosowują się proporcjonalnie do tej wartości.



### Wady:

1. Przez chwilę nieuwagi można spowodować, że tekst w zagnieżdżonych elementach będzie bardzo mały ponieważ rozmiar czcionki jest dziedziczony.
2. Użytkownik może zburzyć układ strony, który nie został przystosowany do zmiany rozmiaru tekstu. Jeśli na przykład użytkownik ustawi rozmiar czcionki na dużą wartość, pływające kolumny mogą poprzesuwać się w różne miejsca, jeśli przestaną mieścić się w tych, które im pierwotnie wyznaczono.

(Na wykładzie przyjrzymy się tworzeniu zaawansowanych układów stron przy użyciu CSS, zajmiemy się także problemem i sposobami jego rozwiązania)

## Rozmiar czcionki

---



Rozmiar czcionki możesz określić za pomocą atrybutu stylu font-size:

**font-size: 16px;**



## Jednostki bezwzględne:

- **px** – punkty ekranu
- **pt** – punkty (1 punkt to 1/72 cala)

Uwaga! Jednostek **pt** używamy do definiowania wydruku. Aby przygotować wydruk ze strony w oddzielnym pliku CSS definiujemy rozmiary czcionki za pomocą pt.

$$1\text{pt} = \frac{96}{72}\text{px} = \frac{4}{3}\text{px} = 1,33\text{px}$$

$$1\text{px} = \frac{72}{96}\text{pt} = \frac{3}{4}\text{pt} = 0,75\text{pt}$$



## Imienne wartości absolutne:

**xx-small** - najmniejsza

**x-small** - bardzo mała

**small** - mała

**medium** - średnia

**large** - duża

**x-large** - bardzo duża

**xx-large** - największa

rozmiar **xx-small**

rozmiar **x-small**

rozmiar **small**

rozmiar **medium**

rozmiar **large**

rozmiar **x-large**

rozmiar **xx-large**



Jednostki względne (odnoszące się do jednostki referencyjnej).

**em** – jednostka pochodzi od szerokości małej litery „m”

Rozmiar czcionki ustalany w oparciu o jednostki em jest relatywny w stosunku do rodzica elementu. Rodzicem wszystkich elementów pierwszego poziomu (uwaga na zagnieżdżanie) jest element `<html>`. Domyślna wartość rozmiaru czcionki dla tego elementu w większości przeglądarek to **16px** (1em = 16px).

**%** - procent – działają na podobnej zasadzie jak jednostki em i odnoszą się do rodzica w ustalaniu rozmiaru czcionki.





Imienne wartości względne:

smaller - mniejsza od bieżącej

larger - większa od bieżącej

rozmiar smaller

czcionka bieżąca

rozmiar larger



Przy wyborze kroju pisma masz do dyspozycji wszystkie kroje zainstalowane na Twoim komputerze. Pamiętaj jednak, że inni użytkownicy wcale nie muszą nimi dysponować! Z tego powodu powinieneś ograniczać się c standardowych rodzin krojów pisma:

- ✓ **serif** — czcionka szeryfowa, np. Times New Roman,
- ✓ **sans-serif** — czcionka bezszeryfowa, np. Arial,
- ✓ **cursive** oraz **fantasy** — ozdobne czcionki pochylę (rzadko używane, gdyż każda przeglądarka może podstawić w ich miejsce całkowicie inne kroje pisma — często w ogóle nieprzystające do zamysłu autora strony),
- ✓ **mono** — czcionka nieproporcjonalna (o stałej szerokości wszystkich znaków) używana najczęściej do zapisywania tekstu programów komputerowych, tworzenia prymitywnych tabel lub wyróżniania poleceń programów; np. Courier New.



Wyboru kroju pisma dokonuje się za pomocą atrybutu stylu o nazwie font-family:

**font-family: „times new roman”, serif;**

/ font Times New Roman, jeżeli nie jest dostępny to inna czcionka szeryfowa/



- ✓ Tekst preformatowany (spacje i końce linii mają znaczenie) –  
`<pre>...</pre>`
- ✓ Tekst pisany czcionką o stałej szerokości (tekst maszynowy) –  
`<tt>...</tt>`
- ✓ Marginesy akapitów –  
`<p style="margin-top: 5pt; margin-bottom: 3pt;">...</p>`  
`<p style="margin: 10px 10px 10px 10px;">`  
/gdzie wartości oznaczają margin: górny prawy dolny lewy;/
- ✓ Marginesy wewnętrzne –  
`<p style="padding: 20px;">`

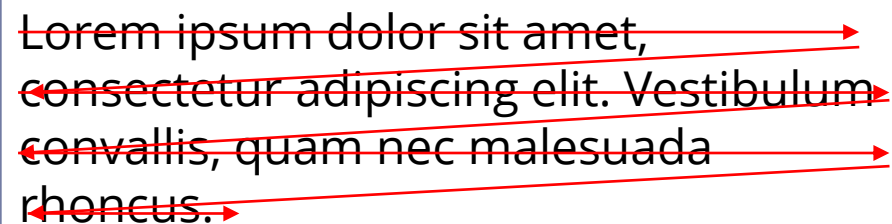


W CSS dostępnych jest osiem własności do formatowania tekstu.

1. `text-indent`
2. `letter-spacing`
3. `word-spacing`
4. `text-decoration`
5. `text-align`
6. `line-height`
7. `text-transform`
8. `vertical-align`

Bardzo ważne jest, aby używając własności tekstu, wiedzieć, że w CSS każda linia tekstu znajduje się w prostokątnej ramce.

Na przykład fragment tekstu w akapicie dla CSS jest długą, cienką linią w prostokątnej ramce, podzieloną na poszczególne wiersze.



Lorem ipsum dolor sit amet,  
consectetur adipiscing elit. Vestibulum  
convallis, quam nec malesuada  
rhoncus.



## Własność text-align:

Przykład:

```
p (text-align:right;)  
Wartości: left, right, center, justify
```

Tekst zostaje wyrównany w poziomie wewnątrz zawierającego go elementu. Dlatego własność tę stosuje się do elementu kontenera. Jeśli chcemy aby nagłówek `<h1>` znajdował się na środku elementu `<div>`, własność `text-align` należy zastosować do elementu `<div>`, a nie `<h1>`.



## Własność text-indent

Przykład:

**Przykład:** `P {text-indent: 3em;}`

**Wartości:** wszystkie jednostki miary (wartości mogą być dodatnie i ujemne)

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum convallis, quam nec malesuada rhoncus, odio metus vestibulum quam, in laoreet ex odio scelerisque mi. Sed sit amet egestas augue. Proin fermentum sit amet nibh eget facilisis. Suspendisse ultrices et lacus at placerat..

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum convallis, quam nec malesuada rhoncus, odio metus vestibulum quam, in laoreet ex odio scelerisque mi. Sed sit amet egestas augue. Proin fermentum sit amet nibh eget facilisis. Suspendisse ultrices et lacus at placerat..





## Własność letter-spacing i word-spacing

Przykład:

Przykład: `p {letter-spacing: 2em;}`

`p {word-spacing: 2em;}`

Wartości: wszystkie jednostki miary (wartości mogą być dodatnie i ujemne)

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum convallis, quam nec malesuada rhoncus, odio metus vestibulum quam, in laoreet ex odio scelerisque mi. Sed sit amet egestas augue.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum convallis, quam nec malesuada rhoncus, odio metus vestibulum quam, in laoreet ex odio scelerisque mi. Sed sit amet egestas augue.

- `<p style="background-color: red; color: blue;"> ..</p>`  
- zdefiniowane kolory dla akapitu (niebieski tekst na czerwonym tle);
- `<body style="color: #ffffff; background-color: #000000">`  
- zdefiniowano kolory dla całej strony (biały tekst na czarnym tle)
- `<span style="color: green;">...</span>` - zmiana koloru fragmentu akapitu, pojedynczego wyrazu lub nawet litery.