

# *Tworzenie Stron Internetowych*

*odcinek 5*

# zawartość nagłówka <head>

---

Nagłówek <head> zawiera podstawowe informacje o dokumencie, takie jak:

- <title> – tytuł strony (**obowiązkowy**)
- <meta> – metainformacje/metadane (**obowiązkowa deklaracja kodowania znaków**)
- <link> – relacje z innymi dokumentami
- <script> – skrypty do interakcji z użytkownikiem (np. JavaScript)
- <style> – arkusze stylów osadzone bezpośrednio w dokumencie
- <base> – podstawowa ścieżka dostępu dla wszystkich ścieżek względnych

Zawartość <head> nie jest wyświetlana przez przeglądarkę.

Pominięcie samego znacznika <head>, z zachowaniem jego zawartości, jest możliwe. Zaleca się jednak jego używanie.

## Tytuł strony <title>

Podanie tytułu strony jest obowiązkowe. Jego brak jest niezgodny ze specyfikacją (x)HTML. Tytuł wyświetlany jest w zakładce przeglądarki, w katalogu „ulubione strony” i w wynikach wyszukiwarek, czyli odpowiedni dobór tytułu jest ważny. Maksymalna długość tytułu nie powinna przekraczać 40 znaków. Jeżeli mamy więcej podstron dobrze jest, aby miały one nieco inny tytuł, odpowiedni do zawartości.

```
<title>Tytuł strony</title>
```

# zawartość nagłówka <head>

---

## Element <meta>

Metadane (dane o danych) nie wpływają na wygląd strony, ale niosą ważne informacje przydatne np. przeglądarkom i wyszukiwarkom internetowym. Użycie <meta /> nie jest obowiązkowe, ale zalecane. Metadane umieszcza się w parach nazwa-wartość (atrybuty **name/http-equiv** i **content**) lub w atrybucie **charset**.

## deklaracja kodowania znaków

Deklaracja podaje jak kodowane są znaki na stronie. Jest konieczne do poprawnego wyświetlania np. polskich znaków. Nie należy używać kodowania ~~windows-1250~~, ponieważ działa ono tylko w systemie Windows. Do kodowania polskich znaków używa się często **iso-8859-2**, jednak zaleca się korzystać z bardziej uniwersalnego standardu **Unicode UTF-8**. *Uwaga:* wybór kodowania zależy od języka, w którym strona jest napisana. Standard **Unicode UTF-16** umożliwia używanie na stronie wielu języków jednocześnie. Wadą UTF-16 jest większy rozmiar plików.

Deklaracje dla języka polskiego:

```
<meta charset="iso-8859-2">
```

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

zalecana

**Uwaga:** Kodowanie znaków tekstu umieszczanego w edytorze HTML musi być takie samo, jak w deklaracji <meta />. Należy na to zwrócić uwagę przy zapisywaniu pliku HTML.

# zawartość nagłówka <head>

---

## Element <meta>

### opis zawartości strony

Krótki (do około 150 znaków) opis zawartości dokumentu. Wyświetlany jest w wynikach wyszukiwarek.

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

### słowa kluczowe

Zestaw kilku, kilkunastu słów, które najlepiej opisują zawartość strony. Słowa kluczowe ułatwiają wyszukiwarkom skatalogowanie danej strony i umieszczenie jej wyżej w liście wyników (pozycjonowanie).

```
<meta name="keywords" content="wyraz1, wyraz2, wyraz3..." />
```

### autor strony

Podaje kto jest autorem strony (informacja dla osób czytających kod źródłowy).

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

# zawartość nagłówka <head>

---

## Element <meta>

### edytor

Podaje nazwę edytora HTML, którego użyto do napisania strony.

```
<meta name="generator" content="nazwa edytora" />
```

### okno podglądu (viewport)

Umożliwia dostosowanie szerokości wyświetlania strony do szerokości wyświetlacza. Element istotny dla stron responsywnych. Więcej informacji na ten temat pojawi się przy omawianiu responsywności.

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Pierwsza część atrybutu **content** wskazuje, że strona ma dostosować się szerokością do wyświetlacza. Druga część ustawia poziom powiększania strony przy jej załadowaniu.

### odświeżanie/przekierowanie

Określa po jakim czasie przeglądarka ma odświeżyć stronę lub przejść na inny adres internetowy. Nie należy nadużywać automatycznego odświeżania.

```
<meta http-equiv="refresh" content="czas w sek." />
```

```
<meta http-equiv="refresh" content="czas w sek.; URL=http://jakiś.adres.internetowy" />
```

# zawartość nagłówka <head>

---

## Element <meta>

### Roboty

Polecenie określa zachowanie programów zbierających informacje o stronach w Internecie (tzw. roboty indeksujące). Z tych informacji korzystają wyszukiwarki. Klucz **robots** pozwala autorom np. wyłączyć wybrane strony z indeksu wyszukiwarek.

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

dyrektywy:

- **"index"** – strona, na której wstawiono polecenie będzie indeksowana przez roboty sieciowe (domyślnie)
- **"follow"** – indeksowanie linków znajdujących się w dokumencie, w celu późniejszego odwiedzenia ich przez roboty (domyślnie)
- **"noindex"** – strona, na której wstawiono polecenie nie będzie indeksowana
- **"nofollow"** – robot nie indeksuje stron wskazywanych przez odsyłacze wstawione w danym dokumencie; *uwaga*: strony te mogą być mimo to zaindeksowane, jeżeli istnieją odnośniki do nich umieszczone w innych dokumentach bez tego zakazu
- **"index, nofollow"** – indeksuje stronę, nie indeksuje stron wskazywanych przez odsyłacze
- **"noindex, follow"** – nie indeksuje strony, na której wstawiono polecenie, indeksuje strony wskazywane przez odsyłacze
- **"all"** (= "index, follow") – indeksuje wszystko (domyślnie)
- **"none"** (= "noindex, nofollow") – nie indeksuje nic, nie podąża za odsyłaczami

Możliwe jest blokowanie indeksowania wybranego linku: <a href="http://jakis\_adres.cos" **Rel="nofollow"**>

# zawartość nagłówka <head>

---

## Element <link>

Element ten podaje relacje z innymi dokumentami, np. wskazanie na arkusze stylów CSS, alternatywne formatowanie strony, ikonę strony.

### arkusz stylów CSS

W poniższy sposób wskazujemy, gdzie znajduje się plik zawierający tzw. zewnętrzny arkusz stylów, który nadaje wygląd naszej stronie WWW. Arkusz podstawowy:

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

i alternatywny (jeśli jest):

```
<link rel="alternate" type="text/css" href="inny-plik.css" title="nazwa" />
```

### arkusz stylów dla innych urządzeń

Możemy udostępnić alternatywny arkusza stylów dla wyświetlenia strony na innych urządzeniach, np. wersję na drukarkę, która będzie mniej rozbudowana kolorystycznie i mniej obciążająca toner przy wydruku strony.

```
<link rel="stylesheet" type="text/css" href="plik.css" media="print" />
```

# zawartość nagłówka <head>

---

## Element <link>

### ikona strony (ulubionych)

Możemy dołączyć ikonę favicon (favorites icon), która będzie widoczna w pasku adresu w przeglądarce, na kartach i w zakładkach. Obrazek powinien być kwadratowy i mieć rozmiar 16x16 pikseli lub 32x32, 48x48, 64x64 pikseli (lista rozmiarów jest dość długa).

```
<link rel="icon" href="ścieżka dostępu do favicon" />
```

```
<link rel="icon" type="image/x-icon" href="ścieżka dostępu do favicon.ico" />
```

```
<link rel="icon" type="image/png" href="ścieżka dostępu do favicon.png" />
```

```
<link rel="icon" type="image/gif" href="ścieżka dostępu do favicon.gif" />
```

podana powyżej *ścieżka dostępu do favicon* może wyglądać na przykład tak:

```
http://www.astro.uni.wroc.pl/favicon.ico
```

```
/favicon.ico
```

```
/images/favicon.ico
```

Jeśli ikona znajduje się na naszym serwerze, to ścieżka dostępu musi być względna (tak, jak przy wstawianiu np. grafiki).

Najlepsze typy plików graficznych, w których możemy zapisać favicon to: .ico, .png, .gif. Typ obsługiwanej grafiki zależy od wersji przeglądarki. W Internecie znajduje się sporo narzędzi on-line umożliwiających tworzenie favicon w formacie .ico, np.: [www.favicon.cc](http://www.favicon.cc).



# zawartość nagłówka <head>

---

## Element <script>

Element ten definiuje skrypty uruchamiane po stronie użytkownika, takie jak JavaScript. Zawiera on polecenia języka skryptowego (*linie oznaczone na niebiesko występują tylko w XHTML*):

```
<script type="application/javascript">
  
    var i=10;
    if (i&lt;5)
    {
      // jakiś kod
    }
  ]&gt;
&lt;/script&gt;</pre></div><div data-bbox="320 402 479 459" data-label="Text"><p>na szaro oznaczony<br/>jest kod w JS</p></div><div data-bbox="32 604 783 636" data-label="Text"><p>lub wskazuje na plik zawierający te polecenia i wtedy wewnątrz elementu jest puste:</p></div><div data-bbox="52 652 588 683" data-label="Text"><pre>&lt;script type="application/javascript" src="plik.js"&gt;&lt;/script&gt;</pre></div><div data-bbox="32 710 937 811" data-label="Text"><p>Umieszczając skrypty (np. JS) warto pamiętać, że przeglądarka użytkownika może nie obsługiwać lub mieć wyłączoną obsługę skryptów. Dla takich użytkowników powinniśmy dodać element &lt;noscript&gt; umożliwiający podanie informacji o tym, że strona wymaga uruchomienia obsługi skryptów.</p></div><div data-bbox="52 827 730 858" data-label="Text"><pre>&lt;noscript&gt;Uwaga, Twoja przeglądarka nie obsługuje JavaScript!&lt;/noscript&gt;</pre></div>
```

# zawartość nagłówka <head>

---

## Element <style>

Element ten służy do osadzania w kodzie HTML tzw. wewnętrznego arkusza stylów CSS. Wnętrze tego elementu to reguły opisujące wygląd poszczególnych elementów HTML. W poniższych przykładach reguły te oznaczone są na szaro.

```
<style>  
  h1 {color:red;}  
  p {color:blue;}  
</style>
```

Możliwe jest też udostępnienie alternatywnego arkusza CSS formatującego stronę na urządzenia inne niż standardowy monitor, np. na drukarkę.

```
<style>  
  h1 {color:#FF0000;}  
  p {color:#0000FF;}  
  body {background-color:#FFefd6;}  
</style>
```

wersja  
ekranowa

```
<style media="print">  
  h1 {color:#000000;}  
  p {color:#000000;}  
  body {background-color:#FFFFFF;}  
</style>
```

wersja  
drukarkowa

# zawartość nagłówka <head>

---

## Element <base>

Element określa podstawową ścieżkę dostępu dla wszystkich ścieżek względnych. Dzięki temu możemy skrócić zapis ścieżek w dalszej części kodu.

```
<base href="http://www.jakaś.domena/foto/" />
```

Dzięki powyższej deklaracji możemy napisać:

```
<body>  
    
  ...  
</body>
```

zamiast

```
<body>  
    
  ...  
</body>
```

Innym zastosowaniem <base /> jest wskazanie sposobu otwierania wszystkich odnośników.

<base target="_blank">	– w nowej karcie/oknie
<base target="_self">	– w tej samej karcie/oknie
<base target="_parent">	– w tej samej ramce (używane z <iframe>)
<base target="_top">	– w pełnym oknie (używane z <iframe>)

Oba argumenty <base /> mogą być używane jednocześnie:

```
<base href="http://www.jakaś.domena/foto/" target="_blank">
```