
HTML - HyperText Markup Language

- HTML czyli tzw. hipertekstowy język znaczników, który opisuje wygląd strony internetowej. Za jego pomocą można osadzić na stronie tekst, grafikę, hiperłącza, tabele, formularze itd. W składni języka HTML wykorzystuje się charakterystyczne znaczniki (tagi) opatrzone z obu stron nawiasami ostrokątnymi.

Extensible Markup Language = XML

- to uniwersalny język formalny przeznaczony do reprezentowania różnych danych w strukturalizowany sposób - XML jest podzbiorem starszego standardu opisu dokumentu o nazwie SGML, który eliminuje jego zbyt skomplikowany charakter tam gdzie tylko można to uczynić;
- Przede wszystkim nie jest to język do tworzenia dokumentów, lecz język do tworzenia innych języków. Dopiero te języki służą do budowania dokumentów w Internecie;
- Jest niezależny od platformy, co umożliwia łatwą wymianę dokumentów pomiędzy różnymi systemami i znacząco przyczyniło się do popularności tego języka w dobie Internetu;
- XML jest standardem rekomendowanym oraz specyfikowanym przez organizację W3C (międzynarodowa organizacja mająca za zadanie ustanawianie i rozwój standardów w sieci Internet)

- Rozwinięcie akronimu HTML
(Hypertext markup language)
niewiele nam tak naprawdę mówi. Możemy
się jedynie dowiedzieć, że jest to:
- **język** – language
- **znaczników** – markup (np. <p> – znacznik
paragrafu)
- **hipertekstowy** – hypertext

- Oczywiście rozwijając skrót Html po polsku, zazwyczaj wypowiada się go jako **hipertekstowy język znaczników**. Jednak w powyższym zestawieniu rozpocząłem od słowa język, z tego względu, że pewnie będzie one na początku najbardziej zrozumiałe. Język html to po prostu **język komputerowy** (służący do komunikacji z komputerem), który używany jest do opisu stron internetowych.
- Co to są te **znaczniki**? Znaczniki to takie wyrażenia w ostrych nawiasach <>, z którymi już się zapewne nie raz spotkaliście np. <head>, dlatego mamy **język znaczników** (nie mylić z językiem programowania).

HTML4, czy HTML5

- Szczególnie osobom, które dopiero zaczynają przygodę z html'em łatwo będzie się pogubić w tym całym nomenklaturowym galimatiasie. O co w ogóle chodzi z tymi cyferkami?

- html jest stale rozwijającym się standardem.
- Za jego rozwój oficjalnie odpowiedzialna jest organizacja **W3C** (World Wide Web Consortium).
- Oficjalnie.., bo od 2004 mamy drugą organizację, która zajmuje się równolegle rozwojem tego standardu mianowicie **WHATWG** (Web Hypertext Application Technology Working Group). Przyczyną powstania tej organizacji był rozłam i nieporozumienie pewnych panów w konsorcjum W3C. W praktyce jedna i druga przyczynia się do rozwoju standardu Html, a obie organizacje częściowo współpracują ze sobą.

- 28 października 2014 r. specyfikacja HTML5 osiągnęła status rekomendacji W3C

- Organizacja **WHATWG** rozwijając nowy standard odnosiła się do niego jako HTML5. Natomiast **W3C** głosowało za przyjęciem nazwy **HTML 5** ze spacją. Od tej pory powstał rozłam nazw, który nie był dobry dla marketingu tegoż nowego standardu. Jednak w końcu ustalono, że jedna i druga organizacja, a także jego oficjalna nazwa będzie bez spacji i będzie brzmiała **HTML5**.

To którego standardu mam się uczyć HTML4 czy HTML5?

- Jak najbardziej HTML5, a to dlatego, że nowy standard jest rozwinięciem poprzedniego. Jeśli ktoś zna HTML w wersji czwartej wystarczy tak na prawdę, że uzupełni swoją wiedzę o wprowadzone zmiany i będzie mógł się bezproblemowo przesiąść oficjalnie na nowy standard. Kilka rzeczy zostało usuniętych ze standardu czwartego, aby poprawić czytelność kodu i kilka rzeczy zostało wprowadzonych, aby wspomóc użytkowników. W gruncie rzeczy te dwa standardy, aż tak mocno się nie różnią od siebie, jeśli chodzi o tworzenie stron internetowych.

Znaczniki języka Html

- Znaczniki są to wyrażenia, które mieszczą się pomiędzy znakami znanymi nam z matematyki. Znaki „mniejsze niż” oraz „większe niż” $<$ $>$. Pomiedzy tymi znaki wstawiamy wyrażenia z języka html i w ten sposób tworzymy znaczniki, np. `<p>`.

Znacznik zamknięcia

- Znaczniki możemy podzielić na **znaczniki otwarcia** oraz **zamknięcia**. Znaczniki otwarcia od znaczników zamknięcia różnią się jedynie pionową kreską pochyłoną w prawo, tzw. **forward slash**, który poprzedza wyrażenie html. Pomiędzy znacznikami otwarcia oraz zamknięcia znajduje się miejsce obowiązywania elementu. Dla przykładu możemy podać znacznik akapitu `<p>`, treść akapitu wstawiamy pomiędzy znacznikami `<p>` oraz `</p>`

Deklaracja Html

- **<!DOCTYPE html>**
- Pierwszą linią kodu jaką napiszemy, będzie deklaracja dokumentu. W skrócie jest to informacja dla przeglądarki z jakim typem dokumentu ma do czynienia.
- Zapis <!DOCTYPE html> informuje przeglądarkę, że jest to dokument html (strona internetowa) i może go przetworzyć (wyświetlić na naszym ekranie).

Znacznik HTML

<html>

- Po deklaracji rodzaju dokumentu, czas na deklarację języka, w którym dokument zostanie napisany. Deklaracji języka html dokonujemy znacznikiem <html>. Dodamy teraz ten znacznik, pod znacznikiem deklaracji dokumentu.

<!DOCTYPE html>

<html>

</html>

Znacznik HEAD

- Oznaczyliśmy już miejsce, w którym będziemy używać języka html, teraz czas na ustawienia dokumentu html, a mianowicie element `<head>` (ang. głowa, sekcja nagłówkowa). Element head opisuje ustawienia dokumentu html.
- Pomędzy znacznikami `<head>` i `</head>` zamieszcza się informacje m.in o rodzaju języka, którym będziemy się posługiwać (polski, angielski, itd.).
- Element head jest elementem języka html, dlatego umieścimy go pomiędzy znacznikami html. W takiej sytuacji mówimy o tym, że znacznik `<html>` jest rodzicem znacznika `<head>` i w drugą stronę, że ten ostatni jest dzieckiem tego pierwszego.

Znacznik Body

`<body>`

- Ustawienia strony trafiają do sekcji wyznaczonej przez element `<head>`, natomiast treść strony internetowej, czyli wszystko to co zostanie wyświetlone na ekranie musi znaleźć się w elemencie `<body>` (ang. ciało strony, dokumentu).
- Pomędzy znacznikami `<body>` oraz `</body>` lokujemy wszystkie informacje, które chcielibyśmy umieścić na stronie. Zaktualizujmy teraz nasz dokument o nowy znacznik.

Struktura Strony Internetowej

- Cały dokument html będzie zawsze zawierać się pomiędzy znacznikami `<html>` oraz `</html>`. Ustawienia strony zawsze umieścimy pomiędzy znacznikami `<head>` oraz `</head>`, a treść strony będziemy wpisywać pomiędzy znacznikami `<body>` oraz `</body>`.

html



```
graph TD; html[html] -- contains --> head[head]; html -- contains --> body[body];
```

The diagram illustrates the structure of an HTML document. It consists of a light blue outer rectangle labeled 'html'. Inside this rectangle, there are two smaller rectangles: a light blue one labeled 'head' at the top and a darker blue one labeled 'body' below it. The 'head' and 'body' rectangles are separated by a thin white horizontal line.

head

body

Polskie Znaki HTML

- **<meta charset="utf-8">**
- Zanim zaczniemy wypełniać treścią naszą stronę internetową, musimy jeszcze dodać jedną linię kodu. Musimy zadeklarować zestaw znaków jakiego będziemy używać.
- Deklaracja ta warunkuje poprawne wyświetlanie polskich liter, w szczególności tych z ogonkami (ąćęłńśóźż). Dokonujemy tej deklaracji w sekcji <head>.

Meta

- Elementu <meta> będziemy używać częściej, gdyż przy jego pomocy tworzy się tzw. **meta tagi**, czy meta elementy. Są to dane informujące przeglądarkę o ustawieniach strony internetowej i dlatego wszystkie meta tagi trafią do sekcji <head>.

Charset

- Natomiast wartość atrybutu charset (ang. character set -zestaw znaków), czyli **utf-8** informuje przeglądarkę o zestawie znaków, którego chcemy użyć. O atrybutach i ich wartościach szerzej powiemy podczas kolejnych lekcji.

- **UTF-8 vs ISO-8859-2**
- Zestaw znaków **utf-8** jest obecnie najczęściej stosowanym standardem i obsługuje on również polskie znaki. Na starych stronach można się jeszcze spotkać z deklaracją *ISO-8859-2*.

Formatowanie Tekstu HTML

- W naszym tekście mogą znaleźć się pojedyncze wyrazy, zdania bądź całe bloki tekstu, które będą trochę różnić się od reszty strony i dlatego będą potrzebowały specjalnego oznaczenia elementami HTML. Możemy dla przykładu **wytłuszczyć** ważniejsze frazy, żeby czytelnik szybciej dotarł do interesujących go treści.

Formatowanie Tekstu – Elementy HTML

- wyraz **pogrubiony**
- wyraz **pogrubiony**
- wyraz *pochylony*
- <i> wyraz *pochylony*
- <small> wyraz pomniejszony
- <mark> podświetlony wyraz
- <sup> indeks ^{górny}
- <sub> indeks _{dolny}
- <s> wyraz ~~nieaktualny~~
- <ins> wyraz wstawiony
- wyraz usunięty

Pogrubienie tekstu HTML

- ** i **
- Pogrubienia tekstu w języku HTML możemy dokonać za pomocą elementu **** oraz ****.
- Znacznik **** (ang. **mocny**, bądź wzmocnienie) jest nowym znacznikiem wprowadzonym w specyfikacji HTML5.
- Znacznik **** (**bold** ang. **pogrubienie**) używany był i jest nadal od dawna.

- `<p>Ten wyraz jest pogrubiony</p>`
`<p>Ten wyraz jest pogrubiony</p>`
- Ten wyraz jest **pogrubiony**
Ten wyraz jest **pogrubiony**

** czy **

- Skoro na ekranie wyglądają tak samo, to jaka jest różnica między nimi i który stosować?
- Element `` został wprowadzony, aby nadać znaczenie wyrażeniu, które znajdzie się pomiędzy jego znacznikami. Wyrażenie to ma niezwykłą wagę w odniesieniu do reszty tekstu.
- Element `` nie nadaje specjalnego znaczenia wyrażeniu, a jedynie formatuje jego wygląd, pogrubiając czcionkę.
- Zaleca się używania elementu ``.

Pochylenie Tekstu HTML

- `` i `<i>`
- Pochylenia tekstu w języku HTML możemy dokonać poprzez elementy `` oraz `<i>`.
- Element `` (ang. **emphasize** – podkreślenie, uwypuklenie) jest nowym znacznikiem wprowadzonym w specyfikacji HTML5.
- Element `<i>` (ang. **italic** – kursywa, tekst pochyły) używany był i jest nadal od dawna w HTML.

- `<p>`Ten wyraz jest ``pochylony`</p>`
- `<p>`Ten wyraz jest `<i>`pochylony`</i></p>`
- **`` czy `<i>`**
- Skoro na ekranie wyglądają tak samo, to jaka jest różnica między nimi i który stosować?
- Element `` według nowych standardów **HTML5** ma wyróżniać tekst poprzez podkreślenie oraz zwiększenie nacisku na wyrażenie, które znajdzie się pomiędzy jego znacznikami.
- Element `<i>` nie ma żadnego semantycznego znaczenia, służy jedynie do formatowania tekstu poprzez jego pochylenie.

Zmniejszenie Tekstu HTML

- **<small>**
- Element `<small>` (ang. mały) zmniejsza czcionkę o jeden rozmiar. Możemy go wielokrotnie zagnieżdżać, aż do najmniejszego rozmiaru czcionki przeglądarki.

- <p>Ten wyraz jest pomniejszony <small>raz,
<small>dwa,
<small>trzy</small></small></small> razy</p>

Podświetlenie Tekstu HTML

- `<mark>`
- Element `<mark>` (ang. zaznaczać) służy do wyróżniania pojedynczych wyrazów, bądź zwrotów poprzez podświetlenie.

Podświetlenie uzyskiwane jest poprzez dodanie tła do elementu.

`<p>`Ten wyraz jest
`<mark>`podświetlony`</mark></p>`

Indeks Górny Dolny HTML

- `<sup>` i `<sub>`
- **Indeks górny** uzyskujemy poprzez zastosowanie elementu `<sup>` (ang. **sup**erscript – indeks górny). Natomiast **indeks dolny** uzyskujemy poprzez zastosowanie elementu `<sub>` (ang. **sub**script – indeks dolny). Żeby nam się nie myliły możemy skojarzyć sobie słówko **sub** ze słówkiem **subway**, czyli z metrem (nie kanapki), a metro zazwyczaj jeździ pod ziemią, czyli na dole...

- `<p>`Ten wyraz jest w `^{`indeksie górnym`}</p>`
- `<p>`Ten wyraz jest w `_{`indeksie dolnym`}</p>`

Przekreślenie Tekstu HTML

- **<s> i **
- Zarówno element **<s>** (ang. **strikethrough** – linia przez), jak i element ****(ang. **delete** – usuwać) służą do przekreślania tekstu. Jaka różnica występuje między nimi?
- **<s>**
- Elementu **<s>** używamy zawsze, gdy coś jest nieaktualne, a nie powinno być usunięte, np. promocja, która już się skończyła.

- `<s>`Dziś promocja na Crocsy w Lidlu`</s>`
Wyprzedane

- ****
- Natomiast elementu używamy, gdy chcemy zwrócić uwagę, że coś zostało usunięte, np. gdy jakiś artykuł został usunięty ze strony do której linkujemy.
- ****Jak zarobić miliony.**** Artykuł niestety usunięty

- **<ins>**
- Element **<ins>** (ang. **insert** – wstawiać) przydaje nam się w momencie, gdy chcemy dodać coś do artykułu, który już opublikowaliśmy.

Spacja niełamliwa HTML

- ** **;
- Hmm... A jak chcemy użyć więcej niż jednej spacji, to co wtedy? Możemy tego dokonać poprzez dodanie znaku specjalnego. **Znak specjalny** to odpowiednik znaku wprowadzanego z klawiatury (np. spacji) posiadający specjalny zapis html. Każdy znak specjalny zaczyna się od znaku **&** (**ampersand**) i kończy się średnikiem **;**. Znak specjalny niełamliwej spacji to ** **; (ang. **non-breaking space** – spacja niełamliwa).

- Znaków specjalnych używamy, gdy chcemy wyświetlić znaki zarezerwowane w języku html, tj. `<` `>` `'` `"` `&`, bądź gdy dany znak jest ciężki do wprowadzenia, bo nie ma go na klawiaturze. Znaków specjalnych jest cała masa, ale nie będziemy ich teraz podawać, żeby nie zaprzętać sobie nimi głowy niepotrzebnie. Wystarczy jak zapamiętamy znak niełamliwej spacji ` `;

Sekcje

- **<div>**
- Do budowy sekcji możemy użyć elementu blokowego **<div>** (ang. **division** – oddział, wydział, sekcja). Element ten nazywany jest również box'em (pudełkiem), albo kontenerem (ang. container, contain – zawierać w sobie), z tego względu, że zawiera on w sobie inne elementy.

Sekcje w HTML do tej pory tworzone były głównie za pomocą elementu `<div>`. W HTML5 wprowadzono nowe elementy blokowe, które mają zastąpić element `<div>` tam, gdzie będzie to miało sens.

Nowe elementy blokowe w HTML5:

- `article`
- `aside`
- `footer`
- `header`
- `main`
- `nav`
- `section`

header



aside

main

footer

<header>

<nav>

<section>

<aside>

<article>

<footer>

- <header> - definiuje nagłówek dla dokumentu lub sekcji
- <Nav> - Określa kontener dla linków nawigacyjnych
- <Section> - Definiuje sekcję w dokumencie
- <Article> - definiuje niezależną samodzielne artykuł
- <asside> - definiuje zawartość : banery linki itp
- <Footer> - Definiuje stopkę do dokumentu lub sekcji
- <details> - definiuje dodatkowe szczegóły

ELEMENTY BLOKOWE

- Elementy blokowe występują niezależnie/odrębnie od siebie. Umieszczamy je jeden pod drugim, nigdy w jednej linii. Wewnątrz siebie mogą zawierać inne elementy blokowe lub zwyczajny tekst. Z elementów blokowych konstruujemy bloki strony większe, niż z elementów wyświetlanych w linii.

ELEMENTY LINIOWE

- elementy liniowe (wierszowe) nie mogą zawierać wewnątrz siebie elementów blokowych, innych elementów liniowych oraz tekstu. Owszem, mogą występować w jednej linii z racji tego, że nie budują nowego wiersza.
- Elementy liniowe zajmą tylko tyle miejsca ile same potrzebują, pozostawiając miejsce na tej samej linii dla kolejnych elementów liniowych

Elementy blokowe:

- `<h1> </h1> <h2></h2>` itd. do 6
- `<p> </p>`
- `
`
- `<hr />`
- `<pre> </pre>`
- `<blockquote> </blockquote>`
- `<address> </address>`
- `<div> </div>`
- ` `
- ` `
- ` `
- `<dl> </dl> <dt></dt> <dd></dd>`

- Liniowe elementy, które poznaliśmy do tej pory:
- `<a>`
- ``
- ``
- `<i>`
- ``
- ``

Znacznik

- Znacznik służy do określenia jednego lub więcej parametrów czcionki:
 - Nazwy
 - koloru
 - lub wielkości.
-
- Znacznik **FONT** jest zdeprecjonowany przez specyfikację HTML 4.01 - zaleca się stosowanie stylów CSS.

Atrybuty width i height

- ``

<input>

- Najbardziej popularnym elementem html pozwalającym na wprowadzanie danych przez użytkownika jest element **<input>** (ang. dane wejściowe).
- **<input type="text">**
- **<input type="text" value="Wartość wpisana przez Nas">**
- **<input type="password">**

text ,password, checkbox, radio

<label>

- Do opisu elementów <input> używamy specjalnie przeznaczonego do tego elementu <label> (ang. etykieta). Element ten wypełniamy opisem danego pola.

```
<label for="price">Cena</label>
```

```
<input type="text" id="price">
```

Placeholder

- Wartość atrybutu placeholder wypełniamy wybranym przez nas tekstem. Tekst ten pojawi się w środku elementu `<input>`, i zniknie jak zaczniemy wpisywać swoje dane.
- `<input type="text" placeholder="Imię i Nazwisko">`

Input Type Radio

- **`<input type="radio">`**
- Kolejnym dość szczególnym elementem input jest radio, określany również radio buttonem (przyciskiem), bądź wykluczającym polem wyboru.

`<p>Opis:<p>`

`<label><input type="radio" name=„tekst"> Opis</label>`

`<label><input type="radio" name=„tekst"> opis2</label>`

`<label><input type="radio" name=„tekst"> opis3</label>`

- **checked**
- Tak jak w przypadku elementu checkbox możemy domyślnie zaznaczyć dany element radio button używając atrybutu checked.

```
<input type="radio" name="job" checked>
```


- **<input type="checkbox">**
- **<textarea>** wieloliniowe pole, w którym możemy wpisywać nie tylko krótkie formy pisemne
- **<textarea cols="50"></textarea>**
- **<textarea cols="50" rows="4"></textarea>**
- **Select – Lista Rozwijana HTML**
 - <select>**
 - <option>Opcja 1</option>**
 - <option>Opcja 2</option>**
 - <option>Opcja 3</option>**
 - </select>**

Formularz HTML

- **<form>**
- **Formularz HTML** tworzymy poprzez zastosowanie elementu **<form>**

```
<form>
```

```
<label>Twój Login <input type="text"></label>
```

```
<label>Twój email <input type="email"></label>
```

```
<label>Twoje hasło <input type="password"></label>
```

```
</form>
```

TABELE

- **<table>**
- Kolejnym sposobem prezentowania i grupowania danych w języku Html jest tabela. Sposób prezentowania danych w tabeli html jest taki sam jak w każdej innej tabeli, którą znamy z życia codziennego. Tabele tworzymy przy użyciu elementu <table>.
- **<table> </table>**

Tabela HTML – Wiersz

- `<tr>`
- Jednak, żeby tabela powstała, musimy do niej dodać jeszcze wiersze oraz umieścić tam komórki. Wiersze do tabeli dodajemy poprzez element `<tr>` (ang. **table row** – wiersz tabeli).

Tabela HTML – Komórka

- **<td>**
- Żeby nasza tabela była kompletna i poprawnie wyświetlała dane, należy jeszcze dodać do niej komórki. Komórki tabeli dodajemy poprzez element **<td>** (ang. **table data**, table cell – dane tabeli, komórka tabeli). Do rozwijania skrótu td powszechnie używa się table cell, jednak nazwa pochodzi od table data.

- `<table> <tr> <td>Trešć</td> </tr> </table>`

- Treść umieszczamy zawsze w elemencie `<td>`. Element ten podobny jest trochę do kolumny.
- Jednak nie zapisujemy jej raz tak jak ma to miejsce w przypadku wiersza, tylko musimy za każdym razem dodać tyle komórek (column) do wiersza ile aktualnie nam potrzeba.

- `<table>`
- `<caption>Etykieta</caption>`
- `<tr>`
- `<th colspan=`
- `<td>1 wiersz 1 kolumna</td>`
- `<td>1 wiersz 2 kolumna</td>`
- `</tr>`
- `<tr>`
- `<td>2 wiersz 1 kolumna</td>`
- `<td>2 wiersz 2 kolumna</td>`
- `</tr>`
- `</table>`

Nagłówek Tabeli HTML

`<th>`

- Mamy już treść tabeli. Przydałoby się jeszcze oznaczyć czego dotyczą nasze dane. W tym celu posłużymy się nagłówkiem tabeli. Nagłówek taki uzyskujemy poprzez zastosowanie elementu `<th>` (ang. **table heading**). Element ten stosujemy tak jak zwykłą komórkę tabeli, czyli `<td>`. Nagłówki również umieszczamy w wierszach tabeli, czyli elementach `<tr>`.

- **Tabela HTML – łączenie komórek**
- W przypadku, gdy jedna komórka odnosi się do więcej niż jednego nagłówka, możemy dokonać połączenia kilku komórek w jedną, żeby kilkakrotnie nie powtarzać jej wartości. Łączenie nie jest jednak najlepszym określeniem, w przypadku tabeli html. W zasadzie nie dokonujemy łączenia komórek, a zwiększenia ich rozpiętości w pionie lub poziomie.

Łączenie Komórek Kolumnami (w poziomie)

- **colspan**
- W celu zwiększenia rozpiętości poziomej komórki używamy atrybutu colspan(ang. **column span** – rozpiętość kolumnowa komórki). Atrybut ten przyjmuje wartości liczbowe, tzn. przekazujemy tam jaką rozpiętość kolumnową (czyli w poziomie) ma posiadać dana komórka tabeli.

- `<table>`
- `<tr>`
- `<th></th>`
- `<th>Poniedziałek</th>`
- `<th>Wtorek</th>`
- `</tr>`
- `<tr>`
- `<th>18:00</th> <td>Kręgle</td> <td>Polo</td>`
- `</tr> <tr> <th>20:00</th> <td colspan="2">Basen
i Sauna</td> </tr> </table>`