

Tworzenie Stron Internetowych

odcinek 2

elementy HTML

Podział funkcyjny elementów HTML:

- **strukturalne** – opis logicznej struktury tekstu
- **hipertekstowe** – zawierają linki do innych dokumentów lub innych części tego samego dokumentu
- **prezentacyjne** – opis wyglądu poszczególnych elementów (usunięte)

Podział semantyczny elementów HTML:

- **semantyczne** – mają przypisane jednoznaczne znaczenie dla zawartości
- **niesemantyczne** – nie mają przypisanego znaczenia dla zawartości

Podział na elementy blokowe i liniowe:

- **Elementy blokowe** (*block elements*)

Elementy blokowe zaczynają się zawsze od nowej linii i zajmują całą dostępną szerokość. Domyślnie poprzedza je i następuje po nich odstęp. Przykładem tego typu elementu jest: akapit, nagłówek, tabela, lista. Mogą zawierać w sobie elementy liniowe i w niektórych przypadkach inne elementy blokowe.

- **Elementy liniowe** (*inline elements*)

Elementy wplecione w tekst, nie rozpoczynają nowej linii i nie zajmują całej dostępnej szerokości. Przykłady: np. wyróżnienie, odnośnik, skrót. Elementy liniowe są dzielone i przenoszone do nowej linii razem z tekstem, który zawierają. Elementy liniowe nie mogą zawierać elementów blokowych.

elementy blokowe

Nagłówki **<h1>**, **<h2>**, ..., **<h6>**

Nagłówki (*headings*) dzielą dokument na logiczne fragmenty (np. tytuły rozdziałów, podrozdziałów, itp.). Występują w sześciu „stopniach”, od **<h1>** (najgrubszy podział) do **<h6>** (najdrobniejszy podział). Nagłówki powinny być ułożone wg hierarchii.

Przykład:

<h1>Tytuł strony**</h1>**

<h2>Tytuł rozdziału 1**</h2>**

<h3>Tytuł podrozdziału 1.1**</h3>**

.....

<h2>Tytuł rozdziału 2**</h2>**

Numeracja nagłówków nie oznacza ich kolejności. Oznacza ich hierarchię.

Nagłówki są ważne – używane są przez wyszukiwarki do indeksowania zawartości stron, a użytkownicy mogą przeglądać stronę po nagłówkach.

elementy blokowe

Akapity `<p>` i łamanie linii `
`

Akapit oznaczany jest elementem `<p>` (*paragraph*). Wielokrotne spacje i złamanie linii w akapicie są ignorowane. Długość linii tekstu jest automatycznie dostosowywana do szerokości okna przeglądarki.

Złamanie linii można uzyskać stosując pusty element `
`. Należy unikać jego zbyt częstego używania. Łamanie linii stosujemy tylko w przypadkach koniecznych, np. wiersze, dialogi (lub używamy bardziej odpowiedniego elementu).

przykład:

```
<p>Na podstawie obserwacji orbitalnych oraz badań meteorytów marsjańskich wydaje się, że powierzchnia Marsa jest złożona głównie z bazaltu. Niektóre dowody sugerują, że część powierzchni jest bogatsza w krzemionkę niż bazalt i mogą ją tworzyć skały podobne do ziemskich andezytów, jednak można to także wytłumaczyć obecnością amorficznej krzemionki.
```

```
<br />
```

```
Znaczna część powierzchni Marsa jest pokryta pyłem tlenku żelaza</p>
```

elementy blokowe

Listy nieuporządkowane, uporządkowane i listy definicji

Listy pozwalają pogrupować powiązane ze sobą elementy.

Listy nie mogą być umieszczone wewnątrz akapitu `<p>`.

Listy nieuporządkowane: ``

Używa się ich, gdy kolejność elementów listy jest bez znaczenia.

Listę definiuje się przez `` (*unordered list*), a poszczególne jej elementy przez `` (*list item*).

Bezpośrednio w `` nie może być żadnych innych elementów niż ``.

Listy można zagnieżdżać. Kod list podrzędnych wpisujemy wewnątrz elementu `...` będącego punktem nadrzędnym.

Przykład:

```
<p>Jadąc na rajd zabierz:</p>
<ul>
  <li>buty - trapery</li>
  <li>nieprzemakalna kurtkę</li>
  <li>smaczny prowiant:
    <ul>
      <li>czekolada</li>
      <li>kanapki</li>
    </ul>
  </li>
</ul>
```

elementy blokowe

Listy uporządkowane: ``

List uporządkowanych używamy, gdy kolejność elementów ma znaczenie dla treści.

Listę definiuje się przez `` (*ordered list*), a poszczególne jej elementy przez `` (*list item*). Elementy listy będą automatycznie ponumerowane przez przeglądarkę. Bezpośrednio w `` nie może być żadnych innych elementów niż ``.

Zasady zagnieżdżania są takie same jak dla ``.

Przykład:

```
<p>Plan rajdu:</p>
<ol>
  <li>spotkanie na dworcu</li>
  <li>przejazd pociągiem</li>
  <li>przejście trasy</li>
  <li>kolacja i nocleg</li>
</ol>
```

elementy blokowe

Listy definicji: <dl>

Lista definicji jest zestawieniem terminów i ich opisów.

Lista definicji <dl> (*description list*) zawierają dwa rodzaje elementów – termin <dt> (*definition term*) i jego objaśnienie <dd> (*definition description*).

Umożliwia tworzenie np. słowników. Jeden termin może mieć wiele definicji i jedna definicja może opisywać wiele terminów.

przykładowy kod źródłowy:

```
<p>Słownik rajdowy:</p>
<dl>
  <dt>KT PKP</dt>
  <dd>&quot; konsorcjum turystyczne &quot; organizujące Rajd Astronomów</dd>
  <dt>Niezbędnik Uczestnika Rajdu</dt>
  <dd>informator rajdowy rozdawany uczestnikom</dd>
</dl>
```

i efekt jego działania:

Słownik rajdowy:

KT PKP

"konsorcjum turystyczne" organizujące Rajd Astronomów

Niezbędnik Uczestnika Rajdu

informator rajdowy rozdawany uczestnikom

elementy blokowe

Cytowany fragment: **<blockquote>**

Używany jest do cytowania długich fragmentów tekstu. Wewnątrz **<blockquote>** należy używać elementów blokowych formatujących tekst (np. paragrafów).

Atrybut **cite** pozwala definiować źródło pochodzenia cytatu (URL)

Krótkie cytaty można dodawać za pomocą elementu **<q>**.

```
<p>Na stronach rajdowych znajdujemy taki oto tekst:</p>
```

```
<blockquote cite="www.astro.uni.wroc.pl/rajdy/rajdy.html">
```

```
<p>Rajd odbywa się dwa razy w roku: pierwszy na wiosnę (edycja wiosenna), drugi na  
jesień (edycja jesienna). Jedynym warunkiem uczestnictwa jest nieodparta chęć  
poważenia się w dobrym towarzystwie i z garbem (czyli plecakiem) po dzikich,  
górkich ostępach.</p>
```

```
</blockquote>
```


elementy blokowe

Oddzielanie elementów: `<hr>`

Pusty element `<hr>` (*horizontal rule*) wstawia poziomą linię. Umożliwia podział dokumentu, np. na różniące się pod względem treści fragmenty. Nie może być umieszczany wewnątrz akapitów.

```
<p>Opis trasy rajdowej ...</p>
<hr>
<p>pytania? napisz do nas: ...</p>
```

Tekst preformatowany: `<pre>`

Element ten służy do wyświetlania tekstu z zachowaniem wszystkich spacji i znaków podziału wiersza dokładnie tak, jak są zapisane w kodzie źródłowym HTML. Można go użyć w połączeniu z `<code>` lub `<samp>` (zapis kodu komputerowego).

```
<pre>
I have come to bring the end
Dry and hopeless my descent
For I am the core of winter
The unforgiving darkness
</pre>
```

elementy blokowe

Dane kontaktowe: `<address>`

Element ten definiuje adres związany ze stroną, na której jest umieszczony, np. dane kontaktowe autora danej strony (adres tradycyjny lub e-mail, numer telefonu, profil w mediach społecznościowych). Może zawierać jedynie elementy liniowe. Wstawiając wielolinijkowy adres należy posłużyć się elementem `
`:

```
<address>
Konsorcjum Turystyczne PKP <br>
ul. Kopernika 11 <br>
51-622 Wrocław
</address>
```

Blok (sekcja): `<div>`

Służy do grupowania elementów i dzielenia strony na bloki, jeśli nie ma bardziej właściwego elementu do tego celu. **Element jest niesemantyczny.**

```
<div>
  <h1>Rajd Astronomów</h1>
  <p>najnowsze informacje</p>
</div>
```

elementy blokowe

Obramowanie **<fieldset>**

Znacznik **<fieldset>** służy do grupowania pól w formularzach. Za jego pomocą możemy dodać obramowanie wokół wybranych elementów powiązanych ze sobą (nie tylko w formularzu). Wygląd obramowania może zależeć od przeglądarki. (więcej o formularzach później)

```
<fieldset>  
  tekst  
</fieldset>
```

Dodając znacznik **<legend>** (musi być wewnątrz **<fieldset>**) można podać tytuł ramki.

```
<fieldset>  
  <legend>opis</legend>  
  tekst  
</fieldset>
```

elementy liniowe

Elementy liniowe (ang. inline)

Elementy liniowe służą do oznaczania tekstu (nie tworzą bloków)

Skrót: `<abbr>` i skrótowiec: ~~`<acronym>`~~

Wskazuje, że dane słowo jest skrótem lub skrótowcem. Użycie atrybutu **title** umożliwia podanie jego rozwinięcia, co ułatwia czytelnikowi (i wyszukiwarkom) zrozumienie tekstu.

```
<abbr title="et cetera (i tym podobne)">etc.</abbr>
```

```
<acronym title="„Studenckie Koło Przewodników Sudeckich">SKPS</acronym>
```

<acronym> został usunięty z HTML5

Cytat: `<q>`

Oznacza cytowany krótki fragment. Dłuższe fragmenty najlepiej oznaczać przez **`<blockquote>`**.

```
<p>Przed pierwszym rajdem większość twierdziła, że <q>trasa rajdowa powinna liczyć 20 - 30 km</q> na jeden dzień.</p>
```

elementy liniowe

Nawiązanie: `<cite>`

Elementem `<cite>` oznaczamy tytuł dzieła, np. tytuł cytowanej książki, filmu, piosenki, itd.

`<p>Na podstawie <cite>Słownika Geografii Turystycznej Sudetów, tom 20</cite>.</p>`

Definicja: `<dfn>`

Pomaga umieścić objaśnienie terminu pojawiającego się w tekście. Objaśniane słowo oznaczamy przez `<dfn>`.

`<p>Skrótowiec <dfn>SGTS</dfn> oznacza Słownik Geografii Turystycznej Sudetów</p>`

Zmiana kierunku tekstu: `<bdo>`

Umożliwiają zmianę wyświetlania kierunku tekstu na przeciwny (*bi-directional override*). Kierunek zadajemy atrybutem `dir`: `ltr` (left-to-right, domyślny), `rtl` (right-to-left).

`<bdo dir="rtl">Ten tekst będzie wypisany z prawej do lewej</bdo>`

elementy liniowe

Kod: `<code>`

Używany do oznaczenia przykładowego kodu komputerowego. Jeśli chcemy zachować podział na osobne linie, to należy umieścić `<code>` w elemencie `<pre>`.

```
<code>a=reform(b)</code>
```

Zmienna: `<var>`

Zmienna (komputerowa, matematyczna). Może być używany wewnątrz `<code>` lub samodzielnie. Element przydatny do wyróżniania zmiennych w cytowanym kodzie lub tekście.

```
<var>y</var> = exp(<var>x</var>)
```

Kombinacja klawiszy: `<kbd>`

Określa działanie, które należy wykonać z klawiatury (skrót klawiszowy).

```
<p>Wciśnij <kbd>Ctrl+Alt+Del</kbd> i postępuj zgodnie z wyświetloną instrukcją.</p>
```

Przykład: `<samp>`

Pozwala oznaczyć wynik działania programu komputerowego. Łącząc ten element z `<kbd>` możemy opisać dialog użytkownika z komputerem.

```
<p>Jeżeli program zapyta nas <samp>Czy na pewno kontynuować?</samp>, musimy kliknąć <samp>Tak</samp> lub wcisnąć <kbd>Enter</kbd>.</p>
```

elementy liniowe

Emfaza: **** i ważny tekst: ****

Umożliwiają podkreślenie, wzmocnienie ważności wybranego fragmentu tekstu.

**** ma mniejszy nacisk niż ****.

```
<p><strong>Uwaga</strong> pamiętajcie, aby zabrać <em>korony czeskie</em> na rajd. </p>
```

Wyróżniony tekst: **<mark>**

Znacznik stosuje się do oznaczenia tekstu, który ma być wyróżniony. Znacznik dostępny tylko w HTML5.

```
<p>Przedpłata <mark>nie podlega zwrotowi</mark> niezależnie od sytuacji.</p>
```

Dawne elementy prezentacyjne: **** (pogrubienie) i **<i>** (kursywa)

Zazwyczaj nie powinno się ich używać. Należy rozważyć czy inny znacznik HTML nie jest w danej sytuacji bardziej odpowiedni. Dzięki CSS można każdy fragment tekstu pogrubić lub pochylić.

```
<p>Tekst <b>pogrubiony</b> lub <i>pochylony</i></p>
```

W HTML5 element **<i>** zmienił swoje znaczenie. Obecnie to tekst pochyłony bez wyróżnionej ważności, np.: wstawka z innego języka, krótka myśl, nazwa własna czegoś, termin techniczny. Można go użyć w sytuacji, gdy inne znaczniki semantyczne nie pasują.

Element **** jest niesemantyczny i nie oznacza tekstu ważnego.

elementy liniowe

Drobny tekst: `<small>`

Stosuje się go do opisu mało ważnego tekstu (przeciwieństwo `` i ``)

```
<small>Przedpłata nie podlega zwrotowi.</small>
```

Indeks dolny: `<sub>` i górny: `<sup>`

Stosujemy, gdy tekst wymaga zapisu z indeksem górnym i dolnym.

```
<p>W skład wiatru słonecznego wchodzi <sup>4</sup>He.</p>
```

Oznaczanie zmian: `<ins>` i ``

Element `<ins>` oznacza tekst, który został wstawiony, natomiast `` tekst, który został (ma zostać) usunięty.

```
<p>Wyjazd o godzinie <ins>7:00</ins> a nie o <del>8:00</del>.</p>
```

Dowolny zakres: ``

Liniowy odpowiednik blokowego `<div>`. Korzystając z niego możemy np. tworzyć własne elementy liniowe, których HTML nie przewiduje i nadać im odpowiedni wygląd używając CSS.

```
<span>Rajd Astro</span> <span>nomów</span>
```

Element `` jest niesemantyczny.

elementy liniowe i przydatne encje

Przełamanie wyrazu: **<wbr>**

Znacznik wstawia punkt przełamania wyrazu (*word break opportunity*). Pozwala on przeglądarce podzielić wyraz we wskazanym miejscu. Szczególnie przydatny w przypadku długich wyrazów. Znacznik dostępny tylko w HTML5.

`<p>Megkápó<wbr>szťahúsító<wbr>ttalanítottátok to jedno z najdłuższych słów w języku węgierskim.</p>`

Niełamliwa spacja ** **

Niełamliwa spacja ** ** (*non-breaking space*), zwana również twardą spacją umożliwia:

- blokadę przełamania wiersza między wybranymi wyrazami
- dodanie więcej niż jednej spacji między wybranymi wyrazami lub spacji na początku akapitu.

Dzięki temu możemy zapobiec np. pojawienia się spójników na końcu linii lub rozdzielenia wartości od jednostek do osobnych wierszy:

biały i czarny

masa wynosi 10 kg

Przydatny może być też niełamliwy myślnik **‑**

nowe elementy w HTML5

Nowododane elementy semantyczne (wybrane):

- **<details>** – zawiera dodatkowe szczegóły, które użytkownik może wyświetlić lub ukryć
- **<summary>** – dodaje nagłówek do treści zawartej w <details>, jest elementem „klikalnym” (ukryj/wyświetl zawartość <details>)

```
<details>
```

```
  <summary>Szczegóły wydania</summary>
```

```
  <p> Wydano w roku 2017 przez UWwr</p>
```

```
</details>
```

treść widoczna

treść ukryta

Uwaga: wykorzystując powyższe elementy należy sprawdzić, które przeglądarki i w jakich wersjach je obsługują (www.w3schools.com/tags/default.asp).

wygląd domyślny i semantyczność

Domyślnie w przeglądarkach różne elementy są wyświetlane podobny sposób graficzny (np. pogrubienie i pochylenie). **Przy wyborze elementu należy brać pod uwagę wyłącznie jego znaczenie** (kod semantyczny).

Dla przykładu:

```
<p><strong>Tekst</strong></p>
```

wygląda tak samo jak nagłówek, ale ma zupełnie inne znaczenie.

Uwaga zamykająca

Każdy fragment tekstu umieszczony w kodzie HTML musi być częścią jakiegoś elementu.