

Nazwa implementacji: Tworzenie prostej strony internetowej **Autor:** michal.czyzewski **Opis implementacji:** Zaczynając od pustego szablonu, stworzymy szkielet strony internetowej zawierającej nagłówek, stopkę, nawigację i treść.

Wprowadzenie

Język **HTML** (ang. HyperText Markup Language) jest językiem opisu treści, co oznacza, że jego podstawowym zastosowaniem jest opis struktury różnych rodzajów zawartości (tekstu, obrazków, klipów wideo, etc.) oraz ich relacji między sobą. Najczęstszym zastosowaniem języka HTML jest tworzenie stron internetowych (a właściwie strukturyzacja informacji zawartych na tych stronach).

Historia języka HTML sięga lat 80-tych ubiegłego wieku (choć pierwsza oficjalna specyfikacja – HTML 2.0 – powstała w roku 1995), nie jest więc zaskoczeniem, że na przestrzeni lat powstało wiele wersji tego języka. Najnowszym standardem używanym do tworzenia stron i aplikacji internetowych jest HTML5, jego też będziemy poznawali w tej serii lekcji.

Tagi

Dokument HTML (m. in. każda strona internetowa) składa się z tzw. tagów i treści w nich zawartych. Tagi mogą mieć również atrybuty, zawierające dodatkowe informacje o ich zawartości. Na przykład:

```
<tag atrybut-tagu="wartość atrybutu">opisywana zawartość strony</tag>
```

</tag> jest oznaczeniem zamknięcia tagu otwartego przez <tag>.

Nazwy oraz atrybuty tagów są ustandaryzowane tak, aby różne przeglądarki internetowe na różnych systemach operacyjnych mogły je jednoznacznie interpretować. Kilka często używanych tagów to:

<p> oznacza akapit tekstu;

<a> oznacza linki (zwykle do innych stron lub miejsc w bieżącym dokumencie), najczęściej jest używany z atrybutem href, który określa adres URL linku:
treść linku

<h1>, <h2>, <h3>, <h4>, <h5>, <h6> oznaczają różne poziomy nagłówków;

<div> oznacza po prostu kawałek dokumentu i może zawierać w sobie dowolne inne tagi, często jest używany do nadania danemu fragmentowi określonej klasy.

 pozwala wstawić do dokumentu obrazek, zwykle jest używany z atrybutami src (położenie obrazka) oraz alt (opis zawartości obrazka):

Czas na przykład kompletnego dokumentu HTML w najbardziej podstawowej wersji:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Tytuł dokumentu</title>
  </head>
  <body>
    <h1>Nagłówek w treści dokumentu</h1>
    <p>Przykładowy akapit. Ale niezbyt długi.</p>
    <p>Drugi przykładowy akapit</p>
  </body>
</html>
```



Należy tu zaznaczyć, że wcięcia na początku linijek (jak każde powtarzające się „białe znaki”, takie jak spacje, tabulatory, znaki nowej linii) są wyłącznie dla wygody programisty i nie wpływają na wygląd strony w przeglądarce. Równie dobrze można by cały dokument HTML umieścić w jednej, bardzo długiej linii, jednak wówczas jego edycja stałaby się niewygodna. Numery linii również zostały dodane dla naszej wygody i nie są częścią dokumentu.

Znaczenie poszczególnych linii:

Deklaracja typu dokumentu, określa ona wersję dokumentu i powinna znaleźć się na początku każdego dokumentu HTML; nasza deklaracja oznacza, że wersja HTML, której używamy to HTML5 (mimo, że nie jest to wyraźnie napisane – w HTML5 uproszczono deklarację typu do absolutnego minimum).

Początek dokumentu HTML (tag zamykający w linii 11).

- Początek sekcji HEAD (koniec w linii 5), zawierającej informacje o dokumencie, np. tytuł, autora, datę publikacji, odnośniki do informacji o stylach (o nich w następnej lekcji) oraz skryptach (a o nich w lekcji 4).
- Tutaj określamy tytuł dokumentu, który będzie pojawiał się m. in. w pasku tytułu przeglądarki.

Otwieramy sekcję BODY (koniec w linii 10), w której znajduje się treść naszej strony.

Przykład zastosowania tagu <h1> (nagłówek najwyższego rzędu).

- (oraz 9) Przykład zastosowania tagu <p>.

Zadanie: Otwórz powyższy kod (plik przyklad1.html) w przeglądarce i porównaj kod z tym, co pokazała przeglądarka. Otwórz plik w edytorze tekstu i dodaj akapit z linkiem do strony e-swoi.pl, a następnie sprawdź jak wygląda w przeglądarce.

Jak widzimy, dokument HTML tworzy hierarchię drzewa – każdy element (poza samozamykającymi) może zawierać jeden lub więcej innych elementów (są pewne wyjątki od tej reguły, ale o nich za chwilę). Element nadrzędny jakiegoś elementu jest zwykle nazywany jego rodzicem (ang. parent), element podrzędny – dzieckiem (ang. child).

Klasy i identyfikatory

Dwoma często używanymi atrybutami dla tagów w treści strony są id oraz class. Oba pozwalają na identyfikację elementów strony, gdy chcemy się do nich odwołać – np. w skrypcie lub arkuszu stylów. Każdy atrybut id może wystąpić dokładnie 1 raz w danym dokumencie – jest używany do jednoznacznego odwołania się do danego elementu strony (tagu). Atrybuty class mogą się powtarzać, co pozwala na operowanie na wielu elementach strony na raz – np. nadanie im tego samego wyglądu lub wywołanie na nich tej samej funkcji skryptu. Co więcej, każdemu elementowi można nadawać wiele klas, oddzielając je spacją. Przykładowy fragment kodu ilustrujący atrybuty id i class:

```
<div id="lista-notatek"> <!-- to jest jedyna na całej stronie lista notatek -->  
<p class="notatka">To jest treść mojej notatki.</p>  
<p class="notatka wazne">To jest treść innej, bardzo ważnej notatki.</p>  
</div>
```

Powyższy przykład służy ilustracji mechanizmu – w praktyce niepotrzebne jest dodawanie klasy notatka dla każdego elementu takiej listy, kiedy możemy się do wszystkich notatek odwołać w bardziej sprytny/elegancki sposób (wszystkie są dziećmi elementu <div> o id lista-notatek). O odwoływaniu się do elementów powiemy trochę więcej w następnej lekcji.

W tym przykładzie widzimy też po raz pierwszy komentarz HTML. Jest to tekst ujęty pomiędzy znacznikami <!-- oraz -->, który jest ignorowany przez przeglądarkę podczas wyświetlania strony.





Komentarze są udogodnieniem dla nas, pozwalają dodawać informacje przydatne przy późniejszym edytowaniu strony.

Tworzenie list

Listy numerowane definiujemy za pomocą tagu ``, a nienumerowane - ``. Elementy list reprezentują tagi ``. Przykład listy nienumerowanej:

```
<ul>
  <li>jakiś element listy</li>
  <li>inny element listy</li>
  <li>element listy <a href="http://example.net">z linkiem</a></li>
</ul>
```

Zadanie: Dodaj do wcześniej utworzonego kodu listę numerowaną z kilkoma elementami.

Nowości w HTML5

HTML5 wprowadził kilka dodatkowych tagów pozwalających dodatkowo podzielić treść strony, jednocześnie określając które elementy należą do nagłówka, menu, stopki, etc. Wcześniej do tego celu używane były tagi `<div>`, jednak prowadziło to do powstawania dużej ilości takich tagów w dokumencie (co powodowało utratę czytelności kodu). Jednocześnie nie były one czytelne dla komputera - np. elementy nawigacyjne różnych stron przyjmowały różne atrybuty `id`; wprowadzenie tagu `<nav>` jednoznacznie określa, gdzie znajdują się linki nawigacyjne, co ułatwia indeksowanie wyszukiwarkom i pozwala na interpretację przez przeglądarki.

Nowe tagi w HTML5 (kilka z nich):

`<section>` - zgodnie z nazwą, zawiera sekcję (kawałek) dokumentu, czyli kilka elementów, które mają ze sobą tematycznie coś wspólnego

`<article>` - oznacza „artykuł”; jest stosowany podobnie jak tag `<section>`, ale dla samodzielnych kawałków treści (np. kolejnych artykułów)

`<header>` - nagłówek; może być umieszczony bezpośrednio wewnątrz `<html>` lub być częścią sekcji lub artykułu

`<footer>` - stopka, umieszczana podobnie jak nagłówek; może zawierać przypisy lub dodatkowe treści związane z sekcją, artykułem

`<nav>` - zawiera linki nawigacyjne, zwykle stosowany do wyświetlania menu nawigacyjnego strony na poziomie dokumentu; wewnątrz

Poniżej znajduje się przykładowy kod wykorzystujący tagi wprowadzone w HTML5.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Tytuł dokumentu</title>
  </head>
  <body>
    <header>Nagłówek całego dokumentu</header>
    <nav>
      <ul>
        <li><a href="adres_postrony_1.html">Podstrona 1</a></li>
        <li><a href="adres_postrony_2.html">Podstrona 2</a></li>
      </ul>
    </nav>
  </body>
</html>
```

3





```
<li><a href="adres_postrony_3.html">Podstrona 3</a></li>
</ul>
</nav>
<article>
  <header>Nagłówek artykułu</header>
  <p>Przykładowy akapit. Ale niezbyt długi.</p>
  <p>Drugi przykładowy akapit</p>
</article>
<footer>
  <p>Zawartość stopki</p>
</footer>
</body>
</html>
```

Zadanie: Otwórz powyższy kod w edytorze (plik przyklad2.html) i zastąp linki do podstron innymi, działającymi adresami. Przekopiuj listę numerowaną z poprzedniego przykładu. Wyniki swojej pracy obejrzyj w przeglądarce.

