2 Podstawy tworzenia stron internetowych

2.1. HTML5 i struktura dokumentu

Podstawą działania wszystkich stron internetowych jest język **HTML** (*Hypertext Markup Language*) – hipertekstowy język znaczników. Dokument napisany w tym języku zawiera znaczniki (tagi) umieszczone w nawiasach ostrych, np.: <html>, , . Zazwyczaj występują one w parach – pierwszy znacznik jest znacznikiem początkowym (lub znacznikiem otwarcia), drugi natomiast – końcowym (znacznikiem zamknięcia). Przeglądarka internetowa odczytuje znaczniki i interpretuje je tak, aby użytkownik zamiast kodu widział właściwie sformatowaną stronę internetową.

Strony internetowe są tworzone według określonych standardów. Odpowiada za nie W3C (*World Wide Web Consortium*). Standardy ustanawiane przez W3C nie mają co prawda mocy prawnej, jednak użycie aktualnych standardów jest zalecane ze względu na prawidłowe wyświetlanie stron www w przeglądarkach internetowych.

Do najpopularniejszych standardów należy **HTML5**, będący najnowszą wersją języka HTML i stanowiący rozwinięcie języków HTML4 i XHTML1.

Projektując stronę internetową, zdecydowanie należy trzymać się szablonu, który daje nam pewność, że strona będzie prawidłowo wyświetlana w przeglądarce internetowej (rys. 1).



Rysunek 1. Przykładowa struktura dokumentu HTML

Mając taką poglądową strukturę dokumentu, sprawdźmy, jak należy opisać poszczególne części za pomocą kodu HTML. Istotny element dokumentów publikowanych w internecie stanowią deklaracje ich typów zalecane przez W3C. Zadaniem deklaracji jest wskazanie przeglądarce internetowej, w jakim języku dokument został napisany. Dzięki temu przeglądarka może odpowiednio zinterpretować kod oraz wyświetlić prawidłowo dokument. W języku HTML5 deklaracja ogranicza się do zaledwie jednej linii kodu umieszczanej na początku dokumentu: <!DOCTYPE html>. Następnie musimy wyznaczyć ramy dokumentu, czyli wprowadzić znacznik <html>, wewnątrz którego zawarte będą wszystkie pozostałe znaczniki.

Większość znaczników ma tzw. atrybuty, dzięki czemu precyzujemy sposób ich działania. I tak przy znaczniku <html> dość ważne jest określenie języka, w jakim ma być prezentowana zawartość witryny. Atrybut wpisujemy wewnątrz nawiasów <>, ostatecznie zatem znacznik powinien przyjąć postać <html lang="pl">>.

Kolejny krok to umieszczenie w dokumencie opisu zawartości strony oraz jej tytułu. Wszystkie te informacje należy wpisać wewnątrz znacznika <head>, czyli w nagłówku dokumentu. Pierwszy znacznik, który wprowadzamy do nagłówka, to <meta>, czyli informacje profilowe strony. Możemy zastosować tu m.in. następujące atrybuty:

- http-equiv="conent-type" służy do określenia sposobu kodowania dokumentu i to właśnie dzięki niemu w przeglądarce będą prawidłowo wyświetlane polskie znaki,
- name="description" pozwala na wprowadzenie opisu strony,
- name="keywords" umożliwia wpisanie słów kluczowych.

Następnie wstawiamy znacznik <title>, dzięki któremu treść, jaką wpiszemy w jego obrębie, będzie widoczna w górnej części okna przeglądarki – jest to szczególnie użyteczne, jeśli mamy otwartych kilka stron w różnych kartach i chcemy przełączać się między nimi.

W nagłówku powinniśmy umieścić także ścieżkę do pliku CSS, w którym opiszemy wygląd strony. Tym jednak zajmiemy się nieco później.

Następnie, już poza nagłówkiem, powinniśmy wyznaczyć tzw. ciało dokumentu, czyli miejsce, w którym będzie umieszczona właściwa część strony. Służy do tego znacznik <body> (rys. 2). Będziemy w nim opisywali wszystkie elementy pokazane na rysunku 1.

Rysunek 2. Określenie ram dokumentu HTML

Kiedy ramy dokumentu są już określone, można przystąpić do definiowania kolejnych elementów. Zacznijmy od znacznika <header>. Przede wszystkim należy pamiętać, że <head> oraz <header> to, mimo nieco mylącej nazwy, dwa różne elementy. Co możemy umieścić w znaczniku <header>? Zazwyczaj takie elementy, jak: logo, motto, hasło reklamowe, formularz wyszukiwania. Warto pamiętać, że bardzo często, będąc w dowolnym miejscu witryny, chcemy szybko wrócić do strony głównej. Dlatego też, jeśli w elemencie <header> wstawiamy logo, powinno ono stanowić odnośnik do tej właśnie strony (rys. 3).

10	<body></body>
11	<pre><header><img <="" pre="" src="logo.png" width="184"/></header></pre>
	height="49" alt="logo">
12	<hl>Podstawą wszystkiego jest właściwe nastawienie</hl>
13	
14	
15	

Rysunek 3. Przykład elementu <header>

W przedstawionym na rysunku 3 kodzie zastosowane zostały znaczniki: <a>, oraz <h1>. Pierwszy z nich to kotwica. Wraz z atrybutem href tworzy hiperłącze (odnośnik, link) do wskazanego dokumentu HTML.

Znacznik służy do umieszczenia w dokumencie obrazu. Najistotniejsze atrybuty to rozmiary oraz alt, czyli tekst alternatywny – w razie, gdyby obraz nie został wczytany, pojawi się przypisana do tego atrybutu treść.

Ostatni użyty znacznik to <h1>, czyli nagłówek poziomu pierwszego. Takich poziomów w języku HTML jest sześć, przy czym <h1> oznacza czcionkę w największym dopuszczalnym dla nagłówków rozmiarze, natomiast <h6> odpowiada za najmniejszą czcionkę nagłówka (rys. 4).



Rysunek 4. Przykład elementu <header> w przeglądarce Google Chrome

Kolejny element to <nav>. Jak można się domyśleć, w tej części strony umieszczamy odnośniki do innych stron lub do wyznaczonych fragmentów bieżącej strony. Załóżmy, że nasza witryna składa się z czterech stron: głównej (czyli domyślnie *index. html*), o nas, oferty oraz kontaktu. Do znacznika <nav> zatem powinniśmy wstawić cztery odnośniki. Najczęściej tworzymy z nich listę wypunktowaną, którą później



Rysunek 5. Przykład elementu <nav>

opisujemy za pomocą stylów CSS. Za listę wypunktowaną odpowiada znacznik , natomiast każdy punkt listy określany jest za pomocą znacznika (rys. 5).

Znacznik <aside> odpowiada za dodanie do strony treści, które niekoniecznie muszą być powiązane ściśle z tematyką naszej witryny. Mogą to być reklamy, przypisy lub linki nawigacyjne (rys. 6).

Element <section> to sekcja dokumentu obejmująca określoną część tematyczną. Zazwyczaj element ten zawiera nagłówek. Dzięki podziałowi treści na sekcje strona jest przejrzystsza, a co za tym idzie, o wiele łatwiejsza w odbiorze. Czytając treść, w której są wyodrębnione pewne fragmenty, bez problemu odnajdujemy na stronie miejsce, w którym skończyliśmy czytać (rys. 7).

W sekcjach użyto kolejnego znacznika . Służy on do tworzenia nowego akapitu. Akapit często mylnie jest postrzegany jako wcięcie w tekście. Przy projektowaniu stron akapit należy rozumieć jako fragment tekstu, zazwyczaj obejmujący kilka wierszy.

Zauważmy, że na razie strona nie do końca oddaje układ, jaki został przedstawiony na rysunku 1. Aby poszczególne elementy zostały ułożone w określonych miejscach, np. <aside> oraz <section> były ustawione w jednym wierszu, musimy sformatować je za pomocą stylów CSS. Aktualny wygląd zatem nie jest skutkiem błędnego projektowania, lecz kwestią tego, że na razie budujemy dopiero strukturę HTML.



Rysunek 6. Przykład elementu <aside>

Element <article> oznacza samodzielny fragment treści dokumentu, np. wpis na blogu lub komentarz. Zmienimy nieco strukturę przykładowego dokumentu. Z tekstów, które umieściliśmy w elementach <section>, utworzymy autonomiczne artykuły. Oczywiście w praktyce będą zazwyczaj dłuższe. Nagłówki zostawimy w takiej formie, w jakiej są, natomiast pozostałą część zamienimy na artykuły (rys. 8).

Struktura HTML prostego dokumentu jest już niemal gotowa. Brakuje tylko elementu <footer>, czyli stopki. W stopce umieszczamy najczęściej informacje na temat autora, praw autorskich, czasami linki. W naszym przykładzie stopka zostanie umieszczona na samym dole witryny (rys. 9). Jednak można ją stosować także dla opisanych wcześniej elementów, czyli: <article>, <aside>, <nav> czy <section>.



- <u>Strona główna</u>
- <u>O nas</u>
- <u>Oferta</u>
- Kontakt

Ciekawe linki

- <u>Dawid Mazur blog osobisty</u>
- <u>Alicja Żarowska-Mazur blog osobisty</u>
- Gryzz.pl blog kulinarny

Potrzebujesz strony internetowej?

Strony skrojone na miarę to gwarancja tego, żeby wyróżnić się wśród innych.

Zastanawiasz się, jaką stronę wybrać?

Tradycyjna strona, strona mobilna, a może responsywna? Przedstawimy Ci wszystkie. Przyjdź do nas, a pokażemy Ci, jak działają.

Rysunek 7. Zastosowanie elementu <section>

30	<section></section>
31	<h2>Potrzebujesz strony internetowej?</h2>
32	<article></article>
33	Strony skrojone na miarę to gwarancja tego, żeby
	wyróżnić się wśród innych.
34	
35	
36	<section></section>
37	<h2>Zastanawiasz się, jaką stronę wybrać?</h2>
38	<article></article>
39	Tradycyjna strona, strona mobilna, a może
	responsywna? Przedstawimy Ci wszystkie. Przyjdź do nas, a pokażemy
	Ci, jak działają.
40	
41	
42	
43	

Rysunek 8. Zastosowanie elementu <article>



Tradycyjna strona, strona mobilna, a może responsywna? Przedstawimy Ci wszystkie. Przyjdź do nas, a pokażemy Ci, jak działają.

Autor: Dawid Mazur

Wszelkie prawa zastrzeżone