

TOMASZ SITEK

Opis tworzenia stron internetowych przy pomocy języka HTML

SPIS TREŚCI

SPIS TREŚCI	1
Czym jest HTML?	2
Jak poprawnie kodować?	2
Struktura dokumentu	2
Nagłówek dokumentu	3
Ciało dokumentu	3
HTML i polskie znaki	4
Formatowanie tekstu	5
Odsyłacze	7
Odsyłacze dalekie	8
Odsyłacze wewnętrzne	8
Mapy na grafice	8
Tabele	9
Listy	10
Formularze	11

Czym jest HTML?

HTML wywodzi się on z grupy języków SGML (Standard Generalized Markup Language – Standardowy Uniwersalny Język Znakowania). Czym są języki znacznikowe? Służą one do oznaczenia pewnych części dokumentu – czyli podziału strukturalnego. Oprócz tego, w niektórych przypadkach (a jest nim HTML) mogą też być stosowane do opisu wyglądu dokumentu. W przypadku HTML'a służą do tego kaskadowe arkusze stylów CSS – dzięki nim możesz nadawać właściwości poszczególnym znacznikom tj. rodzaj czcionki, kolor, podział na kolumny, pozycjonowanie i wiele innych.

Jak poprawnie kodować?

Każdym językiem rządzą pewne zasady. Nie inaczej jest z HTML'em. Każdy znacznik musi posiadać otwarcie i zamknięcie. Wyjątkami od tej reguły są znaczniki nie posiadające zawartości np. `
`, ``,
W przypadku zagnieżdżania znaczników muszą być one zamykane w odwrotnej kolejności niż były otwierane, np.

```
<b>Ala ma <i>Kota</b> a kot ma Alę</i> (ŹLE)  
<b>Ala ma <i>Kota</i></b> a kot ma Alę (POPRAWNIE)
```

Wszystkie parametry znaczników należy podawać w cudzysłowach. Oprócz tego, standard definiuje jakie znaczniki mogą być wzajemnie zagnieżdżane – po szczegóły sięgnij do specyfikacji technicznej.

Struktura dokumentu

Najłatwiej omówić strukturę dokumentu na przykładzie:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">  
<html>  
<head>  
<title>Tytuł strony</title>  
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-2">  
</head>  
<body>  
  
<p>To jest treść dokumentu</p>  
  
</body>  
</html>
```

Dokument HTML można bardzo łatwo poznać po tym, że na samym początku znajduje się znacznik `<html>`, a na jego końcu `</html>`. Przed `<html>` umiesza się jeszcze definicję, określającą w jakim standardzie jest napisany cały dokument – w tym przypadku jest to HTML 4.01 Transitional.

Idąc dalej zauważysz podział dokumentu na dwie części:

Nagłówek (`<head>...</head>`)
Ciało dokumentu (`<body>...</body>`)

Nagłówek zawiera podstawowe informacje na temat dokumentu, tj. tytuł, standard kodowania tekstu, wersję językową, informacje o autorze itp. Oprócz tego możesz tam umieszczać style CSS, o których więcej w kolejnych rozdziałach.

W ciele dokumentu znajduje się cała zawartość, którą później zobaczysz na ekranie. To właśnie tutaj umieszczasz tekst właściwy, grafikę, dźwięki, animacje - wszystko to z użyciem odpowiednich znaczników.

Nagłówek dokumentu

Oto przykładowy nagłówek dokumentu HTML. Przeanalizuj go linia po linii.

1. `<!doctype html public "-//W3C//DTD HTML 4.01 Transitional//EN">`
2. `<html>`
3. `<head>`
4. `<title>Ala i jej kot</title>`
5. `<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-2">`
6. `<meta name="description" content="To jest strona domowa Ali i jej kota">`
7. `<meta name="keywords" content="strona domowa, ala, kot">`
8. `<meta name="generator" content="Macromedia HomeSite 5.0">`
9. `<meta name="author" content="Ala Kowalski">`
10. `</head>`

Definiujesz typ dokumentu - jest to HTML 4.01 Transitional.

Otwierasz dokument HTML.

Otwierasz nagłówek.

Ustalasz tytuł dokumentu na "Ala i jej kot".

Definiujesz rodzaj kodowania tekstu w dokumencie na iso-8859-2.

Opisujesz stronę - w kilku słowach, zdaniach.

Wpisujesz słowa kluczowe, po których strona ma być rozpoznawana w wyszukiwarkach.

Definiujesz, jakiego programu użyłeś do stworzenia dokumentu.

Dodajesz autora strony.

Zamykasz sekcję nagłówka.

Taki opis dokumentu jest kompletny - masz wszystkie informacje, których potrzebujesz. Dokument posiada tytuł, informacje o jego zawartości, autorze i narzędziu w jakim był tworzony. Tak naprawdę większość z tych informacji jest zbędna. Najistotniejsze są następujące:

1. `<!doctype html public "-//W3C//DTD HTML 4.01 Transitional//EN">`
2. `<html>`
3. `<head>`
4. `<title>Ala i jej kot</title>`
5. `<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-2">`
6. `<meta name="keywords" content="strona domowa, ala, kot">`
7. `</head>`

Najważniejsze informacje to rodzaj dokumentu, jego tytuł oraz rodzaj kodowania dokumentu. Dodatkowo można też zamieścić listę słów kluczowych - pomoże to znaleźć stronę w wyszukiwarkach internetowych.

Ciało dokumentu

W ciele dokumentu znajdują się wszystkie informacje, które zostaną wyświetlone na ekranie. Przeanalizuj ten fragment.

1. `<body bgcolor="#FFFFFF" text="#000000" link="#0000FF">`
2. `<div align="center">`
3. `<h1>Witam na mojej stronie</h1>`

Tomaz Sitek – Opis tworzenia stron internetowych przy pomocy języka HTML

4. </div>
5. <div>
6. <p>Znajdziesz tutaj bardzo duzo informacji o Mnie i moim kotku...</p>
7.
8. </div>
9. </body>
10. </html>

Otwarcie ciała dokumentu. W znaczniku <body> zdefiniowałeś kilka dodatkowych parametrów:

bgcolor - kolor tła dokumentu,

text - kolor czcionki,

link - kolor linków.

Centrujesz blok do środka.

Włączasz nagłówek pierwszy (jest on największy) oraz wpisujesz powitanie na ekranie.

Zamykasz blok, który był wyrównywany do środka.

Otwierasz nowy blok, którego zawartość będzie wyrównywana do lewej i prawej.

Otwierasz paragraf i wypisujesz tekst. Na końcu zamykasz paragraf.

Wstawiasz rysunek do dokumentu - podajesz jego ścieżkę, rozmiary oraz podpis.

Zamykasz blok otwarty w linii 5.

Zamykasz ciało dokumentu.

Zamykasz dokument HTML.

Jest to dosyć prosty dokument. W następnym rozdziale dowiesz się więcej na temat różnych znaczników, tabel i stylów. Dzięki nim będziesz mógł tworzyć o wiele bardziej zaawansowane dokumenty.

HTML i polskie znaki

Prawdą jest, że nie ma jednego standardu zapisu polskich znaków na stronach WWW. Istnieją trzy metody kodowania tekstu:

iso-8859-2 - jest to oficjalna norma ISO (International Standardization Organization), określająca kody liter alfabetów krajów Europy Środkowej (w tym Polski),

CP-1250 - standard stworzony przez ANSI (American National Standards Institute). Jest on stosowany w systemach operacyjnych firmy Microsoft,

UTF - (Unicode Transformation Formats) jest uniwersalnym formatem, opisującym znaki wszystkich alfabetów (w tym momencie zdefiniowano ponad 100 000 znaków), za pomocą unikalnego, niepowtarzalnego numeru.

Jaki standard wybrać? Jeśli korzystasz z systemu Windows możesz wybrać CP-1250. Jednakże w naszym kraju obowiązują standardy ISO, a nie ANSI. Dlatego też zdecydowana większość stron jest tworzona w iso-8859-2 i ten standard polecamy.

Większość programów to produkty amerykańskie, wobec czego nie wszystkie obsługują kodowanie iso-8859-2. Aplikacje takie jak: FrontPage, Macromedia DreamWeaver 4+, Adobe GoLive 4+ obsługują je bezproblemowo. Natomiast programy: Macromedia HomeSite lub Notatnik samodzielnie nie umożliwiają tworzenia stron w tym formacie. Problem można rozwiązać na dwa sposoby:

stworzyć stronę np. w CP-1250, a następnie skorzystać z jednego z wielu konwerterów tekstu,

zainstalować na swoim komputerze klawiaturę ISO - można ją znaleźć pod adresem <http://windows.online.pl>

Formatowanie tekstu

`<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>`, `<h6>`, `<h7>` - nagłówki

Znaczniki te służą do definiowania nagłówków. Numery oznaczają ich ważność - `<h1>` jest najważniejszy a `<h7>` najmniej ważny.

Przykład:

```
<h1>Nagłówek 1</h1>
```

```
<h4>Nagłówek 4</h4>
```

`<big>`, `<small>` - duża i mała czcionka

Czasami zachodzi potrzeba, by pewien tekst wyróżnić powiększając go lub pomniejszając. Do tego właśnie służą znaczniki `<big>` i `<small>`.

Przykład:

```
<big>To jest strona Ali</big>
```

```
<small>Ostatnia aktualizacja 14.04.2002</small>
```

`<hr>` - pozioma linia

Powoduje wyświetlenie poziomej linii.

Przykład:

```
<hr width="80%" size="1">
```

`` - pogrubienie

Powoduje pogrubienie czcionki.

Przykład:

```
<b>Ala ma kota</b>
```

`<i>` - pochylenie

Powoduje pochylenie czcionki.

Przykład:

```
<i>Ala ma kota</i>
```

`<u>` - podkreślenie

Dzięki temu znacznikowi możesz podkreślić fragment tekstu.

Przykład:

```
<u>Ala ma kota</u>
```

`<blockquote>` - blok cytowany

Tego znacznika najczęściej używa się przy wyróżnianiu pewnych bloków tekstu - np. cytatów, komentarzy, uwag itp.

Przykład:

```
<blockquote>Ala ma kota</blockquote>
```

Tomaz Sitek – Opis tworzenia stron internetowych przy pomocy języka HTML

<pre> - tekst preformatowany

Czasami zachodzi potrzeba wyświetlenia fragmentu kodu strony lub jakiegoś programu, tak jak zostały zapisane. W tym celu właśnie stosuje się znacznik <pre>. W przypadku, kiedy chcesz wyświetlić kod HTML, PHP należy pamiętać o tym, że kod ten będzie normalnie interpretowany. Dlatego też znaki charakterystyczne dla znaczników - < i > - muszą zostać zapisane w bezpiecznej postaci:

< jako < (od lower then)

> jako > (od greater then)

Przykład:

```
<pre>
&lt;?php
print "Hello world";
?&gt;
</pre>
```


 - złamanie linii

Znacznik ten to zwykły enter. Dzięki niemu tekst, który znajduje się za nim będzie wyświetlany w nowej linii.

Przykład:

```
Ala<br>
ma<br>
kota.
```

<p> - paragraf

Jest to jeden z najczęściej używanych znaczników. Służy on do grupowania wybranych bloków tekstów, grafiki w jeden paragraf/akapit. Dzięki użyciu stylów CSS, można nadawać temu znacznikowi pewne właściwości mające wpływ na wygląd treści w nim zamkniętej.

Przykład:

```
<p>Ala ma kota</p>
```

<div> - wycinek dokumentu

Pozwala na wydzielenie pewnego fragmentu dokumentu np. tekstu, grafiki, listy i nadanie mu rodzaju formatowania - środkowanie lub wyrównanie do prawego marginesu.

Przykład:

```
<div align="center">
<h2>To jest strona Ali</h2>
</div>
Obrazki
```

Obrazki są chyba najczęściej spotykanym elementem wzbogacającym dokumenty HTML. Przyjrzyj się więc znacznikowi umożliwiającemu wstawienie obrazka.

```

```

Najważniejszym parametrem znacznika jest ścieżka do pliku obrazka - src. W przykładzie używamy względnej ścieżki do obrazka. Oznacza to, że obrazek znajduje się

Tomasz Sitek – Opis tworzenia stron internetowych przy pomocy języka HTML

w podkatalogu /gfx (względem katalogu, w którym zapisujemy dokument HTML).

Najczęstszym błędem powodującym niewyświetlenie obrazka jest zła ścieżka. Aby się przed tym zabezpieczyć, podawaj ścieżki bezwzględne do obrazka - od głównego katalogu konta, np:

```

```

Następnie ustal szerokość i wysokość obrazka (width i height). Teoretycznie przeglądarka taka jak Internet Explorer poradzi sobie bez tych parametrów. Jednakże inne przeglądarki nie są aż tak sprytnie.

Używając tych parametrów możesz rozciągać i ścisnąć obrazek - wystarczy tylko, że zmienisz prawdziwe wymiary obrazka na inne.

Parametr alt umożliwia podanie alternatywnego tekstu, który pojawi się zamiast obrazka - np. w takich przeglądarkach jak Lynx.

Kolejnym parametrem jest title - umożliwia on nadanie obrazkowi tytułu, który pojawia się w dymku po najechaniu kursorem na obrazek.

Ostatnim z parametrów jest border czyli ramka. W przypadku, gdy wstawiasz obrazek jako odsyłacz podlega on takim samym zasadom jak odsyłacze tekstowe - czyli pojawia się kolor linku. W przypadku obrazka nie zmienia on koloru. Otrzymujesz jedynie kolorową ramkę. Wpisując parament ramki z wartością zerową, likwidujesz tę ramkę.

Odsyłacze

<a> - kotwica, odsyłacz

Odsyłacze, czyli popularne linki można podzielić na cztery kategorie:

- bliskie,
- dalekie,
- wewnętrzne,
- mapy na grafice.

Odsyłacze bliskie

Odsyłacze bliskie odnoszą się do dokumentów w ramach serwisu.

Przykład:

```
<a href="omnie.html">Więcej informacji o Mnie</a>
```

Efektom tego przykładu będzie stworzenie odsyłacza do pliku omnie.html, gdzie podlinkowanym tekstem będzie "Więcej informacji o Mnie".

Pamiętaj, aby poprawnie podawać nazwę dokumentu, do którego się odwołujesz. W przypadku kiedy strony znajdują się na serwerach UNIX'owych, ważna jest wielkość liter. Dlatego też najlepiej będzie, jeżeli podajesz nazwy plików tylko małymi literami. Dla bezpieczeństwa unikaj też niebezpiecznych znaków, takich jak spacje i znaki lokalnego alfabetu. Stosując się do tych zasad, nie będziesz miał problemów z nie działającymi linkami.

Odsyłacze dalekie

Odsyłacze dalekie odnoszą się do stron na innych serwisach internetowych.

Przykład:

```
<a href="http://home.pl">Firma home.pl</a>
```

Przykład ten jest podobny do poprzedniego, jednakże ścieżka do pliku wygląda trochę inaczej. Rozpoczyna się ona od "http://", który oznacza z jakiego połączenia będziesz korzystać - w tym przypadku będą to strony WWW. Jest jednak możliwość wskazania na inną usługę:

```
http:// - strony WWW,  
ftp:// - serwer FTP,  
news:// - grupy dyskusyjne,  
telnet:// - zdalne podłączenie.
```

Możesz także stworzyć odsyłacz do adresu e-mail.

Przykład:

```
<a href="mailto:info@home.pl">Biuro Obsługi Klienta home.pl</a>
```

Po kliknięciu na taki link zostanie otworzony domyślny program pocztowy.

Odsyłacze wewnętrzne

Odsyłacze wewnętrzne umożliwiają nawigację wewnątrz jednego dokumentu.

Przykład:

```
1. <a name="top">
```

```
2. <p>Ala ma kota, ala ma kota, ala ma kota,
```

```
3. ala ma kota, ala ma kota, ala ma kota</p>
```

```
4. <a href="#top">Powrót do góry strony</a>
```

Definiujesz kotwicę, do której będziesz linkować.

Otwierasz paragraf z przykładową treścią.

Zamykasz paragraf.

Wstawiasz link, który przesuwa okno przeglądarki do miejsca gdzie zdefiniowałeś kotwicę.

Przy klikaniu na linkach wewnętrznych przeglądarka nie przeładuje dokumentu - jedynie przesuwa okno w odpowiednie miejsce.

Mapy na grafice

Czasami zachodzi potrzeba umieszczenia na obrazku więcej niż jednego odsyłacza, bądź też podlinkowania tylko wybranej części grafiki. Nie można tego zrobić przy użyciu znacznika <a>. Istnieje za to możliwość zamapowania obrazka i ustawienia odsyłaczy do tych map.

Przykład:

```
1. 
```

```
2. <map name="mapa">
```

```
3. <area shape="rectangle" coords="7, 10, 20, 20" href="obrazek_01.html">
```


4. `<area shape="circle" coords="100, 120, 25" href="obrazek_02.html">`
5. `<area shape="polygon" coords="241, 213, 153, 67, 138" href="obrazek_03.html">`
6. `</map>`

Wstawiasz obrazek - zauważ, że dodaliśmy parametr usemap - określa on jakiej definicji mapy będziesz używać.

Otwierasz definicję mapy.

Rysujesz kwadracik - opcja rectangle mówi, że będziesz rysować czworobok, dlatego też jako koordynanty podaj cztery wierzchołki figury.

circle określa, że rysujesz okrąg lub elipsę. Parametry, które podajesz to współrzędne środka figury oraz promień.

Jest też możliwość tworzenia nieregularnych figur. Opcja polygon pozwala na tworzenie figury poprzez łączenie liniami kolejnych punktów.

Zamykasz definicję mapy.

Tabele

Tabele powstały pierwotnie z myślą o przedstawianiu wyników badań. Dzisiaj tabel używa się przeważnie do formatowania wyglądu strony - dzięki nim możesz podzielić stronę na kilka kolumn, wierszy i w nich umieszczać treść.

Oto przykład prostej tabeli:

1. `<table width="100%" cellpadding="0" cellspacing="0" border="1">`
2. `<tr>`
3. `<td>Kolumna 1 Wiersz 1</td>`
4. `<td>Kolumna 2 Wiersz 1</td>`
5. `</tr>`
6. `<tr>`
7. `<td>Kolumna 1 Wiersz 2</td>`
8. `<td>Kolumna 2 Wiersz 2</td>`
9. `</tr>`
10. `</table>`

Rozpoczynasz od definicji tabeli - podajesz jej szerokość width (ustalasz, że będzie ona rozciągnięta na całą szerokość okna), podajesz odległości między polami tabelki - cellpadding oraz odległość tekstu od krawędzi pola tabelki - cellspacing. Aby zobaczyć obramowanie tabelki musisz je uaktywnić - border.

Otwierasz definicję wiersza.

Otwierasz kolumnę w pierwszym wierszu i wpisujesz do niej przykładowy tekst.

Podobnie jak wyżej, tylko tym razem operujesz na drugiej kolumnie.

Zamykasz pierwszy wiersz.

Otwierasz drugi wiersz.

Podobnie jak w punkcie 3 tworzysz pierwszą kolumnę w drugim wierszu.

To samo dla drugiej kolumny.

Zamykasz drugi wiersz.

Zamykasz tabelę.

Przy tworzeniu tabel obowiązują pewne zasady:

zawsze po otwarciu tabeli powinno znajdować się otwarcie wiersza `<tr>`, a po nim definicja kolumny. Pomiędzy tymi trzema znacznikami nie można umieszczać tekstu, obrazków itp. To samo dotyczy znaczników zamykających kolumny, wiersze i tabele.

Ilość kolumn w każdym z wierszy tabeli powinna być zawsze taka sama.

Zarówno wierszom, jak i kolumnom można nadawać parametry.

1. `<table width="100%" cellpadding="0" cellspacing="0" border="1">`
2. `<tr bgcolor="#cc3300">`
3. `<td width="30%" valign="top">Kolumna 1 Wiersz 1</td>`

Tomaz Sitek – Opis tworzenia stron internetowych przy pomocy języka HTML

4. `<td width="200" align="center">Kolumna 2 Wiersz 1</td>`
5. `</tr>`
6. `</table>`

Otwierasz tabelę i definiujesz jej parametry.

Otwierasz wiersz i ustalasz kolor tła na czerwony.

Wstawiasz pierwszą kolumnę. Ustalasz, że jest ona szeroka na 30% szerokości całej tabeli oraz jej zawartość będzie wyrównana w poziomie do góry tabeli.

Definiujesz kolejną kolumnę. W jej przypadku ustalasz szerokość na 200 pikseli i całą jej zawartość wyrównujesz w pionie do środka.

Zamykasz wiersz.

Zamykasz tabelę.

Istnieje również możliwość łączenia kolumn - zarówno w pionie jak i w poziomie.

1. `<table width="100%" cellpadding="0" cellspacing="0" border="1">`
2. `<tr>`
3. `<td rowspan="2">Kolumna 1 Wiersz 1 i 2</td>`
4. `<td colspan="3">Kolumna 2,3 i 4 Wiersz 1</td>`
5. `</tr>`
6. `<tr>`
7. `<td>Kolumna 2 Wiersz 2</td>`
8. `<td>Kolumna 3 Wiersz 2</td>`
9. `<td>Kolumna 3 Wiersz 2</td>`
10. `</tr>`
11. `</table>`

Otwierasz tabelę.

Następnie otwierasz wiersz.

Wynikiem tego kodu będzie połączenie pierwszej kolumny z wiersza pierwszego z tą samą kolumną w wierszu drugim - rowspan.

W tej linii łączysz trzy sąsiadujące kolumny w jedną - colspan.

Zamykasz wiersz.

Otwierasz drugi wiersz.

Jako że pierwsza kolumna tego wiersza uległa połączeniu w poprzednim wierszu, nie musisz już jej tutaj definiować. Pozostały Ci jedynie trzy z czterech kolumn. Tutaj definiujesz pierwszą z nich.

Tutaj drugą.

I trzecią ostatnią. W tym punkcie należy policzyć czy ilość kolumn w obu wierszach się zgadza.

Zamykasz drugi wiersz.

Kończysz tabelę.

Listy

Istnieją dwa rodzaje list wyliczeniowych w języku HTML.

Listy nieuporządkowane

1. `<ul type="disc">`
2. `pierwszy punkt`
`drugi punkt`
`trzeci punkt`
3. ``

Otwierasz Unordered List - ``. Jest możliwe ustawienie, jakiego rodzaju znak będzie się pojawiał jako akcent punktu:

disc,

circle,

square.

Wypisujesz kolejno wiersze.
Zamykasz listę.

Listy uporządkowane

1. `<ol type="a" start="e">`
2. `pierwszy punkt`
`drugi punkt`
`trzeci punkt`
3. ``

Otwierasz Ordered List - ``. Definiujesz, że lista jako akcenty ma używać małych liter - `type="a"` oraz, że zaczynasz wyliczanie od litery "e" - `start="e"`. Dostępne są również wyliczenia z akcentami:

- 1 - liczbowe,
 - A - litery duże,
 - a - litery małe,
 - I - duże litery rzymskie,
 - i - małe litery rzymskie,
- Wypisujesz kolejno wiersze.
Zamykasz listę.

Formularze

Formularze służą do zbierania danych, które są później przesyłane do obróbki przez skrypty np. przez skrypt wysyłający dane na wskazany adres e-mail - formmail.

1. `<form action="/cgibin/formmail" method="post">`
2. `<input type="hidden" name="recipient" value="jan-kowalski@home.pl">`
3. `<input type="hidden" name="subject" value="Dane z formularza">`
4. Imie i Nazwisko `<input type="text" name="Imie_Nazwisko" size="10" maxlength="100">`
5. `<input type="reset" value="Wykasuj"><input type="submit" value="Wyślij dane">`
6. `</form>`

Rozpoczynasz od otwarcia formularza. Musisz ustalić, gdzie mają zostać przekazane dane z formularza - action (w przykładzie jest to skrypt formmail) oraz określ metodę przesyłania danych.

Nie wolno zagnieżdżać w sobie formularzy. Powoduje to błędne działanie formularzy bądź też ich brak działania.

input oznacza domyślnie pole formularza - poprzez parametr type określasz, jakie właściwości będzie miało to pole. W tej linii zdefiniowałeś pole jako ukryte - hidden - posłuży ono do konfiguracji skryptu, do którego będą przesłane dane. Następnie nazywasz pole - name - oraz przypisujesz mu wartość - value. W przykładzie pole to określa adresata listu.

Każde z pól formularza (wyjątkiem są przyciski resetowania formularza oraz wysyłania) muszą posiadać unikalną nazwę, inaczej wartości w nich zdefiniowane nie zostaną przesłane przez formularz.

Kolejne pole również służy do konfiguracji skryptu formmail - ustawiasz w nim tytuł listu, który będzie przesłany.

W tej linii pole input przyjmuje typ text - nie jest to już pole ukryte. Na stronie pojawi się pole, w którym można wprowadzić dane. Zauważ, że nie występuje tutaj parametr value - chcesz przecież, aby tę wartość podał użytkownik. Pozostaje jeszcze ustalić rozmiar pola - size oraz maksymalną długość tekstu, jaki można wprowadzić w tym polu - maxlength.

Tomasz Sitek – Opis tworzenia stron internetowych przy pomocy języka HTML

W tej linii znajdują się dwa stałe elementy formularzy - przycisk resetujący formularz (`type="reset"`) oraz zatwierdzający formularz (`type="submit"`). W obu przypadkach wartości podane w parametrze `value` zostaną wypisane na przyciskach.
Zamykasz formularz

Pozostałe typy pola `<input>` to:

`button` - umożliwia stworzenie własnego przycisku,
`checkbox` - tworzy pole akceptacji (można je zaznaczyć bądź nie),
`file` - umożliwia załączenie pliku do formularza,
`image` - dzięki temu parametrowi możesz np. jako przycisk wstawić obrazek,
`password` - gdy zdefiniujesz takie pole, możemy bezpiecznie wprowadzać hasła do formularza - zamiast normalnego tekstu pojawią się gwiazdki,
`radio` - jest to pole wyboru (przeważnie wykorzystuje się go do wyboru np. TAK/NIE).

Inne typy pól, które możesz użyć w formularzach to:

`<textarea></textarea>` - jest to większe pole do wpisywania tekstu. Możesz ustalić ile kolumn i wierszy będzie zawierać,
`<select><option>...</select>` - rozwijana lista. Korzystając ze znacznika `<option>` definiujesz kolejne wiersze listy.