Technologie internetowe

CZĘŚĆ 2

HTML – ZAGADNIENIA PODSTAWOWE

Wykład opracowany na podstawie materiałów z:
http://pl.wikipedia.org/
http://www.kurshtml.edu.pl/
http://www.htmlkurs.yoyo.pl/index.html

Zakład Informatyki WBMiL

Dr inż. Arkadiusz Rzucidło

HTML

- **HTML –** to raczej **ogólne określenie języka** do opisywania **struktury** dokumentów stron WWW. Jak podaje Wikipedia:
  - HTML pozwala **opisać strukturę** informacji zawartych wewnątrz strony internetowej, **nadając znaczenie** poszczególnym fragmentom tekstu – formując **hiperłącz**a, **akapity**, **nagłówki**, **listy** – oraz **osadza w tekście** dokumentu obiekty plikowe np**. multimedia** bądź elementy baz danych np. interaktywne **formularze** danych.
  - HTML umożliwia **określenie wyglądu** dokumentu w przeglądarce internetowej. Do szczegółowego opisu formatowania akapitów, nagłówków, użytych czcionek i kolorów, zalecane jest wykorzystywanie **kaskadowych arkuszy stylów**.
  - W składni HTML wykorzystuje się **pary znaczników** umieszczone w nawiasach ostrokątnych, np. <title> i </title> lub <h1> i </h1>.
  - HTML pozwala również na **osadzanie ciągów instrukcji** języków skryptowych, umieszczonych pomiędzy znacznikami <script> i </script>
  - Należy nadmienić, że HTML, będąc językiem znaczników**, nie jest zaliczany do języków programowania** – w jego składni nie przewidziano wyrażeń obliczeniowych, warunkowych czy iteracyjnych.
  - Ważną cechą HTML-u, która przyczyniła się do popularności systemu WWW oraz Internetu, jest **niezależność od systemu operacyjnego** i wykorzystywanego **sprzętu komputerowego**.

HTML – krótka historia

- 1993 – szkic języka opublikowany przez Internet Engineering Task Force. Potem w 1996 opublikowane jako dokument RFC
- 1997 – publikacja HTML 3.2 z rekomendacją W3C oraz 4.0 (w trzech typach: Strict, Transitional, Frameset)
- 1991 – HTML 4.01
- 2001 – HTML 4.01 Strict jako miedzynarodowy standard ISO/IEC
- 2008 – HTML 5- jako szkic

Konstrukcja HTMLa

- Obecna wersja HTML to 4.01
- W HTMLu można wyszczególnić:
  - Znaczniki (wraz z atrybutami)
  - Typy danych
  - Referencje znakowe
  - Odwołania w postacie encji
  - Deklaracja typu dokumentu
- Pliki z kodem html można oznaczać rozszerzeniem ***.html** lub ***.htm**

Znaczniki – tzw. TAGi

- Są podstawą w języku HTML
- Określone są dwoma cechami: atrybutami i zawartością
- Atrybuty i ich zawartości muszą spełniać wymagania by dokument został poprawnie zweryfikowany pod zgodność ze standardem
- Konstrukcja :
  - <nazwa-znacznika atrybut="wartość">zawartość</nazwa-znacznika>
- Rodzaje znaczników:
  - Strukturalne – opisują logiczną strukturę dokumentu
    - <h1> Treść nagłówka </h1>
  - Prezentacyjne – opisują wygląd poszczególnych elementów
    - treść pochylona
    - wzmocnienie
  - Hipertekstowe – zawierają linki do dokumentów lub ich części oraz tego samego dokumentu
    - <nazwa-znacznika atrybut="wartość">zawartość</nazwa-znacznika>

Atrybuty

- Atrybuty przedstawiane są parami **nazwa-wartość**, rozdzielone są znakiem równości.
- Wartości występują w cudzysłowach pojedynczych lub podwójnych.
  - Znaki te mogą być pominięte ale tylko w specyfikacji HTML – nie w XHTML
- Wiele tagów posiada standardowe atrybuty: **id, class, style, title, …** a także unikatowe
  - **id** – służy do nadania unikatowego identyfikatora aby można było użyć go np. w arkuszu stylów lub w skrypcie do nadania odpowiedniego wyglądu tego elementu
  - **class** – klasyfikuje element do celów prezentacyjnych
  - **style** – może być użyty do nadania konkretnego stylu elementowi w kodzie HTML.
  - **title** – dodaje informację tekstową do znacznika. Wyświetlana ona może być w postaci podpowiedzi po najechaniu myszką na dany element na ekranie.

```
<span id="ZadaneId" class="DanaKlasa" style="color:blue;" title="Hypertext Markup Language">HTML</span>
```

Referencje znakowe

- HTML interpretuje znaki specjalne jak np. **<**, **>** oraz **&** jako elementy składni języka.
  - Znaki **<**, **>** oznaczają rozpoczęcie i zakończenie znacznika zaś **&** oznacza rozpoczęcie ciągu znaków encji bądź symbolu (,& amp,).
  - Zapisanie znaku **&** jako **&#38** pozwoli na użycie go jako wartości elementu lub po prostu treści .
  - Znaki podwójnego cudzysłowia mogą być zapisane także jako np. &quot lub &#x22 lub &#34
- Z uwagi na niedbałość przy wprowadzaniu takich znaków deweloperzy przeglądarek zliberalizowali ich działanie aby tolerowały znaki wprowadzane dosłownie jako kod tylko gdy dalszy ciąg treści wskazuje na użycie kodu.

Semantyka w HTMLu

- Takie określenie jak semantyczny HTML nie istnieje ale istnieją wskazówki jak tworzyć poprawny kod HTMLa
- Jest to praktyka tworzenia kodu, który zwiera jedynie znacznie a nie informacje na temat prezentacji.
  - Inaczej – jak powinien być traktowany tekst (np. jako tytuł nagłówka) a nie jak powinien wyglądać (pogrubiony, pochylony z powiększoną czcionką)
- Praktyka:
  - Unikanie znaczników prezentacyjnych
  - Używanie znaczników do wyodrębnienia znaczeń poszczególnych fraz i struktur w dokumencie
- Semantyczny HTML jest wynikiem zastosowania kaskadowych arkuszy stylów CSS
  - Oddziela warstwę semantyczną od warstwy prezentacji
  - Daje zgodność stylów elementów o tym samym znaczeniu (nagłówki, treść, tabele)
  - Autor nie musi myśleć nad prezentacją dokumentu
  - Dokument staje się niezależny i można go prezentować na wiele sposobów

XHTML

- Extensible HyperText Markup Language – **rozszerzony**
- Specyfikacja określona przez W3C
- XHTML /1.0 Nie jest następcą HTMLa a jedynie przedstawienien HTML 4.01 w postaci XMLa.
- Dokumenty pisane w XHTML są zgodne ze specyfikacją XMLa – są inaczej: poprawnymi dokumentami XML
  - Reguły obejmują m.in. konieczność domykania wszystkich znaczników
  - Dokument nie poprawny składniowo nie może być przetworzony przez parser XML
- Dokument XML nie ma informacji jak należy go wyświetlać w przeglądarce dlatego większość przeglądarek traktuje go jako tekst
- Można jednak do prezentacji dokumentów XML wykorzystać kaskadowe arkusze stylów CSS
  - Umieszcza się wtedy na początku takiego dokumentu informację np.:

```
<?xml-stylesheet type="text/css" href="moj-styl.css"?>
```

Reguły XML

- powinien zawierać deklarację XML, która musi być umieszczona na samym początku pliku (nie może być poprzedzona np. komentarzem) oraz musi posiadać atrybut version (dopuszczalne wartości to 1.0 albo 1.1) oraz opcjonalnie atrybuty:
- encoding – deklaruje zestaw znaków używanych w dokumencie XML, wartością domyślną jest kodowanie UTF-8 w systemie Unicode.
- standalone – określa tryb dokumentu XML, może przyjmować wartość yes lub no. Jeśli ustawimy wartość na yes to będzie oznaczało, że dokument nie zawiera innych plików, które muszą zostać przetworzone wraz z dokumentem. Może to być np. zewnętrzny arkusz stylów lub definicja DTD;
- musi zawierać dokładnie jeden element główny (zwany *root element*), w naszym przypadku elementem głównym jest element o nazwie ksiazka-telefoniczna;
- każdy element musi zaczynać się znacznikiem początku elementu np. <data> oraz kończyć identycznym znacznikiem końca elementu np. </data>, wyjątek stanowią elementy puste (<element-pusty />), czyli takie które nie zawierają żadnych danych, ani innych elementów, mogą zawierać atrybuty;

Reguły XML

- nazwy elementów mogą zawierać znaki alfanumeryczne (litery a-z, A-Z oraz cyfry 0-9), znaki ideograficzne (ą, ó, ń jednak należy unikać takich konstrukcji [potrzebne źródło]) oraz 3 znaki interpunkcyjne (podkreślenie _, łącznik -, kropka.). Znak dwukropka zarezerwowany jest dla identyfikacji przestrzeni nazw, której nazwa dopisywana jest przed nazwą elementu np. <przestrzeń1:element>,
- nazwy elementów nie mogą zaczynać się od znaku łącznika -, kropki, czy cyfry. Dodatkowo nie mogą zaczynać się od xml, XML, xML itp. (wielkość liter bez znaczenia).
- elementy można zagnieżdżać w sobie i wtedy każdy element znajdujący się wewnątrz innego elementu jest nazywany "dzieckiem" tego elementu, a element wewnątrz którego znajdują się inne elementy zwany jest "rodzicem" tych elementów. W naszym przypadku element <newsy> jest rodzicem elementu <news>, element <news> jest dzieckiem elementu <newsy>, a zarazem rodzicem elementów <data>, <autor>, <e-mail>, <tresc>. Nie można stosować konstrukcji takiego typu: <news><data></data><tresc></news></tresc>, ponieważ element <tresc> nie jest prawidłowo zagnieżdżony w elemencie <news>;

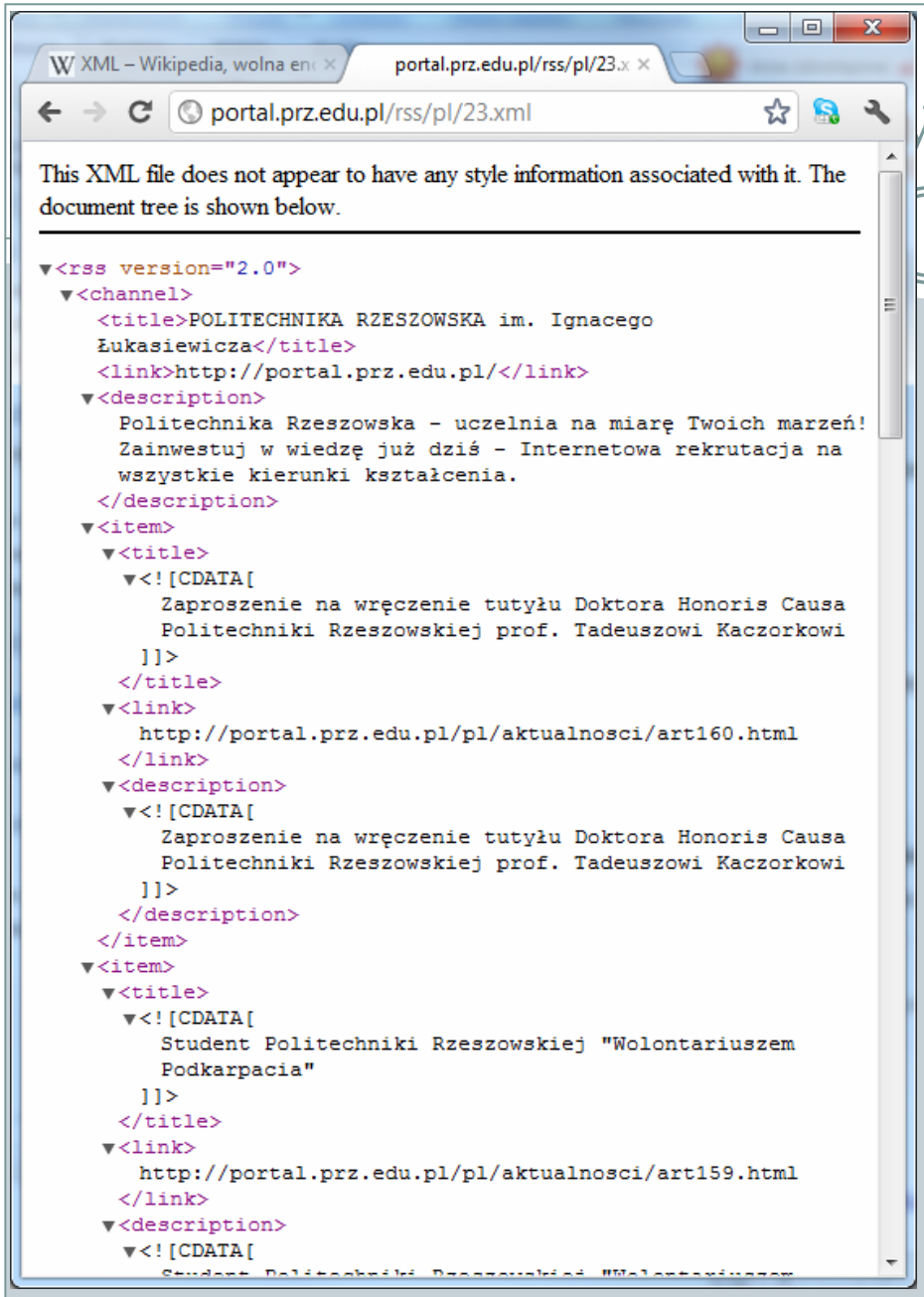
Reguły XML

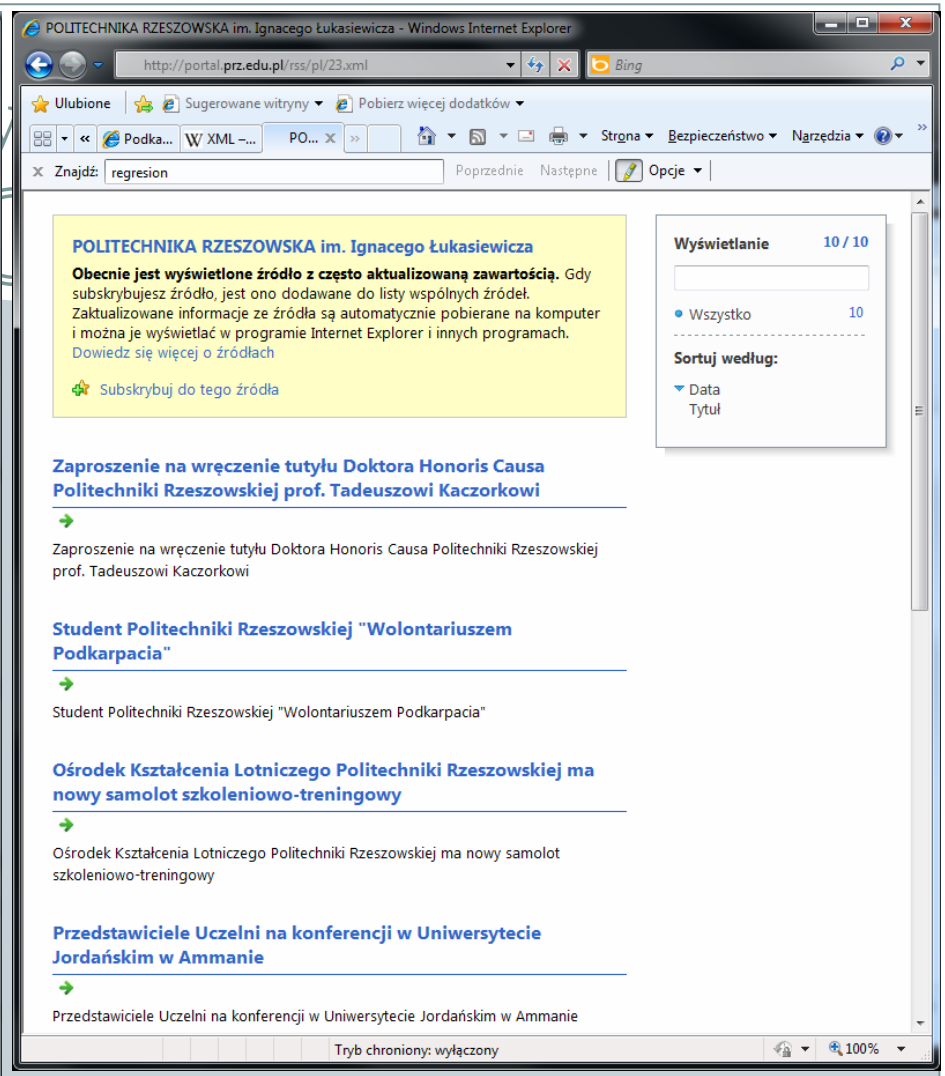
- każdy element może zawierać atrybuty, które definiuje się w znaczniku początku elementu np. atrybutem elementu <news potw="yes"> jest atrybut o nazwie potw oraz wartości yes. Wartości atrybutów podaje się w cudzysłowach albo apostrofach (pojedynczych cudzysłowach).
- informacje, które zawiera element muszą być zapisane pomiędzy znacznikiem początku i końca elementu,
- w danych, atrybutach oraz nazwach elementów nie mogą pojawiać się niektóre znaki. Przykładem może być znak mniejszości (<), lub ampersand (&). Znaków tych nie można używać, ponieważ parsery XML "widząc" np. znak mniejszości wewnątrz elementu stwierdzą, że jest to początek znacznika i dokument zostanie błędnie zinterpretowany. Specyfikacja XML daje możliwość używania takich znaków z wykorzystaniem predefiniowanych odniesień jednostki. Jeśli np. chcemy wstawić znak mniejszości (<) wpisujemy zamiast niego sekwencję < znak ampersand – & znak większości (>) >

Reguły XML

- jeżeli nie chcemy używać predefiniowanych odniesień jednostek możemy część danych, które zawierają np. kod html lub xml zapisać w sekcji danych znakowych, która nie będzie przetwarzana przez analizator składni XML. Znacznik początku sekcji danych znakowych to: <![CDATA[, a znacznik końca:]]>.
- w dokumencie XML możemy wykorzystywać komentarze, które zaczynają się znakami: <!--,, a kończą: -->. Przykład: <!-- To jest komentarz -->,
- specyfikacja XML zezwala na wstawianie instrukcji przetwarzania, które są wykorzystywane do przeniesienia informacji do aplikacji. Instrukcje przetwarzania rozpoczynają się znakami: <?, a kończą: ?>. Przykładem takiej instrukcji może być odniesienie do arkusza stylów, który jest powiązany z dokumentem XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<ksiazka-telefoniczna kategoria="bohaterowie książek">
 <!-- komentarz -->
  <osoba charakter="dobry">
    <imie>Ambroży</imie>
    <nazwisko>Kleks</nazwisko>
    <telefon>123-456-789</telefon>
  </osoba>
  <osoba charakter="zły">
    <imie>Alojzy</imie>
    <nazwisko>Bąbel</nazwisko>
    <telefon/>
  </osoba>
</ksiazka-telefoniczna>
```


XML – Wikipedia, wolna en
portal.prz.edu.pl/rss/pl/23.x
portal.prz.edu.pl/rss/pl/23.xml
This XML file does not appear to have any style information associated with it. The document tree is shown below.
<rss version="2.0">
<channel>
<title>POLITECHNIKA RZESZOWSKA im. Ignacego Łukasiewicza</title>
<link>http://portal.prz.edu.pl/</link>
<description>
Politechnika Rzeszowska - uczelnia na miarę Twoich marzeń! Zainwestuj w wiedzę już dziś - Internetowa rekrutacja na wszystkie kierunki kształcenia.
</description>
<item>
<title>
<![CDATA[
Zaproszenie na wręczenie tutyłu Doktora Honoris Causa Politechniki Rzeszowskiej prof. Tadeuszowi Kaczorkowi
]]>
</title>
<link>
http://portal.prz.edu.pl/pl/aktualnosci/art160.html
</link>
<description>
<![CDATA[
Zaproszenie na wręczenie tutyłu Doktora Honoris Causa Politechniki Rzeszowskiej prof. Tadeuszowi Kaczorkowi
]]>
</description>
</item>
<item>
<title>
<![CDATA[
Student Politechniki Rzeszowskiej "Wolontariuszem Podkarpacia"
]]>
</title>
<link>
http://portal.prz.edu.pl/pl/aktualnosci/art159.html
</link>
<description>
<![CDATA[


POLITECHNIKA RZESZOWSKA im. Ignacego Łukasiewicza - Windows Internet Explorer
http://portal.prz.edu.pl/rss/pl/23.xml
Ulubione
Sugerowane witryny
Pobierz więcej dodatków
Znajdź: regresion
Poprzednie
Następne
Opcje
POLITECHNIKA RZESZOWSKA im. Ignacego Łukasiewicza
Obecnie jest wyświetlone źródło z często aktualizowaną zawartością. Gdy subskrybujesz źródło, jest ono dodawane do listy wspólnych źródeł. Zaktualizowane informacje ze źródła są automatycznie pobierane na komputer i można je wyświetlać w programie Internet Explorer i innych programach.
Dowiedz się więcej o źródłach
Subskrybuj do tego źródła
Wyświetlanie 10 / 10
Wszystko 10
Sortuj według:
Data
Tytuł
Zaproszenie na wręczenie tutyłu Doktora Honoris Causa Politechniki Rzeszowskiej prof. Tadeuszowi Kaczorkowi
Zaproszenie na wręczenie tutyłu Doktora Honoris Causa Politechniki Rzeszowskiej prof. Tadeuszowi Kaczorkowi
Student Politechniki Rzeszowskiej "Wolontariuszem Podkarpacia"
Student Politechniki Rzeszowskiej "Wolontariuszem Podkarpacia"
Ośrodek Kształcenia Lotniczego Politechniki Rzeszowskiej ma nowy samolot szkoleniowo-treningowy
Ośrodek Kształcenia Lotniczego Politechniki Rzeszowskiej ma nowy samolot szkoleniowo-treningowy
Przedstawiciele Uczelni na konferencji w Uniwersytecie Jordańskim w Ammanie
Przedstawiciele Uczelni na konferencji w Uniwersytecie Jordańskim w Ammanie
Tryb chroniony: wyłączony
100%

Edytory kodu HTML

- Pajączek (shareware)
- EdHTML (darmowy)
- kED (darmowy)
- Web Edit (darmowy)
- …
- Notatnik ☺

Darmowe

- Bluefish
- Kate
- gedit ☺

Klasyczne ramy dokumentu

- Standard kodowania znaków **UTF-8** ale nadal używanym i popularnym jest **ISO-8859-2**
- Windows standardowo koduje znaki w **Windows-1250**, trzeba konwertować – notatnik
- Linux standardowo koduje znaki w UTF-8 ale treści można zapisywać również w **ISO**

```
<!DOCTYPE html
                PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
                "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pl" lang="pl">
<head>
                <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-2" />
                <meta name="Description" content="Tu wpisz opis zawartości strony" />
                <meta name="Keywords" content="Tu wpisz wyrazy kluczowe rozdzielone przecinkami" />
                <title>Tu wpisz tytuł strony</title>
</head>
<body>

Tu wpisuje się treść strony

</body>
</html>
```

Ramy dokumentu XHTML

- W dokumencie XHTML – deklaracja zgodności z XML
  - <?xml version="1.0" encoding="iso-8859-2"?>
- Modyfikacja identyfikatora DOCTYPE
  - <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
- Znacznik HTML jest kolejny elementem zmiany
  - <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pl" lang="pl" />

```
<?xml version="1.0" encoding="iso-8859-2"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pl" lang="pl" />
 <head>
<title>Tytuł strony</title>
</head>
<body>

treść strony

</body>
</html>
```

Sekcja HEAD

- Tytuł strony – powinna go posiadać każda strona
  - <title> Wybrany tytuł strony </title>
  - Wyświetlany jest na pasku tytułowym okna przeglądarki
  - Jest również rezultatem wyszukiwania jak połączenie do serwisu.
  - Może zawierać tylko tekst – inne znaczniki nie są dozwolone
  - Optymalna długość to 60-80 znaków
  - Nie można formatować tekstu tytułu
  - Nie używa się samych dużych liter ponieważ niektóre przeglądarki mogą pominąć taką nazwę
  - Można korzystać ze znaków ?,*,! Ale znaki @ i # nie są zalecane
  - Więcej o atrybutach: tutaj
- Opis zawartości strony
  - <meta name ="Descriptionv content="*opis strony*" />
  - Brak opisu spowoduje że wyszukiwarka wyświetli pierwszych kilka wierszy z naszej strony
  - Nie należy uzywac w zawartości cudzysłowów
  - Zaleca się nie przekraczanie 150-200 znaków

Sekcja HEAD

- Wyrazy kluczowe
  - <meta name="Keywords" content=" *wyraz1, wyraz2, wyraz3, wyraz4...*" />
  - Ułatwienie dla sieciowych programów indeksujących
  - Wystarczy kilka, kilkanaście wyrazów – nie przekracza 200
- Autor strony
  - <meta name="Author" content="*imię i nazwisko*" />
- Data utworzenia dokumentu
  - <meta http-equiv="Creation-Date" content="Wed, 14 Nov 2007 14:43:43 GMT" />
- Data ważności dokumentu
  - <meta http-equiv="Expires" content="*data*" />
- Automatyczne przekierowanie
  - <meta http-equiv="Refresh" content="*s*; url= *adres strony lub ścieżka dostępu*" />
  - Wczytanie nowej strony podanej URLem po liczbie sekund - przeniesienie serwisu na inny adres

Sekcja HEAD

- Roboty
  - <meta name="Robots" content="*metoda*" />
  - Pozwala na indeksację witryny
  - Metody:
    - **index** - strona, na której wstawiono polecenie, będzie indeksowana przez *roboty* sieciowe (*indeksery*) - domyślnie,
    - **noindex** - strona, na której wstawiono polecenie, nie będzie indeksowana,
    - **follow** - indeksowanie stron, do których odnoszą się odsyłacze, znajdujące się w dokumencie - domyślnie,
    - **nofollow** - nie indeksuje stron wskazywanych przez odsyłacze wstawione na stronie,
    - **index, nofollow** - indeksuje stronę, nie indeksuje stron wskazywanych przez odsyłacze,
    - **noindex, follow** - nie indeksuje strony, na której wstawiono polecenie, indeksuje strony wskazywane przez odsyłacze,
    - **all** = index, follow - indeksuje wszystko (domyślnie),
    - **none** = noindex, nofollow - nie indeksuje nic.

- Dalsze elementy sekcji HEAD: tutaj

Sekcja BODY – szablon strony

- Sekcja BODY jest miejscem umieszczenia treści strony serwisu
- Każdy serwis powinien posiadać szablon
- Konstrukcja szablonu zależy od przyjętej techniki
  - Tabele – przestarzały model konstruowania serwisu
  - Ramki – FRAMESET – podział strony na kilka fragmentów (Okienek) złożonych z poszczególnych pojedynczych dokumentów (X)HTML

```
<frameset cols="180,*" border="0" frameborder="0" framespacing=„0">
    <frame name="spis" noresize="noresize" frameborder="0" src="spis.html" />
    <frame name="strona" noresize="noresize" frameborder="0" src="home.html" />
    <noframes><body><a href="spis.html">spis treści</a></body></noframes
</frameset>
```

  - Div – bloki – technologia która jest obecnie zalecana jako podstawowa metoda konstrukcji szablonów serwisów.
    - Bloki „div” działają najlepiej z kaskadowym szablonem stylów CSS.

```
<div id="top">
    < div id="NAGLOWEK">Nagłówek szablonu</div>
    <div id="MENU">Menu nawigacyjne</div>
    < div id="TRESC">Treść strony</div>
    <div id="STOPKA">Stopka serwisu</div>
</div>
```

Szablon strony DIV - CSS

- Elementem uzupełniającym szablon na DIVach jest szablon stylów CSS

```
html,
body { background-color: #fff; color: #000; margin: 0; padding: 0; }
#top { width: 780px; }
#NAGLOWEK { background-color: #888; }
#MENU { width: 150px; float: left; overflow: hidden; background-color: #ccc; }
#TRESC { width: 630px; float: left; overflow: hidden; background-color: #fff; }
#STOPKA { clear: both; width: 100%; background-color: #888; }
```

Zalety szablonów na DIVach

- Łatwo zmienić globalny wygląd witryny modyfikując jedynie ustawienia szablonu stylów CSS nie zmieniając żadnej z podstron serwisu.
- Witryna na każdą okazję (Nowy Rok, Święta, ...)
- Personalizacja witryny na zasadzie kilku dostępnych wersji (skórek)
- Mniejsze rozmiary podstron - mniejsze obciążenie kodem
- Inne wersje stron na klasyczny widok komputera, urządzenia mobilne, wydruk, itp.
- Strony poprawne semantycznie są łatwiejsze do indeksacji przez wyszukiwarki
- Ułatwienia korzystania z witryny osobom niepełnosprawnym (niewidomym)
- Rozbudowa serwisu związana jest z jednolitym układem

Dlaczego nie tabele ?

- W tabelach prezentuje się dane tabelaryczne a nie do szablonów – błąd semantyczny
- Zwiększają rozmiary dokumentów
- Ograniczają możliwość stosowania stylizacji CSS
- Utrudniają odbiór treści przez osoby niewidome i korzystanie z przeglądarek tekstowych
- Wpływają negatywnie na pozycjonowanie

- Kiedy zatem stosować tabele ?
  - Tam gdzie można wyszczególnić (logicznie wyznaczyć) nagłówek i pozycje kolumn i wierszy
  - Kiedy prezentujemy przynajmniej dwa egzemplarze podobnych danych (dwa wiesze, dwie kolumny)

Stałe szablony serwisu

- Stały szablon strony to taki, który ma odgórnie ustalony rozmiar w pikslach
  - Szablon jest niezależny od rozdzielczości
  - Szerokość szablonu nie powinna być szersza niż najmniejsza stosowana rozdzielczość ekranowa
  - Przy wyższych rozdzielczościach puste miejsce zagospodarowuje się tłem
  - Efekt porządkowania elementów na stronie jest stały - komfort
    - Tytuły artykułów, treści rozłożone tak jak zaplanowano
  - Popularna rozdzielczość to 1024x768 co powoduje, że standard w tworzeniu szablonów daje dostępną przestrzeń w granicach 1004 px
  - Kolumna bloku tekstu nie powinna być większa niż 780 px
  - Szczegóły

Szablon Płynny

- Szablon płynny zmienia swoją szerokość w zależności od wielkości okna przeglądarki lub rozdzielczości ekranu
- Zazwyczaj dotyczy 100% szerokości okna
- Nie jest popularny ze względu na nie możliwe do przewidzenia zachowanie się elementów na stronie serwisu
- W szablonach płynnych elementem dostosowującym się do szerokości okna jest treść. Pozostałe elementy jak np. menu mają stałe wymiary

- Szczegóły

Elementy blokowe HTML

- Elementy blokowe – zajmują samodzielne miejsce co oznacza że nie można umieścić dwóch takich samych elementów obok siebie
- Mogą zawierać elementy blokowe jak i liniowe
- Przeznaczone są do tworzenia większych struktur niż elementy liniowe.
- Domyślnie zaczynają się od nowej linii

- p
- h1, h2, h3, h4, h5, h6
- ol, ul
- pre
- address
- blockquote
- dl
- div
- fieldset
- form
- hr
- noscript
- Table

Elementy liniowe HTML

- Elementy liniowe – mogą występować obok siebie w jednym wierszu
- Zawierają tekst lub inne elementy liniowe
- Nie tworzą nowego wiersza tak jak elementy blokowe

  - b, big, i, small, tt, s, strike, u
  - abbr, acronym, cite, code, dfn, em, kbd, strong, samp, var
  - a, bdo, img, map, object, q, script, span, sub, sup
  - button, input, label, select, textarea

Zagnieżdżanie elementów

- Zagnieżdżanie elementów to zawieranie się jednych w drugich.
  - Przykładowy etap zagnieżdżania
- Nieprawidłowy zapis
  - Zły zapis zagnieżdżenia
- Dwie proste zasady
  - Znaczników zamykających powinno być tyle samo i le otwierających
  - Kolejność znaczników zamykających musi być odwrotna niż znaczników otwierających

- Charakterystyka elementów HTML
- Wykaz atrybutów
- Elementy zdeprecjonowane

Pytania sprawdzające

- Czym jest HTML ? Czy jest językiem programowania?
- Obecny standard HTML?
- Jakie są rozszerzenia plików HTML?
- co to są znaczniki HTML i jak można je podzielić ?
- Co to są atrybuty i wartości elementów HTML?
- Referencje znakowe?
- Na czym polega poprawność kodu HTML?
- Podstawowe reguły XMLa?
- Kodowanie znaków?
- Klasyczne ramy HTML?
- Ramy dokumentu XHTMKL?
- Budowa sekcji HEAD?
- Szablony strony WWW. Rodzaje?
- Zalety szablonów na DIVach?
- Dlaczego nie stosuje sie szablonów opartych o Tabele?
- Kiedy stosować Tabele?
- Szablon stały, szablon płynny?
- Elementy blokowe, elementy liniowe?
- Zagnieżdżanie elementów?

Netografia

- http://www.kurshtml.edu.pl
- http://pl.wikipedia.com
- http://www.htmlkurs.yoyo.pl/index.html

Koniec

Temat następnego wykładu to:

HTML
Wymiana danych