

# HTML/XHTML/HTML5

## 1. Wprowadzenie

Beata Pańczyk

1

## Bibliografia

- Matthew MacDonald: HTML5. Nieoficjalny podręcznik, O'Reilly 2012
- Meyer Eric A., 'Podręcznik CSS', Helion, Gliwice, 2011
- <http://www.w3schools.com/>
- <http://www.w3.org>
- [http://wazniak.mimuw.edu.pl/index.php?title=Aplikacje\\_WWW](http://wazniak.mimuw.edu.pl/index.php?title=Aplikacje_WWW)

2

## Plan wykładu

- 1. Wprowadzenie:
  - Rys historyczny
  - Adres URL
  - Dokumenty statyczne i dynamiczne
  - Język HTML
  - Przeglądarki WWW
  - Edytory HTML
- 2. Budowa dokumentu HTML
  - Struktura dokumentu HTML i XHTML
  - Struktura dokumentu HTML5
  - Podstawowe znaczniki HTML
- 3. Formularze
  - Osadzanie formularzy w dokumencie HTML
  - Podstawowe pola formularzy HTML
  - Atrybuty pól

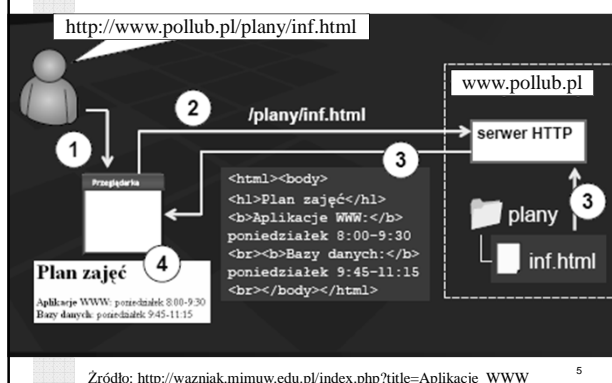
3

## Rys historyczny

- Tim Berners-Lee (1989) - twórca koncepcji hipertekstu, związany ze szwajcarskim CERN (Organisation Européenne pour la Recherche Nucléaire) - opracował pierwszą składnię języka HTML, zaimplementował pierwszy serwer WWW (httpd) i pierwszą przeglądarkę WWW (WorldWideWeb - z edytorem HTML)
- pierwszy serwer WWW - uruchomiony 6 sierpnia 1991 roku pod adresem <http://info.cern.ch/>.
- w 1994 Tim Berners-Lee założył przy MIT (Massachusetts Institute of Technology) organizację World Wide Web Consortium (W3C) - do dziś zajmuje się koordynacją rozwoju technologii WWW

4

## Interakcja w środowisku WWW



5

## Protokół HTTP - przykład

### Składnia URI

- `<scheme>://<authority><path>?<query>`

### Przykład schematu http

`http://my.host.pl:8080/index.php?z1=w1&z2=w2`

- `http:` – schemat, rodzaj protokołu
- `//my.host.pl` – autoryzacja, nazwa hosta
- `8080` – nr portu
- `/index.php` – ścieżka do zasobu
- `z1=w1&z2=w2` – zapytanie

6

# Dokumenty statyczne i dynamiczne

- Dokument statyczny - gotowy do pobrania plik w systemie plików serwera HTTP
- Dokument dynamiczny - dokument generowany na żądanie przez program po stronie serwera HTTP
- Aplikacje WWW – program komputerowy, który pracuje na serwerze i komunikuje się poprzez sieć komputerową z hostem użytkownika komputera z wykorzystaniem przeglądarki internetowej użytkownika (interaktywny klient aplikacji internetowej).

7

## Język HTML

- najważniejszy język definicji dokumentów dla klientów HTTP - umożliwia zapis treści dokumentu i równocześnie opis jego układu
- opracowany przez Tima Berners-Lee w 1990, częściowo w oparciu o język SGMLguid (lokalna odmiana języka SGML stosowana w CERN)
- prawdopodobnie najstarszy dokument HTML dostępny w Internecie - utworzony 13 listopada 1990 roku

## Główne etapy rozwoju HTML

- **SGML** (1986) - fundamentalny pomysł na komputeryzację tekstu w epoce, w której mógł znaleźć zastosowanie w lokalnych systemach, obsługiwanych przez grono profesjonalistów (dokumentacja techniczna, bazy prawne, encyklopedie, itp).
- wersje **HTML 2.0, 3.2, 4.0 i 4.01** (1995-1999), oparte formalnie na SGML, realizowały pomysł na nowe medium publikacji - odniosło wielki sukces, lecz w praktyce nastąpiło zepsucie koncepcji dokumentów elektronicznych i brak perspektyw rozwoju.
- **XML** (1998) - przemyślana adaptacja SGML do warunków sieciowego medium, z naciskiem na automatyzację tworzenia dokumentów, z myślą o złożonych transformacjach treści czy wręcz transakcjach zawieranych przez maszyny w Sieci.

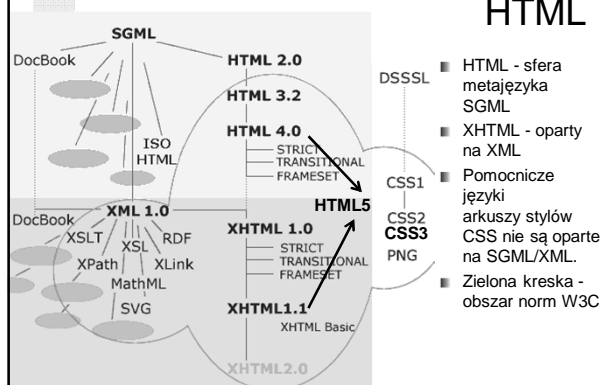
9

## Główne etapy rozwoju XHTML

- **XHTML 1.0** (2000) - XML-owa wersja HTML 4.01, zachowuje ciągłość znaczeń i interpretacji języka przez przeglądarki, ale spełnia już wymogi przetwarzania dokumentów dobrze uformowanych.
- **XHTML 1.1** (2001), standard, który kontynuuje już tylko wersję ścisłą (*strict*) języka HTML, zrywając z kompatybilnością języka wobec wcześniejszych wersji. Jego istotą jest stworzenie wspólnych ram dla modularnych języków rodziny XHTML.
- **Przyszłość:** HTML5 (2008)- rozwinięciem języka HTML4 i jego XML-owej odmiany (XHTML 1), opracowywane w ramach prac grupy roboczej WHATWG (*Web Hypertext Application Technology Working Group*) i W3C

1

## Ewolucja standardu języka HTML



# HTML – język tworzenia stron WWW

- **Dokument HTML** – plik tekstowy (ASCII) z opisem przy pomocy znaczników
- Zależność wyglądu strony od przeglądarki (Mozilla Firefox, Internet Explorer, Safari, Opera, Google Chrome) i narzędzi wspomagających (odtwarzanie dźwięków, animacji, obrazów)
- Problemy: wersja (standard HTML), rozszerzenia specyficzne dla określonych przeglądarek, ustawienia przeglądarek, znaki narodowe

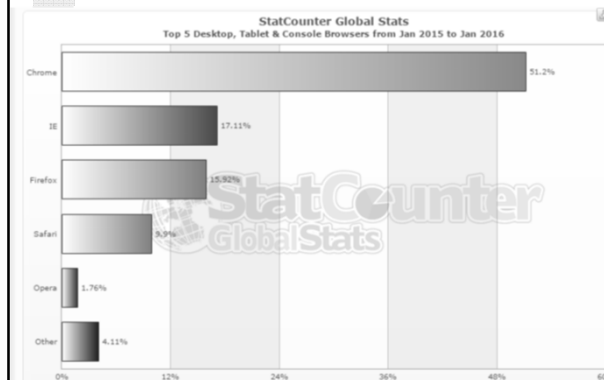
1

## Udział poszczególnych przeglądarek na rynku – 2011, 2014

- Czołówka najpopularniejszych przeglądarek na świecie (dane z sierpnia 2014 i kwietnia 2011 wg serwisu StatCounter: <http://gs.statcounter.com/>):
  - Google Chrome – 46.26% (18,29% - 2011)
  - Internet Explorer – 20.31% (44,52% - 2011)
  - Mozilla Firefox – 17.5% (29,67% - 2011)
  - Safari – 10.81% (5,04% - 2011)
  - Opera – 1,47% (1,91% - 2011)

13

## Udział przeglądarek – I 2016



## 2. Budowa dokumentu HTML/XHTML

### Znaczniki HTML

- **Znaczniki (tagi)** - opis sposobu formatowania strony i jej struktury (wstawienie dodatkowych elementów generowanych lub też z pliku);
- **Zasada ignorowania nieznanych znaczników**
- **Rodzaje znaczników:**
  - Parzyste: `<znacznik>....</znacznik>`
  - Nieparzyste: `<znacznik>`
  - (w XHTML znacznik nieparzysty ma postać: `<znacznik />`)
- **Parametry (atrybuty, własności)** znacznika:
  - wpisywane po nazwie, wartość parametru: po znaku = (ujęta w prosty cudzysłów "..." lub pojedynczy apostrof '...')
  - `<znacznik par1="wart_par1" par2="wart_par2"....>`

16

### Kodowanie polskich liter

- najbardziej polecana norma UTF-8
- zależność od ustawień przeglądarki
- polecenie informujące przeglądarkę o zastosowanym zestawie (w nagłówku dokumentu):
 

```
<meta http-equiv="content-type"
content="text/html; charset=utf-8">
<meta http-equiv="content-type"
content="text/html; charset=iso-8859-2">
```

 (w XHTML znaczniki meta jako nieparzyste mają postać: `<meta ... />`)

17

### Struktura dokumentu HTML

- Tekst ASCII zawarty pomiędzy znacznikami:
 

```
<html> ... </html>
```
- Nagłówek – nie wyświetlana część informacyjna o dokumencie:
 

```
<head>...</head>
```
- Ciało dokumentu, część właściwa:
 

```
<body>...</body>
```
- Ignorowanie: wielokrotnych spacji, przejścia do nowego wiersza (ENTER) => jawność określania końca akapitów, wcięcia w celu wydzielenia struktury dokumentu

18

## Znaczniki w sekcji HEAD

- Tytuł dokumentu wyświetlany w pasku tytułowym przeglądarki:  
`<title> ... </title>`
- Dodatkowe informacje dla przeglądarki, dla wyszukiwarek (atrybut NAME stosowany jest w miejsce atrybutu HTTP-EQUIV)
  - `<meta name="keywords" content="słowa kluczowe">`
  - `<meta name="description" content="opis strony">`
  - `<meta name="generator" content="edytor">`
  - `<meta name="autor" content="autor">`
  - `<meta name="copyright" content="...">`
  - `<meta http-equiv="content-type" content="text/html; charset=iso-8859-2">`
- podstawowy adres URL dla wszystkich adresów względnych  
`<base href="http://www.mojserwer.pl">`

19

## Struktura dokumentu HTML4

```
<!doctype html public "-//w3c//dtd
html 4.01 transitional/en">
<html>
<head>
  <title>tytuł w nagłówku </title>
  <meta ... >
</head>
<body>
  <p>akapit1 <br>
  akapit1 akapit1 ...
  </p>
  <p>akapit2 akapit2
  akapit2 akapit2 ...
  </p>
</body>
</html>
```

- Prolog dokumentu (od wersji HTML 4.01)
- początek dokumentu HTML
- początek nagłówka
- tytuł strony
- informacje dla przeglądarki
- koniec nagłówka
- początek właściwej treści dokumentu
- początek akapitu1 i podział wiersza (w XHTML znacznik `<br />`)
- koniec akapitu 1
- koniec treści
- koniec dokumentu HTML

20

## Co to jest XHTML

- **XHTML = HTML + XML**
- **Standard rekomendowany przez W3C do roku 2014 - XHTML (eXtensible HyperText Markup Language)** to język hipertekstowy używany do tworzenia dokumentów czyli stron internetowych. Polecenia tekstowe (znaczniki i atrybuty) pozwalają na realizację tzw. linków (odsyłaczy), wyświetlanie różnych czcionek, grafiki i innych elementów.
- XML to znacznikowy język do opisu danych (metajęzyk). Język ten to otwarty standard stworzony przez W3C. XML służy do tworzenia aplikacji XML czyli innych języków

21

## XHTML

- XHTML - standard opisu stron WWW zgodny z XML, uniwersalny standard przetwarzania opisu środowiska WWW i dokumentów strukturalnie uporządkowanych informacji
- XHTML jest ściśle związany z kaskadowymi stylami - CSS1 i CSS2, CSS3
- Obecnie zastąpiony standardem HTML5

22

## Podobieństwa z HTML

- Dane są w formie tekstowej
- Dane opisywane są znacznikami (tagami) np: `<img>`
- Znaczniki mogą posiadać atrybuty np: `src="przyk.gif"`
- Atrybuty posiadają wartości (przykład jak wyżej)
- Tagi rozpoczynają się znakiem '`<`' a kończą '`>`' np: `<html>`
- Przykład zamknięcia znacznika (jeśli jest wymagany) `</html>`
- Wartości zawarte są w symbolach '"' np: `"przyk.gif"`
- Komentarze zawierają się między `<!-- -->` i mogą być używane w całym dokumencie

23

## Różnice XHTML w stosunku do HTML 4

- Znaczniki niepuste mają zawsze znacznik końcowy np: `<p>Akapit</p>`
- Znaczniki puste (logiczne) muszą być zawsze zakończone `/>` np: `<br />`
- Wszystkie atrybuty i tagi muszą być pisane małymi literami np: ``
- Atrybuty typu boolean (logiczne) muszą być uzupełniane o wartość np: `selected="selected"`
- Wartości atrybutów muszą być zawarte w symbolach (nawet te jedno wyrazowe) "" np: `<table summary="opis tabeli" width="150">`
- Zlikwidowano znaczniki (dotyczy też HTML 4):
  - `<font>`, `<basefont>`, `<u>` - należy używać stylów
  - `<center>` - należy używać stylów i znacznika `<div>`
  - `<applet>` - należy używać znacznika `<object>`

24

## Różnice XHTML w stosunku do HTML4

- Zlikwidowano atrybuty (zastępowane teraz stylami):
  - *align* (można używać w <col>, <colgroup>, <tbody>, <td>, <tfoot>, <th>, <thead>, <tr>)
  - *alink*, *background*, *bgcolor*
  - *border* (można używać w <table>)
  - *color*
  - *height* (można używać w <iframe>, <img>, <object> ale się nie zaleca)
  - *size* (można używać w <input>, <select>)
  - *target* (zlikwidowano w <a>, <area> i <link>)
  - *text*, *type*, *value*, *version*, *vlink*
  - *width* (można używać w <colgroup>, <col>, <table>, <img>, <object>, <iframe> ale się nie zaleca)

25

## Specyfikacje W3C

- Specyfikacja **XML**: <http://www.w3.org/TR/REC-xml/>
- Specyfikacja **XHTML 1.0**: <http://www.w3.org/TR/xhtml1/>
- Specyfikacja **XHTML 1.1**: <http://www.w3.org/TR/xhtml11/>
- Specyfikacja **HTML5**: <http://www.w3.org/TR/html5/>
- Specyfikacja **CSS**: <http://www.w3.org/TR/CSS/>

26

## Definicje typu dokumentu DTD wersje dla XHTML 1.0

- **Transitional** - najczęściej stosowana wersja DTD; forma przejściowa pomiędzy HTML a XHTML, zezwala na korzystanie ze znaczników i atrybutów w statusie deprecated; nie można stosować ramek  

```
<!DOCTYPE html PUBLIC "-//W3C/DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```
- **Strict** - trudna do zrealizowania wersja DTD (ale należy do niej dążyć); zabrania stosowania tagów i atrybutów w statusie deprecated; nie można korzystać z ramek  

```
<!DOCTYPE html PUBLIC "-//W3C/DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```
- **Frameset** - wersja DTD, która dopuszcza wszystkie elementy (atrybuty i znaczniki deprecated i ramki);  

```
<!DOCTYPE html PUBLIC "-//W3C/DTD XHTML 1.0 Frameset//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
```

27

## Wersja DTD XHTML 1.1

- ```
<!DOCTYPE html
PUBLIC "-//W3C/DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```
- Wersja DTD XHTML 1.1 jest podobna do wersji XHTML 1.0 Strict. Nie wszystkie jednak dokumenty XHTML 1.0 Strict mogą być dokumentami XHTML 1.1.

28

## Struktura dokumentu XHTML

```
<?xml version="1.0" encoding="iso-8859-2"?>
<!DOCTYPE html PUBLIC "-//W3C/DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pl">
<head>
<meta ... /> <!-- znaczniki meta -->
<title>Tytuł strony </title>
<link rel="stylesheet" href="styl.css" type="text/css" />
</head>
<body>
<div>
<p>akapit1 <br /> akapit1 akapit1 ... </p>
<p>akapit2 akapit2 akapit2 akapit2 ... </p>
</div>
<div> ...
</div>
</body>
</html>
```

29

## Struktura dokumentu HTML5

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<meta name="viewport"
content="width=device-width, initial-scale=1.0" />
...
<title> Tytuł strony </title>
</head>
<body>
<!-- Tresc strony – nowe znaczniki -->
<header>...</header>
<nav> ... </nav>
<article> <sections>...</sections>
</article>
<aside>...</aside>
<footer>...</footer>
</body>
</html>
```

30

## Atrybut viewport znacznika

### <meta >

```
<meta name="viewport"
      content="width=device-width, initial-scale=1.0"
/>
```

- Mechanizm jest odpowiednikiem operacji zbliżania za pomocą gestu przybliżenia (ang. pinch-to-zoom), stosowanego w przypadku stron, które nie są przystosowane do urządzeń mobilnych.
- Aplikacja WWW wykorzystuje mechanizm **viewport** do wyśrodkowania zawartości na ekranie.

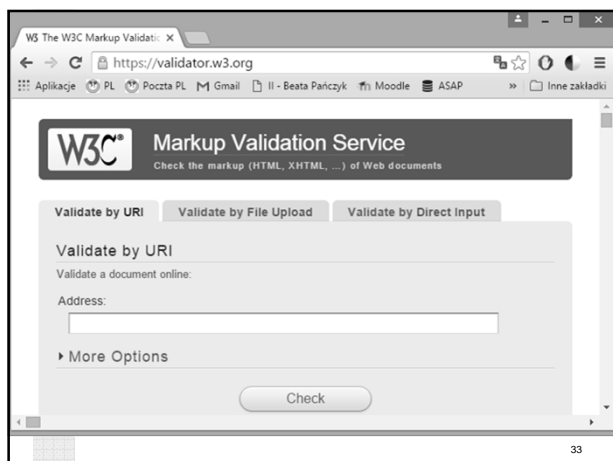
31

## Walidatory

- Walidatory służą do sprawdzania błędów w strukturze dokumentu HTML/XHTML
- Walidatory HTML - W3C Validator, znajduje się pod adresem: <http://validator.w3.org/>
- Walidator CSS sprawdza poprawność stylów pod adresem: <http://jigsaw.w3.org/css-validator/>



32



33

## Przykład HTML5 – przykład1.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>JCSI - strona główna</title>
    <meta charset="UTF-8">
    <meta name="viewport"
          content="width=device-width, initial-scale=1.0">
  </head>
  <body>
    <div> <!-- element blokowy-->
    <p>HTML - Lekcja 1
    <br />1. Podstawowe znaczniki
    <br />2. Struktura dokumentu
    </p>
  </div>
</body>
</html>
```

HTML - Lekcja 1  
1. Podstawowe znaczniki  
2. Struktura dokumentu

34

## Znaki specjalne

Znaki nie należące do standardowego zestawu znaków alfanumerycznych (np. ©), niektóre znaki specjalne (np. &) czy też twardą spację należy określać w dokumentach HTML za pomocą kodów. Kody znaków mogą być definiowane poprzez:

- **nazwę (encję):** &nazwa;
- **wartość liczbowa:** &#nnn;

35

## Tabela wybranych kodów znaków

Wartość	Encja	Symbol
&#034;	&quot;	" (cudzysłów prosty)
&#038;	&amp;	&
&#160;	&nbsp;	twarda spacja
&#167;	&sect;	§ (znak paragrafu)
&#169;	&copy;	© (symbol <i>copyright</i> )
&#174;	&reg;	® (zastrzeżony znak towarowy)
&#178;&#179; &#185;	&sup2; &sup3; &sup1;	<sup>2 3 1</sup> (2, 3, 1 w indeksie górnym)
&#188;&#189; &#190;	&frac14;&frac12; &frac34;	<sup>1 2 3</sup> / <sub>4</sub> ułamki

36

## Akapity, nagłówki, komentarze

- `<p>` - akapit
- `<h1>`, `<h2>`, ... - nagłówki (7 poziomów)
- `<br>` - łamanie wiersza w HTML4
- `<br />` - łamanie wiersza w stylu XHTML
- Komentarz:  
`<!-- komentarz w html/xhtml -->`

37

## Lista numerowana `<ol>`, wypunktowana `<ul>` i lista definicji `<dl>`

- `<ol>`

```
<li>element numerowany 1 </li>
<li>element numerowany 2 </li>
</ol>
```
- `<ul>`

```
<li>element punktowany 1 </li>
<li>element punktowany 2 </li>
</ul>
```
- `<dl>`

```
<dt>termin 1 </dt>
<dd>objaśnienie terminu 1</dd>
<dt>termin 2 </dt>
<dd>objaśnienie terminu 2</dd>
</dl>
```

38

## Przykład - listy

- ♦ Podstawowe znaczniki
- ♦ Struktura dokumentu

1. Podstawowe znaczniki
2. Struktura dokumentu

### HTML

Hyper Text Markup  
Language ? język do  
tworzenia hipertekstu

### HTTP

Hyper Text Transfer  
Protocol ? protokół  
wykorzystywany w WWW

```
<ul> <li>Podstawowe znaczniki</li>
<li>Struktura dokumentu</li>
</ul>
<ol> <li>Podstawowe znaczniki</li>
<li>Struktura dokumentu</li>
</ol>
<dl>
<dt>HTML</dt>
<dd>Hyper Text Markup Language
    - język do tworzenia hipertekstu
</dd>
<dt>HTTP</dt>
<dd>Hyper Text Transfer Protocol
    - protokół wykorzystywany w WWW
</dd>
</dl>
```

39

## Listy zagnieżdżone

- ♦ Podstawowe znaczniki
  1. P
  2. HR
  3. BR
- ♦ Struktura dokumentu
  - ◊ HTML
  - ◊ HEAD
  - ◊ BODY

```
<ul> <!--lista punktowana zewnętrzna -->
<li>Podstawowe znaczniki
  <ol> <!--lista numerowana wewnętrzna -->
    <li>P</li>
    <li>HR</li>
    <li>BR</li>
  </ol>
</li>
<li>Struktura dokumentu
  <ul> <!--lista punktowana wewnętrzna -->
    <li>HTML</li>
    <li>HEAD</li>
    <li>BODY</li>
  </ul>
</li>
</ul>
```

40

## Dodawanie grafiki

``

- **src** – atrybut zawierający URL do pliku graficznego
- **alt** – atrybut zawierający opis tekstowy do wyświetlenia w miejscu elementów graficznych użytkownikom, którzy nie mogą oglądać grafiki;

41

## Przykład – wstawianie obrazka

### HTML - Lekcja 1

1. Podstawowe znaczniki
2. Struktura dokumentu



```
<div>
<h2>
  HTML - Lekcja 1
</h2>
<ol>
  <li>Podstawowe znaczniki</li>
  <li>Struktura dokumentu</li>
</ol>
<p>
  
</p>
</div>
```

42

## Odwołania hipertekstowe (linki)

- kotwica (anchor) `<a href="plik lub url">tekst odnośnika</a>`  
np.  
`<a href="rys.htm">zobacz rys.1</a>`  
`<a href="http://www.google.pl">wyszukaj</a>`
- odwołanie hipertekstowe do pliku na serwerze lokalnym lub URL (łącznie z protokołem http) do serwera odległego
- odwołanie do miejsca (zaznaczonego wcześniej) wewnątrz aktualnego dokumentu  
`<a name="r1">rozdział 1</a>`  
`<a href="#r1">przeczytaj roz. 1</a>`

43

## Przykład – wstawianie linków

### HTML - Lekcja 1

1. [Podstawowe znaczniki](#)
2. [Struktura dokumentu](#)



```
<h2>HTML - Lekcja 1</h2>
<ol>
<li><a href=
"http://webmaster.helion.pl/kurshtml">
Podstawowe znaczniki</a></li>
<li><a href="w1.html">
Struktura dokumentu</a></li>
</ol>
<p>
<a href="http://java.sun.com">

</p>
```

44

## Definiowanie tabeli

- `<table> ... </table>` znacznik tabeli
- `<thead>`, `<tfoot>`, `<tbody>` - opcjonalne elementy tabeli, które mogą być osadzone bezpośrednio wewnątrz tabeli
- `<caption>...</caption>` nagłówek tabeli
- `<tr> ... </tr>` wiersz
- `<td> ... </td>` pojedyncza komórka
- `<th> ... </th>` komórka w postaci nagłówka
- Przykładowa tabela 2 x 2:  
`<table><tbody>`  
    `<tr>     <td> ... </td>   <td> ... </td>   </tr>`  
    `<tr>     <td> ... </td>   <td> ... </td>   </tr>`  
`</tbody></table>`

45

## Przykład – prosta tabela

```
<h3>Tworzenie tabel</h3>
<table>
<tbody>
<tr><td>Ania</td>
<td>Ola</td>
<td>Ela</td>
</tr>
<tr>
<td>Adam</td>
<td>Piotr</td>
<td>Stanisław</td>
</tr>
</tbody>
</table>
```

### Tworzenie tabel

Ania	Ola	Ela
Adam	Piotr	Stanisław

46

## Scalanie komórek td/th

- **colspan** - połączenie komórek (leżących obok siebie w kolumnie)
- Stosowane dla elementu **td** lub **th**:  
np.  
`<td colspan="3">`
- **rowspan** - połączenie komórek (leżących obok siebie w wierszu) np.  
`<td rowspan="2">`

47

## Przykład – scalanie komórek

```
<table border="5">
<caption> Imiona</caption>
<tbody>
<tr>
<td rowspan="2">Ania</td>
<td>Ola</td>
<td>Ela</td>
</tr>
<tr>
<td colspan="2">Adam</td>
<td>Piotr</td>
<td>Stanisław</td>
</tr>
</tbody>
</table>
```

### Tworzenie tabel

Imiona			
Ania	Ola	Ela	
	Adam		Piotr Stanisław

48



## Przykład – scalanie komórek

```
<table border="5">
  <tbody>
    <caption> Imiona</caption>
    <tr>
      <td rowspan="2">Ania</td>
      <td>Ola</td>
      <td>Ela</td>
    </tr>
    <tr>
      <td colspan="2">Adam</td>
    </tr>
  </tbody>
</table>
```

### Tworzenie tabel

Imiona		
Ania	Ola	Ela
	Adam	

49

## Przykład – rozbudowana struktura tabeli

```
<table border="1">
  <caption>Wyniki sesji</caption>
  <thead>
    <tr> <th>Przedmiot</th> <th>Ocena końcowa</th> </tr>
  </thead>
  <tfoot>
    <tr> <td>Średnia</td>
      <td>4.0</td>
    </tr>
  </tfoot>
  <tbody>
    <tr> <td>Matematyka</td> <td>4.0</td> </tr>
    <tr> <td>Fizyka</td> <td>3.0</td> </tr>
    <tr> <td>Informatyka</td> <td>5.0</td> </tr>
  </tbody>
</table>
```

Wyniki sesji	
Przedmiot	Ocena końcowa
Matematyka	4.0
Fizyka	3.0
Informatyka	5.0
Średnia	4.0

50

## 3. Formularze

## Wstawianie formularza

**<form action="plik lub url" method="metoda">**

- action** - określa sposób opracowywania danych pochodzących z formularza (dokąd odesłać dokument)
- method** - określa metodę przesyłania danych do serwera:  
method="POST" - w postaci pliku,  
method="GET" - po dołączeniu do URL-a

**</form>**

**UWAGA!** Wszystkie pola formularza powinny posiadać parametr **name**, jednoznacznie identyfikujący dane pole

52

## Dokument z formularzem

```
<html>
  <head>
    <title> Wstęp do tworzenia formularzy </title> ...
  </head>
  <body>
    <div>
      <form method="post "
        action="mailto:xxx@xxx.xxx"
        enctype="text/plain" >
        <!-- w tym miejscu będą wstawiane pola
        formularza np. pola input, lista select itp. - ->
      </form>
    </div>
  </body>
</html>
```

53

## Pole formularza <input type=...>

**Parametr type:**

- <input type="text"/>** pole tekstu
- <input type="password"/>** pole tekstowe typu hasło
- <input type="radio"/>** przycisk typu radio
- <input type="checkbox"/>** pole wyboru (check box)
- <input type="submit"/>** przycisk potwierdzający formularz i wysyłający go do serwera
- <input type="reset"/>** przycisk do usuwania zawartości formularza

Poza typem pola możliwe są inne parametry pola **input**:

- name, value**
- checked**
- size, maxlength**

54

## Pole formularza <select>

```
<select
  name="..."
  size="n"
  multiple="multiple">
  <option value="w1" >
    wybór1</option>
  <option value="w2">
    wybór2</option>
  <option value="w3 "
    selected="selected">
    wybór3
  </option>
</select>
```

- lista rozwijana (lub przewijana jeśli określono parametr size)
- name** - nazwa listy
- size** - ilość elementów widocznych na liście przewijanej
- multiple** - możliwość wybrania kilku elementów jednocześnie
- <option>** - indywidualne pole wyboru
- selected** - pole zaznaczone jako wartość domyślna

55

## Pole formularza <textarea>

```
<textarea
  name="t1"
  rows="2"
  cols="3" >
  Tu można podać
  tekst domyślny
</textarea>
```

56

## Pole input typu text i password

```
<input type="text"/>
<input type="text" name="adres"/>
<input type="text" name="adres" value="Wileńska 5"/>
<input type="text" name="adres" value="Wileńska"
  size= "10"/>
<input type="text" name="adres" size= "30"
  maxlength= "10" />
<input type="password" name="hasło"> -
parametry size, value i maxlength jak w polu text)
```

57

## Pole input typu radio

```
<input type="radio" name="ulubiona zupa"/>
<input type="radio" name="ulubiona zupa" /> <br />
<input type="radio" name="ulubiona zupa" /> <br />
<input type="radio" name="ulubiona zupa" /> <br />
<input type="radio" name="ulubiona zupa" value="zur" />
  żurek<br />
<input type="radio" name="ulubiona zupa" value="ros" />
  rosół<br />
<input type="radio" name="ulubiona zupa" value="bar" />
  barszcz<br />
```

- żurek
- rosół
- barszcz

58

## Przykład – pole radio

```
<p>
<form>Jaka jest twoja ulubiona
zupa?<br/>
<input type="radio"
  name="ulubiona zupa"
  value="zur" /> żurek <br/>
<input type="radio"
  name="ulubiona zupa"
  value="ros" /> rosół<br/>
<input type="radio"
  name="ulubiona zupa"
  value="bar"
  checked="checked"/> barszcz
</form>
</p>
```

59





## Przykład – pole checkbox

- Każde pole **checkbox** musi mieć różne nazwy (**name**):

```
<p>
<form> Jaka jest twoja
ulubiona zupa?<br />
<input type="checkbox"
  name="zur" value="yes"
  checked="checked" /> żurek <br/>
<input type="checkbox"
  name="ros" value="yes"
  checked="checked" /> rosół <br/>
<input type="checkbox"
  name="bar" value="yes"/>
  barszczyk
</form>
</p>
```



60

## Pole input typu submit i reset

- Standardowo:  
`<input type="submit"/>`   
`<input type="reset"/>` 
- Zmiana tekstu pojawiającego się na przycisku:  
`<input type="submit" value="Wyślij" />`   
`<input type="reset" value="Wyczyść" />` 

61

## Pole <select>

- lista select i opcje wyboru:  
`<select name="zupa" >`  
`<option> żurek </option>`   
`<option> rosół </option>`  
`<option> barszczyk </option>`  
`<option> pomidorowa </option>`  
`</select>`
- wartości dla każdej opcji:  
`<select name="zupa" >`   
`<option value="zur" > żurek </option>`  
`<option value="rosol" > rosół </option>`  
`<option value="barszcz" selected="selected" >`  
`barszczyk </option>`  
`<option value="pomidor" > pomidorowa </option>`  
`</select>`

62

## Przykład – pole select

Jaka jest Twoja ulubiona zupa?



```
<p><form>
Jaka jest twoja ulubiona zupa?<br>
<select name="zupa" size="3">
  <option value="zur" >
    żurek </option>
  <option value="rosol" >
    rosół </option>
  <option value="barszcz"
    selected="selected">
    barszczyk </option>
  <option value="pomidor">
    pomidorowa </option>
  <option value="ogorek">
    ogórkowa </option>
  <option value="grzyb"> grzybowa
  </option>
</select>
</form></p>
```

63

## Pole <button>

- `<button>treść przycisku</button>`
- podobny do klasycznego input, ale pozwala wprowadzić do przycisku znacznie więcej elementów (np. wstawiać grafikę):  
`<button>Kliknij tutaj<br />`  
`</button>`
- Atrybuty przycisku:  
**disabled** - wyłączenie przycisku (nie działa kliknięcie)  
**name** - unikatowa nazwa  
**type** - typ przycisku, submit, reset  
**value** - przypisanie przyciskowi określonej wartości
- Funkcjonalność przycisku można rozbudowywać o skrypty – obsługa zdarzeń w JavaScript

64

## Przykład – pole button

```
<p><form>
A teraz:<br>
<button name="p1">
  kliknij tutaj
</button>
<br /> lub <br />
<button name="p2">
  kliknij tutaj<br />
  
</button>
<br />
</form></p>
```



65

## Przykład formularza

### Formularz do składania zamówień

Nazwisko, imię:   
 Ulica:   
 Kod, Miejscowość:   
 Adres e-mail:

#### Proszę zaznaczyć zamawiane produkty:

☐ Turbo Pascal ☐ C++ ☐ Java

#### Sposób zapłaty:

☐ EuroCard ☐ Visa ☐ Przelew bankowy

66

## Przykład - kod formularza

```
<h3>Formularz do składania zamówień</h3>
<form method="post" action="mailto:xxx@xxx.xx" enctype="text/plain">
  <p>Nazwisko, imię: <input name="nazwisko" size="30" /><br />
  Ulica: <input name="ulica" size="30" /><br />
  Kod, miejscowość: <input name="miejsc" size="30" /><br />
  Adres e-mail: <input name="email" size="30" /><br />
<h4>Proszę zaznaczyć zamawiane produkty:</h4>
  <p><input name="tp" type="checkbox" /> turbo pascal
  <input name="c" type="checkbox" /> c++
  <input name="java" type="checkbox" /> java </p>
<h4>Sposób zapłaty:</h4>
  <p><input name="zapłata" type="radio" value="euro" /> eurocard
  <input name="zapłata" type="radio" value="visa" /> visa
  <input name="zapłata" type="radio" value="przelew" /> przelew
  bankowy<br />
  <input type="submit" value="Wyślij zamówienie" />
  <input type="reset" value="Anuluj zamówienie" /></p>
</form>
```

67

## Przykład – pola formularza w tabeli

```
<table border="1">
<tbody>
<tr>
<td>Nazwisko, imię </td>
<td><input name="nazwisko" size="30" /> </td>
</tr>
<tr>
<td>Ulica </td>
<td><input name="ulica" size="30" /> </td>
</tr>
<tr>
<td>Kod, miejscowość </td>
<td><input name="miejsc" size="30" /> </td>
</tr>
<tr>
<td>Adres e-mail </td>
<td><input name="email" size="30" /> </td>
</tr>
</tbody>
</table>
```

68

## Formularz z tabelą

### Formularz do składania zamówień

Nazwisko, imię	<input type="text"/>
Ulica	<input type="text"/>
Kod, Miejscowość	<input type="text"/>
Adres e-mail	<input type="text"/>

Proszę zaznaczyć zamawiane produkty:

☐ Turbo Pascal ☐ C++ ☐ Java

Sposób zapłaty:

☐ EuroCard ☐ Visa ☐ Przelew bankowy

69

## Dodatkowe znaczniki dla formularzy

- **fieldset** - obramowanie pól formularza
- **legend** - tytuł grupy elementów formularza
- **label** - etykieta pola formularza
- **optgroup** - dla select - listę opcji można dodatkowo grupować

70

## Przykład

```
<form action="..." method="get">
<fieldset> <!-- Pierwsza grupa pól formularza -->
<legend>Pola tekstowe</legend> <!-- Opis pierwszej grupy pól -->
<label for="imie">Imię: </label> <!-- Etykieta pola tekstowego -->
<input type="text" id="imie"
  name="imie"/><br />
<label for="nazwisko">Nazwisko: </label>
<input type="text" id="nazwisko"
  name="nazwisko"/><br />
Pole bez label: <input type="text"/><br />
</fieldset> <!-- koniec 1 grupy pól -->
```

Pola tekstowe	
Imię:	<input type="text"/>
Nazwisko:	<input type="text"/>
Pole bez label:	<input type="text"/>

71

## Przykład – c.d.

```
<fieldset> <!-- Druga grupa pól formularza -->
<legend>Pola wyboru</legend> <!-- Opis drugiej grupy pól -->
<select name="wybor">
<optgroup label="Systemy operacyjne"> <!-- Opcja 1 -->
  <!-- Podopcje dla opcji 1 -->
  <option value="wind">System Windows</option>
  <option value="unix">System Unix</option>
</optgroup> <!-- Koniec opcji 1 -->
<optgroup label="Procesor"> <!-- Opcja 2 -->
  <!-- Podopcje dla opcji 2 -->
  <option value="i">Procesor Intel</option>
  <option value="a">Procesor Amd</option>
</optgroup> <!-- Koniec opcji 2 -->
</select>
</fieldset> <!-- koniec 2 grupy pól -->
<p> <input type="submit" value="Wyślij dane" /> </p>
</form>
```

72

## Przykład – wynik działania

The screenshot displays a web form with two main sections: 'Pola tekstowe' (Text fields) and 'Pola wyboru' (Selection fields). The 'Pola tekstowe' section contains three input fields labeled 'Imie:', 'Nazwisko:', and 'Pole bez label:'. The 'Pola wyboru' section contains a dropdown menu currently showing 'System Windows'. A context menu is open over the dropdown, listing 'Systemy operacyjne' (Operating systems) and 'Procesor' (Processor) as categories. Under 'Systemy operacyjne', 'System Windows' and 'System Unix' are listed. Under 'Procesor', 'Procesor Intel' and 'Procesor Amd' are listed. The 'System Windows' option is currently selected. At the bottom left of the form, the text 'Zakończono' (Finished) is visible. A small number '73' is located at the bottom right of the form area.

Pola tekstowe

Imie:

Nazwisko:

Pole bez label:

Pola wyboru

System Windows

**Systemy operacyjne**

System Windows

System Unix

**Procesor**

Procesor Intel

Procesor Amd

Zakończono

73