

Wprowadzenie

Język HTML jest elementem niezbędnym do budowy własnej strony. Natomiast CSS to zestaw funkcji, za pomocą których możesz kontrolować wygląd poszczególnych elementów na stronie. W skrócie można powiedzieć, że kaskadowe arkusze stylów to pewnego rodzaju uzupełnienie, bez którego można się obejść, wykorzystując rozwiązania dostępne np. w czystym języku HTML. Zapamiętaj jednak, że *nowoczesna strona WWW* to — w dużym uproszczeniu — połączenie języka HTML i CSS.

Konstrukcja języków HTML i CSS jest standaryzowana i określona przez odpowiednią specyfikację. Jest to konieczne w celu zachowania porządku oraz uniezależnienia języków od platform sprzętowych i rozwiązań programowych poszczególnych twórców przeglądarek. Standaryzacją zajmuje się W3C — <http://www.w3.org/>.

Adresy pełnych wersji specyfikacji HTML4, CSS2 oraz projektów HTML5 i CSS3.

- HTML4 — <http://www.w3.org/TR/html4/>,
- HTML5 — <http://www.w3.org/TR/html5/>,
- CSS2 — <http://www.w3.org/TR/CSS21/>,
- CSS3 — <http://www.w3.org/TR/CSS/>.

Niezależnie od tego, z jakiej specyfikacji będziemy korzystać pliki zawierające odpowiednie znaczniki HTML i CSS są dokumentami tekstowymi, a co za tym idzie, mogą być tworzone i przeglądane za pomocą **dowolnego edytora tekstu**, np. systemowego Notatnika. Na ćwiczeniach będziemy używali programu [Bracket](#) lub [Notepad++](#) wykorzystując funkcję kolorowania składni języka, co ułatwi redagowanie dokumentów).

W zależności od zawartości rozszerzenie pliku może się różnić, jednak nadal mamy do czynienia z dokumentem tekstowym. Oto kilka przykładów wraz z opisem ich zawartości:

- **htm** — plik tekstowy zawierający kod HTML, CSS, czasami też kod JavaScript;
- **html** — plik tekstowy zawierający kod HTML, CSS, czasami też kod JavaScript;
- **shtml** — plik tekstowy zawierający kod HTML, CSS oraz instrukcje SSI — pliki takie są przetwarzane przez serwer przed wysłaniem do przeglądarki;
- **php** — pliki tekstowe zawierające kod HTML, CSS połączony ze skryptami PHP;
- **cgi** — pliki tekstowe zawierające skrypty napisane w Perlu bądź Shellu — często połączone z kodem HTML;
- **pl** — pliki tekstowe zawierające skrypty napisane w Perlu — często połączone z kodem HTML;
- **js** — pliki tekstowe zawierające skrypty napisane w języku JavaScript;
- **css** — pliki tekstowe zawierające zewnętrzne arkusze stylów.



Znaczniki i atrybuty

Każdy dokument HTML składa się ze znaczników, które są zapisywane w nawiasach kątowych i wyglądają mniej więcej tak: <znacznik>. Znaczniki pojawiają się prawie zawsze parami, tzn. występuje znacznik otwierający oraz znacznik zamykający:

<znacznik otwierający></znacznik zamykający>

W praktyce może to wyglądać mniej więcej tak:

<p>Przykład akapitu.</p>

Jak widać, znacznik zamykający po nawiasie kątowym zawiera jeszcze /. Każdy znacznik otwierający może mieć szereg dodatkowych elementów — atrybutów — określających jego właściwości, np. formatowanie wyglądu za pomocą CSS. Właściwości atrybutów obowiązkowo powinny zostać zapisane w cudzysłowie " ".

<p class="cytat">Przykład akapitu.</p>

Znaczniki zamykające nigdy nie zawierają dodatkowych parametrów. Większość znaczników występuje w parach, są jednak wyjątki. Typowe przykłady, z którymi się spotkamy, to:
, <hr>, , <meta>, <link>, <base>.

Najważniejsze różnice i zasady

Bezpośrednim poprzednikiem języka HTML5 jest HTML4, a dokładnie jego odsłona oznaczona numerem 4.01. Język ten był rozwijany do pewnego czasu. Następnie twórcy wybrali inną drogę i zaczęli rozwijać XHTML, który doczekał się dwóch głównych specyfikacji. Po wstępnym zachwycie XHTML zdecydowano o zaniechaniu jego dalszego rozwijania i powrocie do prac nad kolejną wersją czystego HTML-a. Nowa odsłona języka HTML5 wprowadza szereg zmian i różnic w stosunku do poprzednich wersji HTML4. Najważniejsze zmiany wypunktowałem poniżej.

- **Kompatybilność** — język HTML5 jest zgodny z poprzednią wersją specyfikacji oznaczoną numerem 4. W praktyce oznacza to, że strona WWW napisana na bazie nowej wersji HTML będzie działać w przeglądarkach, które nie potrafią obsługiwać tego języka. Oczywiście nowe znaczniki i elementy zostaną zignorowane, ale główny szkielet witryny oraz znaczniki znane z poprzednich specyfikacji zadziałają poprawnie. Takie podejście przeważało o popularności wstępnej wersji języka, która z powodzeniem jest używana w wielu projektach.
- **Brak przestarzałych znaczników** — propozycja nowej wersji specyfikacji języka HTML5 usuwa wiele zbędnych i przestarzałych znaczników. W dużym uproszczeniu możemy przyjąć, że wszelkie znaczniki odpowiedzialne za formatowanie wyglądu witryny oraz mające negatywny wpływ na jej funkcjonalność zostały usunięte, ponieważ ich funkcję przejęły kaskadowe arkusze stylów. Usunięte znaczniki to m.in.: `<basefont>`, `<big>`, `<center>`, ``, `<s>`, `<strike>`, `<tt>`, `<u>`, `<frame>`, `<frameset>`, `<no frames>`.
- **Uproszczenie zapisu** — nowa odsłona specyfikacji wprowadza liczne uproszczenia w zapisie znaczników oraz ich atrybutów.

Na przykład w tej chwili określenie strony kodowej wygląda następująco:

```
<meta charset="UTF-8" />.
```

Dla porównania ten sam zapis w poprzedniej wersji specyfikacji miał postać

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />.
```

Uproszczeń jest oczywiście więcej i szczegółowe informacje na ich temat znajdziesz w toku dalszych ćwiczeń.

- **Nowe znaczniki i atrybuty** — jak łatwo się domyślić, nowa specyfikacja języka HTML to również propozycje nowych znaczników, które mają uprościć i zwiększyć kontrolę nad szkieletem strony. Wśród nowych znaczników warto wspomnieć m.in. o: `<section>`, `<article>`, `<header>`, `<footer>`, `<nav>`, `<dialog>`, `<aside>`, `<figure>`. Znaczników i atrybutów jest więcej i szczegółowe informacje będziemy zdobywali na kolejnych zajęciach.
- **API** — rozwiązanie to pozwala na uproszczenie tworzenia rozbudowanych aplikacji internetowych (przykładem takiej aplikacji jest Google Docs).

Zasady tworzenia znaczników:

Zasada 1. Nigdy nie krzyżuj znaczników

Pierwszą i jedną z najważniejszych zasad obowiązujących podczas tworzenia stron WWW jest odpowiednie zagnieżdżanie znaczników. Innymi słowy, chodzi o to, aby unikać krzyżowania znaczników w kodzie strony. Przyjrzyj się poniższemu przykładowi, który prezentuje skrzyżowane znaczniki.

<p>Przykład bloku tekstu ze skrzyżowanymi znacznikami</p>

Teraz przykład, w którym znaczniki zostały prawidłowo zagnieżdżone.

<p>Przykład bloku tekstu ze skrzyżowanymi znacznikami</p>

Zasada 2. Zawsze zamykaj wszystkie znaczniki

Język XHTML wprowadził regułę, w myśl której wszystkie znaczniki powinny być bezwzględnie zamykane. Sprawa jest prosta i oczywista w przypadku większości znaczników, gdyż znaczniki zawsze występują w parze: znacznik otwierający i znacznik zamykający.

<p>Przykład bloku tekstu</p>

HTML5 odchodzi od zasady obowiązkowego zamykania znaczników, zachęcam jednak do stosowania tej zasady bowiem może ona pomóc uniknąć problemów z redagowaniem kodu, zwłaszcza osobom początkującym ;).

Zasada 3. Wpisując znaczniki i atrybuty, zawsze używaj małych liter

Piąta specyfikacja języka HTML porządkuje wiele spraw. Jedną z istotniejszych kwestii jest wprowadzenie obowiązku stosowania wyłącznie małych liter do zapisu nazw znaczników oraz ich atrybutów i wartości.

Poprawny zapis ma więc postać:

<p class="glowny">Przykładowy blok tekstu z własną klasą</p>

Natomiast forma widoczna poniżej uznawana jest za błędną i należy jej unikać.

<P CLASS="glowny">Przykładowy blok tekstu z własną klasą</P>

Zasada 4. Używaj cudzysłowów dla wszystkich atrybutów

Większość znaczników występujących w języku HTML5 może mieć przypisane dodatkowe atrybuty wraz z określonymi wartościami. Tworząc strony na bazie najnowszej odsłony języka HTML5, należy pamiętać o tym, że podczas określania atrybutów wraz z ich wartościami zawsze używamy cudzysłowów. Odpowiedni przykład znajduje się poniżej.

<table border="1">

Zasada 5. Uważaj ze znakami specjalnymi w skryptach

Tworząc kod zgodny z HTML5, zamiast znaków specjalnych, np. < czy &, używaj encji, np. < i & (wyjaśnimy to zagadnienie na konkretnym przykładzie w trakcie redagowania kodu).

Niezbędne narzędzia i programy

Twórca stron WWW w swoim warsztacie powinien mieć kilka narzędzi (programów), które pozwolą mu pracować w znacznie wygodniejszy i szybszy sposób. Tak się składa, że w naszym przypadku wystarczą następujące aplikacje:

- tekstowy edytor HTML i CSS,
- program do tworzenia i obróbki grafiki,
- przeglądarka stron WWW.

Edytory HTML, XHTML i CSS

Wspominałem o tym, że pliki zawierające kod strony WWW to dokumenty tekstowe, a co za tym idzie, do ich edycji możemy wykorzystać dowolny tekstowy edytor, np. systemowy Notatnik. Niestety takie rozwiązanie nie należy do zbyt wygodnych, gdyż każdorazowo wymaga ręcznego wklepywania wszystkich niezbędnych znaczników oraz pamiętania o ich domykaniu. W związku z tym na rynku stosunkowo szybko pojawiły się wyspecjalizowane edytory programistyczne wyposażone w szereg dodatkowych funkcji znacznie ułatwiających pracę webmastera.

Oto kilka przykładów:

- **1st Page 2006** — bardzo rozbudowany edytor stron WWW, który pozwala na pracę nad stroną w trybie tekstowym oraz graficznym. Warto zwrócić uwagę na to, że ten edytor ma mechanizm domykania znaczników, kolorowania składni oraz wygodne podpowiedzi. Duży plus należy się autorowi za dodanie dziesiątek kreatorów ułatwiających utworzenie bardziej skomplikowanego elementu na stronie, np. uzyskanie popularnego efektu rollover albo dodanie tabeli czy galerii ze zdjęciami. Prezentowane narzędzie wspiera pracę z kodem XHTML i HTML. Strona domowa edytora to <http://www.evrsoft.com/>. Minusem jest brak polskiej wersji językowej.
- **PSPad** — zaawansowany edytor programistyczny obsługujący wiele języków programowania, w tym HTML i CSS. Edytor ma wszystkie niezbędne udogodnienia ułatwiające pracę i został wyposażony przez autora w moduł sprawdzania pisowni — również polskiej. Zaletą jest polskojęzyczny interfejs i wygoda obsługi. Warto również wspomnieć o integracji z darmowym edytorem kaskadowych arkuszy stylów, **TopStyle Lite**, oraz sprawdzaniu poprawności kodu i obsłudze wielu dodatków, które można pobrać ze strony domowej — <http://www.pspad.com/en/>.



Przeglądarki stron WWW

Ostatnim ważnym elementem niezbędnym podczas tworzenia stron WWW jest przeglądarka. Oto kilka przykładów:

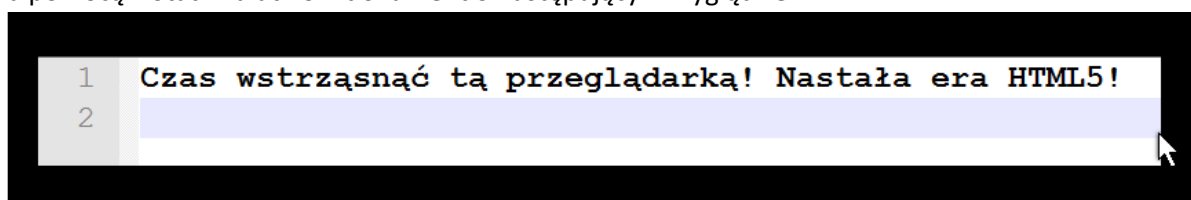
- Google Chrome — <http://www.google.com/chrome/?hl=pl>,
- Mozilla Firefox — <http://www.mozilla.org/pl/firefox/>,
- Opera — <http://www.opera.com/portal/choice/?language=pl>,
- Safari — <http://www.apple.com/pl/safari/>,
- Internet Explorer — <http://windows.microsoft.com/pl-PL/internet-explorer/products/ie/home>.

[Ranking popularności przeglądarek internetowych](#)

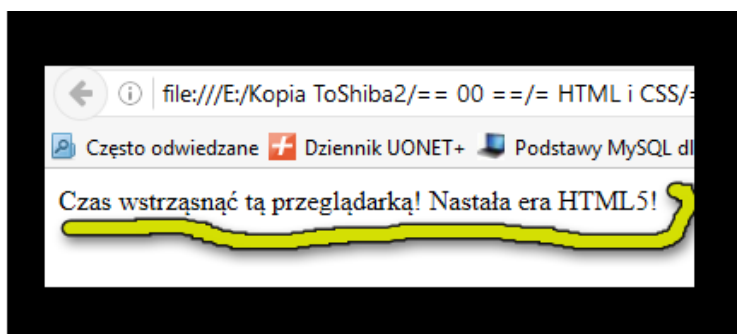
Ćwiczenie 01

Pracę z kodem HTML zaczniemy od maksymalnie uproszczonego dokument zawierającego jeden akapit tekstu. W prawdzie zasada 2 „Zawsze zamykaj wszystkie znaczniki” będzie nas obowiązywała na naszych ćwiczeniach pierwszy tekst maksymalnie uprościmy.

1. W swoim folderze ćwiczeniowym załóż folder o nazwie „01 Super szybko”.
2. Za pomocą Notatnika utwórz dokument o następującym wyglądzie:



3. Plik zapisz jako dokument HTML (komenda Język/H/HTML).
4. Efekt sprawdź w przeglądarce internetowej.

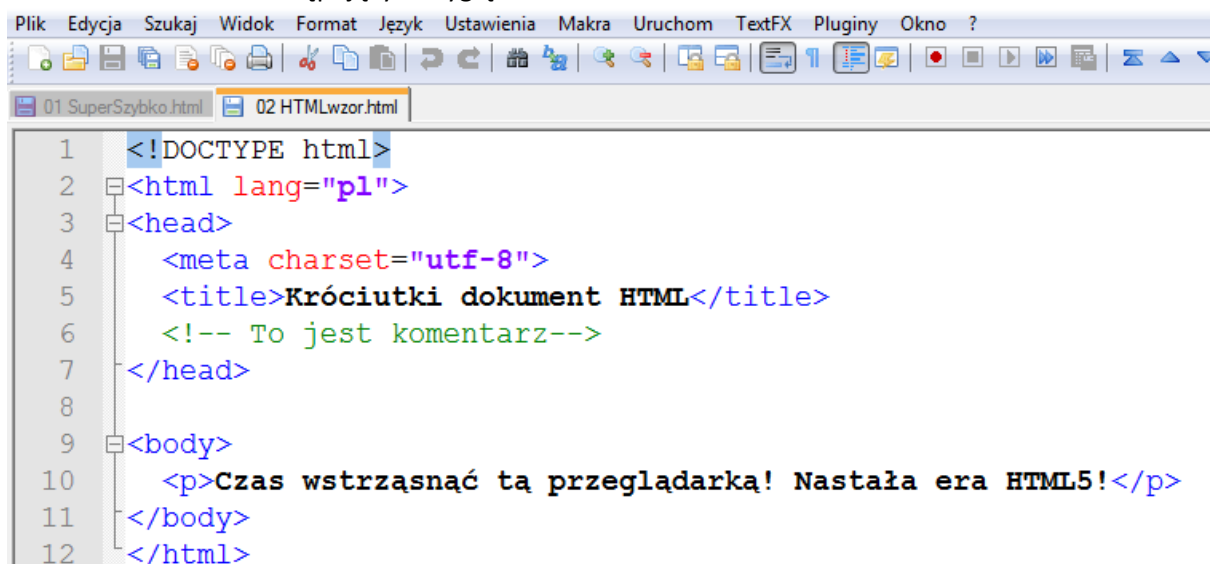


Rysunek 1 Jak widać pomimo braku znaczników tekst został wyświetlony poprawnie

Ćwiczenie 2

W poprzednim ćwiczeniu stworzyliśmy pierwszy tekst, ale znacznie uprościliśmy jego strukturę. W drugim ćwiczeniu utworzymy dokument, który będzie stanowił wzór do naszej dalszej pracy.

1. W swoim folderze ćwiczeniowym załóż folder o nazwie „02 HTML wzór”.
2. Utwórz dokument o następującym wyglądzie:



```
1 <!DOCTYPE html>
2 <html lang="pl">
3 <head>
4   <meta charset="utf-8">
5   <title>Króciutki dokument HTML</title>
6   <!-- To jest komentarz-->
7 </head>
8
9 <body>
10  <p>Czas wstrząsnąć tą przeglądarką! Nastąpiła era HTML5!</p>
11 </body>
12 </html>
```

3. Plik zapisz jako dokument HTML (komenda Język/H/HTML) z zachowaniem kodowania polskich znaków utf-8 (Format/Koduj utf-8).
4. Efekt sprawdź w przeglądarce internetowej.
5. Przeczytaj uwagę zamieszczoną poniżej. Korzystając z podanego hiperłącza do kodów znaków, umieść znak parasolki na końcu zdania „Nastąpiła era HTML5!”.

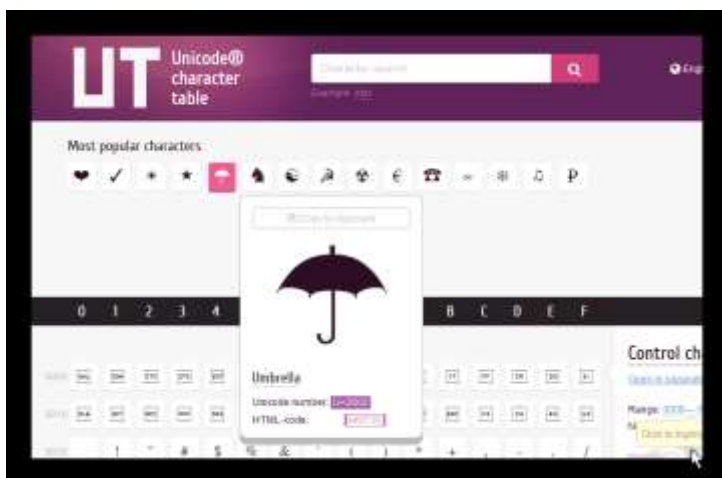
Uwaga do kodu

Pomiędzy znacznikami <head></head> znajduje się kilka elementów ważnych dla konstrukcji całej strony, stanowiących nagłówek dokumentu. W kolejnych ćwiczeniach będziemy wprowadzać w nagłówek kolejne elementy.

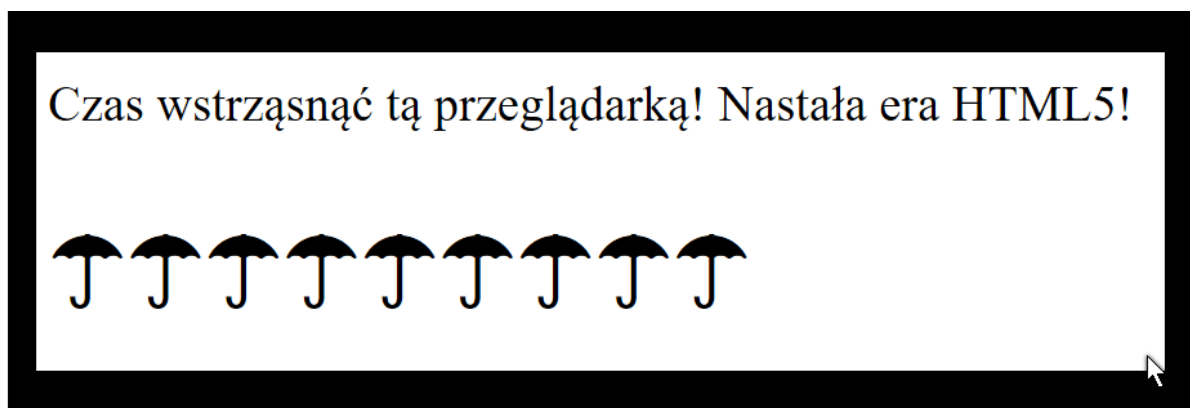
Język HTML jako ponadsystemowy nośnik informacji musi mieć możliwość obsługi różnych języków, a co za tym idzie, różnych znaków narodowych. W związku z tym zostały określone różne strony kodowe dla danych języków i części świata. Dla naszego kraju odpowiednim standardem jest strona kodowa **ISO 8859-2** lub **UTF-8**. Działa ona na wszystkich platformach systemowych: MS Windows, Unix, Linux, Mac OS, iPhone, iPad i wielu innych, dlatego jako świadomy projektant stron musisz stosować się do tej normy. Ktoś może jednak stwierdzić, że przecież wystarczy wpisać polskie znaki w Notatniku, a i tak wszystko zadziała. Oczywiście, nie można odrzucić takiego pomysłu, ale należy zwrócić uwagę na to, że tak zakodowane polskie znaki zadziałają tylko na platformie MS Windows, natomiast pozostałe systemy będą miały z taką stroną kodową problemy.

Pomimo że do dyspozycji mamy dwa sposoby zapisywania polskich znaków, standardem stało się praktyczne i wygodniejsze kodowanie **UTF-8**. Doskonałość tego rozwiązania polega na przypisaniu unikatowego numeru wszystkim znakom charakterystycznym dla różnych alfabetów, np. łacińskiego czy cyrylicy. Poza tym w Unikodzie uwzględniono różne symbole, np. ®, ©.

([To jest przykładowy link do znaków](#)) – wyszukaj kod parasolki, lub innego elementu i umieść go w dokumencie.



Rysunek 2 Przykład znaku parasolki



Rysunek 3 Efekt wielokrotnego wstawienia znaku



Rysunek 4 Przykład kodu z parasolkami

Ćwiczenie 3

W tym ćwiczeniu pokażemy przykład dołączania skryptów w języku JavaScript oraz komentarz dla IE.

1. Utwórz folder o nazwie „03 Dołączanie skryptu i komentarze dla IE”.

2. Utwórz dokument HTML zawierający następującą treść:

```
1 <!DOCTYPE html>
2 <html lang="pl">
3 <head>
4   <meta charset="utf-8">
5   <title>Przykład kodu JavaScript</title>
6 </head>
7 <body>
8   <h1>Umieszczamy na stronie kod JavaScript</h1>
9
10  <p>Jeśli wszystko zadziała prawidłowo, to po wyświetleniu treści
    przeglądarka uruchomi skrypt zapisany w pliku skrypt.js</p>
11
12  <!--Poniżej linia wywołująca skrypt-->
13  <script src="skrypt.js"></script>
14 </body>
15 </html>
16
```

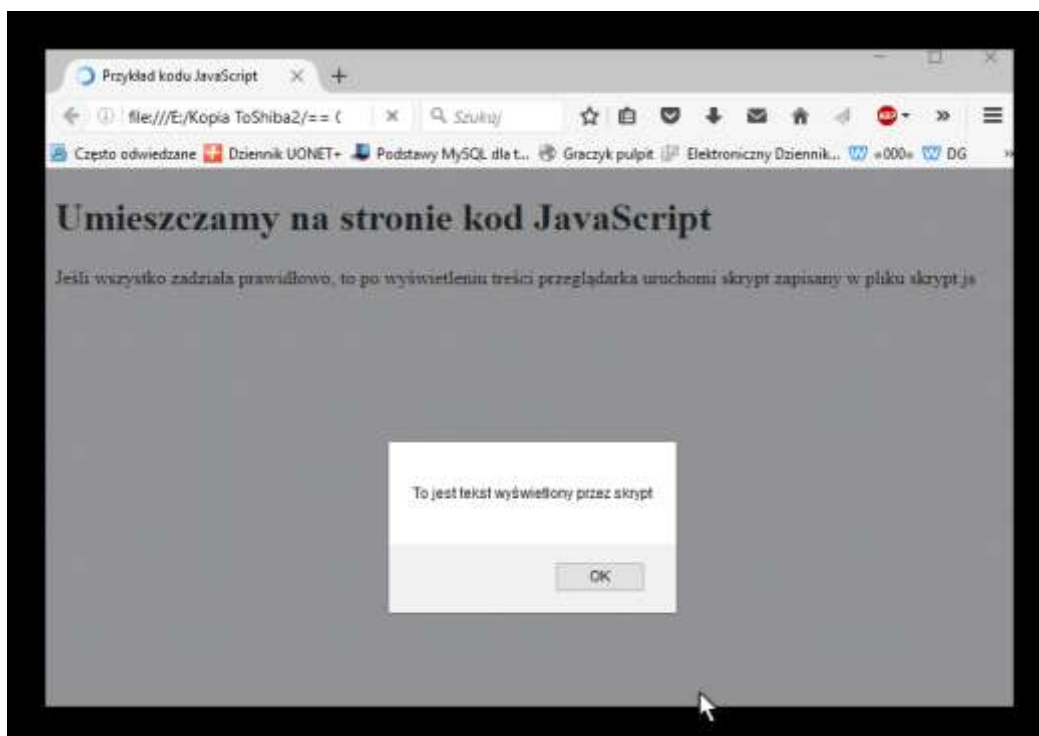
To jest nazwa pliku w której znajduje się kod skryptu.

3. Utwórz plik o nazwie skrypt.js zawierający pokazany na rysunku kod

```
alert("To jest tekst wyświetlony przez skrypt");
```

Uwaga: oba pliki html i js muszą znajdować się w tym samym folderze.

4. Zapisz plik i przetestuj jego działanie na dostępnych przeglądarkach.



Rysunek 5 Efekt działania naszego skryptu na stronie

Ćwiczenie 4

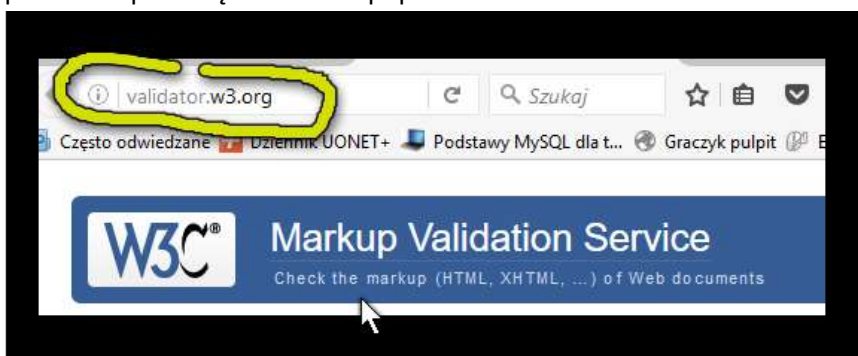
Walidację strony – czyli sprawdzenie poprawności kodu sprawdzimy metodą online pod adresem:

<http://validator.w3.org/>

1. Utwórz folder „04 Walidacja HTML5”.
2. Utwórz dokument html zawierający następującą treść:

```
1  <!DOCTYPE html>
2  <html lang="pl">
3  <head>
4      <meta charset="utf-8"/>
5      <title>Króciutki dokument HTML</title>
6
7  </head>
8
9  <body>
10     <p>Zaraz natkniesz się na błąd. <b>Strzeż </p>się!</b>
11
12 </body>
13 </html>
14
```

3. Sprawdź za pomocą walidatora poprawność kodu.



Rysunek 6 Strona z walidatorem

4. Sprawdź komentarz na stronie.
5. Wykonaj obraz błędu i zapisz go w folderze jako plik graficzny.

Ćwiczenie 5

W tym ćwiczeniu rozbudujemy nagłówek naszej przykładowej strony.

1. Utwórz folder „05 Rozbudowa nagłówka”.
2. Utwórz dokument HTML wg wzoru:

```
1  <!doctype html>
2  <html>
3  <head>
4      <meta charset="UTF-8" />
5      <title>Tytuł przykładowej strony</title>
6      <meta name="keywords" content="Twoje, słowa, kluczowe," />
7      <meta name="description" content="Wstaw opis swojej strony" />
8      <meta name="author" content="imię i nazwisko" />
9      <meta name="copyright" content="imię i nazwisko" />
10     <meta name="robots" content="index, follow" />
11     <meta name="generator" content="Notepad++" />
12     <meta http-equiv="refresh" content="x" />
13     <meta http-equiv="refresh" content="x; url=http://www.wp.pl" />
14     <link rel="stylesheet" href="arkusz.css" type="text/css" />
15
16 </head>
17 <body>
18 </body>
19 </html>
```

3. Zapisz plik i przetestuj jego działanie.

Uwaga do kodu

Warto wiedzieć, że część serwisów indeksujących korzysta z `<meta name="robots" />`, który określa, czy dana strona powinna być indeksowana oraz czy linki na niej zawarte również mają być dodane do bazy serwisu katalogującego. Przy zastosowaniu polecenia robots atrybut content zawiera następujące dyrektywy informujące o dopuszczonych operacjach dla naszej strony:

- 'index' — strona powinna być zaindeksowana;
- 'noindex' — strona nie powinna być zaindeksowana;
- 'follow' — linki z tej strony powinny być zaindeksowane;
- 'nofollow' — linki z tej strony nie powinny być zaindeksowane;
- 'all' równa się 'index, follow' — wartość domyślna;
- 'none' równa się 'noindex, nofollow'.

Ćwiczenie 6

W tym ćwiczeniu zajmiemy się sekcją `<body>`, czyli ciałem dokumentu.

1. Utwórz folder „06 Ciało dokumentu”.

2. Utwórz dokument HTML o następującej treści, wykorzystując plik z poprzedniego ćwiczenia:

```
1 <!doctype html>
2 <html>
3   <head>
4     <meta charset="UTF-8" />
5     <title>Tytuł przykładowej strony</title>
6   </head>
7   <body>
8     <header>To jest nagłówek</header>
9     <nav>Menu</nav>
10    <article>
11      <section>Seksja artykułu </section>
12    </article>
13    <aside>Treść dodatku</aside>
14    <footer>Stopka</footer>
15  </body>
16 </html>
```

3. Plik zapisz w folderze.

Uwaga do kodu:

Twórcy piątej specyfikacji języka HTML dokonali analizy struktury witryny i wyróżnili w niej kilka dodatkowych sekcji, a następnie przypisali do nich nowe znaczniki, za pomocą których będziemy mogli precyzyjniej kontrolować finalny wygląd strony. Nowe znaczniki nie wpływają bezpośrednio na wygląd zawartych w nich danych, ale pozwalają zgrupować szereg innych elementów i łatwiej nimi zarządzać. Lista nowych znaczników wraz z wyjaśnieniami znajduje się poniżej.

- `<header></header>` — pomiędzy znacznikami powinna być zamieszczona część strony, która ma charakter nagłówka i rozpoczyna Twoją stronę.
- `<nav></nav>` — pomiędzy znacznikami powinny być zamieszczone elementy odpowiedzialne za główne menu nawigacyjne witryny.
- `<article></article>` — pomiędzy znacznikami zamieszczamy zestaw elementów tworzących spójny artykuł (np. nagłówek i blok tekstu).
- `<section></section>` — pomiędzy znacznikami zamieszczamy wybraną zawartość części witryny tworzącą spójną sekcję, np. tytuł i wstęp artykułu widoczny na stronie głównej.
- `<aside></aside>` — pomiędzy znacznikami powinna być zamieszczona część strony, która jest elementem uzupełniającym główną strukturę strony.
- `<footer></footer>` — pomiędzy znacznikami powinna być zamieszczona część strony, która ma charakter stopki i zamyka Twoją stronę.



Rysunek 7 Schemat układu strony z wykorzystaniem znaczników

Ćwiczenie 7

W tym ćwiczeniu pokażemy działanie nagłówków.

1. Utwórz folder „07 nagłówki”.
2. Utwórz dokument HTML o następującej treści:

```
1  <!doctype html>
2  <html>
3  <head>
4    <meta charset="UTF-8" />
5    <title>Przykłady nagłówków</title>
6  </head>
7  <body>
8    <h1>Nagłówek stopnia pierwszego</h1>
9    <h2>Nagłówek stopnia drugiego</h2>
10   <h3>Nagłówek stopnia trzeciego</h3>
11   <h4>Nagłówek stopnia czwartego</h4>
12   <h5>Nagłówek stopnia piątego</h5>
13   <h6>Nagłówek stopnia szóstego</h6>
14 </body>
15 </html>
```

3. Plik zapisz w folderze i sprawdź jego działanie w przeglądarce.

Ćwiczenie 8

W tym ćwiczeniu tworzymy akapity. Tekst do naszego ćwiczenia należy przekopiować do dokumentu:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent tristique scelerisque sapien. Aliquam vel ipsum erat, vitae viverra massa. Maecenas accumsan, ipsum eget mollis iaculis, elit velit bibendum enim, at vulputate dui nunc quis eros. Pellentesque non iaculis lacus. Praesent neque eros, tempus nec ultricies id, euismod eget lectus. Pellentesque eu justo non turpis pellentesque tristique eu quis magna. Praesent libero massa, sodales id blandit quis, ornare eu ante. Quisque id justo a ipsum egestas pulvinar. Vestibulum vitae arcu ut est volutpat laoreet ultricies non ipsum. Vestibulum eget massa odio. Morbi sed neque non nisi tincidunt vehicula.

Ut sit amet arcu in velit pulvinar faucibus. Nam ac facilis lectus. Vivamus fringilla interdum dolor, quis gravidas e sodales eu. Aliquam porttitor hendrerit nisi, eu bibendum lorem suscipiteget. Nulla facilis. Nam id urna non neque molestie lobortis. Fusce gravidam nulla ac libero semper vel egestas elit imperdiet. Nullam consequat feugiat lectus non consequat. Aliquam imperdiet sapiens enim gravida et varius arcu molestie. Nullam mollis, metus a convallis rhoncus, lorem felis condimentum nunc, ac mollis lorem tellus id felis. Proin nec elementum justo. Morbi in dictum massa. Phasellus libero libero, lobortis vel faucibus convallis, posuere a turpis. Mauris in nisi et velit ullamcorper varius nec id felis. Pellentesque adipiscing libero ut sem convallis vel elementum felis pulvinar. Cras imperdiet ligula vitae massa pulvinar ornare.

1. Utwórz folder „08 Akapity”
2. Utwórz dokument HTML posługując się przykładowym tekstem:

```
1  <!doctype html>
2  <html>
3  <head>
4      <meta charset="UTF-8" />
5      <title>Tytuł przykładowej strony
6      </title>
7
8  </head>
9  <body>
10     <h1>Nagłówek stopnia pierwszego</h1>
11     <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
        Praesent tristique scelerisque sapien. Aliquam vel ipsum erat,
        vitae viverra massa. Maecenas accumsan, ipsum eget mollis iaculis,
        elit velit bibendum enim, at vulputate dui nunc quis eros.
        Pellentesque non iaculis lacus. Praesent neque eros, tempus nec
        ultricies id, euismod eget lectus. Pellentesque eu justo non turpis
        pellentesque tristique eu quis magna. Praesent libero massa,
        sodales id blandit quis, ornare eu ante. Quisque id justo a ipsum
        egestas pulvinar. Vestibulum vitae arcu ut est volutpat laoreet
        ultricies non ipsum. Vestibulum eget massa odio. Morbi sed neque
        non nisi tincidunt vehicula.
12     </p>
```

```

12     </p>
13     <h2>Nagłówek stopnia drugiego</h2>
14     <p>Ut sit amet arcu in velit pulvinar faucibus. Nam ac facilisis
        lectus. Vivamus fringilla interdum dolor, quis gravida sem sodales
        eu. Aliquam porttitor hendrerit nisl, eu bibendum lorem suscipit
        eget. Nulla facilisi. Nam id urna non neque molestie lobortis.
        Fusce gravida nulla ac libero semper vel egestas elit imperdiet.
        Nullam consequat feugiat lectus non consequat. Aliquam imperdiet
        sapien sed enim gravida et varius arcu molestie. Nullam mollis,
        metus a convallis rhoncus, lorem felis condimentum nunc, ac mollis
        lorem tellus id felis. Proin nec elementum justo. Morbi in dictum
        massa. Phasellus libero libero, lobortis vel faucibus convallis,
        posuere a turpis. Mauris in nisi et velit ullamcorper varius nec id
        felis. Pellentesque adipiscing libero ut sem convallis vel
        elementum felis pulvinar. Cras imperdiet ligula vitae massa
        pulvinar ornare.
15     </p>
16     <h3>Nagłówek stopnia trzeciego</h3>
17     <h4>Nagłówek stopnia czwartego</h4>
18     <h5>Nagłówek stopnia piątego</h5>
19     <h6>Nagłówek stopnia szóstego</h6>
20
21 </body>
22 </html>

```

3. Zapisz plik w folderze i uruchom go w przeglądarce.

Ćwiczenie 9

W tym ćwiczeniu poznamy te znaczniki, które zostały zaadoptowane w HTML5 z poprzednich wersji.

1. Utwórz folder „09 Elementy zaadoptowane”.
2. Utwórz dokument o następującej treści:

```

1 <!DOCTYPE html>
2 <html lang="pl">
3 <head>
4     <meta charset="utf-8">
5     <title>Króciutki dokument HTML</title>
6
7 </head>
8
9 <body>
10     <p>Czas wstrząsnąć tą przeglądarką! Nastała era HTML5!</p>
11     <p>Czas wstrząsnąć tą przeglądarką! Nastała era HTML5!</p>
12     <p>Czas wstrząsnąć tą przeglądarką! Nastała era HTML5!</p>
13 </body>
14 </html>
15

```

3. Wypróbuj działanie następujących znaczników:

- **<small>** - drobny druk używany np. w klauzulach u dołu umowy, których nikt nie czyta ;)
 - **<hr>** - linia pozioma
 - **** - oznaczenie ważnego tekstu.
 - **** - fragmenty pogrubione lecz nie bardziej ważne.
 - **** - nadawanie słowom emfazy-szczególnie w miejscach akcentowanych w trakcie wymawiania.
 - **<i>** - wyróżnienie kursywą tekstu pozbawionego emfazy.
4. Efekt zapisz w folderze i przetestuj w przeglądarce.

Ćwiczenie 10

W tym ćwiczeniu zapoznamy się z oznaczeniem cytatu.

1. Utwórz folder „10 Cytat”.
2. Zmodyfikuj plik HTML z ćwiczenia 08 według wzoru:

```

9  <body>
10  <h1>Zwyczajny blok tekstu</h1>
11  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
    Praesent tristique scelerisque sapien. Aliquam vel ipsum erat,
    vitae viverra massa. Maecenas accumsan, ipsum eget mollis iaculis,
    elit velit bibendum enim, at vulputate dui nunc quis eros.
    Pellentesque non iaculis lacus. Praesent neque eros, tempor nec
    ultricies id, euismod eget lectus. Pellentesque eu justo non turpis
    pellentesque tristique eu quis magna. Praesent libero massa,
    sodales id blandit quis, ornare eu ante. Quisque id justo a ipsum
    egestas pulvinar. Vestibulum vitae arcu ut est volutpat laoreet
    ultricies non ipsum. Vestibulum eget massa odio. Morbi sed neque
    non nisi tincidunt vehicula.
12  </p>
13  <h2>Cytat</h2>
14  <blockquote cite="http://onet.pl">Ut sit amet arcu in velit
    pulvinar faucibus. Nam ac facilisis lectus. Vivamus fringilla
    interdum dolor, quis gravida sem sodales eu. Aliquam porttitor
    hendrerit nisl, eu bibendum lorem suscipit eget. Nulla facilisi.
    Nam id urna non neque molestie lobortis. Fusce gravida nulla ac
    libero semper vel egestas elit imperdiet. Nullam consequat feugiat
    lectus non consequat. Aliquam imperdiet sapien sed enim gravida et
    varius arcu molestie. Nullam mollis, metus a convallis rhoncus,
    lorem felis condimentum nunc, ac mollis lorem tellus id felis.
    Proin nec elementum justo. Morbi in dictum massa. Phasellus libero
    libero, lobortis vel faucibus convallis, posuere a turpis. Mauris
    in nisi et velit ullamcorper varius nec id felis. Pellentesque
    adipiscing libero ut sem convallis vel elementum felis pulvinar.
    Cras imperdiet ligula vitae massa pulvinar ornare.
15  </blockquote>
16  </body>

```

3. Zapisz plik w folderze i przetestuj w przeglądarce.

Uwaga do kodu:

W parze ze znacznikiem **<blockquote></blockquote>** występuje atrybut `cite=""`, który pozwala podać źródło cytowanego tekstu. Na poniższym listingu umieściłem przykład wykorzystania `cite` wraz ze

znacznikami odpowiedzialnymi za cytowanie tekstu. Element ten nie jest widoczny w treści tworzonej strony i można do niego dotrzeć, analizując kod HTML.

```
<blockquote cite="http://adres.pl/plik.html">Cytowany tekst</blockquote>
```

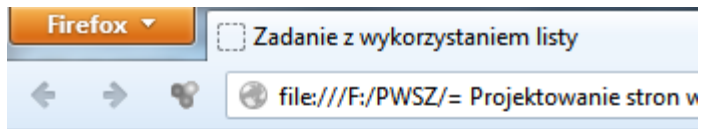
Ćwiczenie 11

W tym ćwiczeniu poznamy jak wyglądają listy.

1. Utwórz folder „10 Lista”.
2. Utwórz dokument HTML z zawartością:

```
1  <!doctype html>
2  <html>
3  <head>
4      <meta charset="UTF-8" />
5      <title>Tytuł przykładowej strony
6      </title>
7  </head>
8  <body>
9      <h1>Lista wypunktowana</h1>
10 <ul>
11 <li>Pierwsza pozycja listy wypunktowanej</li>
12 <li>Druga pozycja listy wypunktowanej</li>
13 <li>Trzecia pozycja listy wypunktowanej</li>
14 </ul>
15
16 <h1>Lista numerowana</h1>
17 <ol>
18 <li>Pierwsza pozycja listy numerowanej</li>
19 <li>Druga pozycja listy numerowanej</li>
20 <li>Trzecia pozycja listy numerowanej</li>
21 </ol>
22
23 <h1>Lista definicji</h1>
24 <dl>
25 <dt>Słowo definiowane</dt>
26 <dd>opis słowa definiowanego</dd>
27 <dt>Kolejne słowo definiowane</dt>
28 <dd>opis kolejnego słowa definiowanego</dd>
29 <dt>Jeszcze jedno słowo definiowane</dt>
30 <dd>opis nowego słowa definiowanego</dd>
31 <dt>Ostatnie słowo definiowane</dt>
32 <dd>opis ostatniego słowa definiowanego</dd>
33 </dl>
34 </body>
35 </html>
```

3. Zapisz plik w folderze.
4. W kolejnym pliku spróbuj wykonać następujący dokument.



Lista zakupów

1. Produkty mleczne
 - mleko 3,2
 - jogurt malinowy
 - śmietanka do kawy
2. Owoce
 - jabłka
 - cytrusy
 - pomarańcze
 - mandarynki
 - brzoskwiwnie
3. Pieczywo
 - chleb
 - bułki
 - rogaliki