

Czas trwania zajęć: 45 minut, przedmiot: TI

**Temat lekcji:** Pierwsze kroki w języku HTML.

Cel ogólny lekcji:

Wprowadzenie do tworzenia stron WWW w języku HTML. Wprowadzenie pojęć: strona WWW, język HTML, dokument HTML.

Cele operacyjne:

**Uczeń wie:**

1. Co to jest strona WWW, język HTML, dokument HTML.
2. Jak skorzystać z edytora Notatnik przy tworzeniu dokumentu HTML.
3. W jaki sposób stworzyć prosty dokument HTML.
4. W jaki sposób działają podstawowe znaczniki formatowania tekstu.

**Uczeń umie:**

1. Wytłumaczyć, co się kryje pod pojęciami: strona WWW, język HTML, dokument HTML.
2. Korzystać z edytora Notatnik przy tworzeniu dokumentu HTML.
3. Stworzyć prosty dokument HTML.
4. Sformatować tekst podstawowymi znacznikami

Metody pracy:

- wykład, pogadanka,
- ćwiczenia praktyczne

Forma pracy:

- indywidualna

Środki dydaktyczne:

- tablica, komputer (z Notatnikiem i przeglądarką www)
- plik przyklad1.txt

## Przebieg lekcji:

### 1. Wstęp:

- ❖ Sprawdzenie obecności.
- ❖ Sformułowanie tematu lekcji: „Pierwsze kroki w języku HTML”.

### 2. Część właściwa:

- ❖ Nauczyciel zadaje pytania uczniom:

#### 1. Czym jest strona WWW?

**Strona WWW** – dokument napisany w języku HTML i w takiej postaci przechowywany w serwerze WWW. Zawiera hipertekst uzupełniony elementami graficznymi i multimedialnymi. Osoba zainteresowana obejrzeniem danej strony może ją pobrać z serwera, posługując się przeglądarką zainstalowaną we własnym komputerze.

#### 2. Czym jest język HTML?

**Język HTML** – (Hypertext Markup Language - Hipertekstowy Język Oznaczenia) język przeznaczony do opisywania dokumentów hipertekstowych, prezentowanych i odczytywanych następnie w sieci WWW za pomocą przeglądarki. Poszczególne elementy składowe strony WWW, takie jak tytuły, akapity, obrazy, dźwięki, łączniki z innymi dokumentami, adresy stron WWW są wyróżniane w tym języku za pomocą znaczników, które pełnią funkcję poleceń dla przeglądarki. Dzięki temu hipertekst może być opisany z użyciem bardzo prostego edytora tekstowego.

#### 3. Czym jest dokument HTML?

**Dokument HTML** – dokument napisany w języku HTML. Taki dokument można utworzyć w najprostszym edytorze tekstu, np. w Notatniku, programie ze środowiska Windows zapisując go z rozszerzeniem .htm lub .html. Do odczytywania dokumentów HTML służą przeglądarki.

#### 4. Co jest potrzebne, żeby napisać stronę internetową?

Teoretycznie może to być dowolny edytor tekstu i dobre chęci. Jak wcześniej wspomnieliśmy dokument HTML jest po prostu plikiem tekstowym, gdzie wpisujemy wszystkie polecenia, dotyczące formatowania tekstu i wstawiania grafiki. Lecz pisanie strony w zwykłym edytorze tekstu, byłoby bardzo

uciażliwe. Dlatego powstało wiele wyspecjalizowanych edytorów, które ułatwią, przyspieszą i uprzyjemnią Twoją pracę. Są to np.:

- Pajaczek: [www.pajaczek.pl](http://www.pajaczek.pl) (shareware - płatny)
- EdHTML:  
<http://binboy.sphere.pl/index.php?show=download&p1=./webdevelopment/editors> (darmowy)
- Edytor znaczników EzHTML (darmowy)

## Język HTML

Dokument HTML zawiera **tekst**, który ma być umieszczony na stronie i **polecenia** (zwane **znacznikami**, „**tagami**”), które stanowią informacje dla przeglądarki, jak ten dokument ma być wyświetlony na ekranie. Polecenia, dla odróżnienia od zwykłego tekstu umieszcza się w nawiasach < >. Prawie każde polecenie języka HTML składa się z dwóch części – otwierającej i zamykającej. Różnią się one tym, że część zamykająca poprzedzona jest znakiem /. Polecenia informują przeglądarkę, co ma zrobić z tekstem, a **parametry** tych poleceń, jak ma to być zrobione.

### 5. Budowa dokumentu HTML. (plik przyklad1.txt)

Dokument w języku HTML jest zawarty między znacznikami:

<HTML> i </HTML>

i dzieli się na dwie części: nagłówek i treść (ciało strony)

Nagłówek rozpoczyna się poleceniem <HEAD>, a treść poleceniem <BODY>. W nagłówku znajduje się tekst tytułu poprzedzony poleceniem <TITLE> - będzie on wyświetlany w pasku tytułowym przeglądarki.

### Schemat dokumentu HTML

```
<HTML>
  <HEAD>
    <TITLE> tytuł strony </TITLE>
  </HEAD>
  <BODY>
    treść strony
  </BODY>
</HTML>
```

```
<html>
  <head>
    <title> tytuł strony </title>
  </head>
  <body>
    treść strony
  </body>
</html>
```

## 6. Polskie znaki – znaczniki meta (kilka wybranych).

```
<META HTTP-EQUIV="Content-type" CONTENT="text/html; charset=iso-8859-2">
```

lub

```
<META HTTP-EQUIV="Content-type" CONTENT="text/html; charset=windows-1250">
```

Polecenia te są deklaracją strony kodowej, czyli sposobu, w jaki będą kodowane znaki na naszej stronie WWW. Zamiast **charset=iso-8859-2** można wpisać: **charset=windows-1250**, ale jest to mocno odradzane, ponieważ polecenie takie deklaruje inną stronę kodową, podczas gdy w całym niemal polskim Internecie przyjęto standard kodowania **iso-8859-2**.

Stronę kodową *Windows* obsługują tylko przeglądarki w systemie *MS Windows* - jeśli użytkownik będzie posiadał inny system operacyjny, prawie na pewno spowoduje to pojawienie się u niego na ekranie zupełnie nieprzewidzianych znaków-krzaczków!.

Z tego powodu nie jest polecane stosowanie innej strony kodowej jak ISO.

```
<meta name="description"content="tu wpisz krótki opis strony">
```

Z tego tekstu korzystają wyszukiwarki internetowe. Pozwala on osobie poszukującej stron na określony temat, zorientować się, czego dana strona dotyczy, zanim jeszcze ją odwiedzi. Wartość atrybutu content wyświetlana jest na liście stron odnalezionych, przez wyszukiwarkę.

```
<meta name="keywords"content="tu wpisz słowa kluczowe">
```

Z tego tekstu korzystają wyszukiwarki internetowe. Jako wartość atrybutu content podaje się tak zwane słowa kluczowe (oddzielone od siebie przecinkami). Słowa te mają zapewnić skuteczność wyszukiwania danej strony w Internecie.

## 7. Poprawność kodu.

Specyfikacja HTML 4.01: [www.w3.org/TR/html4](http://www.w3.org/TR/html4), opracowana przez organizację W3C: <http://www.w3.org> określa elementy i atrybuty, które spełniają dotychczasowe standardy pisania w języku HTML. Niektóre elementy są odradzane, ale przeglądarki internetowe nadal je interpretują (z uwagi na kompatybilność wstecz), lecz w przyszłości mogą stać się zupełnie przestarzałe i nieobsługiwane. Zamiast nich w większości przypadków zaleca się stosowanie odpowiednich poleceń stylów CSS. Jeśli jednak koniecznie chcemy wykorzystać takie elementy, powinniśmy zastosować *deklarację typu dokumentu (DTD - document type declaration) Transitional* (czyli przejściową). *Deklaracja typu dokumentu* określa poziom użytego na stronie języka HTML. Istnieją trzy rodzaje DTD:

1. *Strict DTD* - (ścista) zawiera wszystkie elementy i atrybuty, które nie są *zdeprecjonowane* (nie *deprecated*) oraz nie pojawiają się w dokumentach z [ramkami](#)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN">
```

2. *Transitional DTD* - (przejściowa) lub inaczej *loose DTD* ("luźna") - zawiera wszystko co w *Strict DTD* plus elementy i atrybuty *zdeprecjonowane*

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
```

3. *Frameset DTD* - (dla ramek) zawiera wszystko co w *Transitional DTD* plus elementy odnoszące się do ramek: FRAME, FRAMESET oraz NOFRAMES.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN">
```

Aby wstawić *deklarację typu dokumentu*, należy na samym początku strony wpisać jedną z powyższych deklaracji, jako pierwszą linię w dokumencie - jeszcze przed znacznikiem otwierającym HTML. Podanie adresu wersji DTD (drugi przypadek w każdym z punktów), pozwala przeglądarce pobrać wersję DTD oraz wszystkie potrzebne zestawy znaków.

Jeśli nie podamy żadnej z powyższych wersji DTD, jako domyślna najprawdopodobniej zostanie przyjęta *Transitional DTD* (w IE), ale sposób ten absolutnie nie jest zalecany! W każdym dokumencie HTML, na samym początku, powinna się pojawić deklaracja DTD. Najbezpieczniejszym rozwiązaniem w większości zastosowań wydaje się użycie *Transitional DTD* chyba, że chcemy się ściśle trzymać wytycznych specyfikacji, co do znaczników i atrybutów *zdeprecjonowanych*, wtedy należy zastosować *Strict DTD* i nie używać tych znaczników ani atrybutów.

## 8. Wybrane polecenia.

|          |  |
|----------|--|
| <I>      | ustalenie wyróżnienia czcionki – kursywa                   |
| <B>      | ustalenie wyróżnienia czcionki – pogrubiona                |
| <U>      | ustalenie wyróżnienia czcionki – podkreślona               |
| <CENTER> | ustawienie tekstu i grafiki na środku strony               |
| <BR>     | zmiana wiersza – przeniesienie tekstu o jeden wiersz w dół |
| &nbsp;   | dodatkowy odstęp – wprowadzenie dodatkowej spacji          |
| &lt;     | <  |
| &gt;     | >  |
| &quot;   | "  |
| &amp;    | &  |
| <strike> | wypisuje przekreślony tekst                                |

## 9. Zagnieżdżanie znaczników

Na jeden fragment tekstu może wpływać więcej niż jeden znacznik - i wtedy mówimy o znacznikach zagnieżdżonych. Jednak zawsze należy zachować w takim przypadku kolejność domykania znaczników - jako ostatni musi być domknięty ten, który został otwarty pierwszy. Podobnie, jako pierwszy domykać zawsze trzeba ten znacznik, który był otwarty ostatnio. Tak, więc poprawna będzie konstrukcja:

Fragment **ważnego tekstu** *możemy wyróżnić* **pogrubieniem**.  
natomiast błędna będzie

~~Fragment **ważnego tekstu** *możemy wyróżnić* **pogrubieniem**.~~

### 3. Podsumowanie:

Wspólnie z klasą podsumowanie poznanych informacji.

**Plik, na którym pracują uczniowie.**

**przyklad1.txt**

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<HTML>
<HEAD>
<META HTTP-EQUIV="Content-type" CONTENT="text/html; charset=windows-1250">
<TITLE> tytuł strony </TITLE>
</HEAD>
<BODY>
treść strony
</BODY>
</HTML>
```