

Piotr Chojnacki

<http://www.piotrchojnacki.pl>

IV rok, informatyka chemiczna

Gimnazjum nr 35 we Wrocławiu

Wrocław dn. 20 kwietnia 2006 roku

Czas trwania zajęć: 90 minut, przedmiot: informatyka

Temat lekcji: Style CSS.

Cel ogólny lekcji:

Przypomnienie pojęć: strona WWW, język HTML, dokument HTML. Wprowadzenie stylu CSS jako potężne narzędzie do formatowania stron WWW.

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. Co to są style CSS i do czego służą.

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 za pomocą stylu CSS.

Metody pracy:

- wykład, pogadanka,
- ćwiczenia praktyczne

Forma pracy:

- indywidualna

Środki dydaktyczne:

- tablica, komputer (z Notatnikiem i przeglądarką www)
- plik przykład.html, style.css

Przebieg lekcji:

1. Wstęp: (5 minut)

- Sprawdzenie obecności.
- Sformułowanie tematu lekcji: „Style CSS”.

2. Część właściwa: (80 minut)

- ❖ Przypomnienie wiadomości. Nauczyciel zadaje pytania uczniom: (10 minut)

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 Oznaczania) 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 (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>
```

❖ Czym są *style*? (5 minut)

Co to są style?:

Jest to potężne narzędzie do formatowania stron WWW (CSS - Cascading Style Sheets - Kaskadowe Arkusze Stylów). Dają one ogromne możliwości manipulowania wyglądem dokumentów i zwiększają efektywność narzędzi. Dodatkowo nie powodują problemów w przeglądarkach, które ich nie obsługują. *Styl* jest zespołem poleceń formatujących i pozwala zmieniać wygląd pojedynczych elementów strony lub nawet całej serii dokumentów.

Dlaczego warto używać styli?:

1. *Style* stały się już praktycznie podstawowym narzędziem formatującym. Jeśli poważnie myślisz o zajęciu się projektowaniem stron WWW, koniecznie musisz je poznać.
2. Przeszarżałe atrybuty i znaczniki, znajdujące się bezpośrednio w składni HTML, które dotyczą formatowania, będą stopniowo wycofywane przez producentów przeglądarek internetowych, na rzecz rekomendowanych analogicznych deklaracji *stylów*. Widać już nawet różnice pomiędzy IE4 a IE5!
3. Dokumenty pisane z wykorzystaniem arkuszy stylów są zwykle bardziej przejrzyste i krótsze.
4. *Style* pozwalają w łatwy sposób zarządzać całą serią dokumentów, poprzez stosowanie zewnętrznych arkuszy stylów. Dzięki temu w łatwy i wygodny sposób, można dokonać modyfikacji rodzaju formatowania jednocześnie we wszystkich dokumentach, zmieniając dane tylko w jednym pliku!
5. Dzięki możliwości stosowania klas selektorów, znacznie oszczędzamy sobie pisanie. W jednym miejscu określamy wszystkie atrybuty formatowania (których może być bardzo dużo), odnoszące się do wielu elementów, które mają wyglądać tak samo. Bezpośrednio przy elemencie wystarczy podać tylko nazwę klasy i nie musimy już wypisywać "litanii" poleceń. Znacznie przyspiesza to późniejszą modyfikację strony, ponieważ zmian dokonujemy tylko w jednym miejscu, a wpływają one na wiele elementów jednocześnie.
6. Możliwość stosowania różnorodnych jednostek oraz sposobów definiowania kolorów.

❖ Wstawianie stylów: (10 minut)

- styl lokalny: `<SELEKTOR style="cecha: wartość; cecha2: wartość2...">...</SELEKTOR>`

Styl lokalny pozwala na nadanie formatowania konkretnemu pojedynczemu elementowi strony. Dlatego właśnie styl tego rodzaju nazywa się także *inline* (ang. "w linii"), ponieważ jest wstawiany w tej samej linii, w której znajduje się element formatowany. O tym który to będzie element, decyduje wyraz "**SELEKTOR**" (widoczny powyżej, w *deklaracji stylu*).

- rozciąganie stylu: `...`

Polecenie **SPAN** pozwala na objęcie blokiem pewnego większego fragmentu dokumentu (który znajduje się pomiędzy znacznikami `` oraz ``). Pojedynczym poleceniem **SPAN** możemy objąć kilka różnych elementów, np. paragraf i wykaz.

- wydzielone bloki: `<DIV style="cecha: wartość; cecha2: wartość2...">...</DIV>`

Fragment dokumentu wydzielony za pomocą bloku (znaczniki `<DIV>` oraz `</DIV>`), możemy swobodnie formatować. Metoda ta jest bardzo podobna do **SPAN**, lecz obejmuje zwykle większe fragmenty dokumentu (może zawierać w sobie różne znaczniki jak również inne bloki).

- wewnętrzny arkusz stylów:

```
<HEAD>
(...)
<STYLE TYPE="text/css">
<!--
SELEKTOR { cecha: wartość; cecha2: wartość2... }
SELEKTOR2 { cecha: wartość; cecha2: wartość2... }
(...)
-->
</STYLE>
(...)
</HEAD>
```

- zewnętrzny arkusz styli:

```
<HEAD>
(...)
<LINK REL="Stylesheet" HREF="ścieżka dostępu do pliku *.css" TYPE="text/css">
(...)
</HEAD>
```

- import arkusza styli:

```
<STYLE TYPE="text/css">
<!--
@import url(adres zewnętrznego arkusza stylów);
-->
</STYLE>
```

❖ Kaskadowość styli: (5 minut)

Kaskadowość stylów określa pierwszeństwo w oddziaływaniu na te same elementy strony, stylów z różnych źródeł.

Jeśli w dokumencie znajduje się kilka źródeł stylów, pierwszeństwo mają te, które znajdują się "bliżej" formatowanego elementu. Oddziaływanie stylów z arkuszy zewnętrznych może być modyfikowane przez style zdefiniowane w nagłówku dokumentu, to zaś może być zmieniane przez style zdefiniowane bezpośrednio w ciele dokumentu (inline).

Zatem priorytet ważności stylów (pierwszeństwo) wyglądałby tak:

1. Styl lokalny (inline)
2. Rozciąganie stylu (SPAN) oraz wydzielone bloki (DIV)
3. Wewnętrzny arkusz stylów
4. Zewnętrzny arkusz stylów oraz import arkusza stylów

Można jednak świadomie zmienić zasady *kaskadowości*. Służy do tego polecenie **!important**. Jeżeli pojawi się ono w deklaracji stylu po wartości cechy, spowoduje to, że taka cecha będzie miała pierwszeństwo przed innymi, nawet, jeśli ma niższy priorytet

❖ Selektory specjalne: (5 minut)

- klasy selektorów:

```
<HEAD>
(...)
<STYLE TYPE="text/css">
<!--
(...)
SELEKTOR.klasa { cecha: wartość }
(...)
-->
</STYLE>
(...)
</HEAD>
<BODY>
(...)
<SELEKTOR class="klasa">...</SELEKTOR>
(...)
</BODY>
```

gdzie kolorem **niebieskim** zaznaczono wewnętrzny arkusz stylów, w którym znajduje się deklaracja *klasy* (może się ona również znajdować w zewnętrznym lub importowanym arkuszu). Natomiast kolorem **czerwonym** zaznaczono konkretny przykład użycia zdefiniowanej wcześniej *klasy*.

SELEKTOREM może być dowolny znacznik języka HTML (np.: **P** - paragraf, **Hn** - tytuł określonego rzędu czy **TD** - komórka tabeli i inne)

Przykład:

W wewnętrznym arkuszu stylów tej strony została umieszczona następująca deklaracja:

```
H4.klasa { color: red }
```

Dlatego jeśli teraz w dowolnym miejscu wpiszemy:

```
<H4 class="klasa">To jest tytuł rzędu 4</H4>
```

zostanie wyświetlony w kolorze czerwonym:

- selektor identyfikatora:

```
<HEAD>
(...)
<STYLE TYPE="text/css">
<!--
(...)
SELEKTOR#identyfikator { cecha: wartość }
(...)
-->
</STYLE>
(...)
</HEAD>
<BODY>
(...)
<SELEKTOR id="identyfikator">...</SELEKTOR>
(...)
</BODY>
```

gdzie kolorem **niebieskim** zaznaczono wewnętrzny arkusz stylów, w którym znajduje się deklaracja *identyfikatora* (ID). Natomiast kolorem **czerwonym** zaznaczono konkretny przykład użycia zdefiniowanego wcześniej *identyfikatora*.

SELEKTOREM może być dowolny znacznik języka HTML (np.: **P** - paragraf, **Hn** - tytuł określonego rzędu czy **TD** - komórka tabeli i inne)

Przykład:

W wewnętrznym arkuszu stylów tej strony została umieszczona następująca deklaracja:

```
H5#identyfikator { color: red }
```

Dlatego element o podanym identyfikatorze ID:

```
<H5 id="identyfikator">To jest tytuł rzędu 5</H5>
```

zostanie wyświetlony w kolorze czerwonym:

❖ Omówienie podstawowych poleceń formatowania: **(45 minut)**

Wszystkie poniższe polecenia uczniowie wykonują na przykładowym dostarczonym pliku: **przyklad.html**

SELEKTOR { font-size: rozmiar }

- *xx-smal* - najmniejsza
 - *x-small* - mniejsza
 - *small* - mała
 - *medium* - średnia
 - *large* - duża
 - *x-large* - większa
 - *xx-large* - największa
- bądź px, em, in itp.

SELEKTOR { font-family: rodzaj, rodzaj1, rodzaj2,... }

np.: 'Times New Roman', 'Courier New'

Oprócz podania rodzaju czcionki wprost, możliwe jest także wpisanie rodziny ogólnej:

serif - czcionka szeryfowa (końcówki znaków posiadają "ozdobniki"), np.: 'Times New Roman', Georgia, Garamond, Bodoni

sans-serif - czcionka bezszeryfowa (końcówki znaków są proste), np.: Arial, Verdana, 'Trebuchet MS', Helvetica, Univers, Futura

monospace - czcionka o stałej szerokości znaków - monotypiczna (wygląda, jak pisana na maszynie), np.: Courier, 'Courier New'

cursive - czcionka mająca pewne cechy pochyłej (wygląda, jak pisana ręcznie)

fantasy - czcionka fantazyjna (dekoracyjna)

SELEKTOR { font-style: styl }

Natomiast jako "styl" należy wpisać:

normal - czcionka normalna (podstawowa)

italic - czcionka pochylona

oblique - również czcionka pochylona (podobna jak poprzednio)

SELEKTOR { font-weight: waga }

normal - czcionka normalna (podstawowa)

bold - czcionka pogrubiona 100, 200, 300, 400 (odpowiednik "normal"), 500, 600, 700 (odpowiednik "bold"), 800, 900 - każdy następny numer wskazuje wagę czcionki przynajmniej tak samo wytłuszczonej jak dla poprzedniej wartości w sekwencji

SELEKTOR { color: kolor }

SELEKTOR { text-decoration: dekoracja }

none - bez zmian

underline - podkreślenie

line-through - przekreślenie

overline - nadkreślenie

SELEKTOR { text-align: wyrównanie }

left - wyrównanie tekstu do lewego marginesu (domyślnie)

right - wyrównanie do prawego marginesu

center - do środka (wyśrodkowanie)

justify - do obu marginesów jednocześnie (justowanie).

SELEKTOR { text-indent: wcięcie }

SELEKTOR { background-color: kolor }

SELEKTOR { background-image: url(ścieżka dostępu do obrazka) }

SELEKTOR { background-repeat: powtarzanie }

Natomiast jako "powtarzanie" należy wpisać:

repeat - powtarzanie tła w obu kierunkach (domyślnie)

repeat-x - powtarzanie tła tylko w kierunku poziomym

repeat-y - powtarzanie tła tylko w kierunku pionowym

no-repeat - brak powtarzania tła (zostanie wyświetlone jako pojedynczy obrazek)

```
SELEKTOR { margin-top: rozmiar }
SELEKTOR { margin-bottom: rozmiar }
SELEKTOR { margin-left rozmiar }
SELEKTOR { margin-right: rozmiar }
```

```
SELEKTOR { padding-top: rozmiar }
SELEKTOR { padding-bottom: rozmiar }
SELEKTOR { padding-left: rozmiar }
SELEKTOR { padding-right: rozmiar }
```

```
SELEKTOR { width: szerokość }
SELEKTOR { width: szerokość }
```

❖ Formatowanie odnośników:

przykład:

```
a.odnosnik:link {color: #666666; text-decoration: underline; font-size: 16px; }
a.odnosnik:visited {color: #666666; text-decoration: underline; font-size: 16px; }
a.odnosnik:active {color: #666666; text-decoration: underline; font-size: 16px; }
a.odnosnik:hover { color: black; background: none; text-decoration: none; font-size: 16px; }
```

❖ Ćwiczenie dla uczniów:

Na koniec ćwiczenie do wykonania. Uczniowie tworzą stronę internetową z zastosowaniem CSS. Wzór strony w pliku wzor.jpg.

**Przed Tobą znajduje się kilka prostych
zadań praktycznych dotyczących
języka HTML i CSS.**

1. Utwórz plik o nazwie ***twojeimie.html***.
2. Następnie korzystając z poznanych wcześniej znaczników stwórz następującą stronę internetową (przykład: **wzor.jpg**)
3. Stronę rozpocznij ustawieniem tła jako **czarne** i koloru czcionki na **białe**.

4. Stronę rozpocznij nagłówkiem (**wyrównany do środka**) o wielkości 2 wpisując tekst:

Oto moja pierwsza strona internetowa

5. Następnie niżej w paragrafie (**justowany**) wpisz tekst o wielkości 12px:

Miesiące jesienne to miesiące intensywnego korzystania z Internetu. Znajduje to wyraz w listopadowych wynikach badania Megapanel. W listopadzie polscy użytkownicy Internetu dokonywali więcej odsłon, a także spędzali więcej czasu. Najchętniej oglądanym serwisem był Onet.pl, który przyciągnął 7,36 miliona użytkowników. Dane na dzień 7 lutego 2006 roku.

Uwzględniając przy tym, aby data była pogrubiona pochylona czcionka wielkości 10px oraz koloru żółtego.

6. Utwórz odnośnik do strony Onet.pl, gdzie, a po najechnaniu myszką link podkreśli się i zmieni kolor na zielony

3. Podsumowanie: (5 minut)

Wspólnie z klasą podsumowanie poznanych informacji.

Zawartość pliku dla uczniów:

przyklad.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=windows-1250">
<link rel="stylesheet" href="style.css" type="text/css">
</head>
<body>
<h1>Znaczniki HTML z zastosowaniem CSS</h1>
<div>&lt;!DOCTYPE&gt; standard języka HTML użyty w dokumencie</div>
<div>&lt;HEAD&gt; opis dokumentu</div>
<div>&lt;TITLE&gt; tytuł dokumentu</div>
<div>&lt;BODY&gt; zawartość dokumentu</div>
<div>&lt;A&gt; hiperłącze</div>
<div>&lt;H1&gt; nagłówek</div>
<div>&lt;OL&gt; lista numerowana</div>
<div>&lt;P&gt; akapit</div>

<br><br><br>

<div id="identyfikator">przykład identyfikatora</div>

<br>
<div class="klasa">przykład klasy</div>

<br><br>
przykład formatowania odnośników:

<a class="odnosnik"href="http://www.wp.pl" target="_blank">wp.pl</a>

</body>
</html>
```

style.css

```
body { background-color: black; color: white; }
h1 { text-align: center; }

#identyfikator
{
color: green
}
```

```
.klasa
{
color: yellow;
}

a.odnosnik:link {color: #666666; text-decoration: underline; font-size: 16px; }
a.odnosnik:visited {color: #666666; text-decoration: underline; font-size: 16px; }
a.odnosnik:active {color: #666666; text-decoration: underline; font-size: 16px; }
a.odnosnik:hover { color: black; background: none; text-decoration: none; font-size: 16px; }
```