helion.pl
KSIĘGARNIA INTERNETOWA · WYDAWNICTWO

IDŹ DO
PRZYKŁADOWY ROZDZIAŁ
SPIS TREŚCI

KATALOG KSIĄŻEK
KATALOG ONLINE
ZAMÓW DRUKOWANY KATALOG

TWÓJ KOSZYK
DODAJ DO KOSZYKA

CENNIK I INFORMACJE
ZAMÓW INFORMACJE O NOWOŚCIACH
ZAMÓW CENNIK

CZYTELNIA
FRAGMENTY KSIĄŻEK ONLINE

CSS. Ćwiczenia



Autor: Maria Sokół
ISBN: 83-246-1097-9
Format: A5, stron: 152
Przykłady na ftp: 797 kB

Usprawnij proces projektowania stron WWW

- Jak osadzać style w dokumentach XHTML?
- W jaki sposób formatować tekst, tabele i łącza?
- Jak pozycjonować elementy strony WWW?

Użytkownicy sieci przeglądają witryny WWW nie tylko na monitorach komputerowych. Często także strony internetowe wyświetlane są na telefonach komórkowych, urządzeniach PDA i terminalach Blackberry. Programy ułatwiające osobom niepełnosprawnym korzystanie z sieci odczytują treści witryn WWW za pomocą syntezatorów mowy. Jednak aby witryna WWW działała prawidłowo w połączeniu z różnymi urządzeniami wyświetlającymi ją, niezbędne jest zaprojektowanie jej tak, aby elementy odpowiedzialne za jej wygląd były odseparowane od treści. W tym celu konsorcjum W3C zaproponowało w roku 1996 technologię kaskadowych arkuszy stylów – CSS. Dzięki stylom projektanci witryn mogą zdefiniować kolorystykę, liternictwo i inne aspekty wyglądu strony niezależnie od jej treści.

Książka „CSS. Ćwiczenia" to wprowadzenie do technologii kaskadowych arkuszy stylów. Czytając ją i wykonując zawarte w niej ćwiczenia nauczysz się korzystać ze stylów w projektach witryn WWW. Poznasz sposoby łączenia opisów stylów z dokumentami XHTML, zasady definiowania stylów dla tekstu i grafiki, jednostki miar stosowane w stylach i metody pozycjonowania elementów na stronie. Dowiesz się, jak określić krój i rozmiar czcionki, wyróżnić hiperłącza, sformatować tabele i listy oraz przygotować układ strony WWW oparty na warstwach.

- Najważniejsze atrybuty dokumentów XHTML
- Rodzaje stylów CSS
- Jednostki miary i kolory w stylach
- Formatowanie tekstu
- Pozycjonowanie elementów strony
- Definiowanie wyglądu hiperłączy
- Projektowanie układu strony w oparciu o style CSS

Dołącz do grona profesjonalistów stosujących w pracy najnowsze technologie

Wydawnictwo Helion
ul. Kościuszki 1c
44-100 Gliwice
tel. 032 230 98 63
e-mail: helion@helion.pl

Spis treści

Rozdział 1. XHTML a CSS 5

Kod XHTML 5
Znaczniki 6
Podstawowe wymagania dotyczące kodu XHTML 7
Atrybut id 11
Atrybut lang 11
Obowiązkowe elementy XHTML 12
Przykładowy dokument XHTML 12
Kodowanie polskich znaków diakrytycznych 15
Wyświetlanie strony w przeglądarce 19

Rozdział 2. Kaskadowe arkusze stylów — podstawowe definicje i zasady 21

Zasady ogólne 22
Typy kaskadowych arkuszy stylów 23
Grupowanie kodu 31
Klasy stylu 32
Zewnętrzny arkusz stylów zawierający klasy 37

Rozdział 3. Skalowalność arkusza stylów 43

Jednostki miar CSS 43
Jednostka em 45
Skalowanie obszaru 46
Skalowanie wcięć 48
Skalowanie obrazów 48

Rozdział 4. Kolory 53
Atrybuty definiujące kolor 55
Rozdział 5. Tekst 61
Atrybuty stylu odnoszące się do czcionki 61
Rozmieszczanie tekstu w pionie i w poziomie 69
Marginesy i wcięcia 77
Marginesy wewnętrzne — padding 82
Formatowanie fragmentów tekstu 84
Trójkolumnowa strona WWW utworzona za pomocą formatowania CSS 92
Rozdział 6. Model pudełkowy 97
Obramowania 99
Style pudełek 102
Pozycjonowanie 107
Warstwy 121
Przepełnienie 126
Obrazy 129
Rozdział 7. Odnośniki 133
Zmiana wyglądu odnośników 134
Dodatek A Znaczniki i elementy 143
Dodatek B Oznaczenia i kody kolorów 151



Kaskadowe arkusze stylów — podstawowe definicje i zasady



Kaskadowe arkusze stylów (ang. *Cascading Style Sheets — CSS*), zalecana przez W3C metoda stosowania stylów, to narzędzie, które pozwoli Ci zapanować nad układem stron i serwisów WWW. Bez arkuszy CSS nie ma nowoczesnych stron WWW. Nie tylko jednak o standardy tu chodzi — kaskadowe arkusze stylów to także ogromna oszczędność czasu, o czym bez wątpienia przekonasz się, wprowadzając w kodzie swoich stron różne poprawki.

Dlaczego kaskadowe arkusze stylu nazywają się „kaskadowymi"? Wynika to ze sposobu ich działania; prawie każdy element podrzędny (a więc na przykład element `<span>` zawarty wewnątrz elementu `<p>`) dziedziczy atrybuty stylu elementu nadrzędnego, a wybrane z nich może samodzielnie nadpisywać. W efekcie powstaje swoista kaskada definicji stylu, której wyższe stopnie narzucają wygląd stopniom niższym, a te z kolei mogą anulować wybrane elementy definicji i zastąpić je własnymi, które z kolei spróbują narzucić własnym elementom podrzędnym.

W przypadku występowania różnych arkuszy stylów na jednej stronie stopień ich ważności rośnie w następującej kolejności:

1. Domyślne ustawienia przeglądarki.
2. Zewnętrzny arkusz stylów.
3. Osadzony arkusz stylów (umieszczony między znacznikami `<head> </head>`).
4. Styl wpisany (dotyczący konkretnego elementu HTML).

Arkusze CSS są obsługiwane — w mniej lub bardziej zgodny ze standardem sposób — przez wszystkie nowoczesne, graficzne przeglądarki WWW (Internet Explorer, Opera oraz Mozilla Firefox), przy czym za implementację wzorcową można uważać przeglądarkę Firefox.

Zasady ogólne

1. *Twórz uniwersalne arkusze stylów* — tak aby można było z nich korzystać w różnych dokumentach.
2. *Pamiętaj o skalowalności dokumentów* — ponieważ dokumenty będą wyświetlane w dynamicznych środowiskach i na różnym sprzęcie, szczególnie istotne jest wybranie odpowiednich jednostek miary — jednostki względne powinny mieć przewagę nad absolutnymi (dotyczy to na przykład szerokości marginesów, wielkości czcionki itp.). Więcej informacji na ten temat znajdziesz w rozdziale 3.
3. *Licz się z czytelnikiem* — niektórzy czytelnicy będą korzystali z własnych ustawień. Twoje arkusze stylów powinny to brać pod uwagę. W jaki sposób? Odpowiednie definicje stylów powinny być umieszczane we właściwych miejscach. Na przykład te, które są stosowane do całego dokumentu, umieszczaj w sekcji `<body>` — i tylko tam. W ten sposób czytelnik bez trudu wprowadzi odpowiednie zmiany.
4. *Testuj arkusze stylów* — testuj je w kilku przeglądarkach, aby nie sprawić swoim czytelnikom niemiłej niespodzianki. Jeśli korzystasz z właściwości CSS, która nie jest obsługiwana powszechnie, upewnij się, że strona wygląda względnie poprawnie nawet w mniej zaawansowanych przeglądarkach.

5. *Uwzględniaj struktury dokumentu* — arkusze stylów umożliwiają oddzielenie wyglądu strony od jej struktury logicznej. Zawsze zaczynaj od zapisania kodu strony standardowymi znacznikami nagłówków, akapitów tekstu i tabel, a dopiero później nakładaj na gotowy dokument style modyfikujące jego wygląd.
6. *Dostarczaj nazwy rodzin czcionek* — pamiętaj, że dokument będzie wyświetlany na różnych komputerach, z różnym „wyposażeniem" w czcionki. Podanie nazwy ich rodziny pozwala na zachowanie charakteru strony. Unikaj też „egzotycznych" krojów pisma, jest bowiem wysoce prawdopodobne, że może ich brakować. Najlepiej ograniczać się do czcionek dostępnych standardowo w systemach operacyjnych Windows i Linux.
7. *Zachowaj umiar w stosowaniu ujemnych marginesów* — ujemne marginesy pozwalają uzyskać ciekawe efekty, ale nie wszystkie przeglądarki radzą sobie z nimi. Jeśli stosujesz takie marginesy, testuj stronę w różnych przeglądarkach.
8. *Upraszczaj formy* — konstruując arkusz stylów, zachowaj umiar. Możesz użyć wielu różnych krojów pisma i zapełnić stronę kolorami, lecz nie wpłynie to pozytywnie na przekaz informacji.

Typy kaskadowych arkuszy stylów

Kiedyś, w epoce poprzedzającej wprowadzenie arkuszy, wygląd strony zależał od użytej przeglądarki. To ona decydowała o sposobie rozmieszczenia zawartości dokumentu. Arkusze stylów dają twórcy ścisłą kontrolę nad wyglądem strony. Pozwalają także oddzielić zawartość (treść i konstrukcję strony) od wyglądu i formatowania. Ułatwia to ewentualne późniejsze zmiany.

Istnieją trzy odmiany arkuszy: *osadzone*, *wpisane* i *łączone*.

Styl wpisany

Style wpisane są to atrybuty stylu wprowadzane w ramach atrybutu `style` dowolnego, interesującego Cię znacznika HTML. Zaletą tej metody jest jej bezpośredniość — możesz zmienić wygląd dowolnego, najmniejszego choćby elementu strony całkowicie niezależnie od wyglądu

pozostałych elementów. Z tej zalety wynika również największa wada tej metody — zmiana wyglądu większej liczby elementów wymaga wprowadzania olbrzymiej ilości kodu. Ten kod musi być później pobrany przez przeglądarkę WWW z serwera, wydłuża się zatem czas ładowania strony i zwiększa obciążenie łączy internetowych.

ĆWICZENIE

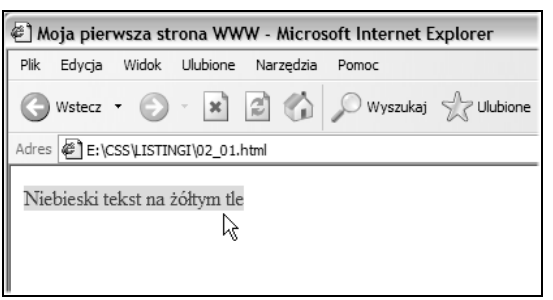
2.1 Definiowanie stylu wpisanego

Zdefiniuj w szablonie strony WWW żółty tekst na niebieskim tle, korzystając ze stylu wpisanego:

1. Otwórz szablon dokumentu *szablon.html* w oknie Notatnika.
2. Umieść w obszarze elementu `body` następującą definicję (oczywiście tekst obejmowany elementem `span` może być dowolny):

```
<span style="color: blue; background-color: yellow;">Niebieski
tekst na żółtym tle</span>
```

3. Zapisz dokument pod nową nazwą i wyświetl w oknie przeglądarki (rysunek 2.1).



Rysunek 2.1. *Do krótkiego tekstu można zastosować styl wpisany, korzystając z elementu span z atrybutem style*



Stosowanie stylu wpisanego ma sens, gdy chcesz zmienić wygląd kilku wyrazów lub linijek tekstu. Zanim jednak zastosujesz styl wpisany, zastanów się, czy nie warto zastosować stylu osadzonego. Choć styl wpisany jest najprostszą formą kaskadowych arkuszy stylu, to jednak w większości przypadków zastosowanie stylu osadzonego pozwala znacznie ograniczyć ilość kodu składającego.

Styl osadzony

Styl osadzony jest najchętniej stosowanym rodzajem kaskadowych arkuszy stylu. Definicję stylu osadzonego tworzy się w całkowitym oderwaniu od rzeczywistego elementu — określamy rodzaj czcionki, kolor tekstu i szerokość marginesu, mając na myśli nie jakiś pojedynczy, konkretny element strony, a **całą klasę** elementów. Definicja stylu może określać wygląd wybranych elementów języka HTML lub też wyłącznie ich wydzielonych podklas.

Raz stworzona definicja stylu osadzonego może być wykorzystana później do zmiany wyglądu dowolnej liczby elementów strony WWW. Co więcej, jeśli nagle zapragniesz zmienić nieco tę definicję, automatycznie zmieni się wygląd wszystkich elementów, których wygląd ta definicja określa! Nie muszę mówić, jak wielkim jest to udogodnieniem.

Trudno się dziwić, że w zasadzie wygląd każdej większej strony WWW opisany jest właśnie za pomocą osadzonego arkusza stylu. Zmniejsza się w ten sposób rozmiar pliku HTML (definicja stylu umieszczona jest tyko w jednym miejscu kodu, a potem używana wielokrotnie), zaś ewentualne zmiany wyglądu strony nie wymagają wprowadzania poprawek w dziesiątkach lub setkach miejsc kodu. Mimo to jeszcze lepsze, oszczędniejsze i zapewniające większą jednolitość dużych serwisów WWW (choć nieco trudniejsze w opanowaniu) jest rozwiązanie oparte na zewnętrznym arkuszu stylu.

W tym rozdziale zajmiemy się więc przede wszystkim stylami osadzonymi, ściśle związanymi z konkretnym plikiem HTML. Wszystkie osadzone arkusze stylów mają tę samą postać definicji i są umieszczane w sekcji `<head>` stron WWW.

Ogólna postać osadzonego arkusza CSS jest następująca:

```
<style type="text/css">
<!--
znacznik { atrybut: wartość; }
-->
</style>
```

Między znacznikami `<style>` i `</style>` umieszczana jest lista znaczników HTML wraz z właściwościami arkusza, które je definiują. Definicja zaprezentowana powyżej zawiera tylko jeden symboliczny znacznik definiowany przez jedną właściwość CSS. Jednak nic nie stoi na przeszkodzie, aby dodać do niej inne.

Zawarta w obrębie elementu `style` definicja ma następującą składnię:

```
nazwa_elementu{atrybut:wartość [; atrybut:wartość] ...}
```

nazwa_elementu to znacznik, konkretny element, klasa lub selektor, który chcesz zdefiniować; *atrybut* to atrybut, który zmieniasz, przypisując mu nową *wartość*. Atrybut i wartość rozdzielone są dwukropkiem oraz zawarte w nawiasach klamrowych:

```
body {color: black}
```

Jeśli wartość ma postać wielowyrazową — na przykład *sans serif* — umieszcza się ją w cudzysłowie:

```
p {font-family: "sans serif"}
```

Oto przykład rzeczywistej definicji:

```
<style type="text/css">
 <!--
  body       {margin:20px; color:black}
  h1         {color:blue; text-align:center}
  h2         {color:blue; text-align:left}
  p          {text-align:justify; text-indent:25px}
 -->
</style>
```

Między znacznikami `<style>` i `</style>` umieszczana jest lista elementów HTML wraz z właściwościami arkusza, które te elementy definiują. Jeśli właściwości jest kilka, wszystkie muszą być umieszczone w nawiasie klamrowym ({}) oraz oddzielone średnikami (;).

Atrybut `type` znacznika `<style>` ma wartość `text/css`. Jest to dla przeglądarki informacja o tym, że następne instrukcje to arkusz stylów. Instrukcje te zawarte są w znaczniku komentarza, `<!-- -->`. Jest to zabezpieczenie na wypadek, gdyby strona trafiła do przeglądarki starego typu, która nie potrafi obsłużyć arkuszy stylów. Wówczas instrukcje formatowania zostaną zignorowane, a nie wyświetlone na ekranie jako tekst.

ĆWICZENIE

2.2 Definiowanie osadzonego arkusza stylów

Umieść w szablonie dokumentu html osadzony arkusz stylów:

1. Otwórz szablon dokumentu w oknie Notatnika.
2. Umieść punkt wstawiania za elementem `meta` i naciśnij klawisz *Enter*.

3. Wpisz prostą definicję osadzonego arkusza stylów:

```
<style type="text/css">
<!--
body {
 background-color: white;
 margin: 0 0 0 0;
 padding: 0 0 0 0;
}
-->
</style>
```

Co oznacza ta definicja? Określa wygląd tła strony WWW — biały kolor podkładu (`background-color: white;`) oraz brak marginesów (`margin: 0 0 0 0;`) i odstępów (`padding: 0 0 0 0;`) na krawędziach strony.

4. Zapisz kod swojego dokumentu pod nową nazwą. Prezentuje go listing 2.1.

Listing 2.1. *Definicja osadzonego arkusza stylów w kodzie dokumentu HTML*

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/
TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pl" lang="pl">
 <head>
  <title>
   Moja pierwsza strona WWW
  </title>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <style type="text/css">
  <!--
  body {
   background-color: white;
   margin: 0 0 0 0;
   padding: 0 0 0 0;
  }
  -->
  </style>

 </head>
 <body>

 </body>
</html>
```

W kolejnym ćwiczeniu przygotujemy dwie definicje stylu CSS — osadzoną i wpisaną — narzucające kolor tła. Dzięki temu porównaniu zorientujesz się, na czym polega różnica.

ĆWICZENIE

2.3 Porównanie wpisanego stylu z osadzonym arkuszem stylów

Porównaj style wpisany i osadzony formatujące tło dokumentu:

1. Otwórz szablon dokumentu w dwóch oknach Notatnika.
2. W pierwszym oknie umieść punkt wstawiania za elementem `meta` (sekcja `head`) i naciśnij klawisz *Enter*.
3. Wpisz prostą definicję osadzonego arkusza stylów:

```
<style type="text/css">
<!--
body {background-color: #FF0000;
}
-->
</style>
```



Kod #FF0000 definiuje kolor. Tablice kodów kolorów znajdziesz w dodatku B.

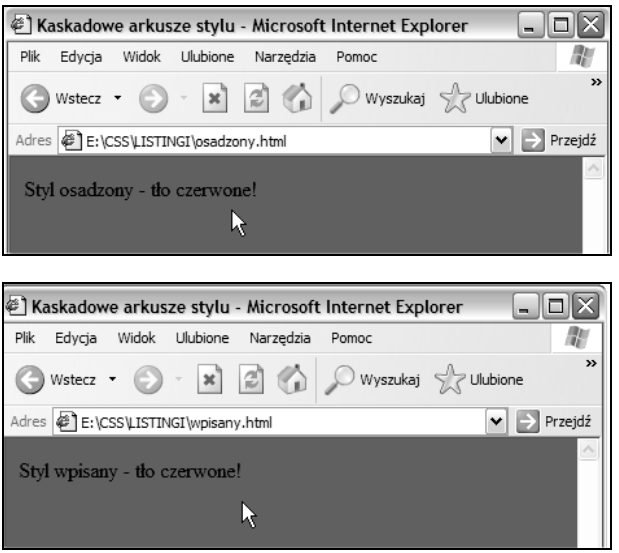
4. Zapisz pierwszy dokument pod nazwą *osadzony.html*.
5. W drugim oknie Notatnika umieść atrybut stylu w znaczniku `<body>`:

```
<body bgcolor="#FF0000">
```

6. Zapisz drugi dokument jako *wpisany.html*.
7. W przeglądarce oba dokumenty prezentują się podobnie (rysunek 2.2).

Zewnętrzny arkusz stylu

Zewnętrzny arkusz stylu nie różni się zasadniczo od arkusza osadzonego; rozszerza tylko jego uniwersalność, przenosząc definicje stylów z wnętrza kodu jednej strony WWW do osobnego pliku, który może być wykorzystany w dziesiątkach, setkach lub nawet tysiącach stron składających się na serwis internetowy.



Rysunek 2.2. *Porównanie efektów działania stylu osadzonego i wpisanego definiującego tło dokumentu*

Zewnętrzny arkusz stylu jest tym dla serwisu internetowego, czym styl osadzony był dla pojedynczej strony — umożliwia scentralizowanie definicji stylu i uniknięcie konieczności wprowadzania poprawek w wielu plikach przy każdej najdrobniejszej zmianie Twojego projektu. Zewnętrzny arkusz stylu jeszcze bardziej ogranicza ilość danych, które czytelnik musi pobrać z sieci — raz pobrany plik arkusza stylu pozostaje w pamięci podręcznej przeglądarki i jest dostępny natychmiast dla każdej kolejnej strony Twojego serwisu.

W kodzie każdej ze stron korzystających z zewnętrznych arkuszy stylów musi zostać zdefiniowane połączenie z plikiem zewnętrznym, który je zawiera. Połączenie to definiuje się za pomocą znacznika `<link />`, który umieszczany jest w sekcji `head` dokumentu:

```
<head>
<link rel="stylesheet" type="text/css"
href="mojestyle.css" />
</head>
```

Definicje stylów znajdują się w pliku *mojestyle.css* — nazwa pliku może być dowolna, ale musi mieć rozszerzenie *.css*. Przeglądarka odczyta plik i sformatuje dokument zgodnie z zawartymi w nim instrukcjami.

W znaczniku `<link />` musi znaleźć się atrybut `rel` o wartości `stylesheet`. Dzięki temu przeglądarka będzie posiadać informację o tym, że atrybut `href` podaje jej adres URL pliku, który zawiera arkusz stylów.

ĆWICZENIE

2.4 Definiowanie zewnętrznego arkusza stylów

1. Uruchom edytor Notatnik.



Plik *.css* można utworzyć w dowolnym edytorze.

2. Wpisz w Notatniku zestaw definicji stylów, na przykład taki jak na listingu 2.2. Plik nie powinien zawierać znaczników HTML.

Listing 2.2. *Definicje stylów w pliku styl_test.css*

```
hr {color: sienna}
p {margin-left: 20px}
body {background-image: url("images/obraz.gif")}
```

3. Zapisz plik z rozszerzeniem nazwy *.css* — na przykład nazwij go *styl_test.css*. Styl jest gotowy do użytku.
4. Aby zastosować zewnętrzny arkusz CSS, umieść w elemencie `head` dokumentu, w którym chcesz zastosować style, znacznik `<link />` o takiej postaci:

```
<link rel="stylesheet" href="styl_test.css" type="text/css" />
```

Atrybut `href` zawiera nazwę pliku (lub ścieżkę dostępu do niego), z którego odczytane zostaną deklaracje stylu. Oczywiście do tego samego pliku deklaracji stylu może odwoływać się wiele plików HTML i o to właśnie nam chodziło, gdyż dzięki takiemu prostemu zabiegowi wygląd wszystkich stron będzie jednolity, a ewentualne zmiany deklaracji będą natychmiast obowiązywały na wszystkich stronach WWW.

Grupowanie kodu

Jeśli różne elementy korzystają z tego samego atrybutu o takiej samej wartości — na przykład `h1 {color: red}`, `h2 {color: red}`, `p {color: red}`, możesz zebrać je w jednej definicji stylu.

ĆWICZENIE

2.5 Upraszczanie definicji stylu

Powiedzmy, że w elemencie `style` umieszczono definicje stylu określające identyczny — czerwony — kolor nagłówków i tekstu:

```
<style type="text/css">
h1 {color: red}
h2 {color: red}
p {color: red}
</style>
```

Aby uprościć definicję:

- Utwórz w sekcji `style` grupę elementów, oddzielając je przecinkami. Wszystkim składowym tej grupy przypisany zostanie kolor czerwony:

```
h1, h2, p {color: red}
```

Inną formą grupowania jest zebranie wszystkich deklaracji stylu dla danego elementu. Innymi słowy, zamiast definiować osobno kolor i wyrównanie, można to zrobić w jednej deklaracji.

ĆWICZENIE

2.6 Rozbudowanie definicji stylu

Aby dla danego elementu określić w jednej deklaracji stylu kilka atrybutów stylu:

- W deklaracji stylu zawartej w nawiasach klamrowych {} umieść wszystkie niezbędne definicje. Musisz je rozdzielić średnikami, tak jak zostało to pokazane w poniższym wyrażeniu przykładowym, w którym do tekstu stosowane jest wyrównanie i kolor:

```
p {text-align: center; color: red}
```

Aby definicja była bardziej czytelna, każdą z właściwości możesz umieszczać w osobnym wierszu:

```
p
{
text-align: center;
color: black;
font-family: arial
}
```

Klasy stylu

Zanim przećwiczymy stosowanie zewnętrznego arkusza stylów na jednym z przykładowych dokumentów, należy poznać klasy.

Klasę określa się w taki oto sposób:

```
element.nazwa_klasy{atrybut:wartość [; atrybut:wartość] ...}
```

lub:

```
.nazwa_klasy{atrybut:wartość [; atrybut:wartość] ...}
```



Nazwy klas mogą być dowolne, pamiętaj jednak, by *nie stosować w nich polskich liter*.

W pierszym przykładzie klasa jest powiązana z elementem danego typu, w drugim przypadku klasa jest niezależna od typu elementu.

Powiedzmy, że chcesz w swoim dokumencie zastosować do akapitów dwa różne sposoby wyrównania tekstu: pewne akapity chcesz dosunąć do prawego marginesu, a pozostałe wycentrować. W takim przypadku przydatny będzie atrybut `class`. Pozwala zdefiniować różne style dla tego samego elementu — inaczej mówiąc, pozwala zdefiniować *klasy stylu*.

ĆWICZENIE

2.7 Definiowanie klas stylu

Zdefiniuj dwie klasy: pierwszą, nazwijmy ją `prawy`, w której ustawimy sposób wyrównania tekstu do prawego marginesu, i klasę `center`, w której tekst będzie wyśrodkowany.