

## 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 według Erica Meyera. Sztuka projektowania stron WWW

Autor: Eric A. Meyer

Tłumaczenie: Andrzej Badowski,

Wojciech Grobel, Joanna Sugiero

ISBN: 83-7361-709-4

Tytuł oryginału: [Eric Meyer on CSS.](#)

[Mastering the Language of Web Design](#)

Format: B5, stron: 322



Ewolucja internetu pociągnęła za sobą również ewolucję technologii wykorzystywanych do tworzenia stron WWW. Witryny są wyświetlane nie tylko na monitorach komputerów, ale także na wyświetlaczach telefonów komórkowych i urządzeń przenośnych, drukowane na drukarkach i odczytywane przez urządzenia wykorzystywane przez osoby z wadami wzroku. Powstała więc konieczność ustalenia standardu formatowania stron, który pozwalałby na oddzielenie treści dokumentu od jego wyglądu. Takie właśnie założenie przyświecało twórcom mechanizmów CSS – kaskadowych arkuszy stylów. Wykorzystując je, możemy zdefiniować wygląd i formatowanie tekstu, kolorystykę dokumentów, układ elementów na stronie, a nawet zachowanie się jej składników w reakcji na różne działania użytkownika. Dzięki technologii CSS możemy całkowicie zmienić wygląd strony, modyfikując jeden plik zawierający definicję stylów.

Książka „CSS według Erica Meyera. Sztuka projektowania stron WWW” to szczegółowe omówienie kaskadowych arkuszy stylów, autorstwa jednego z najbardziej cenionych specjalistów w dziedzinie projektowania stron WWW. Autor przedstawia metodologię projektowania witryn WWW z wykorzystaniem stylów na bazie 13 praktycznych projektów. Wykonując je, poznasz wszystkie tajniki kaskadowych arkuszy stylów.

- Konwertowanie istniejących witryn WWW
- Definiowanie stylów dla artykułów prasowych
- Formatowanie hiperłączy i przycisków nawigacyjnych
- Tworzenie stylów na potrzeby drukowania dokumentów
- Modyfikowanie wyglądu formularzy
- Zmiana układu strony
- Pozycjonowanie elementów strony

**Eric A. Meyer** projektuje witryny WWW od 1993 roku. Obecnie pracuje nad wdrażaniem standardów sieciowych dla Netscape Communications; mieszka w Cleveland w stanie Ohio. Koordynował tworzenie dokumentu CSS1 Test Suite przez konsorcjum W3C. Jest również autorem książek CSS. Kaskadowe arkusze stylów. Przewodnik encyklopedyczny (wydawnictwo Helion), CSS2.0 Programmer's Reference (wydawnictwo Osborne/McGraw-Hill), a także zestawień zgodności przeglądarek ze standardami CSS.

Wydawnictwo Helion  
ul. Chopina 6  
44-100 Gliwice  
tel. (32)230-98-63  
e-mail: [helion@helion.pl](mailto:helion@helion.pl)



# SPIS TREŚCI

<b>O Autorze</b>	<b>7</b>
<b>O konsultantach technicznych</b>	<b>8</b>
<b>Przedmowa</b>	<b>9</b>
<b>Wstęp</b>	<b>11</b>
<b>PROJEKT 1. Konwersja istniejącej strony</b>	<b>17</b>
<hr/>	
Cel projektu .....	18
Przygotowania.....	19
Na początek.....	19
Konwersja dokumentu.....	21
Korzyści z naszej pracy .....	43
Przykładowe modyfikacje.....	45
<b>PROJEKT 2. Style na potrzeby artykułów prasowych</b>	<b>47</b>
<hr/>	
Cel projektu .....	47
Przygotowania.....	48
Na początek.....	48
Style w dokumencie.....	50
Modyfikowanie stylów .....	56
Przekształcanie arkusza stylów na zewnętrzny .....	60
Modyfikacje podstawowego projektu .....	62
Przykładowe modyfikacje.....	65
<b>PROJEKT 3. Formatowanie kalendarza wydarzeń</b>	<b>67</b>
<hr/>	
Cel projektu .....	67
Przygotowania.....	68
Na początek.....	68
Formatowanie dokumentu .....	70
Praca z bardziej szczegółowym kalendarzem .....	80
Przykładowe modyfikacje.....	87
<b>PROJEKT 4. Ożywianie hipertączy</b>	<b>89</b>
<hr/>	
Cel projektu .....	89
Przygotowania.....	90
Na początek.....	90
Formatowanie dokumentu .....	91
Przykładowe modyfikacje.....	101

---

**PROJEKT 5. Jak upiększyć menu** **103**


---

Cel projektu .....	104
Przygotowania.....	104
Na początek.....	104
Formatowanie menu.....	106
Zmiana wyglądu menu .....	116
Przykładowe modyfikacje.....	121

---

**PROJEKT 6. Formatowanie wydruków** **123**


---

Cel projektu .....	124
Przygotowania.....	124
Na początek.....	124
Formatowanie dokumentu do druku.....	126
Style drukowania na potrzeby artykułu prasowego.....	134
Formatowanie wydruku kalendarza.....	138
Przykładowe modyfikacje.....	145

---

**PROJEKT 7. Dbamy o wygląd formularzy** **147**


---

Cel projektu .....	147
Przygotowania.....	148
Na początek.....	148
Formatowanie dokumentu .....	150
Style wykorzystywane do druku.....	155
Wykończenie .....	163
Przykładowe modyfikacje.....	165

---

**PROJEKT 8. Tworzenie kartki internetowej z pozdrowieniami** **167**


---

Cel projektu .....	167
Przygotowania.....	168
Na początek.....	168
Stylizujemy dokument.....	169
Nieco inne spojrzenie.....	177
Jeszcze jeden wariant.....	181
Przykładowe modyfikacje.....	183

---

**PROJEKT 9. Układ wielokolumnowy** **185**


---

Cel projektu .....	186
Przygotowania.....	186
Na początek.....	186
Stylizujemy dokument.....	187
Stylizujemy dokument przez pozycjonowanie.....	193
Stylizujemy trzy kolumny.....	197
Wracamy do pływania.....	205
Przykładowe modyfikacje.....	207

<b>PROJEKT 10. Ucieczka z pudełka</b>	<b>209</b>
Cel projektu .....	210
Przygotowania .....	210
Na początek .....	210
Stylizujemy dokument .....	211
Uwaga, zakrety .....	219
Przykładowe modyfikacje .....	229
<b>PROJEKT 11. Pozycjonowanie sposobem ulepszenia projektu</b>	<b>231</b>
Cel projektu .....	232
Przygotowania .....	232
Na początek .....	232
Stylizujemy dokument .....	234
Przykładowe modyfikacje .....	253
<b>PROJEKT 12. Unieruchamiamy tło</b>	<b>255</b>
Cel projektu .....	256
Przygotowania .....	256
Na początek .....	256
Stylizujemy dokument .....	257
Inne pomysły .....	271
<b>PROJEKT 13. Eric Meyer o CSS w CSS</b>	<b>273</b>
Cel projektu .....	274
Przygotowania .....	274
Na początek .....	274
Określenie stylów w dokumencie .....	275
Analiza po ukończeniu projektu .....	295
Przykładowe modyfikacje .....	297
<b>SKOROWIDZ</b>	<b>299</b>

# 8

## TWORZENIE KARTKI INTERNETOWEJ Z POZDROWIENIAMI

*Kiedy dostajesz kartki na Boże Narodzenie, a sam takowych nie wysyłasz, musisz pogodzić się ze świadomością życia w niełasce przez cztery lata, ale potem już wiesz, że ludzie, którzy przysyłają Ci kartki na święta, robią to, by zrobić **Tobie** przyjemność i nie spodziewają się odpowiedzi.*

— QUENTIN CRISP (CYTOWANY PRZEZ JONA WINOKURA)

JĘŚLI CZYTASZ TĘ KSIĄŻKĘ, to jest bardzo prawdopodobne, że spędzasz dużo czasu w internecie. Jest też bardzo prawdopodobne, że znasz sporo ludzi — przyjaciół, kolegów, klientów itd. — którzy również spędzają mnóstwo czasu w sieci. Postępując więc zgodnie z duchem cyfrowego świata, możesz wysłać im własnoręcznie utworzoną kartkę z pozdrowieniami świątecznymi. Może to być bardzo elektryzujące Boże Narodzenie (albo inne dowolne zimowe święto) dla każdego!

### CEL PROJEKTU

Naszym głównym celem jest utworzenie ładnie wyglądającej kartki, którą można potem wysłać do przyjaciół i kolegów. Dokonamy tego za pomocą podstawowych sposobów pozycjonowania w celu ustawienia tekstu ponad obrazkiem, dowiemy się też, jak zapobiec zlewaniu się tekstu z częściami obrazka podobnymi kolorystycznie. Następnie utworzymy inaczej wyglądające kartki, dokonamy tego



przez odmienne rozmieszczenie elementów tekstowych względem głównego obrazka. Na końcu otrzymamy trzy różne układy kartki.

## PRZYGOTOWANIA

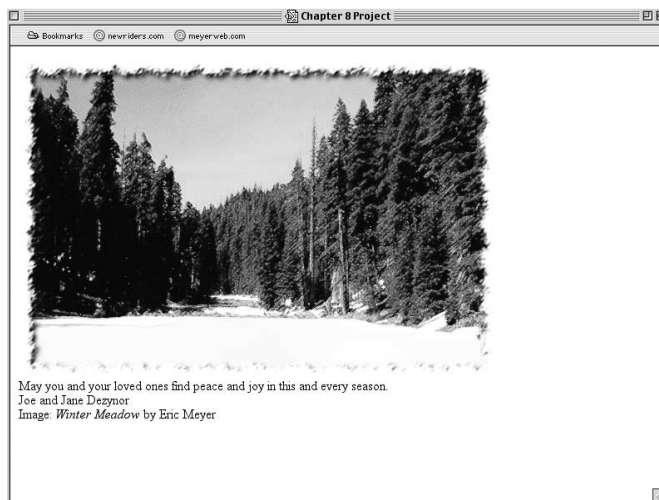
Z serwera FTP należy pobrać pliki do projektu 8. Jeżeli planujesz wprowadzać modyfikacje równoległe z czytaniem książki, otwórz w dowolnym edytorze tekstów plik *ch08proj.html*. Właśnie ten plik będziemy w miarę postępu prac edytować, zapisywać, a następnie odświeżać jego wygląd w przeglądarce.

## NA POCZĄTEK

Potrzebujemy odpowiedniego do pory roku obrazka i życzeń. Na rysunku 8.1 przedstawiono to, od czego musimy zacząć.

**RYСУNEK 8.1.**

Bazowa kartka z pozdrowieniami, jeszcze niezmienną przy użyciu stylów CSS



Aby umożliwić efektywne wykorzystanie stylów, musimy określić strukturę strony. Chcemy zachować maksimum elastyczności, więc każdą linię zawrzemy w osobnym elemencie `div`, a całość z kolei — w nadrzędnym elemencie `div`. W następnym etapie wyrzucimy wszystkie elementy `<br>`, ponieważ już nie będziemy ich potrzebować. Natura blokowa elementów `div` pomoże nam w rozmieszczeniu kolejnych linii tekstu. Na listingu 8.1 pokazano te zmiany.

### LISTING 8.1. Dokument z nowymi elementami `div`

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN"
  "http://www.w3.org/TR/REC-html40/loose.dtd">
<html>
<head>
<title>Chapter 8 Project</title>
</head>
<body>

```

```

<div id="card">

<div id="sentiment">
May you and your loved ones find peace and joy in this and every season.
</div>
<div id="signature">
Joe and Jane Dezynor
</div>
<div id="credit">
Image: <cite>Winter Meadow</cite> by Eric Meyer
</div>
</div>
</body>
</html>

```

Element `<div id="card">` może wydawać się niepotrzebny, siedzi po prostu między znacznikiem body a jego zawartością. Kiedy później zastosujemy nasze style, bez tego elementu o wiele trudniej byłoby powiązać wszystko razem.

## STYLIZUJEMY DOKUMENT

Jedną z najważniejszych rzeczy, które musimy wiedzieć w czasie projektowania tej kartki, jest wielkość bazowego obrazka. W tym przypadku ma on 575 pikseli szerokości i 384 piksele wysokości. Przy projektowaniu stron internetowych szerokość jest zazwyczaj ważniejsza niż wysokość, ponieważ elementy przeważnie są odpowiednio wysokie w stosunku do swojej zawartości. Oznacza to, że autorzy rzadko muszą się martwić o wysokość elementu, a poświęcają o wiele więcej czasu na decydowanie, jak szeroki lub jak wąski ma być element. Wkrótce przekonamy się, że przy rozmieszczeniu wysokość również odgrywa istotną rolę.

## Wyśrodkowanie kartki

Aby nasza elektroniczna kartka była bardziej podobna do zwykłej kartki, powinniśmy jej zawartość wyśrodkować w poziomie.

Najlepszym sposobem, aby to zrobić, jest ustawienie bezpośredniej wartości dla właściwości `width` nadrzędnego elementu `div`, a następnie ustalenie wartości prawego i lewego marginesu na `auto`. Kiedy elementowi blokowemu, takiemu jak `div`, nadaje się bezpośrednią wartość właściwości `width`, a lewy i prawy margines zostają ustawione na `auto`, zostaje im automatycznie nadana identyczna szerokość, co wyśrodkowuje ten element.

```

<style type="text/css" media="screen">
div#card {width: 575px; margin-left: auto; margin-right: auto;}
</style>

```

Problem jest w tym, że przeglądarka Internet Explorer obsługuje tę całkiem prostą operację dopiero od wersji IE6, a nawet w tym przypadku udaje się to tylko wtedy, kiedy pracuje w trybie renderowania „strict”. Na szczęście istnieje sposób, aby zmusić IE5.x/Win do poprawnego wyświetlenia układu, który próbujemy utworzyć.

### Wybór medium

Arkuszy stylów tego projektu zoptymalizujemy pod kątem wyświetlania go na ekranie, ponieważ konstrukcje, które utworzymy, nie nadają się zbyt do drukowania. Projekty oparte na pozycjonowaniu często stwarzają problemy podczas drukowania, a że mamy zamiar umieszczać tekst ponad obrazkami, to drukowanie stanie się jeszcze bardziej problematyczne. W związku z tym, że nasze style będą ograniczone tylko do ekranu, wszelkie wydruki będą „pozbawione stylu”, a tekst pojawi się na nich pod głównym obrazkiem. Zapobiegnie to problemom związanym z elementami nakładającymi się na siebie.

### Renderowanie "strict" i "loose"

Podczas pozycjonowania i wykonywania innych zaawansowanych czynności związanych z CSS warto wiedzieć, w jakim trybie renderowania pracuje przeglądarka. W nowoczesnych przeglądarkach wybór trybu renderowania dokonywany jest za pomocą deklaracji `DOCTYPE`, znajdującej się na początku dokumentu.

Więcej informacji na ten temat można znaleźć w rozdziale „Picking a Rendering Mode” w witrynie WWW tej książki.

W starszych wersjach przeglądarka Internet Explorer przyjmowała, że jeśli właściwości `text-align` nadana była wartość `center`, to i elementy blokowe, takie jak `div`, powinny być wyśrodkowane. Nie jest to sposób, w jaki mają działać style CSS, ale tak działa Internet Explorer. Jeśli więc nadamy styl elementowi `body`, to wyśrodkujemy zewnętrzny element `div` w przeglądarce IE5.x dla Windows i wcześniejszych (nie wspominając o IE6 w trybie luźnego renderowania).

```
<style type="text/css" media="screen">
body {text-align: center;} /* IE5.X workaround */
div#card {width: 575px; margin-left: auto; margin-right: auto;}
```

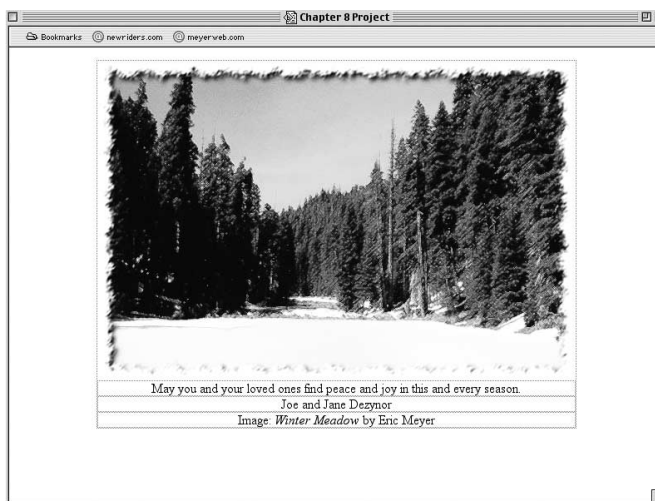
Wykorzystując ten dodatkowy parametr, uniknęliśmy błędów przeglądarki i wyśrodkowaliśmy kartkę. Przy okazji wyśrodkowaliśmy też tekst znajdujący się na kartce, co nie było naszym zamiarem, ale na razie nie będziemy się tym zajmować. Zmienimy to później bez większego problemu.

Aby mieć lepszą kontrolę nad pracą, dodamy teraz do elementów `div` obramowanie. Pozwoli to zobaczyć, gdzie przeglądarka umieszcza te elementy (patrz rysunek 8.2).

```
<style type="text/css" media="screen">
div {border: 1px dotted gray;} /* temporary borders */
body {text-align: center;} /* IE5.X workaround */
```

## RYСУNEK 8.2.

Wyśrodkowana kartka z wyśrodkowanym tekstem i obramowaniem elementów dodanym dla przejrzystości



Wyśrodkowaliśmy więc kartkę i ustawiliśmy pomocne obramowanie elementów `div`. Nie jest to prawdziwe pozycjonowanie — raczej przeformatowanie normalnej zawartości. Aby przejść do prawdziwego pozycjonowania, musimy zbudować fundament.

## Tworzenie kontekstu

Aby pozycjonować element, najpierw musimy utworzyć kontekst. Przecież zawsze pozycjonujemy coś względem czegoś innego. Terminologia CSS określa taki kontekst nazwą **bloku obejmującego**, a z definicji każdy pozycjonowany w sposób absolutny element posiada blok obejmujący. Co składa się na taki blok obejmujący?

### Znaczenie absolutu

Istnieje tendencja do przyjmowania, że element pozycjonowany w sposób absolutny zachowuje się jak ramka i nie da się go przewinąć w obrębie dokumentu.

Tak naprawdę jest to dobra definicja elementu ustalonego (*fixed-position element*).

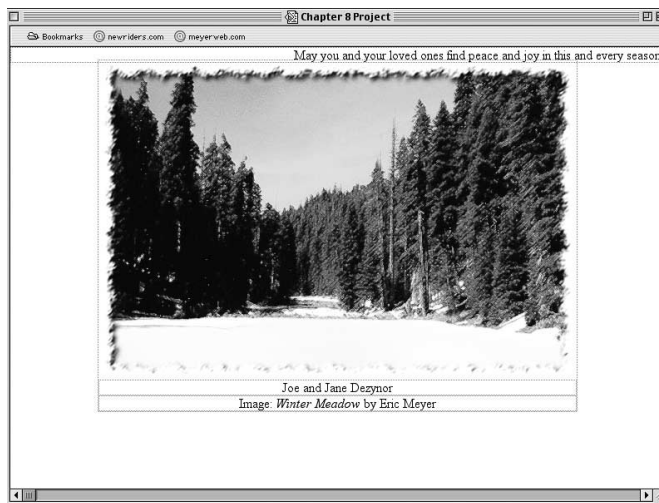
Element pozycjonowany w sposób absolutny umieszczany jest względem swojego bloku obejmującego, więc jeśli da się ten blok obejmujący przewijać razem z dokumentem, tak samo będzie się zachowywał element pozycjonowany w sposób absolutny.



Blok obejmujący każdego pozycjonowanego w sposób absolutny element zdefiniowany jest jako najbliższy element przodek (*ancestor*), który został pozycjonowany. Innymi słowy, jeśli jeden z przodków pozycjonowanego w sposób absolutny elementu został również pozycjonowany w jakiś sposób, to jest to blok obejmujący.

Jeśli nie ma takich pozycjonowanych przodków, to blok obejmujący staje się „elementem podstawowym” (*root element*), który interpretowany jest przez przeglądarkę jako element body lub html. Jeśli po prostu pozycjonowalibyśmy nasze życzenia (blok „sentiment”), to ich blokiem obejmującym byłby element podstawowy (patrz rysunek 8.3).

```
div#card {width: 575px; margin-left: auto; margin-right: auto;}
div#sentiment {position: absolute; top: 0; right: 0; text-align: right;}
</style>
```



Stosując tę zasadę, udało nam się ustawić życzenia w górnym, prawym rogu dokumentu, ale nie kartki. Chcemy jednak, żeby życzenia wyglądały tak, jakby były częścią kartki. Byłoby o wiele wygodniej, aby sama kartka stała się blokiem obejmującym. Na szczęście nie będzie to trudne, ponieważ nasz element znajduje się już na właściwym miejscu. Musimy tylko go spozycjonować

```
body {text-align: center;} /* IE5.X workaround */
div#card {width: 575px; margin-left: auto; margin-right: auto;
position: relative;}
div#sentiment {position: absolute; top: 0; right: 0; text-align: right;}
```

Stosując ten jeden prosty dodatek, sprawiliśmy, że zewnętrzny element `div` stał się blokiem obejmującym (kontekstem pozycjonującym) dla całej zawartości kartki. Zatem podczas pozycjonowania w sposób absolutny jakiegokolwiek elementu `div` w obrębie kartki, a nawet samego obrazka, kontekstem, w którym pozycjonujemy, stanie się element `div` zawierający identyfikator (`id`) `card`. W naszym przypadku życzenia będą teraz pozycjonowane w górnym, prawym rogu kartki, a nie całego dokumentu.

Udaje się to zrobić, ponieważ spozycjonowaliśmy w sposób relatywny zewnętrzny element `div`, ale dokonaliśmy tego bez żadnych przesunięć. Zatem spozycjonowany relatywnie element `div` pozostaje dokładnie tam, gdzie powinien być, jeśli nie zostałyby w ogóle

### RYСУNEK 8.3.

*Pozycjonowanie elementu w obrębie górnego prawego elementu podstawowego*

### Wyłączamy tryb „normal flow”

Zauważ, że po spozycjonowaniu życzeń przestrzeń wokół nich w normalnym układzie dokumentu została „zamknięta”. Spójrz jeszcze raz na rysunek 8.3. Zauważ, że między dolną krawędzią obrazka a elementem `div` napisu „Joe and Jane Dezynor” nie ma wolnego miejsca. Tak właśnie miało być.

spozycjonowany. Wizualnie nie ma żadnej zmiany. Pomimo braku zmiany położenia, ten element `div` staje się blokiem obejmującym dla wszystkich swoich potomków oraz kontekstem, względem którego możemy pozycjonować w sposób absolutny nasz podpis, życzenia itd.

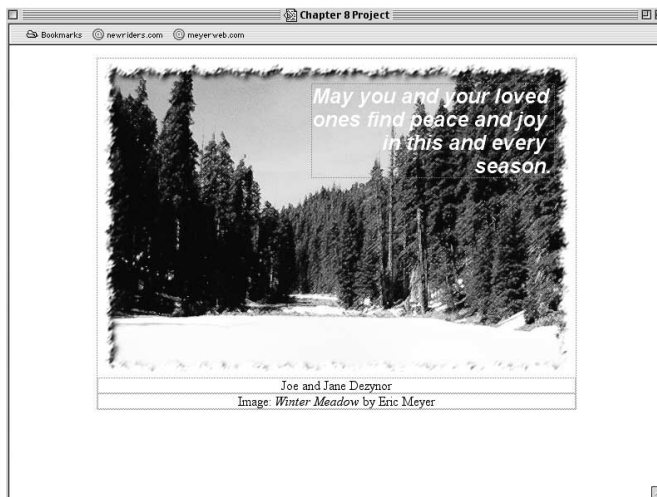
## Zachowanie czytelności

Jeśli zamierzamy umieścić tekst na tle obrazka, musimy zapewnić jego czytelność. Możemy zmienić wielkość liter, ich kolor itd., ale ciągle istnieje niebezpieczeństwo, że kolor i jasność obrazka może być różna, więc część tekstu będzie czytelna, a część nie. Zobaczmy, jakie będą efekty, jeśli przesuniemy tekst nieco w dół i w lewo, wybielimy go i powiększymy (patrz rysunek 8.4).

```
div#card {width: 575px; margin-left: auto; margin-right: auto;
position: relative;}
div#sentiment {position: absolute; top: 30px; right: 25px; width: 50%;
text-align: right; font: italic bold 25px Arial, sans-serif; color: white; }
</style>
```

### RYСУNEK 8.4.

Umieszczenie tekstu na tle obrazka może powodować problemy z kontrastem



### Wielkość czcionki ustawiona w pikselach?

W większości przypadków wielkość czcionek nie powinna być ustawiana w pikselach, ponieważ może to prowadzić do problemów z dostępnością. W tym konkretnym przypadku czysto wizualna natura naszego projektu przemawia za użyciem pikseli. Ale gdybyśmy nawet wykorzystali procenty lub jednostki `em`, nasz projekt nie ucierpiałby na tym.

### Właściwości przesunięcia

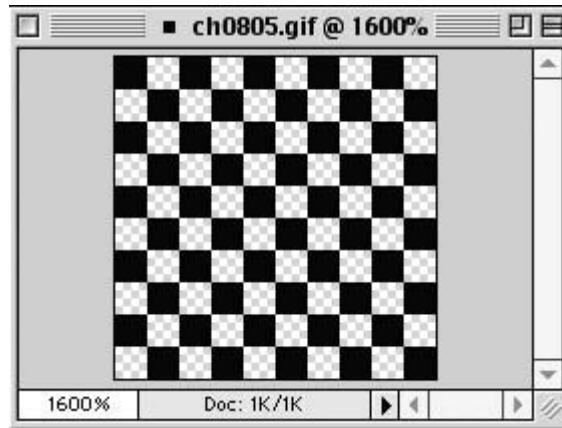
Ponieważ wartości `top`, `right`, `bottom` i `left` określają wielkość przesunięcia między krawędzią pozycjonowanego elementu a jego blokiem obejmującym, nazywane są **właściwościami przesunięcia** (*offset properties*).

Ustawiając właściwość `top` na `30px`, a `right` na `25px`, przesunęliśmy ten element w dół o `30` pikseli i w lewo o `25` pikseli. Zabieg ten przesunie tekst, aby nie zasłaniał obramowania, dekorującego bazowy obrazek. Te dwie właściwości (razem z `bottom` i `left`) definiują przesunięcia względem krawędzi bloku obejmującego. Zatem ustawiliśmy zewnętrzną, górną krawędź życzeń o `30` pikseli poniżej górnej krawędzi bloku obejmującego, a zewnętrzną, prawą krawędź o `25` pikseli w lewo od prawej krawędzi bloku obejmującego.

Mimo że na tym etapie nasz tekst jest całkiem czytelny, to przejście z jasnego do ciemnego tła i z powrotem jest w czasie czytania nieco męczące dla oczu. Musimy wygładzić kontrast między tekstem a obrazkiem.

Aby tego dokonać, musimy dodać do życzeń obrazek tła, ale nie jakiś zwykły, stary obrazek. Wykorzystując trik opracowany przez Todda Fahrnera, doświadczonego projektanta witryn

internetowych i uznanego eksperta w dziedzinie CSS, zastosujemy do naszych życzeń coś, co nazywa się obrazkiem typu **halfscreen**, a wygląda jak szachownica kolorowych i przezroczystych pikseli (patrz rysunek 8.5).

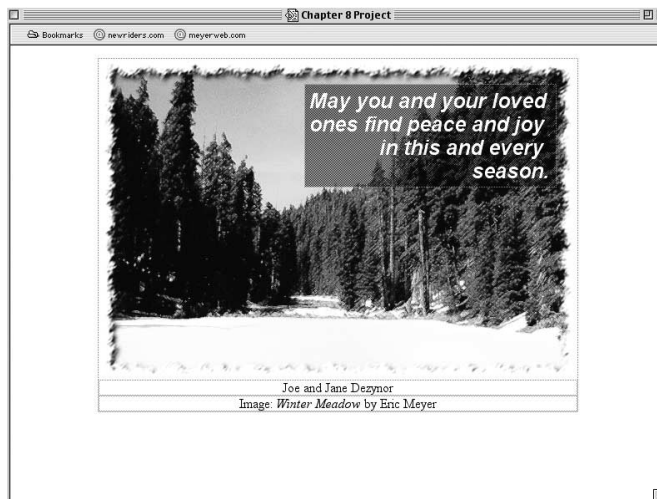


**RYСУNEK 8.5.**

Obrazek tła typu *halfscreen* w powiększeniu 1600%, widziany w programie Photoshop

Każdy czarny piksel widoczny na tym powiększeniu będzie widoczny na ekranie, ale pozostałe piksele pozostaną przezroczyste i pozwolą na wyświetlenie tego, co znajduje się pod nimi. Kiedy do naszych życzeń dodamy ten obrazek tła — razem z odrobiną wypełnienia (*padding*) — otrzymamy czytelniejszy blok tekstu (patrz rysunek 8.6).

```
div#sentiment {position: absolute; top: 30px; right: 25px; width: 50%;
text-align: right; font: italic bold 25px Arial, sans-serif; color: white;
padding: 5px; background: transparent url(halfscreen-black.gif) center repeat;}
</style>
```



**RYСУNEK 8.6.**

Życzenia są teraz o wiele łatwiejsze do przeczytania, a jednocześnie przez nie możemy zobaczyć bazowy obrazek kartki

#### Tymczasowe obramowanie

Pamiętaj, że obramowanie znajdujące się wokół życzeń jest reliktem naszej zasady „pokaż mi wszystkie elementy *div*” i zostanie usunięte w końcowej fazie projektu.

Zauważ, że obrazek typu *halfscreen*, który wykorzystujemy, ma rozmiar 10×10 pikseli. Można zastosować nawet obrazek o wymiarach 2×2, ale ułożenie obok siebie tak małych obrazków tła może znacząco spowolnić pracę przeglądarek w czasie renderowania kartki. Ponieważ czas pobierania obrazka o wymiarach 2×2 jest praktycznie taki sam jak obrazka 10×10 (nawet przez modem), wykorzystujemy większą wersję, aby nieco odciążyć przeglądarki.

## Podpisywanie kartki

Nadszedł czas, aby w kontekście kartki umieścić podpis (blok „signature”). Wyśrodkujemy tekst, napiszemy go czcionką pogrubioną i umieścimy go w dolnej części obrazka. Najłatwiej to zrobić przez ustawienie przesunięcia względem bloku obejmującego i dodanie stylu pogrubionego.

```
div#sentiment {position: absolute; top: 30px; right: 25px; width: 50%;
text-align: right; font: italic bold 25px Arial, sans-serif; color: white;
padding: 5px; background: transparent url(halfscreen-black.gif) center repeat;}
div#signature {position: absolute; top: 335px; left: 0; right: 0; width: 100%;
font-weight: bold;}
</style>
```

Teraz, kiedy spozycjonowaliśmy podpis, okazuje się, że znajduje się on tuż pod obrazkiem. Podpis został umieszczony w taki sposób, że jego górna krawędź jest o 335 pikseli oddalona od górnej krawędzi bloku obejmującego (patrz rysunek 8.7).



### Właściwości left, right, i width

Właściwości left, right i width nadalśmy bezpośrednie wartości w celu uniknięcia błędów Explorera związanych z pozycjonowaniem. Usunięcie jednej lub kilku tych wartości, nawet jeśli da się to zrobić, może prowadzić do przesunięcia względem siebie lub nawet niewyświetlenia przez IE5.X niektórych elementów.

### RYSUNEK 8.7.

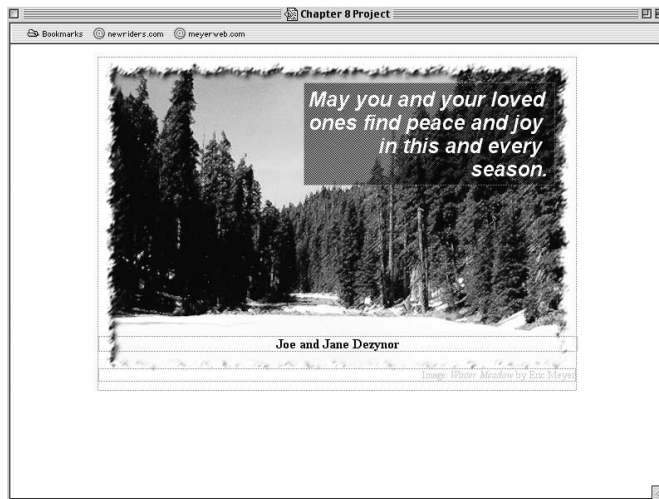
*Podpis został rozmieszczony na podstawie zarówno odziedziczonych, jak i specjalnie określonych stylów*



Tekst jest wyśrodkowany, ponieważ odziedziczona została wartość center właściwości text-align, co było częścią naszego obejścia błędów IE5.X i co jest nawet wygodne w użyciu. Ustawiliśmy wartość szerokości (width) na 100%, ponieważ to gwarantuje, że przeglądarki będą utrzymywały szerokość bloku div równą szerokości całej kartki, a co za tym idzie, umieszczą tekst na środku kartki.

## Stylizowanie informacji o autorze zdjęcia

Ostatnim ważnym elementem, za który musimy się zabrać, jest informacja o autorze zdjęcia (blok „credit”), znajdująca się w dolnej części obrazka. Nie chcemy, żeby zajmowała ona zbyt wiele miejsca na kartce, dlatego zmniejszymy rozmiar czcionki, ustawimy jasnoszary kolor tekstu oraz wyrównamy go do prawej strony. Później podniesiemy go o parę pikseli w górę, aby nakładał się nieco na dolną krawędź obrazka (patrz rysunek 8.8).



RYSUNEK 8.8.

*Informacja o autorze zdjęcia zabiera teraz mniej miejsca i jest mniej widoczna*

```
div#signature {position: absolute; top: 335px; left: 0; right: 0; width: 100%;
font-weight: bold;}
div#credit {color: silver; font-size: 80%; text-align: right; position: relative;
top: -10px;}
</style>
```

Tym razem spozycjonowaliśmy relatywnie element `div` i możemy zobaczyć rezultat — element `div` zawierający informację o autorze zdjęcia został podniesiony. Zauważ jednak, że zostało trochę miejsca między jego dolną krawędzią a dolną krawędzią zewnętrznego elementu `div`. Kiedy element jest pozycjonowany relatywnie, przestrzeń wokół niego **nie zostaje zamknięta**. Zamiast tego dokument zostaje rozmieszczony tak, jakby jego elementy nie były pozycjonowane, a relatywnie pozycjonowany element zostaje podniesiony zgodnie z wartościami właściwości jego przesunięcia (`top`, `left` itd.). Jak widać na rysunku 8.8, oznacza to, że informacje o autorze zdjęcia zostały przesunięte do góry o 10 pikseli.

Wygląda to niebyt ładnie, bo tekst wychodzi poza prawą krawędź obramowania obrazka. Musimy z powrotem przesunąć tekst trochę w lewo. Można to zrobić na trzy sposoby:

- ◆ ustawić wartość właściwości `right` na 25px,
- ◆ ustawić wartość właściwości `margin-right` na 25px,
- ◆ ustawić wartość właściwości `padding-right` na 25px.

W tym konkretnym przypadku wszystkie sposoby dadzą identyczny efekt wizualny. Należy jednak pamiętać, że zwiększenie wartości właściwości wypełnienia powiększy również odstęp — który mógł być ustawiony — między zawartością a krawędziami. Lepiej zatem ustawiać ten rodzaj przesunięcia, stosując albo właściwość `right`, albo `margin-right`. Dla odmiany zastosujemy to drugie rozwiązanie.

```
div#credit {color: silver; font-size: 80%; text-align: right; position: relative;
top: -10px; margin-right: 25px;}
</style>
```

Zabieg ten przesunie zawartość (obramowania też) w prawą stronę, ale pamiętaj, że wartość właściwości `right` wynosi ciągle 0. Marginesy, tak jak wszystko w obrębie elementu, są mierzone względem punktów zdefiniowanych przez właściwości przesunięcia.

## Gotowy produkt

Kiedy wreszcie spozycjonowaliśmy elementy, możemy wreszcie usunąć zasadę „pokaż mi obramowanie”, aby nasza kartka wyglądała tak pięknie, jak zawsze sobie to wyobrażaliśmy (patrz rysunek 8.9).

```
<style type="text/css" media="screen"> /* border-rule deleted */
body {text-align: center;} /* IE5.X workaround */
```

**RYСУNEK 8.9.**

Gotowa cyfrowa kartka  
z pozdrowieniami



Zauważ, że po usunięciu obramowań da się zauważyć pewne przesunięcia w rozmieszczeniu elementów. Dzieje się tak, ponieważ obramowania również zajmują nieco miejsca i przesuwają obszary zawartości trochę dalej od punktów zdefiniowanych przez właściwości przesunięcia. Jest to mało istotny efekt, który nie psuje całego projektu, ale warto o nim pamiętać w bardziej skomplikowanych sytuacjach związanych z pozycjonowaniem.

Za pomocą tych kilku całkiem prostych kroków doszliśmy do arkusza stylów pokazanego na listingu 8.2. Pójdźmy jednak dalej!

### LISTING 8.2. Kompletny arkusz stylów

```
<style type="text/css" media="all">
body {text-align: center;} /* IE5.X workaround */
div#card {width: 575px; margin-left: auto; margin-right: auto;
position: relative;}
div#sentiment {position: absolute; top: 30px; right: 25px; width: 50%;
text-align: right; font: italic bold 25px Arial, sans-serif; color: white;
padding: 5px; background: transparent url(halfscreen-black.gif) center repeat;}
div#signature {position: absolute; top: 335px; left: 0; right: 0; width: 100%;
font-weight: bold;}
```

```
div#credit {color: silver; font-size: 80%; text-align: right; position: relative;
↳top: -10px; margin-right: 25px;}
</style>
```

## NIECO INNE SPOJRZENIE

Przyjmijmy, że po kilku przemyśleniach zdecydowaliśmy, że chropowato wyglądająca krawędź będzie wydawała się naszym odbiorcom trochę zbyt surowa. Może powinniśmy zastosować tu coś nowocześniejszego i kanciastego.

Aby tego dokonać, musimy wrócić do poprzednich znaczników i stylów, jakie przedstawia rysunek 8.2, a następnie utworzyć wokół obrazka czarne obramowanie grubości 2 pikseli. Aby się to udało, musimy dopasować właściwość `width` zewnętrznego elementu `div`, ponieważ obrazek (szerokości 575 pikseli) plus obramowanie (2 piksele z każdej strony) daje razem 579 pikseli szerokości, a nie 575.

```
<style type="text/css" media="screen">
body {text-align: center;} /* IE5.X workaround */
div#card {width: 579px; margin-left: auto; margin-right: auto;
↳position: relative;}
div#card img {border: 2px solid black;}
div#sentiment {position: absolute; top: 30px; right: 25px; width: 50%;
↳text-align: right; font: italic bold 25px Arial, sans-serif; color: white;
↳padding: 5px; background: transparent url(halfscreen-black.gif) center repeat;}
</style>
```

Zauważ, że nie pozbywamy się obejścia błędów IE5 (ciągle chcemy, żeby kartka była wyśrodkowana) oraz stylów życzeń, ale wszystko inne wyrzucamy. Po wprowadzeniu zmian szerokość kartki zostaje dopasowana, a do obrazka dodane obramowanie.

Moglibyśmy zatrzymać resztę stylów z poprzedniej kartki, co pewnie byłoby interesujące, ale zobaczymy, czy potrafimy utworzyć od początku nieco inny wygląd. Patrząc na rysunek 8.10., widzimy trzy rzeczy, które powinny zostać ulepszone:

- ◆ Podpis powinien być umieszczony w interesujący sposób.
- ◆ Trzeba nadać styl informacji o autorze zdjęcia.
- ◆ Życzenia mogłyby znaleźć się bliżej krawędzi obrazka.

Zajmijmy się tymi problemami po kolei.

## Ponowne umieszczanie podpisu

Ponieważ liczy się pomysł, niech nasi odbiorcy wiedzą, kto na niego wpadł. Możemy uzyskać to przez większe niż w poprzednim projekcie uwypuklenie podpisu. Byłoby całkiem sztywnie, gdyby udało się utworzyć efekt w rodzaju wiszącej wizytówki, w którym podpis zostaje umieszczony w ramce, podzielonej na pół przez obramowanie obrazka.



### Nowy obrazek

Zmieniamy także obrazek. Plik `card-image.jpg` został zamieniony na `card-image2.jpg`. Można to sprawdzić w źródle pliku `ch08prog10.html`.

## RYSUNEK 8.10.

Zaczynamy od początku z nieco zmienionym obrazkiem



Najpierw musimy wybrać style czcionki, tła i obramowania podpisu. Jak już to zrobimy (patrz rysunek 8.11), możemy spozycjonować podpis.

```
div#sentiment {position: absolute; top: 30px; right: 25px; width: 50%;
text-align: right; font: italic bold 25px Arial, sans-serif; color: white;
padding: 5px; background: transparent url(halfscreen-black.gif) center repeat;}
div#signature {font-weight: bold; font-size: 18px; border: 1px solid black;
background: silver; width: 275px; margin: 0 auto; padding: 0.25em;}
</style>
```

## RYSUNEK 8.11.

Teraz podpis wygląda bardziej jak wizytówka, ale nie jest jeszcze na swoim miejscu



Kiedy style są już gotowe, musimy zdecydować, jak zamierzamy uzyskać efekt, w którym podpis w połowie nasuwa się na dolną część obrazka. Jednym ze sposobów jest spozycjonowanie relatywne podpisu z przesunięciem ku górze, ale to już robiliśmy z informacjami o autorze zdjęcia w poprzednim projekcie. W takim razie spozycjonujemy go w sposób absolutny względem kartki (która, jak pamiętamy, też jest blokiem obejmującym).



Jest to dobry plan, ale musimy postępować ostrożnie. Możemy spróbować spozycjonowania podpisu, używając ujemnego przesunięcia względem dołu, ale to by go przesunęło względem dołu kartki, a nie dołu bazowego obrazka. Definitywnie chcemy, aby podpis był umieszczony względem obrazka. Na szczęście góra kartki i góra obrazka znajdują się w tym samym miejscu, więc zdefiniujemy przesunięcie względem góry (top).

```
div#signature {font-weight: bold; font-size: 18px; border: 1px solid black;
background: silver; width: 275px; margin: 0 auto; padding: 0.25em;
position: absolute; top: 370px; left: 152px; right: 152px;}
</style>
```

Zauważ, że nadaliśmy wartości właściwościom left i right, co pomaga uniknąć błędów w Explorerze. Wiemy, że obrazek ma 384 piksele wysokości, więc, określając wartość właściwości top na 370px, umieścimy podpis w miejscu, w którym ma się znajdować (patrz rysunek 8.12).



A co się stało z informacją o autorze zdjęcia? Widać tylko sam jej dół, wystający spod podpisu na rysunku 8.12. Wsunęła się pod podpis, ponieważ podpis nie jest już częścią normalnego układu dokumentu. Jeśli nie chcemy całkiem pozbyć się informacji o autorze, musimy ją spozycjonować, aby była widoczna.

## Gdzie umieścić informację o autorze zdjęcia?

Sposób, w jaki nadaliśmy styl informacji o autorze w poprzednim projekcie, nie był zły, ale może umieścić ją na górze obrazka zamiast na dole? A może umieścimy ją tuż nad obrazkiem?

```
div#signature {font-weight: bold; font-size: 18px; border: 1px solid black;
background: silver; width: 275px; margin: 0 auto; padding: 0.25em;
position: absolute; top: 370px; left: 150px; right: 150px;}
div#credit {font: 11pxArial, sans-serif; text-align: right; position: absolute;
top: -13px; left: 0; right: 0; width: 100%; color: #999;}
</style>
```

### RYСУNEK 8.12.

*Wykorzystując przesunięcie względem góry, możemy umieścić podpis bardzo precyzyjnie, ale informacja o autorze zdjęcia zostaje w tym procesie zaciemniona*

Nie ma tu zbyt wielu nowości. Zmieniliśmy kolor tekstu na jasnoszary i wyrównaliśmy go do prawej. Ustawiliśmy wartości właściwości `left`, `right` i `width`, żeby uniknąć kłopotów z Explorerem. Ustawiliśmy rozmiar i rodzaj czcionki. Jediną prawdziwą różnicą jest nadanie właściwości `top` wartości ujemnej, co spowodowało, że element został umieszczony na zewnątrz swojego własnego bloku obejmującego (patrz rysunek 8.13).

**RYСУNEK 8.13.**

Umieszczenie informacji o autorze zdjęcia nad obrazkiem oznacza, że obrazek ten został spozycjonowany na zewnątrz swojego bloku obejmującego



To nie błąd, tylko dokładnie to, co miało się stać. Zasadniczo powiedzieliśmy, że przesunięcie między górą elementu `div`, zawierającego informację o autorze zdjęcia, a górną krawędzią bloku obejmującego powinno wynosić `-13px`. Ponieważ dodatkowo wartości przesunięć powodują dodatnie wartości odstępu między tymi dwoma krawędziami, to jednocześnie powodują one przesunięcie spozycjonowanych elementów w kierunku środka ich bloku obejmującego. Wynika z tego, że ujemne wartości powinny wypychać spozycjonowane elementy na zewnątrz.

#### Inne przesunięcia

W podobny sposób mogliśmybyśmy ustawić informacje o autorze zdjęcia po prawej stronie obrazka, wykorzystując ujemną wartość właściwości `right`.

## Wracamy do życzeń

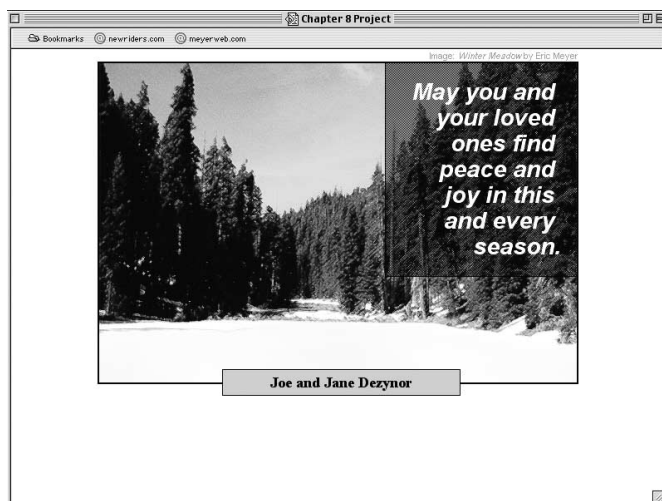
Teraz, kiedy podpis jest bardziej widoczny, musimy coś zrobić, aby poprawić wygląd życzeń. Możemy to osiągnąć, zwiększając rozmiar czcionki elementu, co zrobimy, ale spróbujemy także uwypuklić tło tego elementu w obrębie obramowania obrazka. Będziemy także chcieli zwiększyć wartość wypełnienia elementu z życzeniami. Ta zmiana spowoduje konieczność dodania czarnego obramowania grubości 1 piksela wokół życzeń.

```
div#card img {border: 2px solid black;} div#sentiment {position: absolute; top: 0;
⌘right: 0; width: 33%; text-align: right; font: italic bold 28px Arial,
⌘sans-serif; color: white; padding: 20px; border: 1px solid black;
⌘background: transparent url(halfscreen-black.gif) center repeat;}
```

Równie łatwo mogliśmybyśmy ustawić wartość właściwości `width` życzeń w pikselach, ale bardziej interesujące będzie wykorzystanie procentów. (Jeśli jesteś ambitny, sam oblicz wartość szerokości). Obramowanie pomaga określić region życzeń, nie narażając go na zbytne nakładanie się na inne elementy, oraz zapobiega wystąpieniu efektu chropowatej krawędzi, co poprzednio widzieliśmy przy życzeniach. Modernizm górą!

#### Niezgodność szerokości

Zauważ: mimo że IE5/Mac i Netscape 6.x są zgodne z arkuszami stylów CSS i traktują właściwość `padding` inaczej niż `width` (oraz `height`), to Explorer 4.x, 5.x, i 6.x w trybie luźnego renderowania traktują `padding`, jakby był częścią `width` (oraz `height`) elementu. IE6 w trybie ścisłego renderowania zgadza się z arkuszami stylów CSS.



Wprowadzając te zmiany, doszliśmy do całkiem ciekawego wariantu naszego pierwszego projektu. Na listingu 8.3 pokazano kompletny arkusz stylów tego projektu.

#### LISTING 8.3. Kompletny arkusz stylów wariantu

```
<style type="text/css" media="screen">
body {text-align: center;} /* IE5.x workaround */
div#card {width: 579px; margin-left: auto; margin-right: auto;
position: relative;}
div#card img {border: 2px solid black;}
div#sentiment {position: absolute; top: 0; right: 0; width: 33%;
text-align: right; font: italic bold 28px Arial, sans-serif;
color: white; padding: 20px; background: transparent url(halfscreen-black.gif)
center repeat;}
div#signature {font-weight: bold; font-size: 18px; border: 1px solid black;
background: silver; width: 275px; margin: 0 auto; padding: 0.25em;
position: absolute; top: 370px; left: 150px; right: 150px;}
div#credit {font: 11px Arial, sans-serif; text-align: right; position: absolute;
top: -13px; left: 0; right: 0; width: 100%; color: #999;}
</style>
```

## JESZCZE JEDEN WARIANT

Nie możemy się powstrzymać od jeszcze jednego małego zestawu zmian w projekcie. Będzie to wymagało tylko kilku niewielkich zmian. Najpierw przenieśmy życzenia na lewą stronę i dopasujmy do tego ich style. Będziemy musieli przesunąć je nieco w dół, a także trochę rozszerzyć ramkę z życzeniami.

Zauważ, że w tym wariantcie, usunęliśmy z życzeń czarne obramowanie (patrz rysunek 8.15).

#### RYСУNEK 8.14.

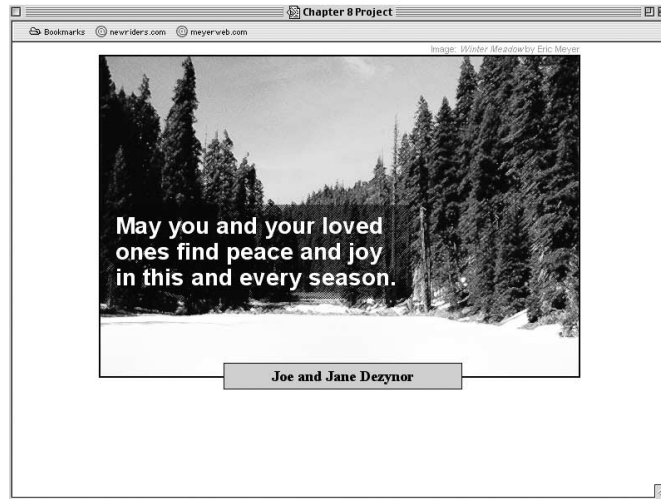
Przesunięcie życzeń w górę względem obramowania obrazka prowadzi do kilku innych zmian



Więcej informacji na temat sposobów obchodzenia problemu niezgodności szerokości można znaleźć w artykule „Tricking Browsers and Hiding Styles”, znajdującym się w witrynie WWW tej książki.

## RYSUNEK 8.15.

*Jeszcze jedna stylizacja życzeń*



```
div#card img {border: 2px solid black;}
div#sentiment { position: absolute; top: 180px; left: 0; width: 350px;
text-align: left; font: bold 28px Arial, sans-serif; color: white;
padding: 10px 5px 10px 20px; background: transparent url(halfscreen-black.gif)
center repeat;}
```

Jak widzisz, przesuwanie spozycjonowanego elementu z jednego miejsca w inne jest proste i wymaga tylko kilku dopasowań stylów. Dodatkową zaletą jest to, że użytkownik będzie nadal mógł doskonale przeczytać tekst, nawet jeśli korzysta ze starszej przeglądarki, która nie obsługuje pozycjonowania.

### Niespozycjonowane pocieszenie

Spróbuj po prostu usunąć style pozycjonujące, aby zobaczyć, jak wygląda teraz kartka — możesz być zaskoczony jej dobrym wyglądem. Symulacja wersji kartki pozbawionej stylów nie wymaga niczego więcej, tylko sprawdzenia rysunku 8.1.

## PRZYKŁADOWE MODYFIKACJE

Jest wiele innych sposobów na stylizowanie formy podobnej do tej, nad którą pracowaliśmy w tym projekcie. Oto kilka sugestii.

1. Spróbuj wyśrodkować życzenia z pierwszego wariantu i nadać im tło w postaci białego obrazka typu halfscreen. Będziesz musiał też zmienić kolor tekstu na jakiś ciemniejszy. Ładnie wyglądałoby też tutaj obramowanie. Może pokusić się o podwójne, grube obramowanie?



2. Przesuń wariant podpisu dosyć daleko w prawą stronę, aby zwiślał z krawędzi obrazka. Powinno to dać Ci dość miejsca, aby umieścić informacje o autorze zdjęcia w dolnym, lewym rogu obrazka, co byłoby wydajniejszym wykorzystaniem miejsca.





- 3.** A teraz dodatkowe wyzwanie: spróbuj wykorzystać szary obrazek typu halfscreen jako tło dla życzeń, biały jako tło dla podpisu, a ciemny z szarym kolorem tekstu na informacje o autorze zdjęcia. Wszystko to umieść na górze obrazka kartki i niech żadne elementy nie nakładają się na siebie. Możesz używać jakiegokolwiek obrazka dla kartki, ale dostaniesz dodatkowe punkty, jeśli wykorzystasz wersję obrazka z chropowatym obramowaniem.