

Maciej Rościszewski



# ZAWÓD FRONT-END DEVELOPER

11 KROKÓW DO ZOSTANIA WEBMASTEREM

Helion 

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz Wydawnictwo HELION dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Redaktor prowadzący: Małgorzata Kulik

Projekt okładki: Studio Gravite / Olsztyn  
Obarek, Pokoński, Pazdrijowski, Zaprucki

Grafika na okładce została wykorzystana za zgodą Shutterstock.com

Wydawnictwo HELION  
ul. Kościuszki 1c, 44-100 GLIWICE  
tel. 32 231 22 19, 32 230 98 63  
e-mail: [helion@helion.pl](mailto:helion@helion.pl)  
WWW: <http://helion.pl> (księgarnia internetowa, katalog książek)

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

<http://helion.pl/user/opinie/twwwwwp>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

ISBN: 978-83-283-3885-2

Copyright © Helion 2019

Printed in Poland.

- [Kup książkę](#)
- [Poleć książkę](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

# Spis treści

<b>Wstęp</b> .....	<b>13</b>
<b>Część I. Wprowadzenie</b> .....	<b>15</b>
<b>Krok 1. Zainstaluj potrzebne programy</b> .....	<b>17</b>
1.1. Przeglądarki internetowe .....	17
1.2. Edytor kodu .....	17
1.3. Prezentacja kodu w książce .....	19
<b>Część II. HTML5 i CSS3</b> .....	<b>21</b>
<b>Krok 2. Poznaj język internetu</b> .....	<b>23</b>
<b>Krok 3. Poznaj HTML5</b> .....	<b>25</b>
3.1. Dokument HTML (pierwsza strona WWW) .....	25
3.1.1. Szkielet dokumentu HTML .....	25
3.1.2. Kodowanie polskich znaków .....	28
3.1.3. Meta tagi .....	30
3.1.4. Podgląd kodu w przeglądarce .....	30
3.1.5. Błędy w kodzie .....	31
3.1.6. Przeglądarka poprawia błędy .....	32
3.2. Nagłówki/treść .....	32
3.2.1. Nagłówek nad tekstem .....	32
3.2.2. Treść na stronie internetowej .....	33
3.2.3. Wyświetlanie znaków specjalnych .....	34
3.2.4. Pogrubienie, pochylenie, podkreślenie tekstu .....	37
3.2.5. Kolejność otwierania i zamykania znaczników .....	38

3.3. Listy uporządkowane oraz nieuporządkowane .....	39
3.3.1. Lista uporządkowana .....	39
3.3.2. Lista nieuporządkowana .....	40
3.3.3. Zagnieżdżanie list .....	40
3.4. Zdjęcia na stronie .....	42
3.4.1. Wstawianie zdjęcia .....	42
3.4.2. Rozmiar zdjęcia .....	43
3.4.3. Tekst alternatywny .....	44
3.5. Znaczniki semantyczne .....	45
3.5.1. nav .....	45
3.5.2. header .....	46
3.5.3. article .....	46
3.5.4. aside .....	47
3.5.5. section .....	47
3.5.6. footer .....	48
3.5.7. Pozostałe znaczniki semantyczne .....	49
3.5.8. Podsumowanie .....	52
3.6. Kontener uniwersalny .....	53
3.7. Tabela .....	54
3.7.1. Tworzenie tabeli .....	54
3.7.2. Podział tabeli .....	55
3.7.3. Tytuł tabeli .....	57
3.7.4. Scalanie komórek .....	57
3.8. Odsyłacze .....	60
3.8.1. Odsyłacz do innej strony .....	60
3.8.2. Odsyłacz do podstrony .....	61
3.8.3. Kotwica .....	64
3.9. Elementy formularza .....	67
3.9.1. Tworzenie formularza .....	67
3.9.2. Pole wieloliniowe .....	68
3.9.3. Pole danych .....	70
3.9.4. Pole hasła .....	71
3.9.5. Pozostałe pola input — wprowadzanie danych .....	72
3.9.6. Pola wielokrotnego wyboru .....	74
3.9.7. Pole jednokrotnego wyboru .....	76
3.9.8. Listy rozwijane .....	77
3.9.9. Przyciski .....	78
3.9.10. Atrybuty elementów formularza .....	79
3.9.11. Przesyłanie danych z formularza .....	83

---

3.10. Multimedia .....	84
3.10.1. Odtwarzacz audio .....	84
3.10.2. Odtwarzacz wideo w HTML5 .....	85
3.10.3. Dodawanie filmu z serwisu YouTube .....	86
3.11. Elementy blokowe i liniowe .....	88
<b>Krok 4. Poznaj CSS3 .....</b>	<b>89</b>
4.1. Styl w dokumencie HTML .....	89
4.1.1. Styl w znacznikach HTML .....	89
4.1.2. Styl w części HEAD .....	90
4.2. Styl w pliku CSS .....	92
4.2.1. Utworzenie i dodanie pliku CSS .....	92
4.2.2. Dodanie stylów do pliku CSS .....	93
4.2.3. Deklaracja stylu według znacznika .....	95
4.2.4. Deklaracja stylu według atrybutu id .....	96
4.2.5. Deklaracja stylu według atrybutu class .....	97
4.2.6. Komentarze w kodzie CSS .....	98
4.2.7. Podgląd kodu w przeglądarce .....	99
4.3. Podstawowe właściwości elementów .....	99
4.3.1. Kolory w CSS .....	99
4.3.2. Obramowanie .....	102
4.3.3. Tło kolorowe .....	106
4.3.4. Tło obrazkowe .....	110
4.3.5. Rozmiar elementu .....	115
4.3.6. Marginesy .....	120
4.3.7. Cień elementu .....	124
4.4. Rozmieszczanie elementów blokowych .....	126
4.4.1. Wyświetlanie .....	126
4.4.2. Sposoby wyświetlania .....	128
4.4.3. Ustawienie .....	129
4.4.4. Rodzaj pozycjonowania .....	130
4.5. Stylizacja tekstu .....	141
4.5.1. Kolor tekstu .....	141
4.5.2. Rozmiar fontu .....	141
4.5.3. Wysokość wiersza .....	142
4.5.4. Odstęp między znakami .....	142
4.5.5. Wyrównanie tekstu .....	143
4.5.6. Dekoracja tekstu .....	143
4.5.7. Waga fontu .....	145
4.5.8. Styl fontu .....	145

4.5.9. Krój pisma .....	146
4.5.10. Fonty niestandardowe .....	147
4.5.11. Fonty niestandardowe od Google .....	148
4.5.12. Cień tekstu .....	151
4.5.13. Stylizacja fragmentu tekstu .....	152
4.6. Stylizacja linków .....	153
4.6.1. Stan linku .....	153
4.6.2. Efekt przejścia .....	155
4.7. Stylizacja list .....	159
4.7.1. Lista uporządkowana .....	159
4.7.2. Lista nieuporządkowana .....	160
4.7.3. Zwijanie tekstu .....	161
4.7.4. Elementy listy w poziomie .....	162
4.7.5. Margines wewnętrzny .....	163
4.8. Styl w wybranych elementach .....	163
4.8.1. Nazwa elementu, klasa, identyfikator .....	163
4.8.2. Grupowanie selektorów .....	164
4.8.3. Selektor potomka .....	164
4.8.4. Selektor dziecka .....	165
4.8.5. Selektor rodzeństwa .....	165
4.8.6. Selektor atrybutu .....	166
4.8.7. Selektor wybranego elementu .....	168
4.9. Stylizacja elementów formularza .....	175
4.9.1. Pole wieloliniowe .....	175
4.9.2. Pola input .....	176
4.9.3. Pole wyboru .....	177
4.9.4. Lista rozwijana .....	185
4.9.5. Przycisk formularza .....	185
4.9.6. Zmiana stylu po atrybucie .....	185
4.9.7. Pseudoklasa focus .....	186
4.10. Animacje w CSS3 .....	187
4.10.1. Klatki animacji .....	187
4.10.2. Nazwa animacji .....	189
4.10.3. Czas trwania animacji .....	189
4.10.4. Tempo animacji .....	190
4.10.5. Powtarzanie animacji .....	191
4.10.6. Opóźnienie animacji .....	191
4.10.7. Zapis skrócony .....	192

4.11. Strona responsywna .....	192
4.11.1. Czym jest strona responsywna? .....	192
4.11.2. Meta viewport .....	194
4.11.3. Zapytania medialne .....	195
4.11.4. Praktyczny przykład media queries .....	198
4.12. Więcej informacji o CSS3 .....	203
4.12.1. Reset stylów przeglądarki .....	203
4.12.2. Nadpisywanie stylów .....	206
4.12.3. Pierwszeństwo stylów .....	206
4.12.4. Dziedziczenie stylów .....	207
4.12.5. Przedrostki przeglądarek .....	209

## **Część III. Budowa strony internetowej ..... 211**

### **Krok 5. Poznaj świat profesjonalistów ..... 213**

5.1. Kto ustala zasady? .....	213
5.2. Przeglądarki internetowe .....	213
5.3. HTML5 i CSS3 .....	213
5.4. Web design .....	214

### **Krok 6. Zbuduj stronę internetową ..... 215**

6.1. Moje zasady .....	215
6.2. Projekt .....	215
6.3. Przygotowanie plików do pracy .....	216
6.4. Top strony .....	217
6.4.1. Co chcemy osiągnąć? .....	217
6.4.2. Utworzenie sekcji strony .....	218
6.4.3. Dodanie nagłówków .....	220
6.4.4. Tworzenie przycisku .....	229
6.4.5. Top strony w wersji mobilnej .....	232
6.5. Menu nawigacyjne .....	239
6.5.1. Co chcemy osiągnąć? .....	239
6.5.2. Utworzenie miejsca na nawigację .....	239
6.5.3. Utworzenie linków dla nawigacji .....	241
6.5.4. Środkowanie nawigacji .....	242
6.5.5. Stylizacja linków .....	244
6.5.6. Menu nawigacyjne w wersji mobilnej .....	249

6.6. Oferta .....	259
6.6.1. Co chcemy osiągnąć? .....	259
6.6.2. Utworzenie sekcji strony .....	260
6.6.3. Środkowanie zawartości strony .....	261
6.6.4. Tytuł działu .....	265
6.6.5. Okienka z ofertą .....	268
6.6.6. Wypełnienie boksów treścią .....	272
6.6.7. Dodanie elementów fontu jako ikon .....	278
6.6.8. Oferta w wersji mobilnej .....	291
6.7. Dietetyk Maria .....	303
6.7.1. Co chcemy osiągnąć? .....	303
6.7.2. Utworzenie sekcji strony .....	303
6.7.3. Zmiana stylu nagłówka oraz podpisu .....	305
6.7.4. Utworzenie miejsca na treść .....	307
6.7.5. Stylizacja treści .....	311
6.7.6. Responsywne obrazy .....	313
6.7.7. „Dietetyk Maria” w wersji mobilnej .....	319
6.8. Mapa Google .....	325
6.8.1. Co chcemy uzyskać? .....	325
6.8.2. Utworzenie sekcji strony .....	325
6.8.3. Klucz API .....	327
6.8.4. Wyświetlenie mapy .....	327
6.8.5. Dodanie markera .....	333
6.8.6. Zmiana stylu mapy .....	336
6.8.7. Mapa Google w wersji mobilnej .....	339
6.9. Formularz kontaktowy .....	339
6.9.1. Co chcemy uzyskać? .....	339
6.9.2. Utworzenie sekcji strony .....	340
6.9.3. Miejsce na pola formularza .....	342
6.9.4. Dodanie pól formularza .....	345
6.9.5. Przycisk formularza .....	353
6.9.6. Naprawa menu górnego .....	357
6.9.7. Formularz kontaktowy w wersji mobilnej .....	359
6.10. Stopka strony .....	363
6.10.1. Co chcemy uzyskać? .....	363
6.10.2. Utworzenie sekcji strony .....	364
6.10.3. Stopka strony w wersji mobilnej .....	366
6.11. Prawie gotowe .....	367



## **Część IV. Dodatkowe możliwości strony internetowej ..... 369**

### **Krok 7. Poznaj JavaScript i jQuery ..... 371**

7.1. Wprowadzenie do języka JavaScript .....	371
7.2. Biblioteki JavaScript .....	372
7.3. Praca z biblioteką jQuery .....	373
7.4. Dodanie biblioteki do strony .....	374
7.5. Działanie menu przy użyciu jQuery .....	376
7.6. Menu w wersji mobilnej .....	384
7.7. Oddzielenie kodu JavaScript od HTML .....	402
7.8. Więcej o języku JavaScript i jQuery .....	404

### **Krok 8. Poznaj PHP7 ..... 405**

8.1. Wprowadzenie do PHP7 .....	405
8.2. Instalowanie serwera XAMPP .....	405
8.3. Utworzenie pliku oraz skryptu PHP .....	409
8.4. Pobieranie danych z formularza .....	411
8.5. Różnica między post a get .....	415
8.6. Przesyłanie danych w tle .....	415
8.6.1. Przygotowanie formularza .....	415
8.6.2. Pobieranie danych za pomocą jQuery .....	419
8.6.3. Walidacja pól za pomocą jQuery .....	420
8.6.4. Wprowadzenie do Ajaxa .....	424
8.6.5. Przygotowanie pliku PHP do odbioru danych .....	431
8.6.6. Usuwanie znaczników HTML .....	435
8.6.7. Przesyłanie znaków specjalnych .....	437
8.6.8. Informowanie użytkownika o przesyłaniu danych .....	438
8.7. Wysyłanie danych z formularza na adres e-mail .....	454

## **Część V. Publikacja strony w internecie ..... 461**

### **Krok 9. Poznaj potrzebne pojęcia ..... 463**

9.1. Domena .....	463
9.2. Serwer .....	463
9.3. Hosting .....	463
9.4. FTP .....	464

<b>Krok 10. Opublikuj stronę w internecie .....</b>	<b>465</b>
<b>Krok 11. Rozpocznij własną przygodę .....</b>	<b>469</b>
<b>Skorowidz .....</b>	<b>471</b>

## Krok 6.

# Zbuduj stronę internetową

## 6.1. Moje zasady

W tym kroku zbudujesz stronę internetową. Stosowany przeze mnie sposób pisania strony internetowej jest moją praktyką, którą wypracowałem przez lata. Pamiętaj, że jest to jedynie sugestia, a nie wytyczne, którymi należy ślepo się kierować. Niektóre z przedstawionych zadań można wykonać na kilka sposobów. Kiedyś spotkałem się w internecie z twierdzeniem: „ilu programistów, tyle sposobów na rozwiązanie problemu”. Jeżeli znajdziesz własne rozwiązanie i będzie ono zgodne ze standardami oraz poprawne technicznie, jak najbardziej możesz je zastosować.

## 6.2. Projekt

Zanim zaczniemy pisać kod strony, musimy wiedzieć, co chcemy stworzyć. Jak wspomniałem wcześniej, tworzona przez nas strona będzie stroną typu *One Page* dla osoby świadczącej usługi dietetyczne. Nie jest to projekt, który można znaleźć w internecie. Na potrzeby książki zaprojektowałem stronę tak, aby zademonstrować w praktyce HTML i CSS oraz ogólne podejście do budowy witryny. Oczywiście niemożliwe jest wykorzystanie wszystkich znaczników czy właściwości CSS w jednej stronie internetowej. Jeszcze trudniej o to na stronie typu *One Page*.

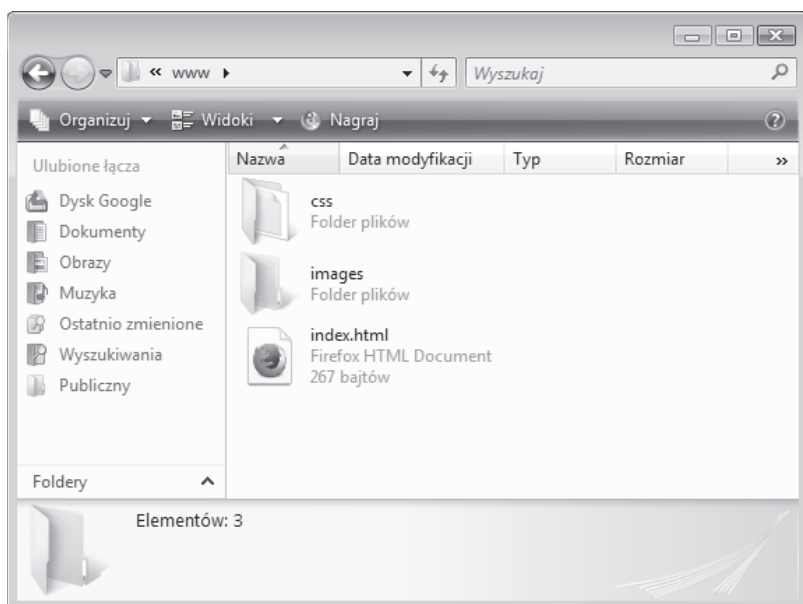
Na początku warto również określić przestrzeń kolorów, jakimi będziemy musieli się posługiwać. Jeżeli firma lub osoba prowadząca działalność posiada logo, to problem z głowy. Wtedy tworzy się stronę pasującą do koloru logo firmy. W naszym projekcie nie ma logo. Zdecydowałem się na wykorzystanie koloru brązowego oraz jego odcieni. W kolorze tym zostaną wykonane dodatki, takie jak ramki czy przyciski. Często w projektach wykorzystuję dwa lub trzy kolory bazowe. Większa ich liczba może spowodować, że strona będzie wyglądać zbyt pstrokato. Tekst pozostawię standardowo czarny lub w odcieniu czerni.

Najważniejszą składową każdej witryny jest treść. Po określeniu typu strony oraz koloru lub kolorów bazowych należy zdecydować, jakie treści będą umieszczone na stronie. W naszym projekcie będą to następujące elementy: oferta diety, informacje o osobie piszącej diety, mapa dojazdowa do gabinetu oraz formularz kontaktowy. Wszystkie elementy należy poukładać w całość. Zazwyczaj mapa dojazdowa oraz formularz kontaktowy znajdują się na samym końcu strony. Oferta zostanie

umieszczona nad informacjami o osobie piszącej diety, a na samym początku stworzymy tzw. top strony. Umieszcza się w nim jasne informacje dotyczące typu prowadzonej działalności, często ze zdjęciem w tle obrazującym wykonywane usługi.

## 6.3. Przygotowanie plików do pracy

Do pracy należy przygotować sobie miejsce, w którym będziemy przechowywać wszystkie pliki strony. Możesz kontynuować pracę w folderze *www*, w którym znajdują się foldery: *css* oraz *images*. W pierwszym umieściliśmy plik *style.css*, w drugim natomiast będziemy przechowywać grafikę. Oprócz wymienionych folderów w folderze *www* znajduje się plik *index.html*. Całość powinna wyglądać jak na rysunku 6.1.



**RYСУNEK 6.1.** Przygotowanie plików do pracy

W pliku *index.html* stwórz pusty szkielet dokumentu HTML jak na listingu 6.1. Mamy już w nim połączenie z plikiem CSS oraz znacznik `<meta>` odpowiedzialny za stronę responsywną.

**LISTING 6.1.** Przygotowanie pliku *index.html* do pracy

```
<!DOCTYPE HTML>
<html lang="pl">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,initial-scale=1.0">
    <title>Moja strona WWW</title>
    <link href="css/style.css" rel="stylesheet">
```

```
</head>
<body>

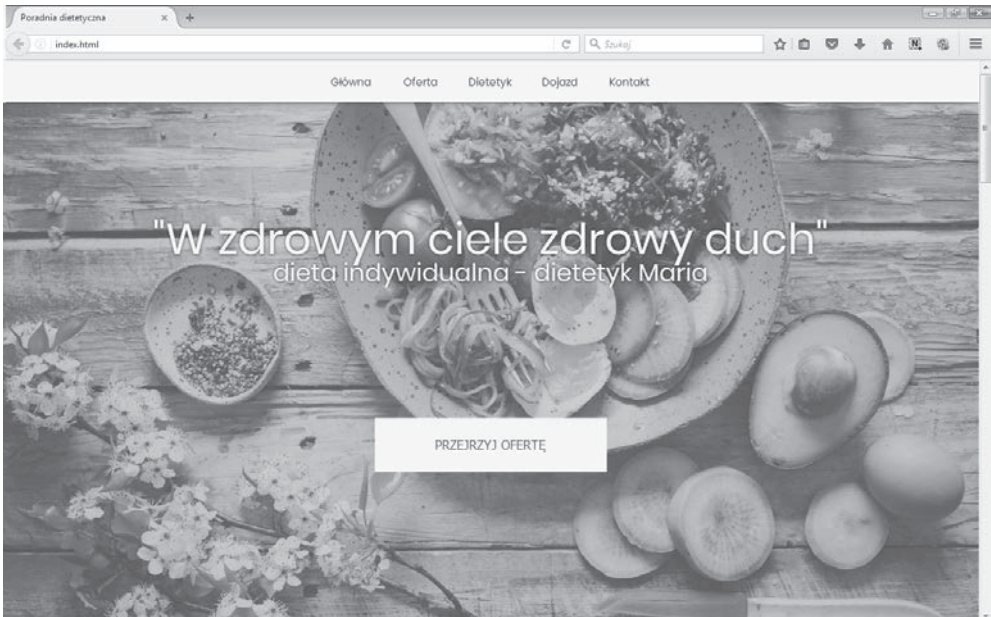
</body>
</html>
```

W pliku *style.css* skasuj wszystkie deklaracje, które w nim są, i zostaw plik całkowicie pusty.

## 6.4. Top strony

### 6.4.1. Co chcemy osiągnąć?

Pierwsze (dobre) wrażenie jest bardzo ważne. Top strony to część witryny, którą użytkownik zobaczy zaraz po wczytaniu strony. Dlatego zawsze staram się, aby ta część przykuła uwagę użytkownika oraz spełniła rolę informacyjną, mówiącą, jakiego rodzaju treści może się on spodziewać. Tym, co najbardziej rzuca się w oczy, jest grafika na całą szerokość i wysokość okna przeglądarki. Następnie na zdjęciu umieszczone zostały dwa teksty. Pierwszy z nich to cytat, a pod nim jest informacja opisująca świadczone usługi. Informację tę dodatkowo obrazuje zdjęcie w tle. Nieco niżej znajduje się przycisk prowadzący do oferty. Menu znajdujące się na samej górze zbudujemy nieco później. Na razie elementów jest niewiele, a efekt jest całkiem dobry. Na rysunku 6.2 pokazuję top strony, którą będziemy budować.



**RYСУNEK 6.2.** Top strony

## 6.4.2. Utworzenie sekcji strony

By utworzyć top strony jak na rysunku 6.2, musimy mieć odpowiednią grafikę. Jeżeli nie dysponujesz odpowiednim zdjęciem, możesz je pobrać z internetu. Jest kilka stron udostępniających darmową, dobrą grafikę. Możesz skorzystać np. ze strony <https://stocksnap.io/>, która ma w swojej bazie wiele zdjęć o różnej tematyce. **Przed ich wykorzystaniem zapoznaj się z licencją, z której dowiesz się, na jakich zasadach możesz ich używać.**

Często po zdobyciu odpowiedniego zdjęcia trzeba je poddać drobnej obróbce w programie graficznym. Na pewno jednym z takich zabiegów będzie zmniejszenie rozmiaru do 1920×1080 px. Obecnie bardzo trudno spotkać wyższą rozdzielczość na komputerze. W przypadku grafiki o wskazanej rozdzielczości można spróbować obniżyć jej jakość, aby uzyskać jak najmniejszą wagę pliku przy zachowaniu ostrości zdjęcia. Dodatkowym niezbędnym zabiegiem graficznym może się okazać lekkie przyciemnienie grafiki w celu uzyskania lepszej czytelności umieszczonego na niej tekstu.

Jeżeli już masz pasującą grafikę, możemy dodać ją do naszej strony. Pamiętaj, aby plik umieścić w folderze *images*. Przed dodaniem grafiki w pliku *index.html* dodajmy znaczniki semantyczne HTML5 `<header></header>`, które służą do umieszczenia treści początkowej na stronie. Więcej o tych znacznikach znajdziesz w podrozdziale 3.5 „Znaczniki semantyczne”. Do znacznika dodaj identyfikator, po którym będziemy odwoływać się do niego w pliku CSS. Plik powinien wyglądać jak na listingu 6.2.

### LISTING 6.2. Dodanie znaczników `<header></header>`

```
<!DOCTYPE HTML>
<html lang="pl">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,initial-scale=1.0">
    <title>Moja strona WWW</title>
    <link href="css/style.css" rel="stylesheet">
  </head>
  <body>
    <header id="top_strony">

    </header>
  </body>
</html>
```

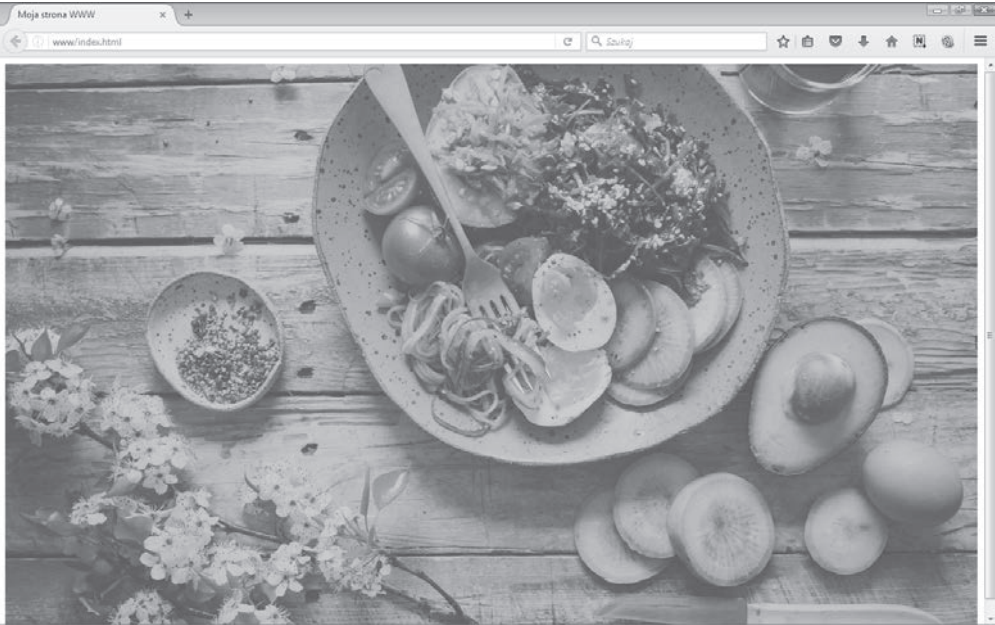
Przygotowaną grafikę musimy dodać jako tło dla elementu o nazwie identyfikatora `#top_strony`. Oprócz dodania grafiki element ten musimy rozciągnąć na całą szerokość i wysokość okna przeglądarki. Do pliku *style.css* dodaj kod z listingu 6.3.

### LISTING 6.3. Kod CSS — rozciągnięcie grafiki na całą szerokość oraz wysokość okna przeglądarki

```
html, body {
  height: 100%;
}
/* TOP STRONY */
#top_strony {
```

```
width: 100%;
height: 100%;
background: url(../images/top_strony.jpg) no-repeat center;
background-size: cover;
}
```

Na początku deklaracji elementom `<html></html>` oraz `<body></body>` nadajemy wysokość o wartości 100%. Inaczej nie uda się nam wyświetlić zdjęcia na całą wysokość okna przeglądarki. Następnie dodajemy odwołanie do elementu `<header></header>` poprzez nazwę identyfikatora. Element ten rozciągamy na całą szerokość oraz wysokość okna przeglądarki. W kolejnej linii kodu dla elementu ustawiamy grafikę jako tło. Skorzystałem z zapisu skróconego. Najpierw podaję ścieżkę do pliku, następnie ograniczam powielanie grafiki, a na samym końcu środkiem obraz. W ostatniej linii nakazujemy przeglądarce proporcjonalne rozciągnięcie obrazu w elemencie. Jeśli grafika jest większa od elementu, zostanie „ucięta”. Całość widać na rysunku 6.3. Więcej o grafice w tle w punkcie 4.3.4 „Tło obrazkowe”.



**RYСУNEK 6.3.** Top strony wypełniony grafiką wstawioną w tle

Pomiędzy jedną a drugą deklaracją wstawiłem komentarz, aby łatwiej było odszukać style odpowiednie dla danego fragmentu strony. Kiedy zawartość pliku CSS się „wydłuży”, łatwiej będzie odnaleźć się w kodzie.

Mimo iż ustawiliśmy wszystko na wartość 100%, wokół grafiki znajduje się białe pole. Jest to domyślny margines elementu `<body></body>`, który należy usunąć. W tym celu wystarczy odwołać się do elementu `<body></body>` i skorzystać z właściwości `margin` o wartości 0 (jak na listingu 6.4). Przypominam, że gdy przypisuje się wartość zero, nie ma konieczności podawania jednostki.

**LISTING 6.4.** Kod CSS — zerowanie marginesów dla `<body></body>`

```
html, body {
  height: 100%;
}
body {
  margin: 0;
}
/* TOP STRONY */
#top_strony {
  width: 100%;
  height: 100%;
  background: url(../images/top_strony.jpg) no-repeat center;
  background-size: cover;
}
```

Zapisz wszystkie pliki i odśwież stronę. Teraz zobaczysz, że grafika rozciąga się od krawędzi do krawędzi okna przeglądarki.

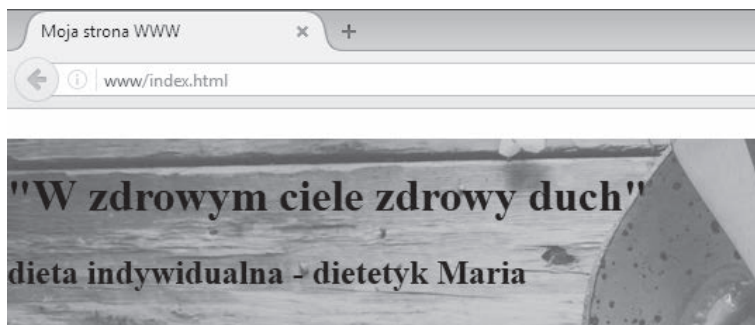
### 6.4.3. Dodanie nagłówków

Mamy gotowe tło, możemy więc umieścić na nim dwa zdania tak jak w przykładzie. Cytat zostanie umieszczony w nagłówku `<h1></h1>`, a informacja o świadczonej usłudze w nagłówku `<h2></h2>` jak na listingu 6.5.

**LISTING 6.5.** Kod HTML — dodanie dwóch akapitów

```
<header id="top_strony">
  <h1>"W zdrowym ciele zdrowy duch"</h1>
  <h2>dieta indywidualna - dietetyk Maria</h2>
</header>
```

Po zapisaniu pliku i odświeżeniu strony nasza grafika w tle nie przylega do górnej części okna przeglądarki, co widać na rysunku 6.4.



**RYSUNEK 6.4.** Nagłówek `<h1></h1>` swoim domyślnym marginesem przesuwca całą zawartość w dół



By to zmienić, dla elementu `<header></header>` należy dodać właściwość `overflow` z wartością `auto` jak na listingu 6.6. Więcej o właściwości `overflow` w podrozdziale 4.3 „Podstawowe właściwości elementów”. Dodatkowo dla tego elementu wyródkujemy tekst tak, aby nasze nagłówki, a w późniejszym czasie przycisk, znalazły się na środku strony.

---

**LISTING 6.6.** *Kod CSS — środkowanie tekstu w topie strony*

---

```
html, body {
    height: 100%;
}
body {
    margin: 0;
}
/* TOP STRONY */
#top_strony {
    width: 100%;
    height: 100%;
    background: url(../images/top_strony.jpg) no-repeat center;
    background-size: cover;
    text-align: center;
    overflow: auto;
}
```

---

Po tych dwóch deklaracjach tło ponownie rozciąga się od krawędzi do krawędzi okna przeglądarki, a tekst zostanie wyródkowany. Na razie nasza treść nie prezentuje się zbyt dobrze. Zabiegiem poprawiającym jakość tekstu będzie zmiana fontu. Użyłem w projekcie fontu Google o nazwie Poppins, wybierając dodatkowo pogrubienie o wartości 700 (standardowe pogrubienie). Krój czcionki jest bardzo ważnym elementem całej strony, gdyż treści będziemy wyświetlać tylko w tym jednym foncie. Jeżeli tak jak ja wybrałeś font Google, to zgodnie z ich dokumentacją w pliku HTML należy podać link do fontu (jak na listingu 6.7).

---

**LISTING 6.7.** *Dodanie linku do czcionki od firmy Google*

---

```
<!DOCTYPE HTML>
<html lang="pl">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,initial-scale=1.0">
    <title>Moja strona WWW</title>
    <link href="https://fonts.googleapis.com/css?family=Poppins:400,700" rel="stylesheet">
    <link href="css/style.css" rel="stylesheet">
  </head>
  <body>
    <header id="top_strony">
      <h1>"W zdrowym ciele zdrowy duch"</h1>
      <h2>dieta indywidualna - dietetyk Maria</h2>
    </header>
  </body>
</html>
```

---

W pliku CSS dla elementu `<body></body>` należy przypisać font zgodnie z zaleceniami Google. Po przypisaniu kroju czcionki do `<body></body>` jak na listingu 6.8 tekst na całej stronie zostanie wyświetlony w tym samym foncie. Szczegółowe informacje na temat dodania fontu od Google w podrozdziale 4.5 „Stylizacja tekstu”.

---

**LISTING 6.8.** *Kod CSS — przypisanie kroju fontu do elementu `<body></body>`*

---

```
html, body {
    height: 100%;
}
body {
    margin: 0;
    font-family: 'Poppins', sans-serif;
}
/* TOP STRONY */
#top_strony {
    width: 100%;
    height: 100%;
    background: url(../images/top_strony.jpg) no-repeat center;
    background-size: cover;
    text-align: center;
    overflow: auto;
}
```

---

Dla całej strony internetowej ustawmy również wielkość fontu na 16px. Obecnie każda przeglądarka wyświetla domyślnie tekst właśnie w takiej wielkości, jednak warto ją zdefiniować w przypadku zmiany domyślnych wartości. Jeżeli chcesz, możesz ustawić dowolną wielkość fontu. Do zmiany rozmiaru służy właściwość `font-size`. Przypisz ją do `<body></body>` z podaną wcześniej wartością jak na listingu 6.9.

---

**LISTING 6.9.** *Kod CSS — ustawienie rozmiaru fontu na stronie*

---

```
html, body {
    height: 100%;
}
body {
    margin: 0;
    font-family: 'Poppins', sans-serif;
    font-size: 16px;
}
/* TOP STRONY */
#top_strony {
    width: 100%;
    height: 100%;
    background: url(../images/top_strony.jpg) no-repeat center;
    background-size: cover;
    text-align: center;
    overflow: auto;
}
```

---

Przypisanie rozmiaru fontu dla całej strony jest istotnym zabiegiem, gdyż następne rozmiary będziemy podawać w jednostkach `em`. Co ma wspólnego rozmiar fontu przypisany do `<body></`

↳body> z jednostką em? Najlepiej będzie to zobrazować na przykładzie. W pliku CSS odwołaj się do nagłówka <h1></h1>, podając nazwę identyfikatora <header></header>, a za nią nazwę znacznika nagłówka stopnia pierwszego. Następnie przypisz właściwość font-size z wartością 3em jak na listingu 6.10.

**LISTING 6.10.** Kod CSS — zmiana rozmiaru czcionki w nagłówku przy użyciu jednostki em

```
html, body {
    height: 100%;
}
body {
    margin: 0;
    font-family: 'Poppins', sans-serif;
    font-size: 16px;
}
/* TOP STRONY */
#top_strony {
    width: 100%;
    height: 100%;
    background: url(../images/top_strony.jpg) no-repeat center;
    background-size: cover;
    text-align: center;
    overflow: auto;
}
#top_strony h1 {
    font-size: 3em;
}
```

Zapisz plik i odśwież stronę. Zobaczysz, że nagłówek znacznie się zwiększył. Teraz zamiast wartości 3em przypisz 48px. Ponownie zapisz plik i odśwież stronę. Zauważysz, że absolutnie nic się nie zmieniło. Stało się tak dlatego, że po przypisaniu do elementu nadrzędnego rozmiaru fontu o wartości 16px jednostka 1em dla elementu podrzędnego automatycznie wynosi 16px. W skrócie  $16px = 1em$ . Dlatego wartość  $3em = 48px$ . Wynika to z prostego mnożenia  $16px \times 3em = 48px$ . Mamy wszędzie zdefiniowaną wielkość fontu w jednostce em, wystarczy więc zmienić wartość właściwości font-size dla elementu <body></body>, aby zmienić rozmiar tekstu na całej stronie.

Powracając do naszego przykładu dla elementu <h1></h1>, przypisz rozmiar 3.5em, a dla nagłówka <h2></h2> ustaw rozmiar na 2em, jak na listingu 6.11.

**LISTING 6.11.** Kod CSS — zmiana rozmiaru czcionki dla nagłówków w topie strony

```
html, body {
    height: 100%;
}
body {
    margin: 0;
    font-family: 'Poppins', sans-serif;
    font-size: 16px;
}
/* TOP STRONY */
#top_strony {
    width: 100%;
```

```
height: 100%;
background: url(../images/top_strony.jpg) no-repeat center;
background-size: cover;
text-align: center;
overflow: auto;
}
#top_strony h1 {
  font-size: 3.5em;
}
#top_strony h2 {
  font-size: 2em;
}
```

---

Teraz zmienimy kolor fontu. Najlepiej go zdefiniować w elemencie `<header></header>`, co spowoduje zmianę koloru tekstu w jego elementach nadrzędnych, czyli w nagłówkach. Dopisz kolor tekstu jak na listingu 6.12.

**LISTING 6.12.** *Kod CSS — zmiana koloru tekstu w topie strony*

---

```
html, body {
  height: 100%;
}
body {
  margin: 0;
  font-family: 'Poppins', sans-serif;
  font-size: 16px;
}
/* TOP STRONY */
#top_strony {
  width: 100%;
  height: 100%;
  background: url(../images/top_strony.jpg) no-repeat center;
  background-size: cover;
  text-align: center;
  color: rgb(245,245,245);
  overflow: auto;
}
#top_strony h1 {
  font-size: 3.5em;
}
#top_strony h2 {
  font-size: 2em;
}
```

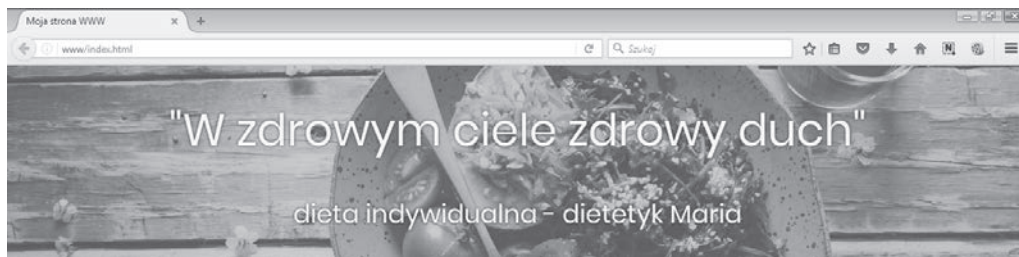
---

Ostatnim zabiegiem przy edycji fontu będzie usunięcie domyślnego pogrubienia (`font-weight`) oraz nadanie delikatnego cienia (`text-shadow`) dla tekstu w obu nagłówkach. W przypadku cienia zastosowałem zapis skrócony, którego pierwsze dwie wartości odpowiadają za położenie cienia, trzecia wartość za jego intensywność, a ostatnia to kolor cienia. Więcej na temat tekstu w podrozdziale 4.5 „Stylizacja tekstu”. Na listingu 6.13 widać opisane zmiany.

**LISTING 6.13.** Kod CSS — usunięcie pogrubienia oraz nadanie cienia dla nagłówków

```
html, body {
    height: 100%;
}
body {
    margin: 0;
    font-family: 'Poppins', sans-serif;
    font-size: 16px;
}
/* TOP STRONY */
#top_strony {
    width: 100%;
    height: 100%;
    background: url(../images/top_strony.jpg) no-repeat center;
    background-size: cover;
    text-align: center;
    color: rgb(245,245,245);
    overflow: auto;
}
#top_strony h1 {
    font-size: 3.5em;
    font-weight: normal;
    text-shadow: 0 0 3px rgb(10,10,10);
}
#top_strony h2 {
    font-size: 2em;
    font-weight: normal;
    text-shadow: 0 0 3px rgb(10,10,10);
}
```

Po wszystkich zmianach całość powinna wyglądać jak na rysunku 6.5.



**RYСУNEK 6.5.** Stylizacja nagłówków w topie strony

Do osiągnięcia zamierzonych efektów brakuje tylko ustawienia nagłówków bliżej siebie oraz odsunięcia ich od górnej krawędzi. Odwołaj się do nagłówków po ich nazwach i dla obu wyzeruj margines wewnętrzny oraz zewnętrzny jak na listingu 6.14.

**LISTING 6.14.** Kod CSS — zerowanie marginesów dla nagłówków

```
html, body {
    height: 100%;
}
```

```
body {
  margin: 0;
  font-family: 'Poppins', sans-serif;
  font-size: 16px;
}
h1, h2 {
  margin: 0;
  padding: 0;
}
/* TOP STRONY */
#top_strony {
  width: 100%;
  height: 100%;
  background: url(../images/top_strony.jpg) no-repeat center;
  background-size: cover;
  text-align: center;
  color: rgb(245,245,245);
  overflow: auto;
}
#top_strony h1 {
  font-size: 3.5em;
  font-weight: normal;
  text-shadow: 0 0 3px rgb(10,10,10);
}
#top_strony h2 {
  font-size: 2em;
  font-weight: normal;
  text-shadow: 0 0 3px rgb(10,10,10);
}
```

---

Deklaracje dla nagłówków dodałem tuż za deklaracją dla `<body></body>`, gdyż zerowanie marginesów będzie dotyczyło wszystkich nagłówków, które zostaną utworzone na stronie. Jeżeli użyłeś stylów resetujących, nie musisz dodawać zerowania marginesów dla nagłówków. Mimo usunięcia marginesów oba teksty nie przylegają do siebie. Tak samo jak tekst w akapicie, tak i nagłówki mają wysokość wiersza. Tylko dla nagłówka `<h2></h2>` do właściwości `line-height` przypisz wartość zero jak na listingu 6.15.

---

**LISTING 6.15.** *Kod CSS — zerowanie wysokości wiersza dla nagłówka `<h2></h2>`*

---

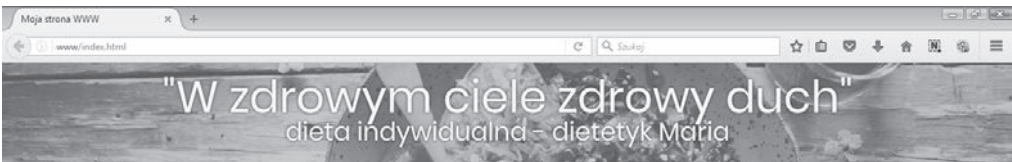
```
html, body {
  height: 100%;
}
body {
  margin: 0;
  font-family: 'Poppins', sans-serif;
  font-size: 16px;
}
h1, h2 {
  margin: 0;
  padding: 0;
}
/* TOP STRONY */
#top_strony {
  width: 100%;
  height: 100%;
```

```

background: url(../images/top_strony.jpg) no-repeat center;
background-size: cover;
text-align: center;
color: rgb(245,245,245);
overflow: auto;
}
#top_strony h1 {
font-size: 3.5em;
font-weight: normal;
text-shadow: 0 0 3px rgb(10,10,10);
}
#top_strony h2 {
line-height: 0;
font-size: 2em;
font-weight: normal;
text-shadow: 0 0 3px rgb(10,10,10);
}

```

Oba akapity wyglądają dokładnie tak samo jak w przykładzie, co widać na rysunku 6.6.



**RYСУNEK 6.6.** Nagłówki wyglądające tak jak w przykładzie

Jedynie, co pozostało, to odsunąć całość od górnej krawędzi okna przeglądarki. W tym celu dla nagłówka `<h1></h1>` wystarczy przypisać margines górny wynoszący 15% (jak na listingu 6.16). Jeżeli chcesz, możesz przypisać inną wartość dla marginesu. Pamiętaj jedynie, że w topie strony będzie znajdować się jeszcze menu oraz przycisk pod nagłówkami. Dlatego margines musi zostać odpowiednio dobrany. Na wielu stronach internetowych tego typu tekst zazwyczaj jest środkowany względem okna przeglądarki. Uważam, że lepiej wygląda, kiedy taki tekst jest wyżej niż w połowie wysokości okna przeglądarki.

**LISTING 6.16.** Kod CSS — odsunięcie nagłówków od górnej krawędzi okna przeglądarki

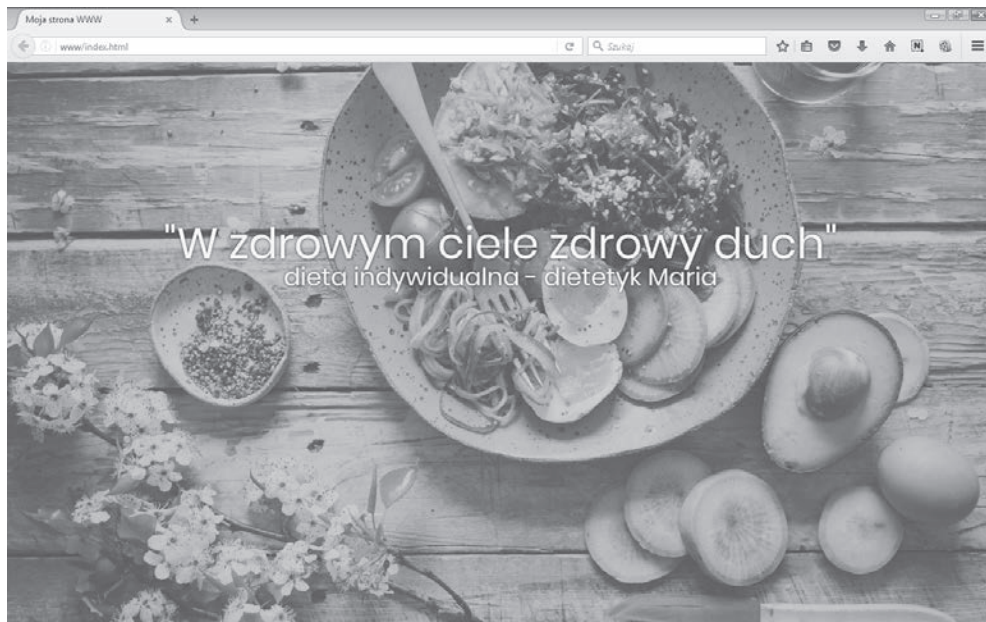
```

html, body {
height: 100%;
}
body {
margin: 0;
font-family: 'Poppins', sans-serif;
font-size: 16px;
}
h1, h2 {
margin: 0;
padding: 0;
}
/* TOP STRONY */
#top_strony {
width: 100%;

```

```
height: 100%;
background: url(../images/top_strony.jpg) no-repeat center;
background-size: cover;
text-align: center;
color: rgb(245,245,245);
overflow: auto;
}
#top_strony h1 {
margin-top: 15%;
font-size: 3.5em;
font-weight: normal;
text-shadow: 0 0 3px rgb(10,10,10);
}
#top_strony h2 {
line-height: 0;
font-size: 2em;
font-weight: normal;
text-shadow: 0 0 3px rgb(10,10,10);
}
```

Wcześniej marginesy zewnętrzne dla obu nagłówków były zerowane. Deklaracje te znajdują się tuż pod deklaracjami dla elementu `<body></body>`. Przypisując górny margines dla elementu `<h1></h1>`, który znajduje się w elemencie `<header></header>`, dokonaliśmy nadpisania stylów. Wcześniej pisałem, że przeglądarka wykonuje plik CSS linijka po linijce. Dlatego zerowy margines górny nagłówka zadeklarowany wcześniej zostanie zastąpiony przez ten, który dodaliśmy przed chwilą. Całość będzie wyglądać jak na rysunku 6.7.



**RYСУNEK 6.7.** Położenie nagłówków na stronie internetowej



Przypominam, że wartości procentowe są naliczane od wielkości elementu nadrzędnego. Elementem nadrzędnym nagłówka jest `<header></header>`, który rozciągamy na całą szerokość i wysokość okna przeglądarki. Dlatego margines górny nagłówka `<h1></h1>` jest liczony względem wielkości okna przeglądarki. Jego wielkość natomiast uzależniona jest od rozdzielczości monitora, na którym użytkownik przegląda stronę.

## 6.4.4. Tworzenie przycisku

Ostatnim elementem brakującym w topie strony jest duży, animowany przycisk. *Button* tworzy się za pomocą znaczników `<button></button>`, które służą do wysyłania danych z formularza. Przycisk, który znajduje się w topie strony, wykonamy z linku za pomocą znaczników `<a></a>`. Po kliknięciu zostaniemy przeniesieni do odpowiedniej sekcji strony. Zawsze będzie możliwość dodania linku do innego pliku HTML czy strony internetowej, jeżeli będzie taka konieczność. W pliku *index.html* pod nagłówkami utwórz znacznik akapitu, wewnątrz niego umieść znaczniki `<a></a>` jak na listingu 6.17.

**LISTING 6.17.** Kod HTML — dodanie linku prowadzącego do oferty

```
<header id="top_strony">
  <h1>"W zdrowym ciele zdrowy duch"</h1>
  <h2>dieta indywidualna - dietetyk Maria</h2>
  <p><a href="#">PRZEJRZYJ OFERTĘ</a></p>
</header>
```

Przypomnę, że link standardowo jest koloru niebieskiego i ma podkreślenie. Taki też wygląd linku znajdującego się pod nagłówkami powinieneś zobaczyć na swojej stronie. Pierwszą deklaracją będzie dodanie górnego marginesu dla akapitu (jak na listingu 6.18). Margines powinien być na tyle duży, aby cały akapit znajdował się nieco poniżej połowy strony.

**LISTING 6.18.** Kod CSS — odsunięcie linku od nagłówków

```
html, body {
  height: 100%;
}
body {
  margin: 0;
  font-family: 'Poppins', sans-serif;
  font-size: 16px;
}
h1, h2 {
  margin: 0;
  padding: 0;
}
/* TOP STRONY */
#top_strony {
  width: 100%;
  height: 100%;
  background: url(../images/top_strony.jpg) no-repeat center;
  background-size: cover;
  text-align: center;
```

```
    color: rgb(245,245,245);
    overflow: auto;
}
#top_strony h1 {
    margin-top: 15%;
    font-size: 3.5em;
    font-weight: normal;
    text-shadow: 0 0 3px rgb(10,10,10);
}
#top_strony h2 {
    line-height: 0;
    font-size: 2em;
    font-weight: normal;
    text-shadow: 0 0 3px rgb(10,10,10);
}
#top_strony p {
    margin-top: 18%;
}
```

---

Zdecydowałem się na nadanie górnego marginesu dla akapitu w celu odsunięcia linku. Możesz usunąć tę deklarację i przypisać margines dolny dla nagłówka `<h2></h2>` o tej samej wartości, aby uzyskać ten sam rezultat. Miej na uwadze, że do uzyskania jakiegoś efektu można wykorzystać kilka sposobów.

Zauważ, że każde odwołanie do elementów wewnątrz `<header></header>` poprzedzam nazwą identyfikatora tego elementu. Można by do znacznika `<p></p>` lub dowolnego nagłówka dodać `id` lub `class` i za pomocą nazwy odwołać się do elementu w pliku CSS. Taka konstrukcja byłaby jak najbardziej prawidłowa. Jednak nadanie `id` dla elementów nietworzących ramy strony lub elementów bardzo istotnych jest mało praktyczne. Elementy w topie strony bardzo często są stylizowane osobno i dokładnie ten sam styl nie jest już nigdzie powtarzany na stronie. Dlatego identyfikacja z wykorzystaniem selektora potomka jest najbardziej praktycznym rozwiązaniem.

Jeżeli chcemy, aby link wyglądał jak przycisk, musimy z linku usunąć domyślne podkreślenie, zmienić kolor tekstu oraz dodać tło. Korzystając z marginesu wewnętrznego, zwiększymy zakres aktywności linku, co ostatecznie sprawi, że link będzie wyglądał jak przycisk. Tak jak poprzednio, i tym razem odwołamy się do elementu `<a></a>`, korzystając z selektora potomka, co widać na listingu 6.19.

---

**LISTING 6.19.** *Kod CSS — stylizacja linku, która upodobni go do przycisku*

---

```
html, body {
    height: 100%;
}
body {
    margin: 0;
    font-family: 'Poppins', sans-serif;
    font-size: 16px;
}
h1, h2 {
    margin: 0;
    padding: 0;
}
```

```

/* TOP STRONY */
#top_strony {
    width: 100%;
    height: 100%;
    background: url(../images/top_strony.jpg) no-repeat center;
    background-size: cover;
    text-align: center;
    color: rgb(245,245,245);
    overflow: auto;
}
#top_strony h1 {
    margin-top: 15%;
    font-size: 3.5em;
    font-weight: normal;
    text-shadow: 0 0 3px rgb(10,10,10);
}
#top_strony h2 {
    line-height: 0;
    font-size: 2em;
    font-weight: normal;
    text-shadow: 0 0 3px rgb(10,10,10);
}
#top_strony p {
    margin-top: 18%;
}
#top_strony p a {
    padding: 25px 80px;
    background: rgb(245,245,245);
    color: rgb(83,62,26);
    text-decoration: none;
}

```

Na początku za pomocą właściwości `padding` zwiększamy obszar działania linku. Skorzystałem z zapisu skróconego dla marginesu wewnętrznego, którego pierwsza wartość odpowiada za margines górny i dolny, a wartość druga za margines prawy i lewy. Następnie zdefiniowane zostaje tło i kolor tekstu. Na samym końcu pozbywamy się domyślnego podkreślenia linku. Całość powinna wyglądać jak na rysunku 6.8.

Za pomocą pseudoklasy `hover` nadamy aktywność dla przycisku jak na listingu 6.20. Po umieszczeniu kursora na przycisku kolor tła zmieni się na kolor tekstu, a kolor tekstu na kolor tła. Dla uzyskania płynnego przejścia kolorów użyjemy właściwości `transition`.

**LISTING 6.20.** *Kod CSS — nadanie aktywności przyciskowi po umieszczeniu na nim kursora*

```

html, body {
    height: 100%;
}
body {
    margin: 0;
    font-family: 'Poppins', sans-serif;
    font-size: 16px;
}

```

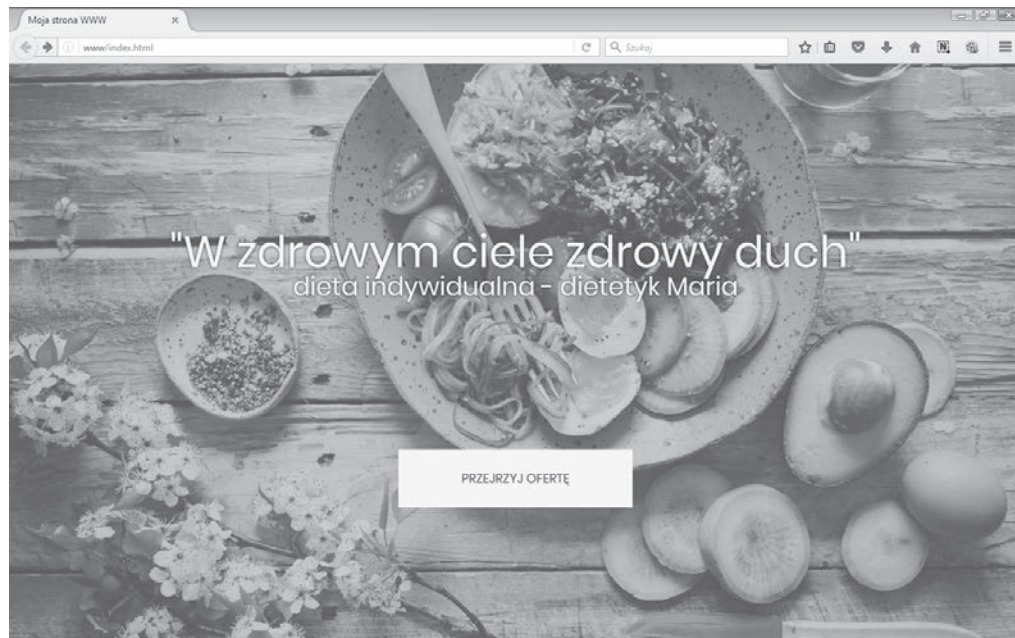
```
h1, h2 {
    margin: 0;
    padding: 0;
}
/* TOP STRONY */
#top_strony {
    width: 100%;
    height: 100%;
    background: url(../images/top_strony.jpg) no-repeat center;
    background-size: cover;
    text-align: center;
    color: rgb(245,245,245);
    overflow: auto;
}
#top_strony h1 {
    margin-top: 15%;
    font-size: 3.5em;
    font-weight: normal;
    text-shadow: 0 0 3px rgb(10,10,10);
}
#top_strony h2 {
    line-height: 0;
    font-size: 2em;
    font-weight: normal;
    text-shadow: 0 0 3px rgb(10,10,10);
}
#top_strony p {
    margin-top: 18%;
}
#top_strony p a {
    padding: 25px 80px;
    background: rgb(245,245,245);
    color: rgb(83,62,26);
    text-decoration: none;
    transition: background .9s, color .9s;
}
#top_strony p a:hover {
    background: rgb(83,62,26);
    color: rgb(245,245,245);
}
```

---

Do wcześniejszej deklaracji przypisaliśmy efekt przejścia dla background oraz color, ustawiając czas przejścia na niecałą sekundę. Kolejne deklaracje z pseudoklasą hover informują przeglądarkę, jakie kolory mają zostać wyświetlone dla tła oraz tekstu przycisku.

## 6.4.5. Top strony w wersji mobilnej

Mając na uwadze, że coraz więcej osób przegląda stronę na urządzeniach mobilnych, musimy przygotować stronę na mniejszą rozdzielczość. Zmniejsz okno przeglądarki lub przejdź do narzędzia *Widok trybu responsywnego*, które jest dostępne w przeglądarce Firefox. Zobacz, jak wygląda strona na ekranie o rozdzielczości 360×640 px. Dlaczego akurat taka rozdzielczość? Według strony <http://www ranking.pl> jest to najczęściej wykorzystywana rozdzielczość do przeglądania internetu na



**RYSUNEK 6.8.** Wygląd strony po stylizacji linku

urządzeniach mobilnych. Wcześniej mówiłem, że progi do zapytań medialnych powinny być uzależnione od układania się elementów oraz czytelności treści. Jednak rozdzielczość tę możemy potraktować jako punkt wyjścia i sprawdzić, jak wygląda nasza strona. Na rysunku 6.9 pokazano widok obecnej strony w rozdzielczości 340×640 px. Widok został ustawiony w przeglądarce Firefox przy użyciu narzędzia *Widok trybu responsywnego*.

Na podstawie symulacji strony na ekranie o mniejszej rozdzielczości możemy zrobić listę elementów do poprawy i koniecznych zmian:

- zmniejszenie czcionki dla obu nagłówków;
- optymalne ustawienie wysokości wiersza dla nagłówka `<h2></h2>`;
- obniżenie nagłówków;
- uzyskanie większego odstępu między nagłówkiem a przyciskiem;
- zmniejszenie wymiarów przycisku.

Dodatkowym zabiegiem będzie ustawienie na sztywno wysokości topu strony. Zabieg ten pozwala na uniknięcie kłopotu z przeskakiwaniem tła w trakcie przewijania strony na urządzeniu mobilnym. Dodatkowo możemy przygotować mniejszy rozmiar grafiki, aby zaoszczędzić transfer internetu mobilnego, co jednocześnie spowoduje szybsze wczytanie obrazu. Otwórz grafikę w programie graficznym i zmniejsz obraz, aby jego wysokość była równa 600px. Plik zapisz pod nazwą *top\_strony\_mobilne.jpg* w folderze *images*.



**RYSUNEK 6.9.** Układ elementów strony w symulacji rozdzielczości na urządzeniu mobilnym

Dla stylów mobilnych utwórz osobny plik CSS i podpisz go *style\_mobile.css*. Plik ten powinien znajdować się w tym samym folderze co plik *style.css*. Po utworzeniu pliku CSS musimy odwołać się do niego w pliku *index.html* jak na listingu 6.21.

**LISTING 6.21.** Dodanie drugiego arkusza stylów do pliku *index.html*

```
<!DOCTYPE HTML>
<html lang="pl">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,initial-scale=1.0">
    <title>Moja strona WWW</title>
    <link href="https://fonts.googleapis.com/css?family=Poppins:400,700" rel="stylesheet">
    <link href="css/style.css" rel="stylesheet">
    <link href="css/style_mobile.css" rel="stylesheet">
  </head>
</body>
```

```
<header id="top_strony">
  <h1>"W zdrowym ciele zdrowy duch"</h1>
  <h2>dieta indywidualna - dietetyk Maria</h2>
  <p><a href="#">PRZEJRZYJ OFERTĘ</a></p>
</header>
</body>
</html>
```

Arkusze stylów, w którym będziemy umieszczać style dla urządzeń mobilnych, został dodany pod arkuszem stylów, który już stworzyliśmy. W przypadku zamiany kolejności przeglądarka nie byłaby w stanie nadpisać deklaracji stylów. Otwórz plik *style\_mobile.css* i wprowadź pierwsze zapytanie medialne dla rozdzielczości 768px jak na listingu 6.22. Urządzenia poniżej tej rozdzielczości można traktować jako urządzenia mobilne. Więcej o zapytaniach medialnych w podrozdziale 4.11 „Strona responsywna”.

**LISTING 6.22.** *Kod CSS Mobile — utworzenie pierwszego zapytania medialnego*

```
@media screen and (max-width: 768px) {
}
```



Listing zaczynający się od „Kod CSS Mobile” umieszczamy w pliku *style\_mobile.css*, a kod zaczynający się od „Kod CSS” w pliku *style.css*.

Osobny plik CSS do zapytań medialnych pozwoli nam na utrzymaniu porządku w kodzie i szybsze odszukanie deklaracji, które będziemy chcieli zmienić. Pamiętaj, że plik CSS z zapytaniami medialnymi musi znajdować się poniżej podstawowego pliku CSS. Jeżeli tworzenie osobnych deklaracji dla urządzeń mobilnych jest dla Ciebie nadal niezrozumiałe, cofnij się do podrozdziału 4.11 „Strona responsywna”.

Wewnątrz nawiasów klamrowych umieścimy deklaracje umożliwiające podmianę grafiki dla elementu `<header></header>`, ustawimy stałą wysokość dla tego elementu oraz zmniejszymy rozmiar fontu dla nagłówków. Dodatkowo dla nagłówka `<h2></h2>` musimy zwiększyć wysokość wiersza, aby tekst nie nakładał się na siebie. Kod w pliku *style\_mobile.css* powinien wyglądać jak na listingu 6.23.

**LISTING 6.23.** *Kod CSS Mobile — deklaracje mobilne dla tła topu strony oraz nagłówków*

```
@media screen and (max-width: 768px) {
  #top_strony {
    height: 600px;
    background: url(../images/top_strony_mobilne.jpg) no-repeat center;
  }
  #top_strony h1 {
    font-size: 2.5em;
  }
  #top_strony h2 {
```

```
font-size: 1.5em;  
line-height: 30px;  
}  
}
```

Zapisz wszystkie pliki i odśwież stronę w rozdzielczości 360×600 px. Teraz wygląda to znacznie lepiej, co widać na rysunku 6.10.



**RYSUNEK 6.10.** Wygląd strony po wprowadzeniu zapytania medialnego dla urządzeń mobilnych

Strona na urządzeniu mobilnym wygląda już dobrze. Aby upewnić się, że dopasowuje się do każdej rozdzielczości, zrób test polegający na zwiężaniu i rozszerzaniu okna przeglądarki. Według mnie elementy strony układają się dobrze niezależnie od jej szerokości. Prawdziwym testem będzie sprawdzenie strony na urządzeniu mobilnym. Jeżeli masz już serwer, możesz na nim przetestować wyniki swojej pracy. Drugim sposobem będzie zgranie wszystkiego na telefon lub tablet i uruchomienie pliku *index.html*. Zrzut z ekranu smartfona na rysunku 6.11.





**RYSUNEK 6.11** Zrzut ekranu mojego urządzenia mobilnego

Porównaj wygląd elementów na ekranie o rozdzielczości symulowanej przez przeglądarkę Firefox oraz z urządzenia mobilnego — wyniki są do siebie bardzo podobne. Pozostały jeszcze trzy kwestie do poprawienia z naszej listy. Musimy nadać większy margines górny dla nagłówka `<h1></h1>` i akapitu, w którym znajduje się przycisk. Następnie zmniejszyć wymiar przycisku, aby prezentował się bardziej estetycznie. Wartości, jakie można przypisać do poszczególnych właściwości, na listingu 6.24.

**LISTING 6.24.** *Kod CSS Mobile — poprawienie układu elementów w wersji mobilnej*

```
@media screen and (max-width: 768px) {  
  #top_strony {  
    height: 600px;  
    background: url(../images/top_strony_mobilne.jpg) no-repeat center;  
  }  
  #top_strony h1 {  
    margin-top: 25%;  
    font-size: 2.5em;  
  }  
}
```

```
#top_strony h2 {
  font-size: 1.5em;
  line-height: 30px;
}
#top_strony p {
  margin-top: 35%;
}
#top_strony p a {
  padding: 20px 60px;
}
}
```

Zmiany, których dokonaliśmy, są zabiegami kosmetycznymi. Zwiększyliśmy tylko dwa marginesy oraz zmniejszyliśmy obszar przycisku, aby przycisk lepiej prezentował się na mniejszym ekranie. Jak widać na listingu 6.24, jeśli tworzy się stronę z myślą o działaniu na urządzeniu mobilnym, wystarczy wprowadzić kilka zmian w zapytaniu medialnym w stosunku do podstawowych właściwości. Strona po delikatnych korektach prezentuje się znacznie lepiej, co widać na rysunku 6.12.



**RYСУNEK 6.12.** Gotowy top strony na urządzeniu mobilnym

# Skorowidz

## A

- Ajax, 424
- animacje, 187
  - czas trwania, 189
  - klatki, 187
  - nazwa, 189
  - opóźnienie, 191
  - tempo, 190
  - zapis skrócony, 192
- atribut
  - checked, 80
  - class, 97
  - colspan, 57
  - controls, 84
  - disabled, 82
  - id, 66, 96
  - maxlength, 79
  - placeholder, 80
  - required, 83
  - rowspan, 59
  - selected, 81
  - src, 84
  - style, 89
  - type, 74
  - value, 82
- atributy
  - elementów formularza, 79
  - zawierające dowolny tekst, 167
  - zawierające myślnik, 168
  - zawierające określony wyraz, 167

## B

- biblioteka jQuery, 373, 404
- biblioteki JavaScript, 372
- błędy, 31
- boksy z treścią, 272

## C

- cień
  - elementu, 124
  - tekstu, 151
- CSS3, 213

## D

- data i czas, 74
- deklaracja stylu
  - według atrybutu class, 97
  - według atrybutu id, 96
  - według znacznika, 95
- dekoracja tekstu, 143
- dodanie
  - biblioteki jQuery, 374
  - deklaracji języka, 29
  - filmu, 86
  - stylów do pliku, 93
- dokument HTML, 25
- domena, 463
- działanie menu, 376
- dziedziczenie stylów, 207

## E

- edytor kodu, 17
- efekt przejścia, 155, 187
- elementy
  - blokowe, 88, 126
  - cień, 124
  - formularza, 67
    - stylizacja, 175
  - liniowe, 88
  - listy w poziomie, 162
  - rozmiar, 115
  - sposoby wyświetlania, 128

## elementy

- ustawienie, 129
- właściwości, 99
- wyświetlanie, 126

email, 72

**F**

flash, 84

fonty, 145, 278

- niestandardowe, 147
- niestandardowe od Google, 148

formularz, 67, 339

- atrybuty elementów, 79
- lista rozwijana, 77, 185
- pobieranie danych, 411
- pola, 342, 345
  - danych, 70
  - hasła, 71
  - input, 176
  - jednokrotnego wyboru, 76
  - wielokrotnego wyboru, 74
  - wieloliniowe, 68, 175
  - wyboru, 177

- przesyłanie danych, 83
- przyciski, 78, 185
- pseudoklasa focus, 186
- tworzenie, 67
- w wersji mobilnej, 359
- wprowadzanie danych, 72

FTP, 464

**G**

gradient

- liniowy, 107
- radialny, 109

grafika jako tło, 110, 111

grupowanie selektorów, 164

**H**

hosting, 463

HTML5, 213

**I**

identyfikator, 163

ikony, 278

informacja o przesyłaniu danych, 438

instalowanie serwera XAMPP, 405

**J**

JavaScript, 371, 404

jQuery, 373, 404

- działanie menu, 376
- pobieranie danych, 419
- walidacja pól, 420

**K**

klasa, 163

klatki animacji, 187

klucz API, 327

kod

- HTML, 402
- JavaScript, 402

kodowanie polskich znaków, 28

kolor tekstu, 141

kolory, 99

- format dziesiętny RGB, 100
- format szesnastkowy RGB, 100

komentarze, 98

konfigurator stylów, 19

kontener uniwersalny, 53

kotwica, 64

krój pisma, 146

**L**

linki

- dla nawigacji, 241
- stylizacja, 153, 244

listy

- nieuporządkowane, 40, 160
- rozwijane, 77, 185
- stylizacja, 159
- uporządkowane, 39, 41, 159
- wyboru, 79
- zagnieżdżone, 40

**M**

mapa Google, 325  
 w wersji mobilnej, 339  
 margines, 120  
 wewnętrzny, 123, 163  
 zewnętrzny, 120  
 marker, 333  
 media queries, 195, 198  
 menu  
 górne, 357  
 nawigacyjne, 239, 246  
 użycie jQuery, 376  
 w wersji mobilnej, 249, 384  
 meta  
 tagi, 30  
 viewport, 194  
 metoda  
 get, 415  
 post, 415  
 multimedia, 84

**N**

nadpisywanie stylów, 206  
 nagłówki, 220  
 nad tekstem, 32  
 nawigacja, 239  
 nazwa  
 atrybutu, 166  
 elementu, 163  
 Notepad++, 18

**O**

obramowanie, 102  
 dowolnej krawędzi, 104  
 szerokość, 103  
 zaokrąglone narożniki, 105  
 zapis skrócony, 104  
 zmiana grubości, 103  
 obrazy responsywne, 313  
 oddzielenie kodu, 402  
 odstępy między  
 akapitami, 34  
 znakami, 142  
 odsyłacz do strony, 60, 61  
 odtwarzacz  
 audio, 84  
 wideo, 85

oferta, 259  
 w wersji mobilnej, 291

**P**

PHP7, 405  
 pobieranie danych, 411, 431  
 przesyłanie danych, 415  
 wysyłanie danych na adres e-mail, 454  
 pierwsza strona WWW, 25  
 pierwszeństwo stylów, 206  
 pliki  
 CSS, 92  
 dodanie stylów, 93  
 komentarze, 98  
 podgląd kodu, 99  
 tworzenie, 92  
 PHP, 409  
 pochylenie, 37  
 podgląd kodu, 30, 99  
 podkreślenie, 37  
 pogrubienie, 37  
 pole  
 danych, 70  
 formularza, 342  
 hasła, 71  
 input, 176  
 jednokrotnego wyboru, 76  
 typu email, 72  
 typu number, 72  
 typu range, 74  
 typu tel, 72  
 wielokrotnego wyboru, 74  
 wieloliniowe, 68, 175  
 wyboru, 177  
 połączenie FTP, 465  
 poprawianie błędów, 32  
 powielanie tła, 112  
 pozycjonowanie, 130  
 absolute, 132  
 fixed, 136  
 relative, 131  
 static, 131  
 program  
 Notepad++, 18  
 Total Commander, 466  
 XAMPP, 406  
 przedrostki przeglądarek, 209  
 przeglądarki internetowe, 17, 213

przesyłanie

danych w tle, 415

znaków specjalnych, 437

przezroczyste tło, 110

przycisk formularza, 185, 353

przyciski, 78

pseudoklasa

first-child, 173

first-of-type, 172

focus, 186

nth-child, 171

nth-of-type, 168, 173

only-child, 175

only-of-type, 173

## R

reset stylów przeglądarki, 203

responsywne obrazy, 313

rodzaj pozycjonowania, 130

rozmiar

cienia, 127

elementu, 115

fontu, 141

grafiki w tle, 110

zdjęcia, 43

rozmieszczanie elementów blokowych, 126

## S

sekcje strony, 218, 260, 303, 340, 364

selektor

atrybutu, 166

dziecka, 165

potomka, 164

rodzeństwa, 165

wybranego elementu, 168

serwer, 463

XAMPP, 405

skrypt PHP, 409

sposoby wyświetlania, 128

stan linku, 153

stopka strony, 363

w wersji mobilnej, 366

strona responsywna, 192

styl

font-size, 89

fontu, 145

obramowania, 102

w części HEAD, 90

w pliku CSS, 92

w znacznikach, 89

style domyślne przeglądarki, 204

stylizacja

elementów formularza, 175

fragmentu tekstu, 141, 152

linków, 153, 244

list, 159

listy nieuporządkowanej, 278

treści, 311

szerokość obramowania, 103

szkielet dokumentu HTML, 25

## Ś

środkowanie

nawigacji, 242

zawartości strony, 261

## T

tabela, 54

podział, 55

scalanie komórek, 57

tytuł, 57

technologia

Ajax, 424

flash, 84

tekst

alternatywny, 44

pochylenie, 37

podkreślenie, 37

pogrubienie, 37

stylizacja, 141, 311

tło

kolorowe, 106

obrazkowe, 110

powielanie, 112

przezroczyste, 110

ustawianie, 113

z gradientu, 107

top strony, 217

w wersji mobilnej, 232

Total Commander, 466

połączenie FTP, 465

treść, 33

tworzenie

formularza, 67

przycisku, 229

strony, 215

dodanie biblioteki jQuery, 374  
 dodanie markera, 333  
 dodanie nagłówków, 220  
 działanie menu, 376  
 fonty jako ikony, 278  
 formularz kontaktowy, 339  
 formularz w wersji mobilnej, 359  
 klucz API, 327  
 linki dla nawigacji, 241  
 mapa Google, 325  
 mapa Google w wersji mobilnej, 339  
 menu nawigacyjne, 239  
 menu w wersji mobilnej, 249, 384  
 miejsce na pola formularza, 342  
 miejsce na treść, 307  
 oferta, 259  
 oferta w wersji mobilnej, 291  
 okienka z ofertą, 268  
 pliki strony, 216  
 pola formularza, 345  
 projekt, 215  
 przycisk formularza, 353  
 przyciski, 229  
 responsywne obrazy, 313  
 sekcje, 218, 260, 303, 325, 340, 364  
 stopka, 363  
 stopka w wersji mobilnej, 366  
 stylizacja linków, 244  
 stylizacja nagłówka, 305  
 stylizacja treści, 311  
 środkowanie nawigacji, 242  
 środkowanie zawartości, 261  
 top strony, 217  
 top strony w wersji mobilnej, 232  
 tytuł działu, 265  
 w wersji mobilnej, 319  
 widoczność menu, 358  
 wypełnienie boków, 272  
 wyświetlenie mapy, 327  
 zasady, 215  
 zmiana stylu mapy, 336  
 tabeli, 54

## U

układ elementów, 273  
 usuwanie znaczników HTML, 435

## W

waga fontu, 145  
 walidacja pól, 420  
 wartość atrybutu zaczynająca się od, 167  
 web design, 214  
 właściwości elementów, 99  
 wstawianie zdjęcia, 42  
 wybór ikon, 279  
 wyrównanie tekstu, 143  
 wysokość wiersza, 142  
 wyświetlanie
 

- elementów, 126
- znaków specjalnych, 34

 mapy, 327

## X

XAMPP, 405

## Y

YouTube, 86

## Z

zagnieżdżanie list, 40  
 zaokrąglone narożniki, 105  
 zapis pliku do dokumentu, 26  
 zapytania medialne, 195  
 zawartość atrybutu, 166  
 zdjęcia, 42
 

- rozmiar, 43
- wstawianie, 42

 zmiana stylu
 

- mapy, 336
- nagłówka, 305
- po atrybucie, 185

 znacznik
 

- <a>, 60
- <article>, 46, 307
- <aside>, 47
- <audio>, 84
- <button>, 78
- <caption>, 57
- <details>, 49
- <div>, 53
- <figure>, 50
- <footer>, 48

## znacznik

- <header>, 46, 218
- <input>, 70
- <meta>, 194
- <nav>, 45
- <option>, 77
- <p>, 283
- <script>, 329, 402
- <section>, 47
- <select>, 77
- <source>, 84
- <style>, 91
- <summary>, 49
- <tbody>, 55
- <textarea>, 68, 69
- <thead>, 55
- <title>, 93
- <video>, 85

## znaczniki

- deklaracja stylu, 95
- kolejność otwierania, 38
- semantyczne, 45, 49
- usuwanie, 435
- zamykanie, 38
- znaki specjalne, 34, 36, 437
- zwijanie tekstu, 161



# PROGRAM PARTNERSKI

— GRUPY HELION —

1. ZAREJESTRUJ SIĘ
2. PREZENTUJ KSIĄŻKI
3. ZBIERAJ PROWIZJĘ

Zmień swoją stronę WWW w działający bankomat!

**Dowiedz się więcej i dołącz już dzisiaj!**

<http://program-partnerski.helion.pl>

GRUPA  
**Helion** 

**Stron internetowych jest w sieci bardzo dużo** i z każdym dniem ich przybywa. Obecnie każda firma i wiele prywatnych osób chce lub wręcz musi mieć w internecie przynajmniej prostą wizytówkę. Nic zatem dziwnego, że w portalach firm rekrutujących pracowników aż roi się od ofert dla ludzi tworzących serwisy, a profesjonalści z doświadczeniem w tej dziedzinie mogą liczyć na naprawdę niezłe zarobki. Może i Ty chciałbyś dołączyć do tego grona, ale nie masz odpowiedniej wiedzy? A może obawiasz się, że na przeszkodzie stanie Ci brak ukończonych studiów informatycznych? O nic się nie martw, po prostu zacznij lekturę!

**Książka *Zawód front-end developer. 11 kroków do zostania webmasterem*** bezboleśnie wprowadzi Cię w świat tworzenia stron internetowych. Dzięki niej szybko przyswoisz podstawowe pojęcia branżowe i informacje niezbędne, aby rozpocząć przygodę z tą branżą oraz dołączyć do elitarnego grona rozchwytywanych specjalistów – autorów stron WWW. Poznasz warsztat pracy zawodowca, nauczysz się języka HTML5 i dowiesz się, jak tworzyć profesjonalne arkusze stylów CSS3. Odkryjesz możliwości, jakie oferują JavaScript i jQuery, a także zapoznasz się z podstawami języka PHP7. Dowiesz się też, jak zaprojektować i zbudować atrakcyjny serwis WWW oraz umieścić go w internecie.

- Wyposażenie warsztatu pracy
- Język HTML i tworzenie kodu strony
- Najważniejsze znaczniki i konstrukcje języka HTML
- Osadzanie treści multimedialnych na stronie
- Zastosowanie arkuszy stylów CSS3
- Formatowanie elementów graficznych i tekstowych
- Zasady tworzenia responsywnych stron WWW
- Projektowanie i budowanie stron oraz ich składników
- Korzystanie ze skryptów i z bibliotek
- Zastosowanie języka PHP7
- Publikowanie strony w internecie

**Tworzenie stron WWW? Z tą książką to nic trudnego!**

<b>Helion</b> 	
	<b>helion.pl</b>
	<b>HELION SA</b> ul. Kościuszki 1c 44-100 Gliwice tel.: 32 230 98 63 helion@helion.pl

Sprawdź nasze szkolenia!

**SZKOLENIA**



**AKADEMIA IT & BUSINESS**

[WWW.SZKOLENIA.HELION.PL](http://WWW.SZKOLENIA.HELION.PL)

**KOD KORZYŚCI**  
Sięgnij po więcej! ▶



ISBN 978-83-283-3885-2



**INFORMATYKA W NAJLEPSZYM WYDANIU**

Cena: 79,00 zł