



TECNOLOGIA NA START!

Bootstrap

Syed Fazle Rahman

TWORZENIE INTERFEJSÓW STRON WWW

Tytuł oryginału: Jump Start Bootstrap

Tłumaczenie: Piotr Cieślak

ISBN: 978-83-283-0511-3

© 2015 Helion S.A.

Authorized Polish translation of the English edition Jump Start Bootstrap,
ISBN 9780992279431 © 2014 SitePoint Pty. Ltd.

This translation is published and sold by permission of O'Reilly Media, Inc.,
which owns or controls all rights to publish and sell the same.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

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.

Wydawnictwo HELION

ul. Kościuszki 1c, 44-100 GLIWICE

tel. 32 231 22 19, 32 230 98 63

e-mail: 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/bootst>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Pliki z przykładami omawianymi w książce można znaleźć pod adresem:

<ftp://ftp.helion.pl/przyklady/bootst.zip>

Printed in Poland.

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

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

Spis treści

Przedmowa	9
Kto powinien przeczytać tę książkę?	9
Przyjęte założenia	9
Fragmenty kodu.....	10
Wskazówki i uwagi.....	10
Materiały dodatkowe	10
Rozdział 1. Bliskie spotkanie z Bootstrapem	11
Czym jest Bootstrap?	11
Po co powstał?	11
W czym ułatwi mi pracę?	12
Historia platform CSS.....	12
Zalety prototypowania CSS	13
Początki platformy Bootstrap	13
Konkurencja Bootstrapa	15
Kto używa Bootstrapa?	15
Ogólne założenia projektowania responsywnego	16
Zmiana układu strony w oparciu o rozmiar ekranu	16
Przygotowanie do pracy z Bootstrapem	18
Rozdział 2. System siatkowy Bootstrapa	25
Czym jest system siatkowy?.....	25
Tworzenie prostej siatki	26
Analiza przypadku: tworzenie dynamicznego układu strony.....	31
Projektowanie dla zwykłych komputerów	33
Projektowanie dla tabletów	36
Projektowanie dla smartfonów	38
Zagnieżdżanie kolumn.....	40
Przesuwanie kolumn	43
Ręczna zmiana układu siatki	45
Podsumowanie.....	45
Rozdział 3. Omówienie komponentów Bootstrapa	47
Komponenty strony	47
Nagłówki stron	47
Panele	50
Obiekt mediów	51

Miniatury.....	52
Listy.....	55
Komponenty nawigacyjne	59
Menu nawigacyjne.....	59
Pasek nawigacji.....	61
Nawigacja okruszkowa	65
Ekspozowanie treści.....	66
Etykiety.....	66
Przyciski.....	67
Glifikony	68
Ramki	69
Odznaki	70
Frajda z formularzami	71
Prosty formularz.....	71
Poziomy układ opisów formularzy.....	73
Formularz w jednej linii.....	74
Klasy pomocnicze w formularzach.....	75
Sterowanie wielkością	76
Podsumowanie.....	77

Rozdział 4. Rozszerzenia Bootstrapa: ciekawe i użyteczne 79

Zwiększanie funkcjonalności	80
Rozwijane menu	80
Rozwijane listy z użyciem JavaScriptu	82
Komunikaty	86
Komunikaty z użyciem JavaScriptu	88
Przyciski.....	88
Przyciski z użyciem JavaScriptu	90
Zarządzanie wyświetlaniem treści.....	91
ScrollSpy.....	91
ScrollSpy z użyciem JavaScriptu.....	94
Zakładki	95
Zakładki z użyciem JavaScriptu.....	97
Harmonijka	97
Harmonijka z użyciem JavaScriptu.....	100
Okienka podpowiedzi	102
Okienka podpowiedzi z użyciem JavaScriptu	103
Dymki	105
Dymki z użyciem JavaScriptu.....	106
Ozdobniki.....	107
Karuzela	107
Karuzela z użyciem JavaScriptu	110
Okienka modalne.....	111
Okienka modalne z użyciem JavaScriptu	113
Podsumowanie.....	115

Rozdział 5. Na głęboką wodę: modyfikowanie Bootstrapa	117
Dostosowywanie arkuszy stylów Bootstrapa.....	117
Dostosowywanie Bootstrapa przy użyciu CSS.....	118
Dostosowywanie Bootstrapa przy użyciu Less.....	121
Zmienne Bootstrapa i domieszki	121
Pliki resetujące.....	121
Pliki główne.....	121
Pliki komponentów	122
Pliki rozszerzeń	123
Pliki pomocnicze	123
Zastępowanie stylów przy użyciu Less	123
Kompilowanie plików Less.....	127
Konfigurowanie Bootstrapa przed pobraniem	127
Zapytania o media a Bootstrap	128
Podsumowanie.....	129
Rozdział 6. Optymalizowanie Bootstrapa	131
Techniki optymalizacji.....	131
Optymalizowanie CSS	131
Optymalizowanie JavaScriptu	133
Optymalizacja grafiki	135
Unikanie typowych błędów	136
Kolejny krok	137
Skorowidz	139

System siatkowy Bootstrapa

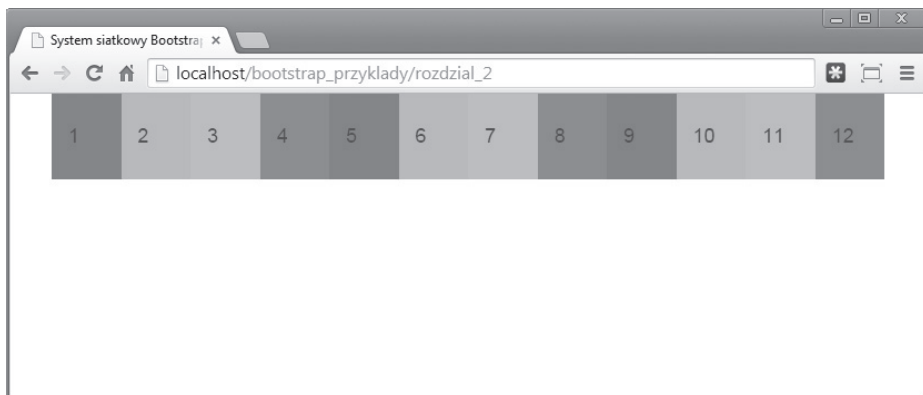
W tym rozdziale zapoznasz się z jedną z najważniejszych funkcji Bootstrapa: **systemem siatkowym**. Przekonasz się, na czym polega jego działanie, i zobaczysz, w jaki sposób można go wykorzystać we własnych projektach. Opracujemy też kilka prostych szablonów stron, aby poznać ów system w praktyce.

Czym jest system siatkowy?

System siatkowy umożliwia poprawne rozmieszczenie zawartości strony WWW. Dzieli on ekran na rzędy i kolumny, które następnie można wykorzystać do projektowania różnych układów treści. Po zdefiniowaniu rzędów i kolumn możemy zdecydować o rozlokowaniu poszczególnych elementów HTML.

System siatkowy dzieli ekran na kolumny — do 12 w każdym rzędzie. Szerokość kolumny jest uzależniona od wielkości ekranu, na którym jest wyświetlana strona. Właśnie to sprawia, że system siatkowy Bootstrapa jest **responsywny**, bo kolumny dynamicznie dopasowują się do każdej zmiany okna przeglądarki. Liczba rzędów może być dowolna; wszystko zależy od wymogów projektu. Przecięcia rzędów i kolumn tworzą prostokątną siatkę, w której rozmieszcza się treść strony.

W ramach przykładu na rysunku 2.1 utworzyłem jeden rząd elementów i za pomocą systemu siatkowego podzieliłem go na 12 kolumn. Tło każdej z kolumn zmieniłem tak, by wyraźnie się od siebie różniły. Dodatkowo każda kolumna jest oznaczona kolejną liczbą.



Rysunek 2.1. Przykładowa strona przedstawiająca 12-kolumnową siatkę

Tworzenie prostej siatki

W tej części rozdziału zaprojektujemy szablon naszej pierwszej strony internetowej opierającej się na systemie siatkowym Bootstrapa. Za punkt wyjścia potraktujemy projekt w folderze *bootstrap_przyklady*, opracowany w poprzednim rozdziale. Skopiuj wszystkie pliki znajdujące się w folderze *bootstrap_przyklady*, a następnie utwórz nowy folder o nazwie *rozdzial_2* i wklej do niego skopiowane pliki.

Otwórz w edytorze dokument *index.html*, zmień tytuł strony na System siatkowy Bootstrapa i usuń znacznik `<h1>` wraz z treścią. Powinieneś otrzymać dokument HTML o następującej zawartości:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <title>System siatkowy Bootstrapa</title>
    <link rel="stylesheet" type="text/css" href="css/bootstrap.css">

    <!--[if lt IE 9]>
      <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
      <script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
    <![endif]-->

  </head>
  <body>
    <!-- Treść strony -->

    <script src="js/jquery.js"></script>
    <script src="js/bootstrap.js"></script>
  </body>
</html>
```


Autorzy Bootstrapa zalecają umieszczenie wszystkich rzędów i kolumn w kontenerze, który gwarantuje ich poprawne wyrównanie i zachowanie odstępów. W Bootstrapie istnieją dwie klasy kontenerów: `container` oraz `container-fluid`. Ta pierwsza służy do tworzenia kontenerów o stałej szerokości w oknie przeglądarki, zaś druga umożliwia tworzenie kontenerów o płynnie zmieniających się rozmiarach. Kontener o stałej szerokości jest zaprojektowany tak, by znajdował się na środku ekranu, w równych odstępach od obydwu boków. Z tego względu warto zadbać o zawijanie tekstu i innych treści w obrębie kontenera.

W kolejnym przykładzie utworzymy kontener klasy `container`, czyli o stałej szerokości. Dodaj poniższy kod do strony HTML:

```
<div class="container">
</div>
```

Następnie wewnątrz kontenera utworzymy rząd. Po zdefiniowaniu rzędu będziemy mogli przystąpić do tworzenia kolumn. Do tworzenia rzędów siatki w Bootstrapie służy klasa `row`:

```
<div class="container">
  <div class="row">
  </div>
</div>
```

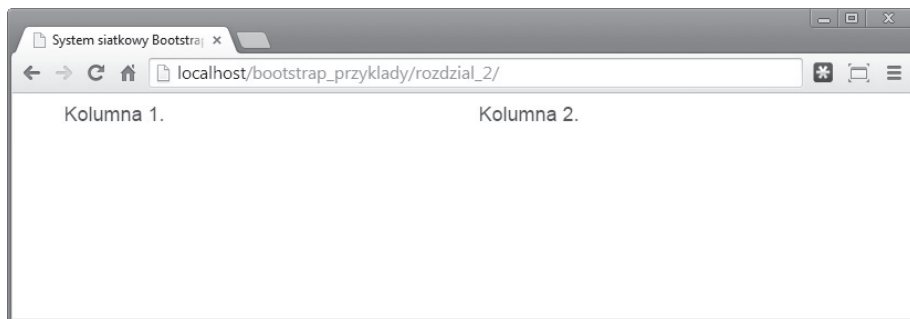
Rzędów można utworzyć dowolnie dużo, ale wszystkie muszą zostać umieszczone w kontenerze. W celu uzyskania optymalnych efektów zaleca się utworzenie jednego kontenera ze wszystkimi rzędami.

W Bootstrapie kolumny są tworzone pośrednio, poprzez określenie, jak wiele spośród 12 standardowych kolumn chciałbyś połączyć. Przypuśćmy, że potrzebujesz tylko jednej dużej kolumny, rozciągającej się na całą szerokość okna. Taka kolumna powinna obejmować wszystkie dwanaście kolumn Bootstrapa. Aby uzyskać taki efekt, należy użyć klasy o nazwie `col-xs-12`, gdzie liczba 12 określa liczbę kolumn do połączenia. (Na razie nie zastanawiaj się nad członem `xs` w nazwie klasy; wrócimy do niego później).

Na tej samej zasadzie aby utworzyć w danym rzędzie dwie kolumny o równej szerokości, należy użyć klasy `col-xs-6` do zdefiniowania każdej z nich. Taka składnia oznacza dla Bootstrapa, że potrzebujemy dwóch kolumn, obejmujących po sześć kolumn Bootstrapa każda:

```
<div class="container">
  <div class="row">
    <div class="col-xs-6">
      <h4>Kolumna 1.</h4>
    </div>
    <div class="col-xs-6">
      <h4>Kolumna 2.</h4>
    </div>
  </div>
</div>
```

Efekt został pokazany na rysunku 2.2.



Rysunek 2.2. Tę siatkę tworzą dwie kolumny o równej szerokości

Aby lepiej wyodrębnić kolumny, zmieńmy kolor każdej z nich. Utwórzmy nowy plik CSS o nazwie *styles.css* w folderze ze stylami CSS projektu. Podana nazwa pliku nie jest obowiązkowa; możesz ją zmienić, jak tylko chcesz. Dołączmy ten plik do dokumentu *index.html* przy użyciu znacznika `link` w elemencie `head`:

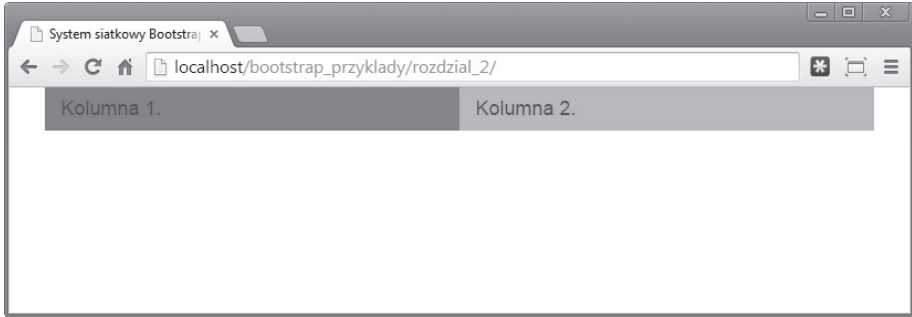
```
<link href="css/styles.css" rel="stylesheet">
```

Zdefiniujmy teraz w tym pliku reguły CSS, dzięki którym każda z kolumn będzie miała inny kolor:

```
.col1{
  background: #5C7080;
}
.col2{
  background: #6BC0FF;
}
```

Musimy jeszcze dodać klasy `col1` i `col2` do kodu, aby kolumny zostały wyświetlone zgodnie ze zdefiniowanymi dla nich regułami (rysunek 2.3). Po modyfikacji ten fragment kodu wygląda tak:

```
<div class="container">
  <div class="row">
    <div class="col-xs-6 col1">
      <h4>Kolumna 1.</h4>
    </div>
    <div class="col-xs-6 col2">
      <h4>Kolumna 2.</h4>
    </div>
  </div>
</div>
```



Rysunek 2.3. Siatka z dwóch kolumn z różnymi kolorami tła

Wróćmy do członu `xs` w nazwie klasy `col-xs-6`. Co on oznacza? W Bootstrapie są cztery rodzaje prefiksów klas służących do tworzenia kolumn przeznaczonych dla ekranów o różnych wielkościach:

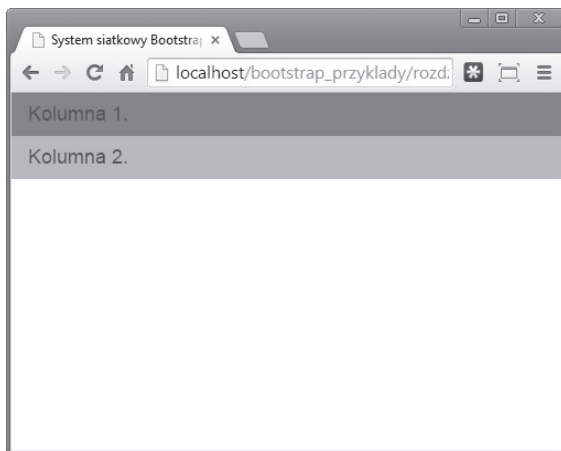
1. `col-xs` dla bardzo małych wyświetlaczy (szerokość ekranu < 768 pikseli),
2. `col-sm` dla małych wyświetlaczy (szerokość ekranu ≥ 768 pikseli),
3. `col-md` dla średnich wyświetlaczy (szerokość ekranu ≥ 992 piksele),
4. `col-lg` dla dużych wyświetlaczy (szerokość ekranu ≥ 1200 pikseli).

Jeśli używamy klasy `col-xs-12`, to znaczy, że na bardzo małych ekranach dany element powinien obejmować wszystkie 12 dostępnych kolumn Bootstrapa. A co z większymi wyświetlaczami? W powyższym kodzie nie określiliśmy, w jaki sposób `div` powinien się zachowywać na dużych ekranach. Na szczęście Bootstrap automatycznie stosuje dla nich szablon zdefiniowany dla najmniejszego rodzaju ekranu. Z tego względu nasz `div` będzie obejmował 12 kolumn niezależnie od wielkości wyświetlacza.

Przeanalizujmy poniższy kod:

```
<div class="container">
  <div class="row">
    <div class="col-xs-12 col-sm-6 col1">
      <h4>Kolumna 1.</h4>
    </div>
    <div class="col-xs-12 col-sm-6 col2">
      <h4>Kolumna 2.</h4>
    </div>
  </div>
</div>
```

Tym razem określiliśmy, by klasa `col-xs-12` była używana na bardzo małych wyświetlaczach, a klasa `col-sm-6` na tych trochę większych. Z tego względu na bardzo małych wyświetlaczach każda kolumna będzie obejmowała wszystkie 12 kolumn Bootstrapa, co przełoży się na ułożenie kolumn jedna nad drugą (rysunek 2.4). Na trochę większych ekranach każda kolumna będzie jednak zajmować tylko sześć kolumn Bootstrapa i w rezultacie zostaną one wyświetlone obok siebie, jak na rysunku 2.3.



Rysunek 2.4. Układ strony na bardzo małym wyświetlaczu

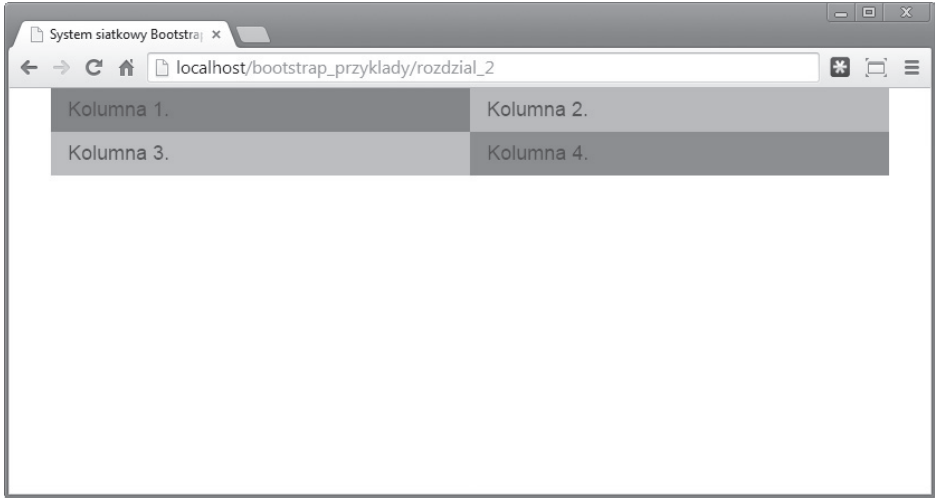
Rozbudujmy dotychczasowy kod o drugi rząd kolumn. Aby to zrobić, po prostu powtórzmy kod pierwszego rzędu. Gotowy kod dla siatki składającej się z dwóch rzędów i (łącznie) czterech kolumn wygląda następująco:

```
<div class="container">
  <div class="row">
    <div class="col-xs-12 col-sm-6 col1">
      <h4>Kolumna 1.</h4>
    </div>
    <div class="col-xs-12 col-sm-6 col2">
      <h4>Kolumna 2.</h4>
    </div>
  </div>
  <div class="row">
    <div class="col-xs-12 col-sm-6 col3">
      <h4>Kolumna 3.</h4>
    </div>
    <div class="col-xs-12 col-sm-6 col4">
      <h4>Kolumna 4.</h4>
    </div>
  </div>
</div>
```

Dodałem ponadto dwie nowe klasy, col3 i col4, aby nadać nowym kolumnom inne kolory:

```
.col3{
  background: #E8AA4C;
}
.col4{
  background: #FF384E;
}
```

A oto efekt, pokazany na rysunku 2.5.



Rysunek 2.5. Dwa rzędy, cztery kolumny

Analiza przypadku: tworzenie dynamicznego układu strony

Przyjrzyjmy się, w jaki sposób można wykorzystać system siatkowy w praktyce, i zaprojektujemy dynamiczny układ strony, który będzie się dopasowywał do wielkości urządzenia, na jakim jest wyświetlany.

Przypuśćmy, że chcemy utworzyć szablon bloga przy użyciu systemu siatkowego Bootstrapa. Wiemy, że na ekranie zwykłego komputera szkielet strony powinien się wyświetlać jak na rysunku 2.6.



Rysunek 2.6. Szablon projektowanej strony oglądany na ekranie zwykłego komputera

Jak widać, szablon zawiera nagłówek rozciągający się na całą szerokość strony. Pod nim zaś mamy trzykolumnowy układ z wpisami z bloga. Jeśli szablon w tej postaci wyświetlimy na tablecie (w układzie pionowym), to strona będzie sprawiała wrażenie bardzo „ciasnej”. Z tego względu siatka strony dla tabletu została zmodyfikowana tak jak na rysunku 2.7.



Rysunek 2.7. Szablon projektowanej strony oglądany na ekranie tabletu

W tej wersji projektu nagłówek wygląda tak samo jak na ekranie zwykłego komputera. Wpisy poniżej zostały jednak wyświetlone w układzie dwukolumnowym zamiast dotychczasowych trzech kolumn. Przypuśćmy, że tę samą stronę chcemy wyświetlić na urządzeniu mobilnym. Szablon takiego wariantu został pokazany na rysunku 2.8.



Rysunek 2.8. Szablon projektowanej strony oglądany na ekranie smartfona

Innymi słowy na potrzeby urządzeń mobilnych układ dwukolumnowy został przekształcony w jednokolumnowy.

Zastanówmy się, w jaki sposób osiągnąć taki efekt w kodzie.

Projektowanie dla zwykłych komputerów

Jak już wspomniałem, za wyświetlacze średniej wielkości przyjęto w Bootstrapie takie, których szerokość wynosi ponad 992 piksele. Ekran komputerów stacjonarnych na ogół są większe. Z tego względu do rozmieszczenia kolumn na typowych ekranach będziemy używali prefiksu `col-md`. Ten sam układ będzie powielony w przypadku wyświetlaczy szerszych niż 1200 pikseli; w przypadku tego konkretnego projektu możemy więc pominąć klasy z prefiksem `col-lg`, bo nie będą one miały wpływu na wygląd strony.

Utwórz nowy dokument HTML i nadaj mu nazwę *blog.html*. Wklej do dokumentu podstawową strukturę HTML ze wstępnie skonfigurowanym Bootstrapem, zgodnie ze wskazówkami z poprzedniego rozdziału. Zmień zawartość znacznika `<title>` na Mój pierwszy blog w Bootstrapie i usuń z treści znacznik `<h1>`. Kod dokumentu *blog.html* powinien teraz wyglądać następująco:

```
<!DOCTYPE html>
<html lang="pl">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <title>Mój pierwszy blog w Bootstrapie</title>
    <link rel="stylesheet" type="text/css" href="css/bootstrap.css">

    <!-- [if lt IE 9]>
      <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
      <script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
    <![endif]-->

  </head>
  <body>
    <!-- miejsce na treść dokumentu -->
    <script src="js/jquery.js"></script>
    <script src="js/bootstrap.js"></script>
  </body>
</html>
```

Najpierw musimy utworzyć kontener na treść bloga. Użyjmy kontenera o stałej szerokości, utworzonego przy użyciu klasy `container`.

```
<div class="container">
</div>
```

Następnie w makiecie wariantu dla zwykłych monitorów mamy nagłówek. Utwórzmy rząd z pojedynczą kolumną obejmującą wszystkie 12 kolumn Bootstrapa.

```
<div class="container">
  <div class="row">
    <div class="col-md-12 text-center">
      <h1>Mój pierwszy blog w Bootstrapie</h1>
    </div>
  </div>
</div>
```

W powyższym kodzie użyłem klasy pomocniczej `text-center` Bootstrapa, aby wyśrodkować tekst w kolumnie. Nagłówek jest gotowy.

Teraz zajmijmy się trzykolumnowym układem treści, w którym mają być publikowane wpisy z bloga. Ponieważ mamy do dyspozycji w sumie 12 kolumn Bootstrapa, każda z kolumn bloga powinna obejmować po cztery takie kolumny. W rezultacie otrzymamy trzy kolumny o jednakowej szerokości. Utwórzmy nowy rząd i zdefiniujmy kolumny za pomocą klasy `col-md-4`.

```
<div class="container">
  <div class="row">
    <div class="col-md-12 text-center">
      <h1>Mój pierwszy blog w Bootstrapie</h1>
    </div>
  </div>

  <hr>

  <div class="row">
    <div class="col-md-4">

    </div>
    <div class="col-md-4">

    </div>
    <div class="col-md-4">

    </div>
  </div>
</div>
```

Pomiędzy dwoma rzędami wstawiłem znacznik `<hr>`, aby utworzyć poziomą linię pod nagłówkiem.

Teraz możemy wypełnić kolumny tymczasową treścią. Użyjemy w tym celu znaczników `<h3>` oraz `<p>` i skorzystamy z fragmentu tekstu *lorem ipsum*.

```
<div class="container">
  <div class="row">
    <div class="col-md-12 text-center">
```



```

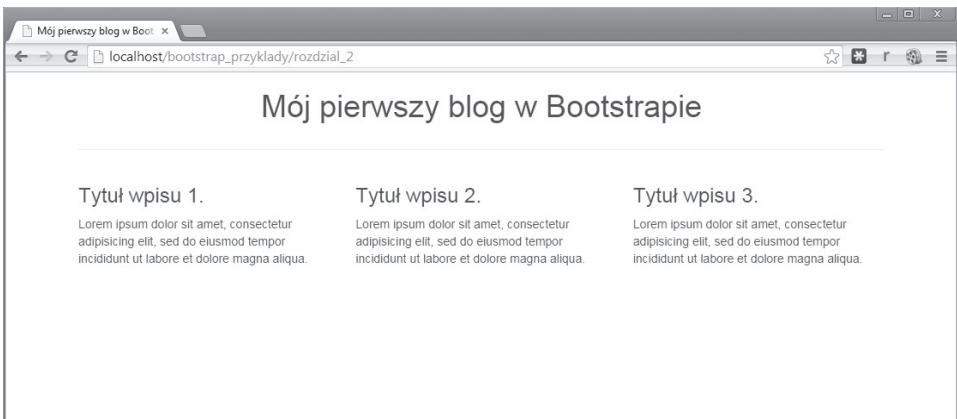
        <h1>Mój pierwszy blog w Bootstrapie</h1>
    </div>
</div>

<hr>

<div class="row">
    <div class="col-md-4">
        <h3>Tytuł wpisu 1.</h3>
        <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod
        ↪tempor incididunt ut labore et dolore magna aliqua. </p>
    </div>
    <div class="col-md-4">
        <h3>Tytuł wpisu 2.</h3>
        <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod
        ↪tempor incididunt ut labore et dolore magna aliqua. </p>
    </div>
    <div class="col-md-4">
        <h3>Tytuł wpisu 3.</h3>
        <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod
        ↪tempor incididunt ut labore et dolore magna aliqua. </p>
    </div>
</div>

```

Teraz dokument *blog.html* powinien wyglądać tak jak na rysunku 2.9.



Rysunek 2.9. Nasz niekompletny na razie blog, w wersji dla zwykłych monitorów

Zgodnie z makietą dla zwykłych monitorów powinniśmy dodać jeszcze trzy kolumny z wpisami z bloga. Tym razem jednak nie utworzymy na te trzy kolumny osobnego rzędu, lecz dołączymy je bezpośrednio do istniejącego rzędu, w dotychczasowym kodzie. Być może zastanawiasz się, w jaki sposób zmieszczą się w jednym rzędzie 24 kolumny Bootstrapa (sześć kolumn bloga po cztery kolumny Bootstrapa), skoro Bootstrap obsługuje ich tylko 12 na rząd. Rzeczywiście tak jest, ale jeśli spróbujemy przekroczyć tę wartość, reszta kolumn zostanie przeniesiona do nowego rzędu. Ten nowy rząd ponownie będzie miał szerokość 12 kolumn Bootstrapa. W ten sposób możemy umieścić wszystkie kolumny z wpisami z bloga w jednym elemencie klasy `row`.

Uzupełnijmy więc istniejący kod o trzy nowe kolumny.

```
<div class="container">
  <div class="row">
    <div class="col-md-12 text-center">
      <h1>Mój pierwszy blog w Bootstrapie</h1>
    </div>
  </div>

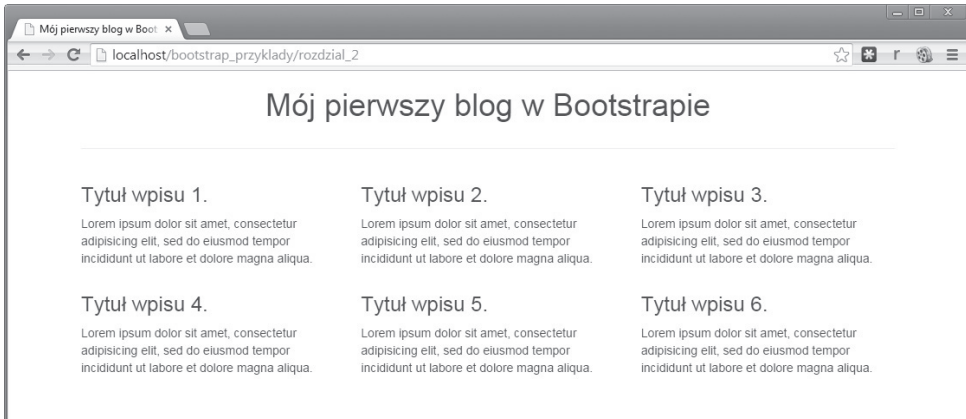
  <hr>

  <div class="row">
    <div class="col-md-4">
      <h3>Tytuł wpisu 1.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
    <div class="col-md-4">
      <h3>Tytuł wpisu 2.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
    <div class="col-md-4">
      <h3>Tytuł wpisu 3.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
    <div class="col-md-4">
      <h3>Tytuł wpisu 4.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
    <div class="col-md-4">
      <h3>Tytuł wpisu 5.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
    <div class="col-md-4">
      <h3>Tytuł wpisu 6.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
  </div>
</div>
```

Wreszcie udało się nam osiągnąć w HTML efekt zakładany we wstępnej makiecie strony w wersji dla zwykłych monitorów (patrz rysunek 2.10).

Projektowanie dla tabletów

Zmodyfikujmy teraz kod tak, by osiągnąć rezultat, jaki zakładaliśmy w makiecie strony dla tabletów. W odróżnieniu od większości zwykłych monitorów na tabletach można pracować w dwóch układach: **portretowym** (pionowym) i **krajobrazowym** (poziomym). Układ poziomy w przypadku tabletu jest traktowany jako wyświetlacz średniej wielkości (**szerość ekranu ≥ 992 piksele**), a sprawę z takim ekranem mamy w zasadzie załatwioną dzięki zastosowaniu klas `col-md-*`. Mamy jednak do opracowania układ pionowy, który jest odpowiednikiem małego wyświetlacza. Ten układ możemy obsłużyć za pomocą klas `col-sm-*`.



Rysunek 2.10. Kompletny układ bloga w wersji dla zwykłych monitorów

Ponieważ w przypadku mniejszych wyświetlaczy zależy nam na uzyskaniu struktury dwukolumnowej, musimy wymusić taki układ, w którym każda nasza kolumna obejmuje sześć kolumn Bootstrapa. W ten sposób otrzymamy dwie kolumny w każdym rzędzie (2 razy 6 kolumn Bootstrapa = 12 kolumn Bootstrapa). Mamy wprawdzie tylko jeden rząd, ale to nie szkodzi: po zajęciu wszystkich 12 kolumn Bootstrapa następne kolumny bloga zostaną przeniesione do nowego rzędu i tak dalej.

Rozszerzymy zatem kod przeznaczony dla zwykłych monitorów o kolejną klasę, `col-sm-6`.

```
<div class="container">
  <div class="row">
    <div class="col-md-12 text-center">
      <h1>Mój pierwszy blog w Bootstrapie</h1>
    </div>
  </div>

  <hr>

  <div class="row">
    <div class="col-md-4 col-sm-6">
      <h3>Tytuł wpisu 1.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
    <div class="col-md-4 col-sm-6">
      <h3>Tytuł wpisu 2.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
    <div class="col-md-4 col-sm-6">
      <h3>Tytuł wpisu 3.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
    <div class="col-md-4 col-sm-6">
      <h3>Tytuł wpisu 4.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
    <div class="col-md-4 col-sm-6">

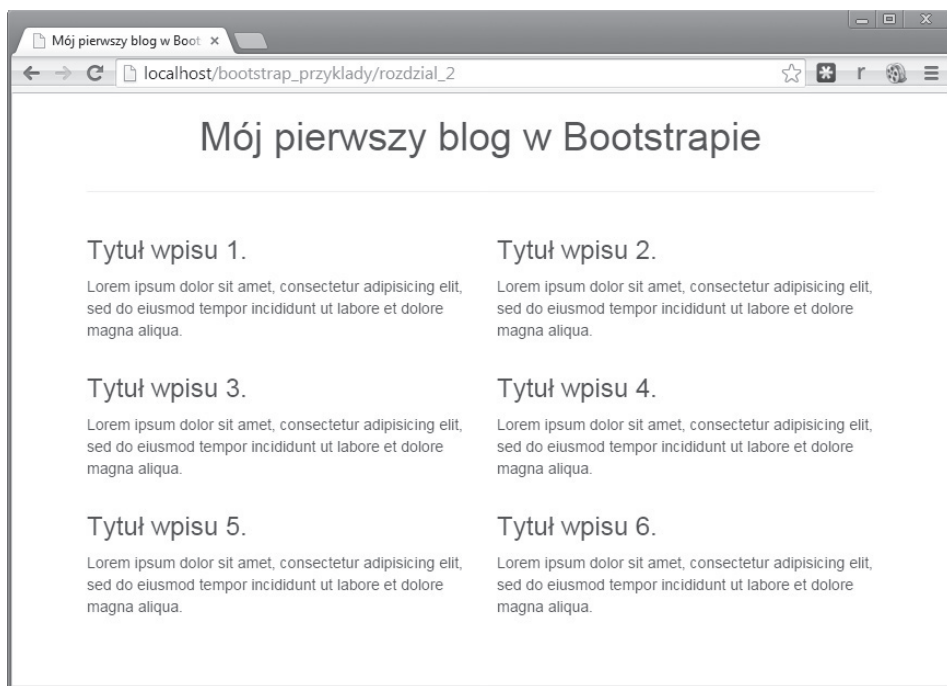
```

```

    <h3>Tytuł wpisu 5.</h3>
    <p>Lorem ipsum dolor sit amet ... </p>
  </div>
  <div class="col-md-4 col-sm-6">
    <h3>Tytuł wpisu 6.</h3>
    <p>Lorem ipsum dolor sit amet ... </p>
  </div>
</div>
</div>

```

Jak widać na rysunku 2.11, otrzymaliśmy w ten sposób dwa różne układy strony dla tabletów: trzykolumnowy w trybie krajobrazowym i dwukolumnowy w trybie portretowym.



Rysunek 2.11. Układ dwukolumnowy dla tabletów

Projektowanie dla smartfonów

Tak jak tablety, smartfony również można użytkować pionowo albo poziomo. W przypadku większości smartfonów zarówno tryb krajobrazowy, jak i portretowy klasyfikuje się do bardzo małych wyświetlaczy (*szerokość ekranu < 768 pikseli*). W przypadku większych smartfonów, niekiedy zwanych phabletami, takich jak Samsung Galaxy Note 3 i Motorola Droid Razr HD, tryb poziomy można potraktować jako wyświetlacz mały.

Jeśli chodzi o układ strony dla urządzeń mobilnych, to musi on zawierać tylko jedną kolumnę. Mam nadzieję, że wiesz już, jak to osiągnąć na podstawie istniejącego kodu: trzeba użyć klas z prefiksem `col-xs`, dla bardzo małych wyświetlaczy. W tym przypadku zależy

nam na tym, by każdy wpis na blogu zajmował wszystkie 12 kolumn Bootstrapa. W ten sposób w jednym rzędzie będzie się znajdował tylko jeden wpis. To sugeruje użycie klasy `col-xs-12`, dodajmy ją więc do istniejącego kodu.

```
<div class="container">
  <div class="row">
    <div class="col-md-12 text-center">
      <h1>Mój pierwszy blog w Bootstrapie</h1>
    </div>
  </div>

  <hr>

  <div class="row">
    <div class="col-md-4 col-sm-6 col-xs-12">
      <h3>Tytuł wpisu 1.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
    <div class="col-md-4 col-sm-6 col-xs-12">
      <h3>Tytuł wpisu 2.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
    <div class="col-md-4 col-sm-6 col-xs-12">
      <h3>Tytuł wpisu 3.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
    <div class="col-md-4 col-sm-6 col-xs-12">
      <h3>Tytuł wpisu 4.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
    <div class="col-md-4 col-sm-6 col-xs-12">
      <h3>Tytuł wpisu 5.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
    <div class="col-md-4 col-sm-6 col-xs-12">
      <h3>Tytuł wpisu 6.</h3>
      <p>Lorem ipsum dolor sit amet ... </p>
    </div>
  </div>
</div>
```

W ten sposób otrzymaliśmy kompletną stronę HTML, responsywną i możliwą do obejrzenia na dowolnym wyświetlaczu — układ jednokolumnowy został pokazany na rysunku 2.12. Tak przygotowany dokument możesz umieścić w jednej z darmowych usług przechowywania danych w chmurze, na przykład Google Drive¹, i przetestować go na prawdziwym tablecie albo smartfonie. Ewentualnie po prostu ręcznie przeskaluj okno przeglądarki i przyjrzyj się zmianom w układzie strony. Mam nadzieję, że analiza tego przykładu ułatwi Ci zrozumienie systemu siatkowego Bootstrapa.

¹ <https://drive.google.com/>



Rysunek 2.12. Jednokolumnowy układ dla małych wyświetlaczy

Zagnieżdżanie kolumn

Nowy zestaw 12 kolumn Bootstrapa można utworzyć w obrębie dowolnej kolumny istniejącego układu. Taki efekt można osiągnąć poprzez wstawienie nowego elementu row w istniejącej kolumnie i wypełnienie tego elementu nowymi kolumnami. Ponieważ jest to nowy rząd, czyli element typu row, dowolna umieszczona w nim kolumna może obejmować 12 kolumn Bootstrapa, ale szerokość całego rzędu będzie ograniczona szerokością elementu, w którym został on umieszczony.

Przyjrzyjmy się temu na przykładzie. Utwórz nowy dokument HTML i zapisz go pod nazwą *nested.html*. Wypełnij dokument kodem z inicjalizacją Bootstrapa zgodnie ze wskazówkami podanymi w poprzednim rozdziale. Ponadto dołącz do dokumentu plik CSS *styles.css*, utworzony wcześniej w tym rozdziale. Kod nowego dokumentu HTML powinien wyglądać następująco:

```
<!DOCTYPE html>
<html lang="pl">
  <head>
    <meta charset="utf-8">
```

```

<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Moja pierwsza strona w Bootstrapie</title>
<link rel="stylesheet" type="text/css" href="css/bootstrap.css">
<link rel="stylesheet" type="text/css" href="css/styles.css">

<!--[if lt IE 9]>
  <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
  <script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
<![endif]-->

</head>
<body>

  <script src="js/jquery.js"></script>
  <script src="js/bootstrap.js"></script>
</body>
</html>

```

Utwórzmy kontener, a w nim rząd:

```

<div class="container">
  <div class="row">

    </div>
</div>

```

Przypuśćmy, że zamierzamy obsługiwać wyświetlacz o średniej wielkości, chcemy więc opracować układ dwukolumnowy. Powinieneś już wiedzieć, że w celu utworzenia takiego układu każda z naszych kolumn powinna obejmować sześć kolumn Bootstrapa. Takie kolumny można zaś uzyskać przy użyciu klasy `col-md-6`. Dodajmy zatem dwie kolumny do istniejącego kodu:

```

<div class="container">
  <div class="row">
    <div class="col-md-6 col1">
      <h3>Kolumna 1.</h3>
    </div>
    <div class="col-md-6 col2">
      <h3>Kolumna 2.</h3>
    </div>
  </div>
</div>

```

W tym kodzie skorzystaliśmy ponadto z dwóch klas zdefiniowanych w pliku *styles.css*, a mianowicie `col1` i `col2`. Te dwie klasy umożliwią zmianę koloru tła kolumn. Na tym etapie dokument HTML powinien wyglądać tak jak na rzucie ekranu na rysunku 2.13.



Rysunek 2.13. Prosty układ dwukolumnowy

Weźmy teraz pierwszą kolumnę (Kolumna 1) i umieścimy w niej nowy rząd:

```
<div class="container">
  <div class="row">
    <div class="col-md-6 col1">
      <h3>Kolumna 1.</h3>
      <!-- początek zagnieżdżenia -->
      <div class="row">

        </div>
      </div>
    <div class="col-md-6 col2">
      <h3>Kolumna 2.</h3>
    </div>
  </div>
</div>
```

Mamy nowy rząd, w którym ponownie możemy umieścić dwie kolumny:

```
<div class="container">
  <div class="row">
    <div class="col-md-6 col1">
      <h3>Kolumna 1.</h3>
      <!-- początek zagnieżdżenia -->
      <div class="row">
        <div class="col-md-6 col3">
          <h3>Kolumna 4.</h3>
        </div>
        <div class="col-md-6 col4">
          <h3>Kolumna 5.</h3>
        </div>
      </div>
    </div>
  </div>
```

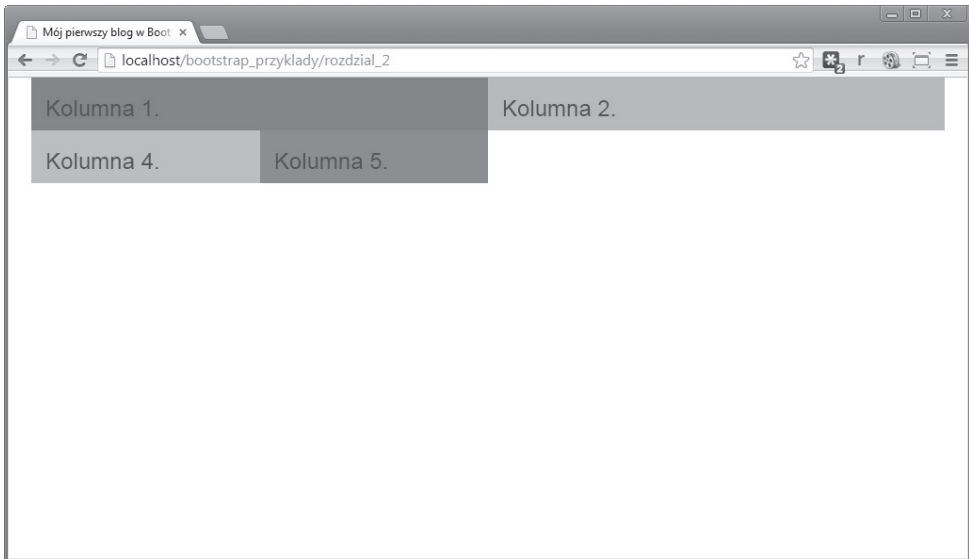


```

        </div>
    </div>
    <div class="col-md-6 col2">
        <h3>Kolumna 2.</h3>
    </div>
</div>
</div>

```

Jak widać na rysunku 2.14, dwie nowe kolumny są umieszczone poniżej pierwszej. Do zmiany koloru ich tła użyłem klas `col3` oraz `col4` z pliku `styles.css`.



Rysunek 2.14. Zagnieżdżony układ dwukolumnowy

Możliwość zagnieżdżania kolumn bardzo się przydaje przy projektowaniu stron o skomplikowanej strukturze. W jednej z otrzymanych wyżej wewnętrznych kolumn można by-łoby wstawić nowy rząd i utworzyć w nim kolejny zestaw kolumn. Taki proces można po-wtarzać aż do osiągnięcia planowanego efektu.

Przesuwanie kolumn

Kolejną niezwykle przydatną funkcją systemu siatkowego Bootstrapa jest możliwość przesuwania kolumn względem siebie. Zwykle służy ona do zwiększenia odstępu po lewej stronie kolumny. Jeśli przykładowo chcesz utworzyć kolumnę znajdującą się w odległości trzech kolumn Bootstrapa od lewej krawędzi okna, możesz skorzystać ze wspomnianej funkcji.

Klasy służące do przemieszczania mają następujące nazwy:

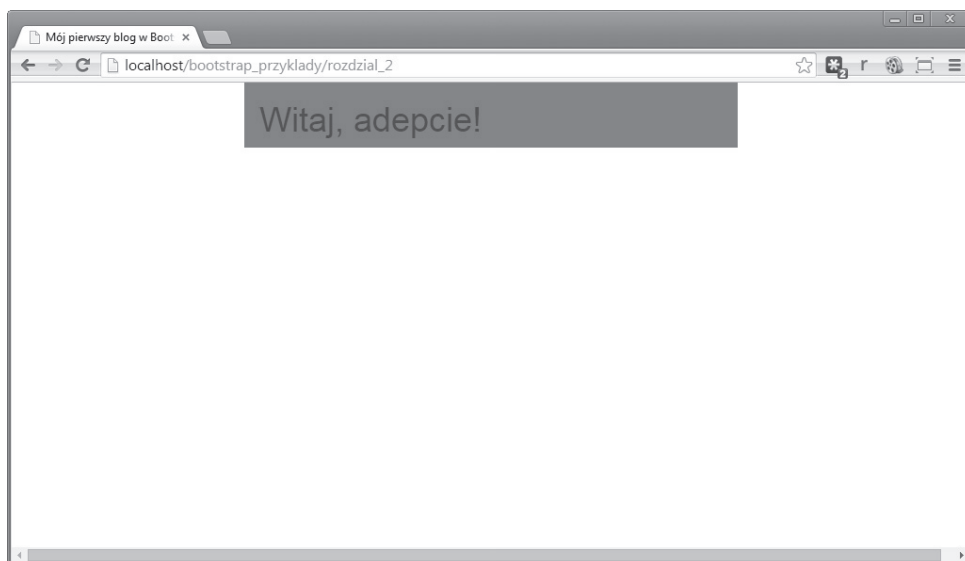
- `col-xs-offset-*`
- `col-sm-offset-*`

- col-md-offset-*
- col-lg-offset-*

Przypuśćmy, że chcemy przenieść kolumnę obejmującą trzy kolumny Bootstrapa w prawą stronę. W przypadku stron dla bardzo małych wyświetlaczy można w tym celu użyć klasy "col-xs-offset-3", na przykład tak:

```
<div class="row">  
  <div class="col-xs-6 col-xs-offset-3 col1">  
    <h1>Witaj, adeptcie!</h1>  
  </div>  
</div>
```

Powyższy kod spowoduje utworzenie kolumny obejmującej sześć kolumn Bootstrapa, przesuniętej o trzy kolumny Bootstrapa w prawo, jak na rysunku 2.15.



Rysunek 2.15. Przesunięta kolumna



Wyśrodkowanie kolumny

Zauważ, że po lewej i po prawej stronie przykładowej kolumny zostały odstępy o szerokości trzech kolumn Bootstrapa. To jeden z najlepszych sposobów na wyśrodkowanie w oknie przeglądarki kolumny o szerokości wynoszącej 50% szerokości okna.

Ręczna zmiana układu siatki

Istnieje możliwość zmiany układu kolumn niezależnie od tego, w jakiej kolejności zostały one ujęte w kodzie. Jeśli najpierw napiszesz kod dla kolumny `col-md-9`, a potem dla kolumny `col-md-3`, to możesz bez trudu zamienić miejscami ich kolejność przy wyświetlaniu w przeglądarce. Można to zrobić za pomocą klas `pull` i `push` Bootstrapa.

Do przenoszenia kolumn w lewą albo w prawą stronę o określoną liczbę pozycji służą klasy takie jak `col-xs-pull-*` i `col-xs-push-*`. Na przykład:

```
<div class="row">
  <div class="col-xs-9 col-xs-push-3">
    <h1>Kolumna przesunięta w prawo</h1>
  </div>
  <div class="col-xs-3 col-xs-pull-9">
    <h1>Kolumna przesunięta w lewo</h1>
  </div>
</div>
```

Efekt powyższego kodu jest taki, że kolumna `col-xs-9` jest przesunięta o trzy kolumny w prawo („popchnięta”). Z kolei kolumna o nazwie `col-xs-3` została przesunięta o dziewięć kolumn w lewo („pociągnięta”). W rezultacie na ekranie zostaną one wyświetlone tak, jakby były zamienione miejscami.

Istnieje kilka wariantów klas `push` i `pull`, dostosowanych do wyświetlaczy o różnych wielkościach:

- `col-xs-pull-*` oraz `col-xs-push-*` dla bardzo małych wyświetlaczy,
- `col-sm-pull-*` oraz `col-sm-push-*` dla małych wyświetlaczy,
- `col-md-pull-*` oraz `col-md-push-*` dla średnich wyświetlaczy,
- `col-lg-pull-*` oraz `col-lg-push-*` dla dużych wyświetlaczy.

Znak `*` można zastąpić liczbą całkowitą od 1 do 12, w zależności od tego, na jak dużą odległość (w kolumnach) chcesz „popchnąć” (`push`) albo „pociągnąć” (`pull`) daną kolumnę.

Podsumowanie

Za pomocą systemu siatkowego Bootstrapa można opracować niemal dowolną stronę internetową. Prawidłowo użyty, system ten umożliwia projektowanie eleganckich, responsywnych stron WWW, możliwych do wyświetlenia na prawie każdym ekranie, niezależnie od jego rozmiarów. Funkcje takie jak zagnieżdżanie, przesuwanie i zmiana kolejności kolumn sprawiają, że system ten jest bardzo elastyczny w obsłudze.

Szczegółową dokumentację systemu siatkowego Bootstrapa znajdziesz pod adresem <http://getbootstrap.com/css/#grid>.

Skorowidz

A

Adaptive Images, 135
animacja, 13
audio, 51

B

biblioteka jQuery, *Patrz:* jQuery
blog, 31
Bootstrap
 domieszki, 121
 historia, 13, 14
 instalowanie, 18
 konfigurowanie przed pobraniem, 127
 modyfikacja domyślnych stylów, 117, 118
 Less, 121, 123
 optymalizacja, 131
 grafika, 135
 JavaScript, 133, 134
 wersja, 14
 zmienna, 121
Bootstrap Colorpicker, 138
Bootstrap Image Gallery, 138
Bootstrap Lightbox, 138
Bootstrap Markdown, 138
Bootstrap Notifications, 138
Bootstrap Star Rating, 138

C

Chrome, 22
CSS, 13, 28, 117
 klasa, *Patrz:* klasa CSS
 optymalizacja, 131, 133
czytnik ekranu, 121

D

drzewo DOM, 95
dymek, 105
 z JavaScript, 106

E

ekran
 rozmiar, 16, 21, 29, 33, 36, 38
 układ
 krajobrazowy, 36, 38
 pionowy, *Patrz:* ekran układ portretowy
 portretowy, 36, 38
 poziomy, *Patrz:* ekran układ krajobrazowy
element
 a, 57, 67, 80
 atrybut, 80
 button, 67
 head, 28
 HTML5, 21
 input, 67
 li, 57, 64, 96
 row, 40
 ul, 57, 63, 80
etykieta, 66

F

film, *Patrz:* wideo
formularz, 64, 71, 122
 pole
 wejściowe, 71, 76
 wyboru, *Patrz:* pole wyboru
 tworzenie, 71, 73, 74
front-end framework, 11
FuelUX, 137

G

glifikona, 68, 81
 usuwanie, 132
Google Drive, 39

H

harmonijka, 97, 98, 99, 101
 z JavaScript, 100

I

interfejs dotykowy, 16
Internet Explorer 8, 20, 21, 110, 136

J

Jasny Bootstrap, 138
JavaScript, 12, 79
jQuery, 20, 90

K

karuzela, *Patrz:* pokaz slajdów
klasa, *Patrz też:* komponent
 .clearfix, 123
 .hidden-sm, 123
 .hide, 123
 .img-circle, 121
 .img-responsive, 121
 .visible-lg, 123
active, 59, 88, 89, 96
alert, 86
alert-dismissable, 87
alert-link, 88
badge, *Patrz:* odznaka
btn, 12, 55, 67, 89, 120
 btn-block, 68
 btn-default, 89
 btn-group, 89
 btn-lg, 89
 btn-primary, 12, 55, 120
caption, 54
caret, 81
carousel, 108
 carousel-caption, 109
 carousel-control, 109
 carousel-inner, 109
checkbox, 75
collapse, 63, 99
col-lg, 29, 33
col-md, 29, 33, 36
col-sm, 29, 36
col-xs, 29, 38
 col-xs-6, 27, 29
 col-xs-12, 27, 29
 col-xs-pull, 45
 col-xs-push, 45
container-fluid, 61

CSS, 12
 dropdown, 65, 82
 dropdown-menu, 65
 fade, 96
 form, 71
 form-control, 71
 form-group, 71
 form-horizontal, 73
 form-inline, 75
 glyphicon, 68
 has-error, 76
 has-success, 76
 has-warning, 76
img-responsive, 135
in, 99
input-lg, 76
input-sm, 76
label, 66
 label-default, 66
list-group, 57
 list-group-item, 57, 58
 list-group-item-heading, 58
 list-group-item-text, 58
 media, 51
 media-heading, 51
modal, 111
 modal-body, 112
 modal-dialog, 111
 modal-title, 112
nav, 59, 63
 navbar, 61
 navbar-brand, 62
 navbar-collapse, 63
 navbar-default, 61
 navbar-header, 61
 navbar-nav, 63
 navbar-toggle, 62
 nav-pills, 60
 nav-stacked, 60
 nav-tabs, 59
 panel-body, 98
 panel-collapse, 99
 panel-danger, 50
 panel-default, 50
 panel-group, 98
 panel-heading, 98
 panel-info, 50
 panel-primary, 50
 panel-success, 50

panel-title, 98
 panel-warning, 50
 pull, 45
 pull-left, 51
 pull-right, 51
 push, 45
 radio, 75
 row, 27, 35
 slide, 108
 tab-content, 96
 tab-pane, 96
 tooltipButton, 102
 well, 69, 75
 klasaLthumbnail, 54
 kod JavaScript, 12
 kolumna

- kolor, 28
- przesuwanie, 43
- szerokość, 27
- wyśrodkowanie, 44
- zagnieżdżanie, 40

 komentarz, 47, 51
 komponent, 47, *Patrz też*: klasa

- automatycznie ukrywany, 70
- badge, 57
- glyphicon, *Patrz*: glifikona
- label, *Patrz*: etykieta
- list group, 55
- media-body, 51
- navbar, *Patrz*: pasek nawigacji
- nav-tabs, 95
- tab-content, 95
- thumbnails, 52
- well, 69

 komunikat, 86, 102, 111, 123

- odsyłacz, 88
- ukrywanie, 86
- w konsoli, 83
- zamykanie, 86, 87

 kontener

- container, 27
- container-fluid, 27
- div, 48

 Kovařík Jan, 69

L

Less, *Patrz*: metajęzyk Less
 lista, 55

kolor pozycji, 58
 odsyłaczy, 57
 rozwijana, 65, 83, *Patrz też*: menu rozwijane

M

media queries, *Patrz*: zapytanie o media
 menu, 47

- nawigacyjne, 59
- rozwijane, 13, 64, 80, 82

 metajęzyk

- Less, 12, 118
 - kompilacja plików, 127
- Sass, 12, 118

 metajęzyk Less, 121
 metoda

- alert, 88
- button, 90
- carousel, 110
- collapse, 100, 101
- dropdown, 82, 83
- modal, 113
- popover, 106
- scrollspy, 94
- tooltip, 103, 104

 minifikacja, 19, 132, 134
 multimedia, 51

N

nagłówek, 47, 122

- strony, 47

 nawigacja okruszkowa, 59, 65
 Node.js, 127

O

obiekt mediów, 51
 obraz, 51

- miniatura, 52

 odznaka, 70
 okienko modalne, 107, 111, 112, 123

- wielkość, 113
- z JavaScript, 113

 okienko odpowiedzi, 102, 104

- inicjowanie, 102
- z JavaScript, 103

 Otto Mark, 13

P

- panel, 50
 - kolor, 50
 - plynne zanikanie, 96
 - z zakładkami, 95
- pasek nawigacji, 59, 61, 64, 80, 91
- pigułka, 59
- pill, *Patrz*: pigułka
- platforma
 - Blueprint, 12
 - Foundation, 15
 - Gumby, 15
 - YUI, 12
- plik
 - alerts.less, 122
 - badges.less, 122
 - bootstrap.css, 19
 - bootstrap.js, 134
 - bootstrap.less, 132
 - bootstrap.min.css, 19
 - bootstrap.min.js, 132, 134
 - breadcrumbs.less, 122
 - button-groups.less, 122
 - buttons.less, 122
 - carousel.less, 123
 - close.less, 123
 - code.less, 122
 - component-animations.less, 122
 - dropdowns.less, 122
 - forms.less, 122
 - glyphicons.less, 122
 - grid.less, 122
 - input-groups.less, 122
 - jquery.js, 20
 - jumbotron.less, 122
 - labels.less, 122
 - Less, 121, 122, 123, 127
 - list-group.less, 122
 - media.less, 122
 - mixins.less, 121
 - modals.less, 123
 - navbar.less, 122
 - navs.less, 122
 - normalize.less, 121
 - pager.less, 122
 - pagination.less, 122
 - panels.less, 122
 - popovers.less, 123
 - print.less, 121
 - progress-bars.less, 122
 - responsive-utilities.less, 123
 - scaffolding.less, 121
 - tables.less, 122
 - thumbnails.less, 122
 - tooltip.less, 123
 - type.less, 122
 - utilities.less, 123
 - variables.less, 121
 - wells.less, 123
- plugin, *Patrz*: rozszerzenie
- pokaz slajdów, 107
 - Internet Explorer, 110
 - sekcja, 108
 - z JavaScript, 110
- pole wyboru, 75, 89, 122
- projektowanie responsywne, 16
- przeglądarka, 121
 - Google Chrome, *Patrz*: Chrome
- przełącznik, 75, 122
- przycisk, 12, 47, 59, 67, 122
 - aktywny, 88
 - działanie, 68
 - kolor, 67, 90
 - nieaktywny, 88, 90
 - niebieski, 12
 - okienko podpowiedzi, *Patrz*: okienko podpowiedzi
 - przełączanie za pomoca jQuery, 90
 - rozmiar, 68
 - w stanie loading, 90
 - wyświetlenie informacji tekstowej, 90, 91
 - z JavaScript, 90
 - zestaw, 89
 - zgrupowanie, 90
- Pure, 15

R

- ramka, 69
- rozszerzenie, 79, 91, 94
 - Bootstrap Colorpicker, 138
 - Bootstrap Image Gallery, 138
 - Bootstrap Lightbox, 138
 - Bootstrap Markdown, 138
 - Bootstrap Notifications, 138
 - Bootstrap Star Rating, 138
 - carousel, 107

collapse, *Patrz:* harmonijka
 dymek, *Patrz:* dymek
 FuelUX, 137
 Jasny Bootstrap, 138
 modal, 107, 111
 popover, *Patrz:* dymek
 zakładka, *Patrz:* zakładka

S

Sass, *Patrz:* metajęzyk Sass
 ScrollSpy, 91, 92, 94
 Semantic UI, 15
 silnik renderujący, 21
 skrypt
 html5shiv.js, 21, 22
 respond.js, 21, 22
 smartfon, 38
 strona responsywna, 16
 system siatkowy, 12, 25, 31, 122
 wewnątrz elementu formularza, 73
 szablon bloga, 31

T

tablet, 36
 Thronton Jacob, 13
 TimThumb, 135
 tryb zgodności, 21
 tweet, 51

W

wideo, 51
 miniatura, 52
 właściwość CSS3, 21

Y

Yahoo User Interface Library, *Patrz:*
 platforma YUI

Z

zakładka, 95
 z JavaScript, 97
 zapytanie o media, 128, 129
 zdarzenie
 activate.bs.scrollspy, 95
 close.bs.alert, 88
 closed.bs.alert, 88
 hidden, 83
 hidden.bs.collapse, 101
 hidden.bs.modal, 114
 hide, 83
 hide.bs.collapse, 101
 hide.bs.modal, 114
 loaded.bs.modal, 114
 show, 83
 show.bs.collapse, 101
 show.bs.dropdown, 86
 show.bs.modal, 114
 show.bs.tab, 97
 shown, 83
 shown.bs.collapse, 101
 shown.bs.modal, 114
 shown.bs.tab, 97
 slid.bs.carousel, 110
 slide.bs.carousel, 110
 znacznik
 a, 51, 57, 67, 80
 h1, 48, 122
 h3, 34
 hr, 34
 link, 28
 meta, 20
 p, 34

Ż

żądanie Ajax, 86

PROGRAM PARTNERSKI

GRUPY WYDAWNICZEJ 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 WYDAWNICZA

 **Helion SA**

PRZEKONAJ SIĘ, JAK SPRAWNIE TWORZYĆ ATRAKCYJNE INTERFEJSY STRON WWW!

Bootstrap to obecnie jeden z najpopularniejszych szkieletów do tworzenia atrakcyjnych interfejsów stron WWW. Projekt zapoczątkowany przez programistów Twittera zyskał licznych zwolenników ze względu na bogactwo komponentów, świetną dokumentację i zgodność z różnymi przeglądarkami internetowymi. Nie bez znaczenia jest również fakt, że Bootstrap wspiera budowę responsywnych stron WWW.

Ta książka to rewelacyjny przewodnik, który wprowadzi Cię w świat Bootstrapa. Na samym początku poznasz system siatkowy (ang. *grid*), który jest fundamentem tworzenia responsywnych układów. Na kolejnych stronach znajdziesz omówienie poszczególnych komponentów tego szkieletu. Po opanowaniu podstaw przejdziesz do sposobów rozszerzania Bootstrapa, modyfikowania go oraz optymalizowania. Książka ta jest doskonałą lekturą dla wszystkich osób chcących błyskawicznie poznać i wykorzystać ten szkielet!

Dzięki tej książce:

- ✓ **POZNASZ DOSTĘPNE KOMPONENTY**
- ✓ **NAUCZYSZ SIĘ BUDOWAĆ RESPONSYWNE STRONY**
- ✓ **ROZSZERZYSZ MOŻLIWOŚCI BOOTSTRAPA**
- ✓ **ZOPTYMALIZUJESZ JEGO DZIAŁANIE**

Helion

28912 numer katalogowy
księgarnia internetowa

<http://helion.pl>

zamówienia telefoniczne



0 801 339900



0 601 339900

Sprawdź najnowsze promocje:
• <http://helion.pl/promocje>
Książki najchętniej czytane:
• <http://helion.pl/bestsellery>
Zamów informacje o nowościach:
• <http://helion.pl/nowosci>

Helion SA
ul. Kościuszki 1c, 44-100 Gliwice
tel.: 32 230 98 63
e-mail: helion@helion.pl
<http://helion.pl>

sięgnij po **WIĘCEJ**



KOD KORZYSCI

ISBN 978-83-283-0511-3



9 788328 305113

informatyka w najlepszym wydaniu

cena: 34,90 zł