

第 9 章

Bootstrapで モックアップを作る

本章では、Bootstrap を使用して Mr. M COFFEE（ミスターMコーヒー）というカフェのWebサイトのモックアップを作成します。これまで数々のBootstrap のコンポーネントやユーティリティについて説明を行ってきましたが、あとはグリッドレイアウトや基本スタイリングに、ユーティリティで調整したコンポーネントを組み込むだけです。自分でコードを書くよりも、格段に速くレスポンシブなWebサイトを構築できることを実感してみてください。

9

1

サイト概要とファイルの準備

実際の作業に入る前に、どのようなサイトを作るか確認しておきましょう。サンプルサイトの概要や設計内容を確認した後、Web ページを各コンテンツのエリアに分け、ヘッダー、ナビゲーション、スライドというように上から下に向かって順番に作成していきます。

9.1.1

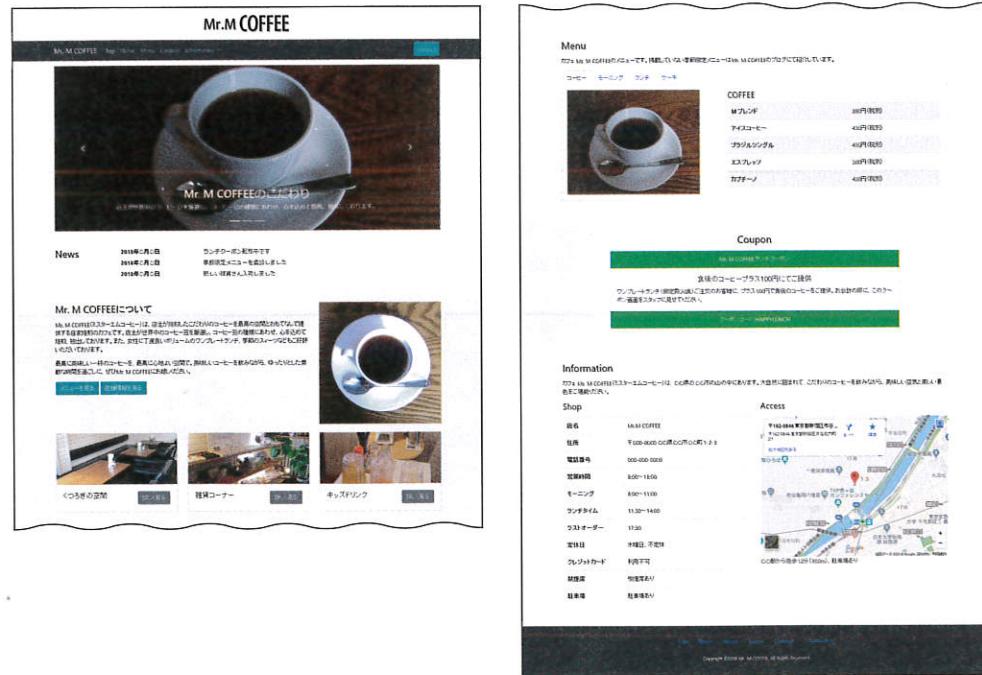
サイト概要

このサイトは、トップページと下層ページの合計 2 ページで構成する Web サイトです。

トップページ

トップページには、新着情報、カフェ紹介、メニュー表やクーポン、店舗概要やアクセスマップといったコンテンツが含まれます（図9-1）。

▼図 9-1 トップページのデザイン（デスクトップ PC 用）



下層ページ

下層ページには、お問合せフォームが含まれます（図 9-2）。

▼図 9-2 下層ページのデザイン（デスクトップ PC 用）

The screenshot shows the 'Contact' page of the Mr. M COFFEE website. At the top, there's a header with the site name 'Mr. M COFFEE' and a 'Contact' button. Below the header, a breadcrumb navigation shows 'Top / Contact'. The main title is 'Contact' with a sub-instruction: 'カツエ Mr. M COFFEE(ミスター・エム・コーヒー)へのお問合せは、こちらのフォームをご利用ください。' Below this, there's a section titled 'お問合せフォーム' (Contact Form). It contains several input fields: 'お名前' (Name), 'メールアドレス' (Email Address), and 'お問い合わせ種類' (Type of Inquiry). There are also checkboxes for 'ロコモ' (Locomotion), '検索エンジン' (Search Engine), and '検索エンジン' (Search Engine). A large text area for 'ご予めについて' (Details about your inquiry) is followed by a '確認する' (Check) button. At the bottom, there's a footer with links to 'Top', 'News', 'About', 'Menu', 'Coupon', 'Information', and 'Contact', along with a copyright notice: 'Copyright ©2016 Mr. M COFFEE. All Rights Reserved.'

9.1.2 ワイヤーフレームの確認

ワイヤーフレームとは、Web ページの「どこに、何を、どのように」配置するかを枠線で示した Web ページの設計図です。レイアウトの方向性を定めるためのもので、ワイヤーフレームの段階では、色や装飾を含めずにページの骨組みを作ります。

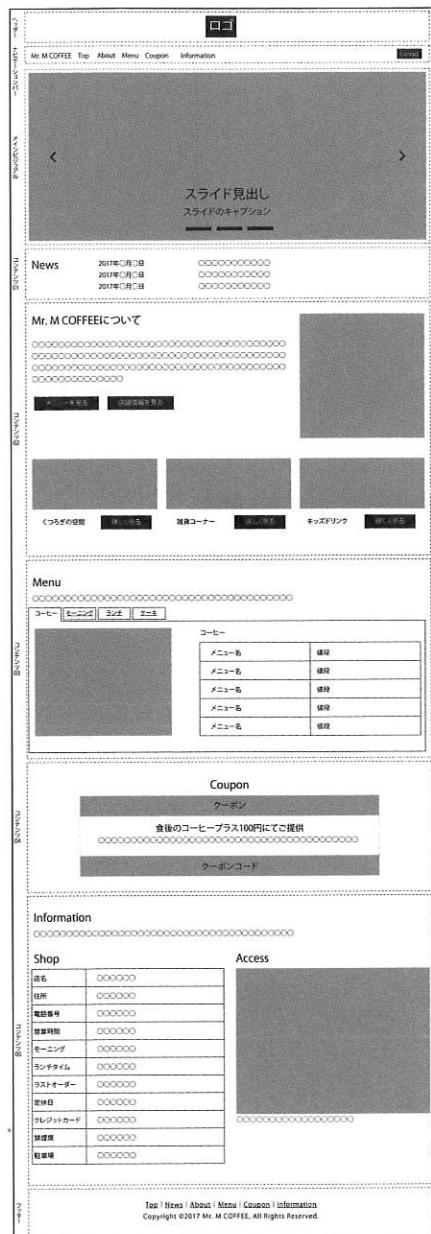
本章のサンプルサイトのワイヤーフレームでは、各コンテンツのエリア分けや、レスポンシブ対応のレイアウト変更などが設計されています。

1
2
3
4
5
6
7
8
9
10

トップページのワイヤーフレーム

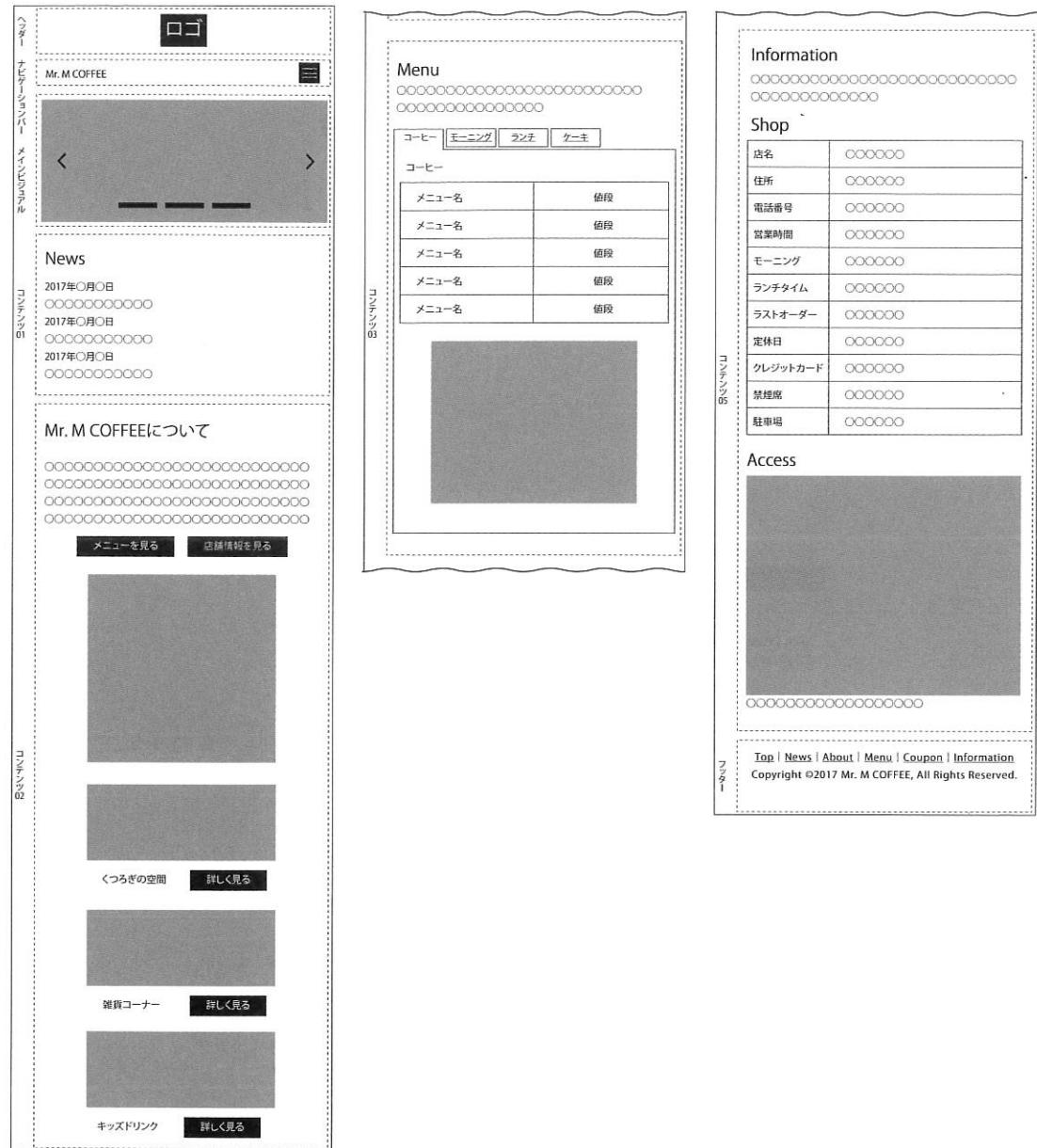
まずはトップページのワイヤーフレームでWebページのレイアウトの仕様を確認しましょう。上からヘッダー、ナビゲーション、メインビジュアルのスライドショー、コンテンツ01「News」、コンテンツ02「About」(Mr.M COFFEEについて)、コンテンツ03「Menu」、コンテンツ04「Coupon」、コンテンツ05「Information」、フッターなどのコンテンツエリアがあります(図9-3)。

▼図9-3 トップページのワイヤーフレーム(デスクトップPC用)



デスクトップPC（画面幅 md 以上）での閲覧時と、モバイル端末（画面幅 sm 以下）での閲覧時とで、図9-4のようにレイアウトが変更されるような設計です。

▼図9-4 トップページのワイヤーフレーム（モバイル端末用）



下層ページのワイヤーフレーム

次に、下層ページのワイヤーフレームを確認しましょう。

ヘッダーとフッターはトップページと共に、パンくずリストやフォームがレイアウトされています（図9-5）。

▼図 9-5 下層ページのワイヤーフレーム（デスクトップ PC 用）

デスクトップPC（画面幅 md 以上）での閲覧時と、モバイル端末（画面幅 sm 以下）での閲覧時とで、図 9-6 のようにレイアウトが変更されるような設計です。



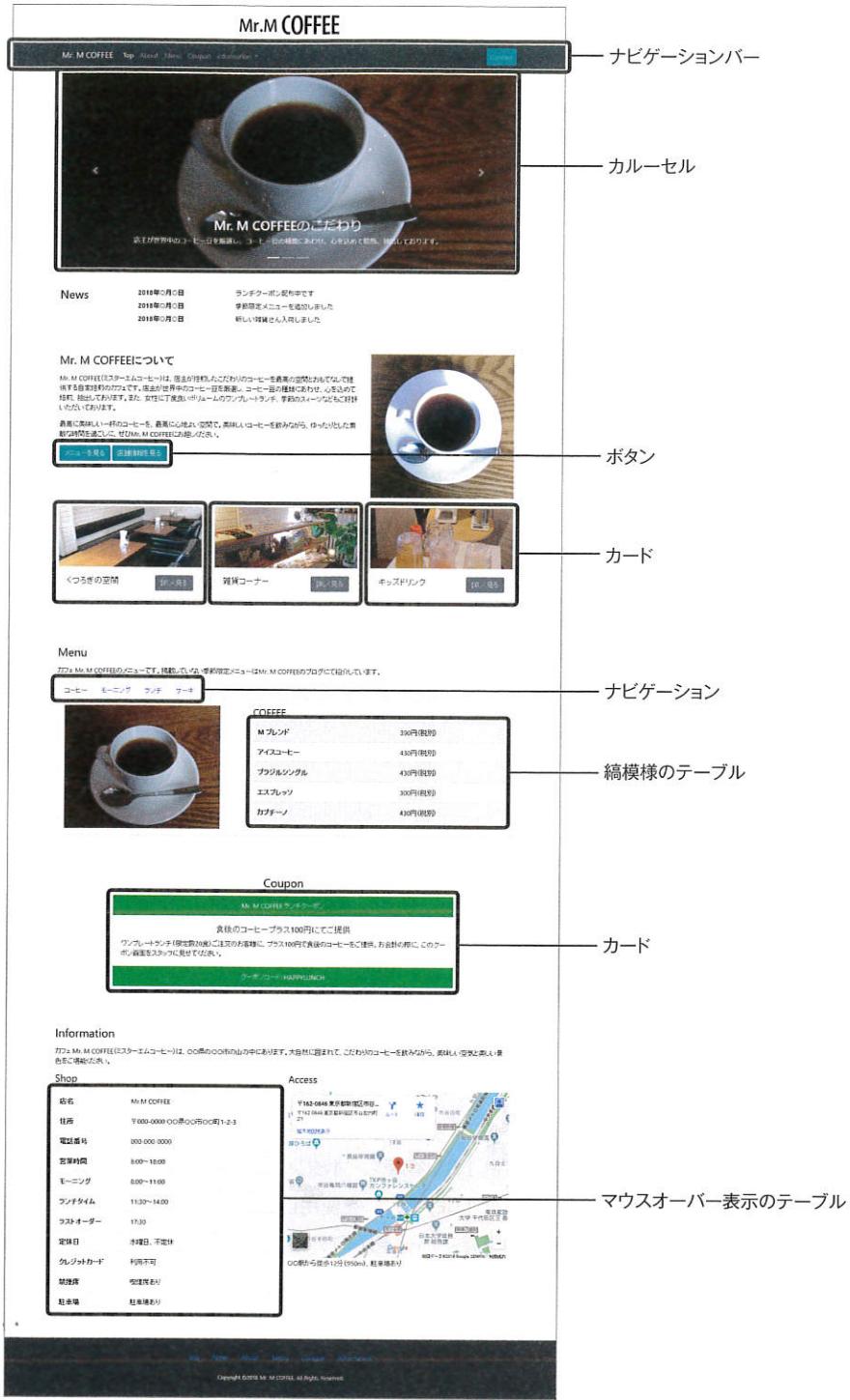
9.1.3 使用する主なコンポーネント

各ページに使用する主なコンポーネントを確認しておきましょう。

トップページに使用するコンポーネント

トップページには、ページ上から、ナビゲーションバー（P.161 参照）、ボタン（P.233 参照）、カルーセル（P.262 参照）、カード（P.124 参照）、ナビゲーション（P.150 参照）を使用します。また、カフェのメニュー表には縞模様のテーブル（P.77 参照）、店舗概要にはマウスオーバー表示のテーブル（P.80 参照）を使用します（図9-7）。

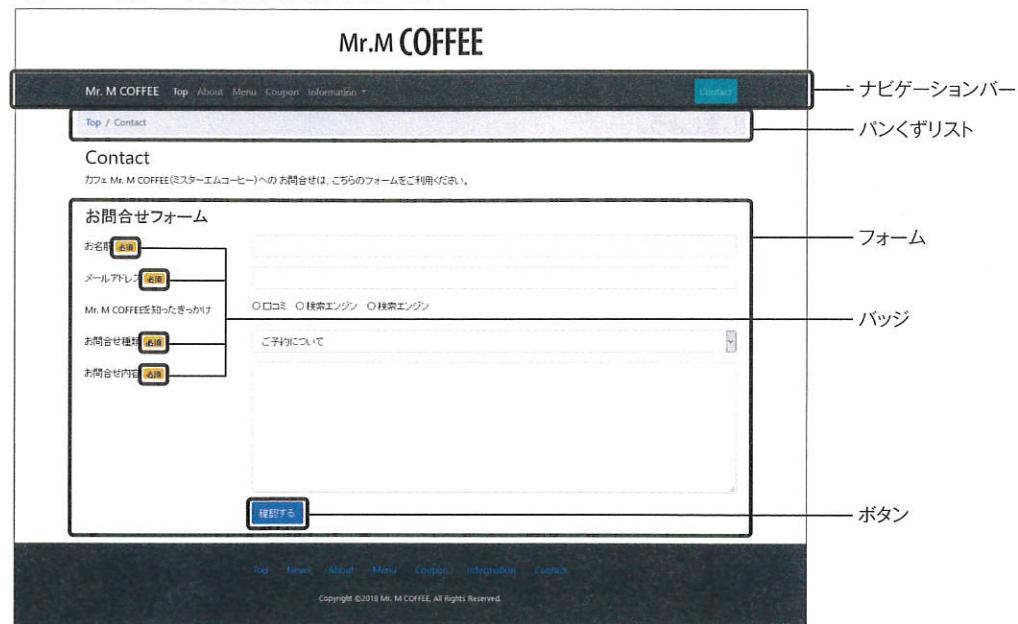
▼図9-7 トップページに使用する主なコンポーネント



下層ページに使用するコンポーネント

下層ページには、パンくずリスト（P.179 参照）とフォーム（P.196 参照）を使用します。ナビゲーションバーはトップページと共通です（図 9-8）。

▼図 9-8 下層ページに使用する主なコンポーネント



1
2
3
4
5
6
7
8
9
10

新規ファイル作成

「HTML の雛形」(P.17 参照) を参考に、必要なディレクトリを作成し、「Starter template」をもとにして新規ファイル「index.html」を作成しましょう。

9.2.1 head 要素の修正

Starter template 内の言語設定や、CSS や JavaScript など関連ファイルへのパスを修正した後、Bootstrap に定義されていないスタイルを追加するための CSS ファイルを「CSS」フォルダ内に作成し、title 要素の変更を行います（リスト 9-1）。

▼リスト 9-1 Starter template の修正（replace.html）

```
<!doctype html>
<html lang="ja">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<link rel="stylesheet" href="css/bootstrap.min.css">
<!-- 追加CSS -->
<link rel="stylesheet" href="css/custom.css"> _____①
<title>カフェ Mr. M COFFEE (ミスター・エム・コーヒー) </title> _____②
</head>
<body>
<script src="js/jquery-3.3.1.slim.min.js"></script>
<script src="js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

まず、Bootstrap に定義されたスタイル以外に、カスタムで必要に応じたスタイルを追加するための CSS ファイルを作成しておきましょう。本章のサンプルで追加するスタイルについては、後の「ページ内リンクの位置調整」(P.421 参照) で説明します。css フォルダー内にスタイルを追加するための CSS ファイル **custom.css** を作成し、head 要素内に link 要素を追加して、追加スタイルを読み込めるようにします（①）。

次に、title 要素の内容をサンプルサイトに合わせて「カフェ Mr. M COFFEE (ミスター・エム・コーヒー)」に変更しておきましょう（②）。

9.2.2 基本構造の入力

Bootstrapでのモックアップ作成は、Webページを各コンテンツのエリアに区切って基本構造を作成し、その中にグリッドを作成してコンポーネントを組み込んでいくという流れをイメージすると良いでしょう。では、ワイヤーフレームを参考に、基本構造の入力を行います。

■ 各コンテンツのエリア分け

まずは基本的なエリア分けを行います（リスト9-2）。

▼リスト9-2 基本構造の入力（structure.html）

```
<body>
<!-- ヘッダー -->
<header class="py-4">

</header>
<!-- /ヘッダー -->
<!-- ナビゲーションバー -->
<nav>

</nav>
<!-- /ナビゲーションバー -->
<!-- メイン -->
<main>
<!-- メインビジュアル -->
<div class="py-4">

</div>
<!-- / メインビジュアル -->
<!-- コンテンツ01 -->
<div class="py-4">

</div>
<!-- / コンテンツ01 -->
<!-- コンテンツ02 -->
<div class="py-4">

</div>
<!-- / コンテンツ02 -->
<!-- コンテンツ03 -->
<div class="py-4">

</div>
<!-- / コンテンツ03 -->
<!-- コンテンツ04 -->
<div class="py-4">
```

1
2
3
4
5
6
7
8
9
10

```
</div>
<!-- /コンテンツ04 -->
<!-- コンテンツ05 -->
<div class="py-4">

</div>
<!-- /コンテンツ05 -->
</main>
<!-- /メイン -->
<!-- フッター -->
<footer class="py-4">

</footer class="py-4">
<!-- /フッター -->
…中略…
</body>
```

ヘッダー部分を header 要素、ナビゲーションバー部分を nav 要素、メイン部分を main 要素、フッター部分を footer 要素でマークアップします。main 要素内には、メインビジュアル、コンテンツ01「News」、コンテンツ02「About」(Mr.M COFFEEについて)、コンテンツ03「Menu」、コンテンツ04「Coupon」、コンテンツ05「Information」の6つのコンテンツを作成するため、div 要素で6つのエリアに分けます。各エリアの上下パディングのサイズをあらかじめ調整し、コンテンツ間が詰まって見えないようにします。この例では、Spacing ユーティリティ(P.318 参照)の **py-* クラス**を追加し、各エリア内に上下パディングを設定しています。

9

SECTION

3

ヘッダーの作成

ヘッダーのエリアにはロゴ画像を水平方向中央に配置します（リスト 9-3）。本章のサンプルでは、ルートディレクトリの直下に「img」フォルダを作成し、必要な画像を格納しています。

▼リスト 9-3 ヘッダーの作成（header.html）

```
<!-- ヘッダー -->
<header class="py-4">
  <div class="container text-center"> ①
    <h1><a href="index.html"></a></h1>
  </div>
</header>
<!-- /ヘッダー -->
```

header 内のコンテンツをページの水平中央に配置するために、div 要素に **container クラス**（P.23 参照）を追加し、レイアウト設定します（①）。また、ボックス内のインライン要素を水平方向中央揃えにする Text ユーティリティ（P.347 参照）の **text-center クラス**を設定し、ロゴの画像を中央揃えにします。ロゴ画像には、一般的な Web ページのように img 要素を h1 要素でマークアップし、トップページ（index.html）へのリンクを設定しています（図 9-9）。

▼図 9-9 ロゴ画像が水平方向中央に配置されたヘッダー



9

4

ナビゲーションバーの作成

続いてナビゲーションバー（P.161 参照）のコンポーネントを使用して、ナビゲーションバーを作成します。

9.4.1 ナビゲーションバーのレイアウト

まず、ワイヤーフレームでナビゲーションバーのレイアウトを再確認しておきましょう。画面幅 md 以上（デスクトップ PC での閲覧時）では、メニューが横に広がって表示されます（図 9-10）。

▼図 9-10 ナビゲーションバーのワイヤーフレーム（画面幅 md 以上）



画面幅 sm 以下（モバイル端末での閲覧時）では、メニューが折り畳まれて表示されます（図 9-11）。

▼図 9-11 ナビゲーションバーのワイヤーフレーム（画面幅 sm 以下）



9.4.2 ナビゲーションバーの基本構成

本サンプルでは、暗い背景色で、スクロールするとページ上部に固定配置されるナビゲーションバーを作成します（リスト 9-4）。

▼リスト 9-4 ナビゲーションバーの基本構成（mockup-navbar-base.html）

```
<!-- ナビゲーションバー -->
<nav class="navbar navbar-expand-md navbar-dark bg-dark sticky-top"> ①
  <!-- サブコンポーネント -->
  <div class="container"> ②
    ここにサブコンポーネントが入ります
  </div>
  <!-- /サブコンポーネント -->
</nav>
<!-- /ナビゲーションバー -->
```

nav 要素に **navbar クラス**を指定してナビゲーションバーを作成します（①）。さらに、**navbar-dark クラス**と **bg-dark クラス**を追加して暗い背景色と明るい文字色を設定し、**sticky-top クラス**を追加してスクロールするとページ上部に固定される動きを設定します。ただし、sticky-top クラスで使用されるスタイル **position: sticky;** は、Internet Explorer などのブラウザではサポートされておらず、ナビゲーションが上部に固定されないことに注意してください。

また、navbar クラスが設定された nav 要素に **navbar-expand-md クラス**を追加することで、画面幅 sm 以下では折り畳み、画面幅 md 以上では広げて表示されるナビゲーションの動きを設定しています（②）。

ナビゲーションバー内のサブコンポーネントを水平方向中央に配置するために、nav 要素内の div 要素に **container クラス**（P.23 参照）を追加します（③）。

9.4.3 サブコンポーネントの組み込み

ナビゲーションバー内に組み込むサブコンポーネントとして、**ブランド**、**切り替えボタン**、**ナビゲーション**を使用します。

■ ブランドと切り替えボタンの作成

「Mr. M COFFEE」を表記する**ブランド**、画面幅によってナビゲーションメニューを折り畳み表示する**切り替えボタン**を作成します（リスト 9-5）。

▼リスト 9-5 サブコンポーネントの基本構成（mockup-navbar-subcomponent.html）

```
<!-- サブコンポーネント -->
<div class="container">
  <!-- ブランド -->
  <a class="navbar-brand" href="index.html">Mr. M COFFEE</a> ①
  <!-- 切り替えボタン -->
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbar-content" aria-controls="navbar-content" aria-expanded="false" aria-label="Toggle navigation"> ②
    <span class="navbar-toggler-icon"></span> ③
  </button>

  <!-- ナビゲーション -->
  ここにナビゲーションのメニューが入ります
</div>
</div>
<!-- /サブコンポーネント -->
```

まず、トップページ（index.html）へのリンクを設定した a 要素に **navbar-brand クラス**を追加し、店名「Mr. M COFFEE」のブランドを作成します（①）。

1
2
3
4
5
6
7
8
9
10

次に、button要素に**navbar-toggler**クラス、属性**type="button"**、**data-toggle="collapse"**、**data-target="**(ナビゲーションのID)を追加し、ナビゲーションバー内のナビゲーションの表示を切り替えるボタンを作成します(②)。また、アクセシビリティへの配慮として**aria-***属性を追加し、スクリーンリーダーなどの支援技術に対してコンポーネントの状態を伝えましょう。

button要素内には、span要素に**navbar-toggler-icon**クラスを追加し、ハンバーガーアイコン☰を表示させます(③)。

■ ナビゲーションの作成

切り替えボタンの後に、表示切り替えの対象となるナビゲーションを作成します(リスト9-6)。

▼リスト9-6 ナビゲーションメニューの組み込み(mockup-navbar-navmenu.html)

```
<!-- ナビゲーション -->
<div class="collapse navbar-collapse" id="navbar-content"> _____ ①
  <!-- ナビゲーションメニュー -->
  <!-- 左側メニュー：トップページの各コンテンツへのリンク -->
  <ul class="navbar-nav mr-auto"> _____ ②
    <li class="nav-item active"> _____ ③
      <a class="nav-link" href="#">Top <span class="sr-only">(current)</span></a> _____ ④
    </li>
    <li class="nav-item"> _____
      <a class="nav-link" href="#">About</a>
    </li>
    <li class="nav-item"> _____
      <a class="nav-link" href="#">Menu</a>
    </li>
    <li class="nav-item"> _____
      <a class="nav-link" href="#">Coupon</a>
    </li>
    <li class="nav-item"> _____
      <a class="nav-link" href="#">Information</a>
    </li>
  </ul>

  <!-- 右側メニュー：Contactページへのリンク -->
  <ul class="navbar-nav"> _____ ②
    <li class="nav-item"> _____ ③
      <a href="contact.html" class="nav-link btn btn-info">Contact</a>
    </li>
  </ul>
<!-- /ナビゲーションメニュー -->
</div>
```

まず、div要素に**collapseクラス**と**navbar-collapseクラス**を追加し、ナビゲーションバー内に切り替え表示されるナビゲーションの枠を作成します（①）。このナビゲーションの枠にはid属性を設定し、先に作成した切り替えボタンのdata-target属性と値を一致させることで、表示切り替えの対象とします。

次に、ul要素に**navbar-navクラス**を追加し、ナビゲーションのメニュー部分を作成します（②）。本サンプルでは、2つのメニューを作成し、左側メニューにSpacingユーティリティ（P.318参照）の**mr-autoクラス**を追加して右側に自動マージンを設定します。これによって、右側メニューは右寄せにレイアウトされます。なお、左側メニューにはトップページの各コンテンツへのリンクを表示し、右側メニューには下層ページ（contact.html）へのリンクをボタン（P.233参照）として表示します。

ナビゲーション内のli要素には**nav-itemクラス**を追加し、メニューの各項目を作成します（③）。現在位置にある項目には**activeクラス**を追加します。li要素内のa要素に**nav-linkクラス**を追加し、メニュー内のリンクを作成します（④）。また、そのリンク先が現在位置の場合は、スクリーンリーダー用ユーティリティ（P.354参照）の**sr-onlyクラス**を加えた非表示要素を使ってスクリーンリーダーなどに伝えましょう。

■ ドロップダウンの組み込み

ナビゲーションメニューのうち、「Information」の項目にドロップダウンを組み込みます（リスト9-7）。

▼リスト9-7 ドロップダウンの組み込み（mockup-navbar-navitem-dropdown.html）

```
<!-- 左側メニュー：トップページの各コンテンツへのリンク -->
<ul class="navbar-nav mr-auto">
  ...中略...
  <!-- ドロップダウン -->
  <li class="nav-item dropdown"> _____ ①
    <a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false"> _____ ②
      Information
    </a>
    <div class="dropdown-menu" aria-labelledby="navbarDropdown"> _____ ③
      <a class="dropdown-item" href="#">Shop</a> _____ ④
      <a class="dropdown-item" href="#">Access</a>
    </div>
  </li>
</ul>
```

まず、li要素に**nav-itemクラス**、**dropdownクラス**を追加し、メニュー項目内にドロップダウン（P.244参照）を作成します（①）。li要素内のnav-linkクラスが設定されたa要素には、**dropdown-toggleクラス**、属性**data-toggle="dropdown"**などを追加し、ドロップダウンの切り替えボタンを作成します（②）。また、アクセシビリティへの配慮として**role属性**と**aria-*属性**を追加し、スクリーンリーダーなどの支援技術に対してコンポーネントの役割や状態を伝えましょう。

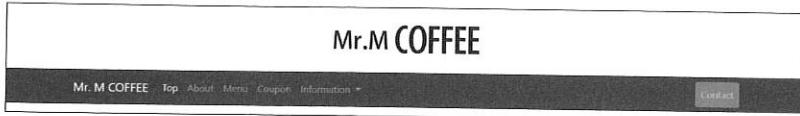
次に、div要素に**dropdown-menuクラス**を追加して、ドロップダウン表示される部分のメニューを作成します（③）。各メニュー項目は、a要素に**dropdown-itemクラス**を追加して作成します（④）。

1
2
3
4
5
6
7
8
9
10

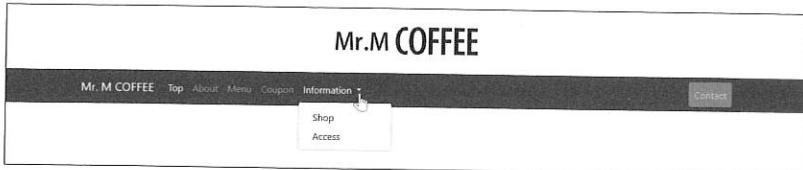
9.4.4 ナビゲーションバーの完成図

以上で、画面幅 md 以上になるとメニューが横に広がり、sm 以下になるとメニューが折り畳まれてアイコン表示されるナビゲーションバーが完成します（図 9-12～図 9-15）。

▼図 9-12 作成されたナビゲーションバー（画面幅 md 以上）



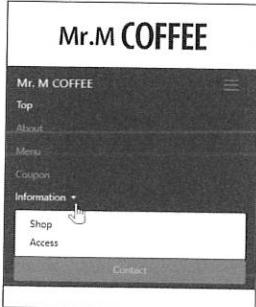
▼図 9-13 画面幅 md 以上でのドロップダウン表示



▼図 9-14 作成されたナビゲーションバー（画面幅 sm 以下）



▼図 9-15 画面幅 sm 以下のナビゲーションメニューの表示切り替え



9

SECTION

5

メインビジュアルの作成

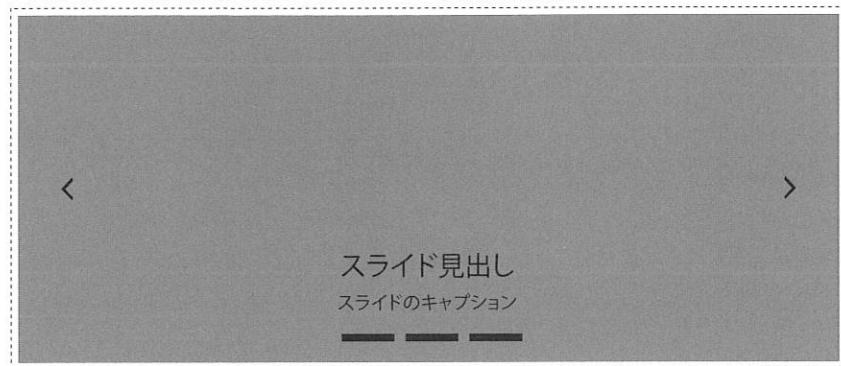
Bootstrap には、メインビジュアルの表示に便利なコンポーネントとして、**ジャンボトロン** (P.108 参照) や**カルーセル** (P.262 参照) が用意されています。本章ではカルーセルを使用し、メインビジュアルをスライドショーで表示させるモックアップを作成します。

9.5.1

メインビジュアルのレイアウト

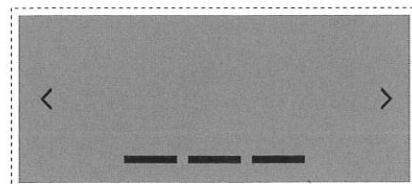
まず、ワイヤーフレームでメインビジュアルのレイアウトを再確認しておきましょう。画面幅 md 以上（デスクトップ PC での閲覧時）では、各スライドにキャプションが表示されます（図 9-16）。

▼図 9-16 メインビジュアルのワイヤーフレーム（画面幅 md 以上）



画面幅 sm 以下（モバイル端末での閲覧時）では、各スライドのキャプションが非表示になります（図 9-17）。

▼図 9-17 ナビゲーションバーのワイヤーフレーム（画面幅 sm 以下）



9.5.2 メインビジュアルの基本構成

メインビジュアル内にカルーセルの基本構成を作成します。本章で作成するメインビジュアルでは、カルーセル内に3枚のスライドとキャプション、インジケーター、コントローラーを表示します（リスト9-8）。

▼リスト9-8 メインビジュアルの基本構成（mainvisual-base.html）

```
<!-- メインビジュアル -->
<div class="py-4">
  <div class="container"> _____①
    <!-- カルーセル外枠 -->
    <div id="main_visual" class="carousel slide" data-ride="carousel"> _____②
      <!-- インジケーター -->
      <ol class="carousel-indicators"> _____③
        各インジケーターが入ります
      </ol>
      <!-- / インジケーター -->
      <!-- カルーセル内枠 -->
      <div class="carousel-inner"> _____④
        <!-- スライド01 -->
        <div class="carousel-item active"> _____
          1枚目のスライド画像とキャプションが入ります
        </div>
        <!-- / スライド01 -->
        <!-- スライド02 -->
        <div class="carousel-item"> _____⑤
          2枚目のスライド画像とキャプションが入ります
        </div>
        <!-- / スライド02 -->
        <!-- スライド03 -->
        <div class="carousel-item"> _____
          3枚目のスライド画像とキャプションが入ります
        </div>
        <!-- / スライド03 -->
      </div>
      <!-- / カルーセル内枠 -->
      <!-- コントローラー -->
      <a class="carousel-control-prev" href="#main_visual" role="button" data-slide="prev"> _____⑥
        前に戻るコントローラー「<」が入ります
      </a>
      <a class="carousel-control-next" href="#main_visual" role="button" data-slide="next"> _____⑦
        次に進むコントローラー「>」が入ります
      </a>
      <!-- / コントローラー -->
    </div>
    <!-- / カルーセル -->
  </div>
</div>
```

```
<!-- / メインビジュアル -->
```

まず、メインビジュアル内のコンテンツをページの水平方向中央に配置するために、div要素に**containerクラス**を追加します（①）。

次に、div要素に**carouselクラス**、**slideクラス**と属性**data-slide="carousel"**を追加し、カルーセル外枠を作成します（②）。カルーセル外枠には、インジケーターやコントローラーのリンク先として指定するためのid属性を設定しておきましょう。

続けてカルーセル内部を構成していきましょう。ol要素に**carousel-indicatorsクラス**を追加し、インジケーターの枠を作成します（③）。インジケーターの内部の作成については、次項で解説します。

インジケーターの後に、複数のスライドを含むカルーセル内枠を作成します。カルーセル内枠は、div要素に**carousel-innerクラス**を追加して作成します（④）。

カルーセル内枠の中に、画像とキャプションを含む各スライドを作成します。各スライドは、div要素に**carousel-itemクラス**を追加して作成します（⑤）。初期表示させるスライドには**activeクラス**の追加を忘れないようにしましょう。各スライドの作成については、後の項で解説します。

カルーセル内枠の後に、コントローラーの枠を作成します。a要素に**carousel-control-prevクラス**、属性**data-slide="prev"**を追加して、<アイコン（前に戻る）を表示するコントローラーを作成します（⑥）。同様に、**carousel-control-nextクラス**、属性**data-slide="next"**を追加して、>アイコン（次に進む）を表示するコントローラーを作成します（⑦）。各コントローラーのa要素のhref属性値は、カルーセル外枠のid属性値と一致させておきましょう。コントローラー内部の作成については、後の項で解説します。

9.5.3 インジケーターの組み込み

インジケーター内部の作成を行います。各インジケーターは、li要素に属性**data-target="（インジケーター外枠のID）"**、**data-slide-to="（0からはじまるスライド番号）"**を追加して作成します。初期表示させるスライドを示すインジケーターには**activeクラス**の追加を忘れないようにしましょう（リスト9-9）。

▼リスト9-9 インジケーターの組み込み（mainvisual-indicator.html）

```
<!-- インジケーター -->
<ol class="carousel-indicators">
  <li data-target="#main_visual" data-slide-to="0" class="active"></li>
  <li data-target="#main_visual" data-slide-to="1"></li>
  <li data-target="#main_visual" data-slide-to="2"></li>
</ol>
<!-- / インジケーター -->
```

9.5.4 各スライドの組み込み

カルーセル内枠に3枚のスライドを組み込みます（リスト9-10）。

▼リスト9-10 各スライドの組み込み (mainvisual-slide.html)

```

<!-- カルーセル内枠 -->
<div class="carousel-inner">
  <!-- スライド01 -->
  <div class="carousel-item active">
    —————①—————
    <div class="carousel-caption d-none d-md-block">—————②—————
      <h2>Mr. M COFFEEのこだわり</h2>
      <p>店主が世界中のコーヒー豆を厳選し、コーヒー豆の種類にあわせ、心を込めて焙煎、抽出しております。</p>
    </div>
  </div>
  <!-- / スライド01 -->
  <!-- スライド02 -->
  <div class="carousel-item">
    
    <div class="carousel-caption d-none d-md-block">
      <h2>Mr. M COFFEEのメニュー</h2>
      <p>コーヒーはもちろん、モーニングやワンプレートランチ、季節のスイーツなどもご好評いただいております</p>
    </div>
  </div>
  <!-- / スライド02 -->
  <!-- スライド03 -->
  <div class="carousel-item">
    
    <div class="carousel-caption d-none d-md-block">
      <h2>Mr. M COFFEEの空間</h2>
      <p>座り心地の良いソファと丁度良い高さのテーブル。くつろぎの空間を満喫してください。</p>
    </div>
  </div>
  <!-- / スライド03 -->
</div>
<!-- / カルーセル内枠 -->

```

carousel-item クラスが設定された各スライドの要素内に、画像とキャプションを配置します。

まず、親要素のサイズによってサイズが変化するレスポンシブ対応の画像を配置します。レスポンシブ対応の画像は img 要素に **img-fluid クラス** (P.69 参照) を追加して作成します (①)。

次に、見出しと段落を含むキャプションを配置します。キャプションは、div 要素に **carousel-caption クラス**を追加して作成します (②)。

なお本章のサンプルでは、ディスプレイサイズの小さい画面幅 sm 以下ではキャプションの表示が省略されるように設定します。carousel-caption クラスが設定された div 要素に、Display ユーティリティ (P.310 参照) の **d-none クラス**を追加して、いったんキャプションを非表示にします。その上で **d-md-block クラス** (P.311 参照) を追加し、画面幅 md 以上でブロックレベルの表示になるように設定し、キャプションが表示されるようにします。

9.5.5 各コントローラーの組み込み

コントローラーの内部を作成します（リスト 9-11）。

▼リスト 9-11 各スライドの組み込み（mainvisual-control.html）

```
<!-- コントローラー -->
<a class="carousel-control-prev" href="#main_visual" role="button" data-slide="prev">
  <span class="carousel-control-prev-icon" aria-hidden="true"></span> ①
  <span class="sr-only">前に戻る</span> ③
</a>
<a class="carousel-control-next" href="#main_visual" role="button" data-slide="next">
  <span class="carousel-control-next-icon" aria-hidden="true"></span> ②
  <span class="sr-only">次に進む</span> ③
</a>
<!-- / コントローラー -->
```

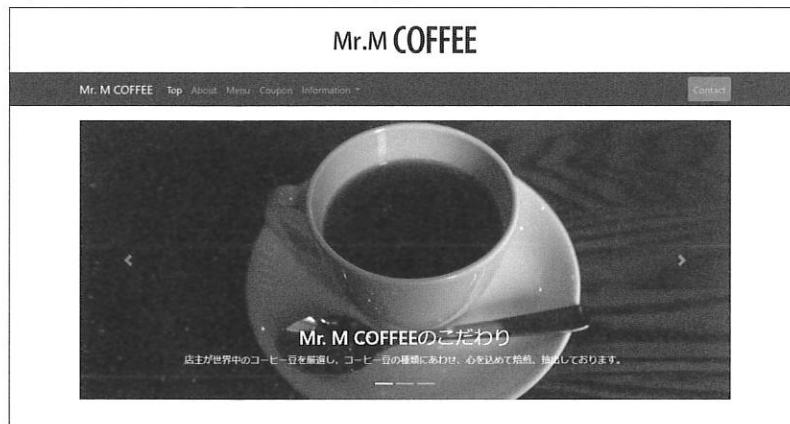
まず、**carousel-control-prev** クラスが設定された a 要素内に、「<」（前に戻る）アイコンを作成します。このアイコンは、span 要素に **carousel-control-prev-icon** クラスを追加して作成します。次に、**carousel-control-next** クラスが設定された a 要素内に、「>」（次に進む）アイコンを作成します。このアイコンは、span 要素に **carousel-control-next-icon** クラスを追加して作成します。なおそれぞれのアイコンには、アクセシビリティへの配慮としてスクリーンリーダー用ユーティリティ（P.354 参照）の **sr-only** クラスを追加して、「前に戻る」「次に進む」などの非表示テキストを加えておきましょう（③）。

9.5.6 メインビジュアルの完成図

以上で、3枚のスライドショーによるメインビジュアルが完成します（図 9-18～図 9-21）。

■ メインビジュアル（画面幅 md 以上）

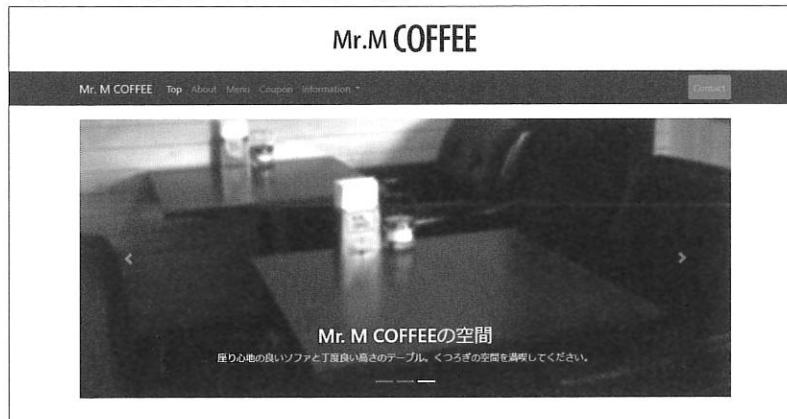
▼図 9-18 メインビジュアルのスライド 1 枚目



▼図9-19 メインビジュアルのスライド2枚目

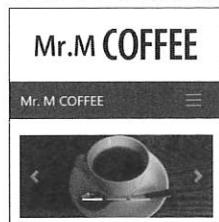


▼図9-20 メインビジュアルのスライド3枚目



■ メインビジュアル（画面幅sm以下）

▼図9-21 画面幅sm以下のメインビジュアル



9

SECTION

6

コンテンツ 01 (News) の作成

新着情報を表示するコンテンツ 01 「News」を作成します。このコンテンツにはグリッドレイアウトを利用した定義リスト（P.63 参照）を使用します。

9.6.1 コンテンツ 01 のレイアウト

まず、ワイヤーフレームでコンテンツ 01 のレイアウトを再確認しておきましょう。画面幅 md 以上（デスクトップ PC での閲覧時）では、コンテンツの見出し（h3 要素）、日付（dt 要素）、説明（dd 要素）が水平方向に横並びになります（図 9-22）。

▼図 9-22 コンテンツ 01 のワイヤーフレーム（画面幅 md 以上）

News	2017年〇月〇日	〇〇〇〇〇〇〇〇〇〇〇〇〇
	2017年〇月〇日	〇〇〇〇〇〇〇〇〇〇〇〇〇
	2017年〇月〇日	〇〇〇〇〇〇〇〇〇〇〇〇〇

画面幅 sm 以下（モバイル端末での閲覧時）では、コンテンツの見出し（h3 要素）、日付（dt 要素）、説明（dd 要素）が垂直方向に縦並びになります（図 9-23）。

▼図 9-23 コンテンツ 01 のワイヤーフレーム（画面幅 sm 以下）

News
2017年〇月〇日
〇〇〇〇〇〇〇〇〇〇〇〇〇
2017年〇月〇日
〇〇〇〇〇〇〇〇〇〇〇〇〇
2017年〇月〇日
〇〇〇〇〇〇〇〇〇〇〇〇〇

9.6.2 コンテンツ 01 の構成

ではコンテンツ 01 を作成していきましょう。コンテンツ 01 は、Bootstrap の柔軟なグリッドシステムを確認できるシンプルな例になっています（リスト 9-12）。

1
2
3
4
5
6
7
8
9
10

▼リスト9-12 コンテンツ01の作成(contents-01.html)

```

<!-- コンテンツ01 -->
<div class="py-4">
  <section id="news"> _____ ①
    <div class="container"> _____ ②
      <div class="row"> _____
        <!-- 左側カラム (画面幅md以上) -->
        <div class="col-md-2"> _____
          <h3>News</h3>
        </div>
        <!-- 右側カラム (画面幅md以上) -->
        <div class="col-md-10"> _____
          <dl class="row"> _____ ③
            <dt class="col-md-3">2017年〇月〇日</dt> _____ ④
            <dd class="col-md-9">ランチクーポン配布中です</dd>
            <dt class="col-md-3">2017年〇月〇日</dt>
            <dd class="col-md-9">季節限定メニューを追加しました</dd>
            <dt class="col-md-3">2017年〇月〇日</dt>
            <dd class="col-md-9">新しい雑貨さん入荷しました</dd>
          </dl>
        </div>
      </div>
    </div>
  </section>
</div>
<!-- /コンテンツ01 -->

```

まず、コンテンツ01内をsection要素でマークアップし、id属性値を**news**と設定します（①）。このようにコンテンツごとにid名を付けておくと、アンカーリンクを設定したり、スタイルを調整したりする際に便利です。

次に、section要素内のコンテンツを水平中央に配置するために、div要素に**containerクラス**を追加します（②）。

このcontainerクラスを設定した要素内に、グリッドレイアウト（P.22参照）を組み込みます。グリッドレイアウトは、div要素にグリッドレイアウトを行うための行を形成する**rowクラス**を追加し、子要素には**col-{ブレイクポイント}-* クラス**を追加して作成します。本章のサンプルでは、画面幅がsm以下のときに1カラムレイアウトに、画面幅がmd以上のときに左側が2列カラム（col-md-2）、右側が10列カラム（col-md-10）の2カラムレイアウトになるように設定しています（③）。

さらに右側カラム内には、グリッドレイアウトを利用した定義リストを入れ子にします。右側カラム内に、**rowクラス**を設定したdl要素を配置し、dt要素とdd要素に**col-{ブレイクポイント}-* クラス**を追加します。サンプルでは、3列カラム（col-md-3）+9列カラム（col-md-9）のレイアウトになるように設定しています（④）。

9.6.3 コンテンツ01の完成図

以上で、新着情報を表示するコンテンツ01「News」が完成です。

画面幅 md 以上では、コンテンツの見出し (h3 要素)、日付 (dt 要素)、説明 (dd 要素) が水平方向に横並びになります (図 9-24)。

▼図 9-24 コンテンツ 01 (画面幅 md 以上)

The screenshot shows the Mr. M COFFEE homepage. At the top is a navigation bar with links for 'Mr. M COFFEE' (the logo), 'Top', 'About', 'Menu', 'Coupon', 'Information', and 'Contact'. Below the navigation is a large image of a coffee cup and saucer. Overlaid on the image is the text 'Mr. M COFFEEのごだわり' and a subtitle: '店主が世界中のコーヒー豆を厳選し、コーヒー豆の種類にあわせ、心を込めて焙煎、抽出しております。'. Underneath the image is a news section titled 'News' with three items:

News	2018年〇月〇日	ランチクーポン配布中です
	2018年〇月〇日	季節限定メニューを追加しました
	2018年〇月〇日	新しい雑貨さん入荷しました

画面幅 sm 以下では、垂直方向に縦並びになります (図 9-25)。

▼図 9-25 コンテンツ 01 (画面幅 sm 以下)

The screenshot shows the same Mr. M COFFEE homepage as above, but viewed on a smaller screen (sm). The layout is now vertical. The news section is displayed as a single column:

News
2018年〇月〇日 ランチクーポン配布中です
2018年〇月〇日 季節限定メニューを追加しました
2018年〇月〇日 新しい雑貨さん入荷しました

1
2
3
4
5
6
7
8
9
10



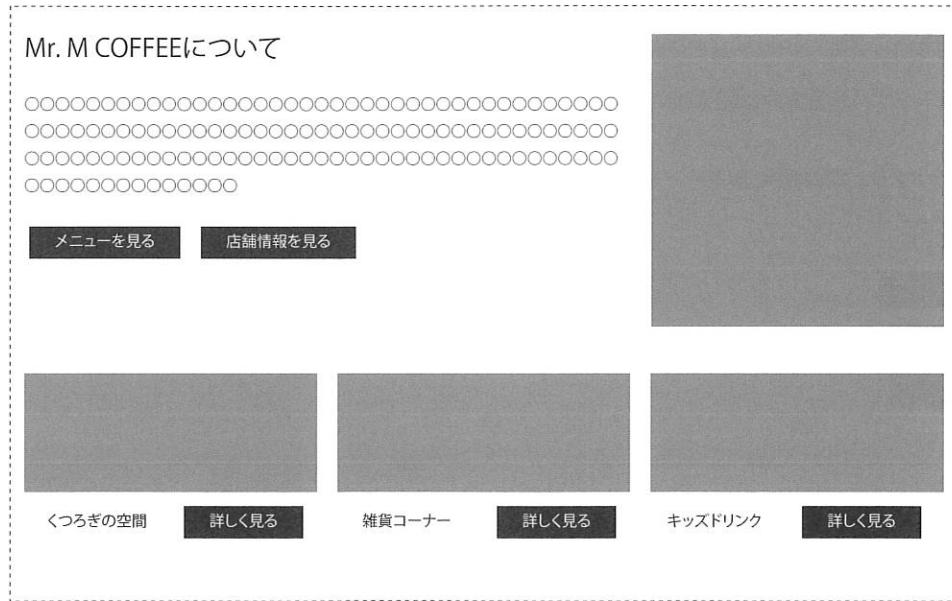
コンテンツ 02 (About) の作成

カフェの店舗や特長を紹介するコンテンツ 02 「About」を作成します。このコンテンツには、ボタン (P.233 参照) やカード (P.124 参照)、モーダル (P.283 参照) などのコンポーネントを使用します。

9.7.1 コンテンツ 02 のレイアウト

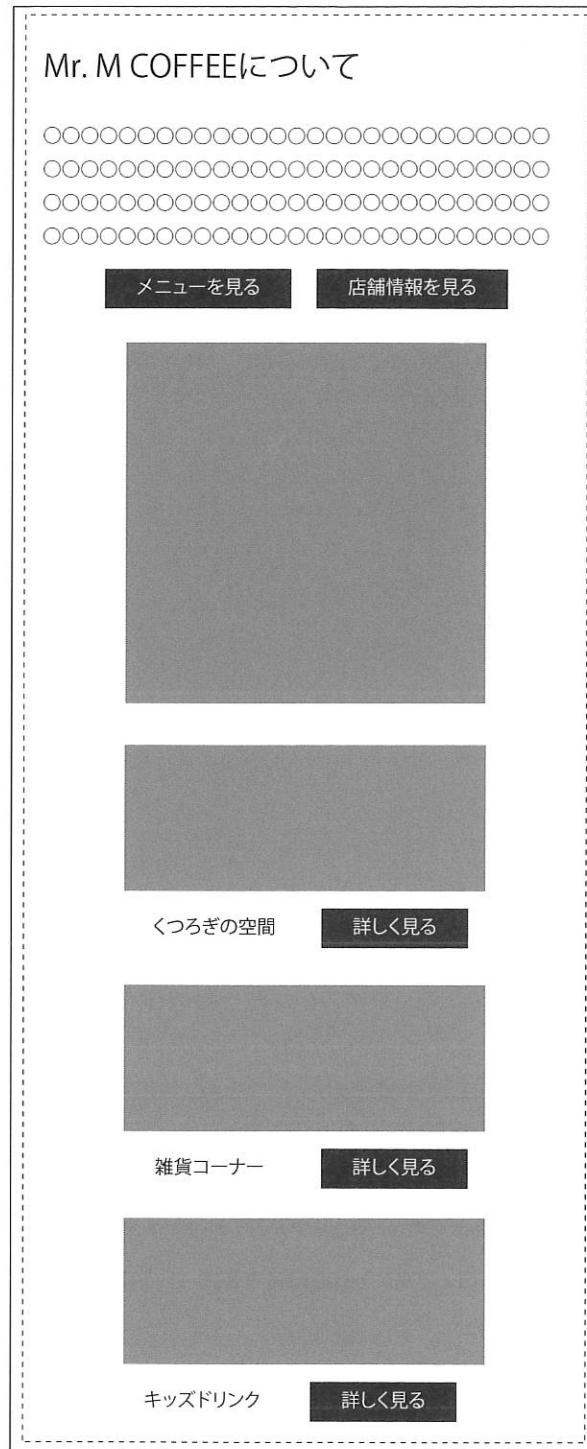
まず、ワイヤーフレームでコンテンツ 02 のレイアウトを再確認しておきましょう。画面幅 md 以上（デスクトップ PC での閲覧時）では、上段にコンテンツの見出しおよび紹介文と写真とが水平方向に横並びに、下段には 3 つのカードが水平方向に横並びになります（図 9-26）。

▼図 9-26 コンテンツ 02 のワイヤーフレーム（画面幅 md 以上）



画面幅 sm 以下（モバイル端末での閲覧時）では垂直方向に縦並びになります（図 9-27）。

▼図 9-27 コンテンツ 02 のワイヤーフレーム（画面幅 sm 以下）



9.7.2 コンテンツ02の構成

ではコンテンツ02を作成していきましょう。

■ コンテンツ02全体のレイアウト

上段にコンテンツの見出しおよび紹介文と写真、下段に3つのカードを配置します（リスト9-13）。

▼リスト9-13 コンテンツ02全体のレイアウト（contents-02-layout.html）

```
<!-- コンテンツ02 -->
<div class="py-4 bg-light"> _____
  <section id="about"> _____
    <div class="container"> _____
      <!-- 上段 -->
      <div class="row mb-4"> _____
        <div class="col-md-8 mb-3"> _____
          コンテンツ見出しおよび紹介文が入ります
        </div>
        <div class="col-md-4"> _____
          画像が入ります
        </div>
      </div>
      <!-- /上段 -->
      <!-- 下段 -->
      <div class="row"> _____
        <div class="col-md-4"> _____
          カードが入ります
        </div>
        <div class="col-md-4"> _____
          カードが入ります
        </div>
        <div class="col-md-4"> _____
          カードが入ります
        </div>
      </div>
      <!-- /下段 -->
    </div>
  </section>
</div>
<!-- /コンテンツ02 -->
```

ページ全体の背景色が単調にならないように、エリアを分けるdiv要素に**bg-lightクラス**（P.304参照）を追加し、背景色を明るいグレー（#f8f9fa）に設定します（①）。

コンテンツ02内をsection要素でマークアップし、id属性値を**about**と設定します（②）。section要素内のコンテンツを水平中央に配置するために、div要素に**containerクラス**を追加します（③）。このcontainerクラスを設定した要素内を上段と下段に分けて、それぞれにグリッドレイアウト（P.22参照）を組み込みます。グリッドレイアウトは、div要素にグリッドレイアウトを行うための行を形成する**rowクラス**を追加し、子要素には**col-1ブレイクポイント}-*クラス**を追加して作成します。

まず、上段のグリッドレイアウトは、画面幅がmd以上のときに左側が8列カラム（col-md-8）、右側が4列カラム（col-md-4）のレイアウトになるように設定しています（④）。また、Spacingユーティリティ（P.318参照）の**mb-*クラス**を追加し、下要素との間のスペースが詰まりすぎないようにマージンを適宜調整します。上段内の構成については、後の「コンテンツ上段の作成」で説明します。

次に、下段のグリッドレイアウトは、画面幅がmd以上のときに4列カラム（col-md-4）が3つ並ぶレイアウトになるように設定しています（⑤）。下段内の構成については、後の「コンテンツ下段の作成」で説明します。

■ コンテンツ上段の作成

コンテンツ上段には、見出しおよび紹介文と写真のレイアウトを作成していきます（リスト9-14）。

▼リスト9-14 コンテンツ02上段の作成（contents-02-upper.html）

```
<!-- 上段 -->
<div class="row mb-4">
  <div class="col-md-8 mb-3">
    <h3 class="mb-3">Mr. M COFFEEについて</h3> ①
    <p>Mr. M COFFEE（ミスター・エム・コーヒー）は、店主が焙煎したこだわりのコーヒーを最高の空間とおもてなしで提供する自家焙煎のカフェです。店主が世界中のコーヒー豆を厳選し、コーヒー豆の種類にあわせ、心を込めて焙煎、抽出しております。また、女性に丁度良いボリュームのワンプレートランチ、季節のスイーツなどもご好評いただいております。</p>
    <p>最高に美味しい一杯のコーヒーを、最高に心地良い空間で。美味しいコーヒーを飲みながら、ゆったりとした素敵な時間を過ごしに、ぜひMr. M COFFEEにお越しください。</p>
    <a href="#" class="btn btn-info">メニューを見る</a> ②
    <a href="#" class="btn btn-info">店舗情報を見る</a>
  </div>
  <div class="col-md-4">
     ③
  </div>
</div>
<!-- /上段 -->
```

このコンテンツの見出しどなるh3要素には、Spacingユーティリティ（P.318参照）の**mb-*クラス**を追加し、下要素との間のスペースが詰まりすぎないようにマージンを適宜調整します（①）。a要素に**btnクラス**と**btn-{色の種類}クラス**（P.233参照）を追加し、他コンテンツに移動するためのボタンを作成します（②）。img要素には**img-fluidクラス**（P.69参照）を追加し、画像サイズをレスポンシブ対応させます（③）。

1
2
3
4
5
6
7
8
9
10

■ コンテンツ下段の作成

コンテンツ下段には、3つのカードを作成していきます（リスト9-15）。

▼リスト9-15 コンテンツ02下段の作成（contents-02-lower.html）

```

<!-- 下段 -->
<div class="row">
  <div class="col-md-4">
    <!-- カード01 -->
    <div class="card mb-3"> _____①
       _____②
      <!-- カードの本文エリア -->
      <div class="card-body d-flex justify-content-between"> _____③
        <h4 class="card-title">くつろぎの空間</h4> _____④
        <button type="button" class="btn btn-secondary"> _____⑤
          詳しく見る
        </button>
      </div>
    </div>
  </div>
  <div class="col-md-4">
    <!-- カード02 -->
    <div class="card mb-3"> _____①
       _____②
      <!-- カードの本文エリア -->
      <div class="card-body d-flex justify-content-between"> _____③
        <h4 class="card-title">雑貨コーナー</h4> _____④
        <button type="button" class="btn btn-secondary"> _____⑤
          詳しく見る
        </button>
      </div>
    </div>
  </div>
  <div class="col-md-4">
    <!-- カード03 -->
    <div class="card mb-3"> _____①
       _____②
      <!-- カードの本文エリア -->
      <div class="card-body d-flex justify-content-between"> _____③
        <h4 class="card-title">キッズドリンク</h4> _____④
        <button type="button" class="btn btn-secondary"> _____⑤
          詳しく見る
        </button>
      </div>
    </div>
  </div>
</div>
<!-- /下段 -->
```

まず、div 要素に **card クラス** (P.124 参照) を追加し、カードを作成します (①)。各カードには Spacing ユーティリティ (P.318 参照) の **mb-* クラス**を追加し、下要素との間のスペースが詰まりすぎないようにマージンを適宜調整します。img 要素には **img-fluid クラス** (P.69 参照) を追加し、画像サイズをレスポンシブ対応させます (②)。

次に、カードの本文エリアを作成します。本文エリアは、div 要素に **card-body クラス** (P.124 参照) を追加して作成します (③)。また Flex ユーティリティ (P.322 参照) の **d-flex クラス**、**justify-content-between クラス**を追加し、子要素の h4 要素と button 要素を本文エリアの両端から等間隔に揃えてレイアウトします。

本文エリア内の h4 要素には、**card-title クラス** (P.124 参照) を追加し、カードの見出しを作成します (④)。button 要素には **btn クラス**、**btn-{ 色の種類 } クラス** (P.233 参照) と属性 **type="button"** を追加し、「詳しく見る」ボタンを作成します (⑤)。このボタンには、後でモーダルの切り替えボタンとして設定を追加します。

■ モーダルの組み込み

コンテンツ下段内の「詳しく見る」ボタンを押すと、店舗の詳細情報がモーダルウィンドウに表示されるように、モーダル (P.283 参照) を組み込みます。コンテンツ下段の後にモーダルを作成していきます (リスト 9-16)。

▼リスト 9-16 モーダルの組み込み (contents-02-modal-window.html)

```
<!-- コンテンツ02 -->
<!-- 上段 -->
<div class="row mb-4">
  ...中略...
</div>
<!-- /上段 -->
<!-- 下段 -->
<div class="row">
  ...中略...
</div>
<!-- /下段 -->
<!-- モーダル -->
<!-- モーダル01 -->
<div class="modal fade" id="modal01" tabindex="-1" role="dialog" aria-labelledby="modal01-label" aria-hidden="true"> ❶
  <div class="modal-dialog modal-dialog-centered" role="document"> ❷
    <div class="modal-content"> ❸
      <div class="modal-header"> ❹
        <h5 class="modal-title" id="modal01-label">くつろぎの空間</h5> ❺
        <button type="button" class="close" data-dismiss="modal" aria-label="Close"> ❻
          <span aria-hidden="true">&times;</span> ❼
        </button>
      </div>
      <div class="modal-body"> ❽
        <p class="text-center"></p>
      </div>
    </div>
  </div>
</div>
```

1
2
3
4
5
6
7
8
9
10

```

<p>店主がこだわった家具たちです。座り心地の良いソファと丁度良い高さのテーブル。</p>
<!-- つろぎの空間を満喫してください。 -->
</div>
<div class="modal-footer"> _____ ⑨
    <button type="button" class="btn btn-secondary" data-dismiss="modal">Close</button> _____ ⑩
</div>
</div>
</div>
</div>
<!-- モーダル② -->
<div class="modal fade" id="modal02" tabindex="-1" role="dialog" aria-labelledby="modal01-label" _____ ①
aria-hidden="true"> _____ ①
...中略...
</div>
<!-- モーダル③ -->
<div class="modal fade" id="modal03" tabindex="-1" role="dialog" aria-labelledby="modal01-label" _____ ①
aria-hidden="true"> _____ ①
...中略...
</div>
<!-- / モーダル -->
<!-- / コンテンツ② -->

```

まずモーダルウインドウの外枠を、div要素に**modalクラス**を追加して作成します（①）。また**fadeクラス**を追加することで、モーダルウインドウがページ上部からスライドしながらフェードインするアニメーションを設定しています。

次に、div要素に**modal-dialogクラス**と**modal-dialog-centeredクラス**を追加し、ウインドウの垂直方向中央に表示されるモーダルのダイアログ本体を作成します。さらに属性**role="dialog"**を追加して、この要素の役割がダイアログであることをスクリーンリーダーに伝えます（②）。

modal-dialogクラスが設定された要素内に、div要素に**modal-contentクラス**を追加してダイアログのコンテンツ部分を作成します（③）。ダイアログのコンテンツ内部には、ヘッダーと本文とフッターを構成します。

ヘッダーの部分は、div要素に**modal-headerクラス**を追加して作成します（④）。ヘッダー内には、見出しと閉じるボタンを配置しましょう。本章のサンプルでは、h5要素に**modal-titleクラス**を追加して見出しを作成しています（⑤）。閉じるボタンは、ボタン（P.233参照）のコンポーネントに、**closeクラス**と属性**data-dismiss="modal"**を追加して作成します（⑥）。またこのボタンに**aria-label属性**を追加して、スクリーンリーダーなどの支援技術に対するラベル付けを行います。閉じるボタン内には「×」アイコンを表示させます。このアイコンは、span要素に属性**aria-hidden="true"**を追加してスクリーンリーダーの対象から外しておきます（⑦）。

本文の部分は、div要素に**modal-bodyクラス**を追加して作成します（⑧）。

フッターの部分は、div要素に**modal-footerクラス**を追加して作成します（⑨）。フッター内には、閉じるボタンを配置します。閉じるボタンは、ボタンのコンポーネントに属性**data-dismiss="modal"**を追加して作成します（⑩）。

最後に、コンテンツ下段の「詳しく見る」ボタンをモーダルの切り替えボタンとして設定します（リスト9-17）。

▼リスト9-17 モーダルの組み込み(contents-02-modal-button.html)

```

<!-- コンテンツ02 -->
<!-- 上段 -->
<div class="row mb-4">
…中略…
</div>
<!-- /上段 -->
<!-- 下段 -->
<div class="row">
  <div class="col-md-4">
    <div class="card mb-3">
      …中略…
      <div class="card-body d-flex justify-content-between">
        <h4 class="card-title">くつろぎの空間</h4>
        <button type="button" class="btn btn-secondary" data-toggle="modal" data-target="#modal01">
          詳しく見る
        </button>
      </div>
    </div>
    <div class="col-md-4">
      <div class="card mb-3">
        …中略…
        <div class="card-body d-flex justify-content-between">
          <h4 class="card-title">雑貨コーナー</h4>
          <button type="button" class="btn btn-secondary" data-toggle="modal" data-target="#modal02">
            詳しく見る
          </button>
        </div>
      </div>
      <div class="col-md-4">
        <div class="card mb-3">
          …中略…
          <div class="card-body d-flex justify-content-between">
            <h4 class="card-title">キッズドリンク</h4>
            <button type="button" class="btn btn-secondary" data-toggle="modal" data-target="#modal03">
              詳しく見る
            </button>
          </div>
        </div>
      </div>
    </div>
  <!-- /下段 -->
  <!-- モーダル -->
  <!-- モーダル01 -->
<div class="modal fade" id="modal01" tabindex="-1" role="dialog" aria-labelledby="modal01-label" ↪
  aria-hidden="true">
  …中略…

```

1
2
3
4
5
6
7
8
9
10

```

</div>
<!-- モーダル02 -->
<div class="modal fade" id="modal02" tabindex="-1" role="dialog" aria-labelledby="modal01-label" aria-hidden="true">
...中略...
</div>
<!-- モーダル03 -->
<div class="modal fade" id="modal03" tabindex="-1" role="dialog" aria-labelledby="modal01-label" aria-hidden="true">
...中略...
</div>
<!-- / モーダル -->
<!-- / コンテンツ02 -->

```

「詳しく見る」ボタンの button 要素の **data-target** 属性値と、各モーダル外枠の div 要素の id 属性値とを一致させて、表示切り替えの対象とします。さらに、button 要素には属性 **data-toggle="modal"** を追加して、JavaScript 経由でモーダルウインドウを表示する機能を有効化します。

9.7.3 コンテンツ02の完成図

以上で、カフェの店舗や特長を紹介するコンテンツ02「About」が完成です。画面幅 md 以上では、上段にコンテンツの見出しおよび紹介文と写真とが水平に並び、下段に3つのカードが水平に並びます（図 9-28）。

▼図 9-28 コンテンツ02（画面幅 md 以上）



画面幅 sm 以下では、垂直方向に縦並びになります（図 9-29）。

1
2
3
4
5
6
7
8
9
10

▼図 9-29 コンテンツ02 (画面幅 sm 以下)

Mr. M COFFEEについて

Mr. M COFFEE(ミスターイムコーヒー)は、店主が焙煎したこだわりのコーヒーを最高の空間とおもてなしで提供する自家焙煎のカフェです。店主が世界中のコーヒー豆を厳選し、コーヒー豆の種類にあわせ、心を込めて焙煎、抽出しております。また、女性に丁度良いボリュームのワンプレートランチ、季節のスイーツなどもご好評いただいております。

最高に美味しい一杯のコーヒーを、最高に心地よい空間で。美味しいコーヒーを飲みながら、ゆったりとした素敵な時間を過ごしに、ぜひMr. M COFFEEにお越しください。

[メニューを見る](#) [店舗情報を見る](#)





くつろぎの空間 [詳しく見る](#)



雑貨コーナー [詳しく見る](#)



キッズドリンク [詳しく見る](#)

カード内のボタンをクリックすると、詳細情報を記載したモーダルが開きます（図 9-30）。

▼図 9-30 モーダルの組み込み (画面幅 md 以上)

