

ステップ30

リゅう がく せい  
留学生のための

# HTML5 & CSS3

ワークブック



Step 01 HTMLファイルとWebサーバー ..... 8

- 1.1 HTMLとは？
- 1.2 HTMLファイルの作成手順
- 1.3 HTMLファイルをWebブラウザで表示する
- 1.4 HTMLファイルを再編集する
- 1.5 インターネットに公開するには？

Step 02 タグの基本と改行 ..... 12

- 2.1 改行を指定する <br>
- 2.2 タグを記述するときのルール
- 2.3 <html>、<head>、<body>の記述
- 2.4 <head> ~ </head>に記述する内容
- 2.5 DOCTYPEの記述
- 2.6 正しいHTMLファイルの例

Step 03 見出しと段落 ..... 17

- 3.1 見出しの指定 <h1>、<h2>、……、<h6>
- 3.2 段落の指定 <p>
- 3.3 ヘアラインの描画 <hr>

Step 04 文字の装飾 ..... 22

- 4.1 太字 <b>、斜体 <i>、マーカー強調 <mark>、取り消し線 <del>
- 4.2 上付き文字 <sup>、下付き文字 <sub>

Step 05 画像の掲載 ..... 26

- 5.1 Webページに利用できる画像
- 5.2 画像ファイルの拡張子の確認
- 5.3 画像の配置 <img>
- 5.4 altテキストの指定
- 5.5 属性の記述ルール

## Step 06 リンクの作成 - 1 ..... 31

- 6.1 リンクの作成 <a>
- 6.2 画像リンクの作成
- 6.3 別サイトへのリンク

## Step 07 リンクの作成 - 2 ..... 35

- 7.1 元のページを維持したままリンク先を開く
- 7.2 パスの記述
- 7.3 ページ内リンクの作成

## Step 08 CSSの基本 - 1 ..... 40

- 8.1 CSSとは？
- 8.2 CSSを記述するときのルール
- 8.3 CSSの記述方法
- 8.4 指定方法① style属性でCSSを指定

## Step 09 CSSの基本 - 2 ..... 45

- 9.1 <head> ~ </head> にCSSを記述する場合
- 9.2 指定方法② 要素に対してCSSを指定
- 9.3 指定方法③ クラスに対してCSSを指定
- 9.4 指定方法④ IDに対してCSSを指定

## Step 10 文字書式のCSS - 1 ..... 52

- 10.1 文字サイズの指定 font-size
- 10.2 文字色の指定 color
- 10.3 文字の太さの指定 font-weight
- 10.4 斜体の指定 font-style
- 10.5 装飾線の指定 text-decoration
- 10.6 フォントの指定 font-family

## Step 11 文字書式のCSS - 2 ..... 57

- 11.1 行間の指定 line-height
- 11.2 文字書式の一括指定 font
- 11.3 行揃えの指定 text-align

## Step 12 いろ してい CSSの色指定 ..... 61

- 12.1 ひかり げんしよく 光の3原色
- 12.2 しんすう 16進数とは？
- 12.3 しんすう へんかん 16進数と10進数の変換
- 12.4 しんすう いろ してい RGBの16進数で色を指定するには？
- 12.5 もじいろ しんすう してい 文字色を「RGBの16進数」で指定
- 12.6 りよう いろ してい rgb()やrgba()を利用した色指定

## Step 13 はいけい 背景のCSS ..... 65

- 13.1 はいけいしよく してい 背景色の指定 background-color
- 13.2 はいけいがぞう してい 背景画像の指定 background-image
- 13.3 はいけいがぞう はいち 背景画像の配置 background-repeat、background-position
- 13.4 はいけいがぞう ちようせい 背景画像のサイズ調整

## Step 14 わくせん サイズと枠線のCSS ..... 70

- 14.1 してい サイズの指定 width、height
- 14.2 わくせん してい 枠線の指定 border
- 14.3 じようげ さゆう わくせん こべつ してい 上下左右の枠線を個別に指定

## Step 15 よほく 余白のCSS ..... 76

- 15.1 ないぶ よほく してい 内部余白の指定 padding
- 15.2 じようげ さゆう ないぶ よほく こべつ してい 上下左右の内部余白を個別に指定
- 15.3 がいぶ よほく してい 外部余白の指定 margin
- 15.4 ボックスのCSSのまとめ

## Step 16 かどまる かげ ほんとうめい 角丸、影、半透明のCSS ..... 83

- 16.1 かどまる してい 角丸の指定 border-radius
- 16.2 かげ してい 影の指定 box-shadow
- 16.3 ほんとうめい してい 半透明の指定 opacity

## Step 17 ようそ ようそ div要素とspan要素 ..... 89

- 17.1 ようそ つか かた div要素の使い方
- 17.2 ようそ つか かた span要素の使い方

## Step 18 回り込みのCSS ..... 95

- 18.1 回り込みの指定 float
- 18.2 回り込みの解除 clear
- 18.3 回り込みを使った要素の配置

## Step 19 フレックスボックスを使った配置 ..... 102

- 19.1 フレックスボックスの構成
- 19.2 アイテムの配置 (水平方向)
- 19.3 アイテムの配置 (垂直方向)
- 19.4 アイテムの折り返し

## Step 20 リンクのCSS ..... 108

- 20.1 リンクの書式指定
- 20.2 訪問済みリンク、マウスオーバー時の書式指定
- 20.3 閲覧履歴の削除について

## Step 21 CSSのまとめ ..... 112

- 21.1 CSSの記述方法
- 21.2 CSSの優先順位
- 21.3 文字書式のCSS
- 21.4 背景のCSS
- 21.5 ボックスのCSS
- 21.6 角丸、影、半透明のCSS
- 21.7 回り込みのCSS
- 21.8 フレックスボックスのCSS
- 21.9 リンクのCSS

## Step 22 表の作成 ..... 119

- 22.1 表作成の基本 <table>、<tr>、<td>
- 22.2 見出しのセル <th>
- 22.3 表内に画像を配置
- 22.4 キャプションの配置 <caption>

Step **23** ひょう しやう してい **表のCSS指定** ..... 124

- 23.1 セルの書式指定 しよしきしてい
- 23.2 キャプションの配置 ほいち caption-side
- 23.3 セルとセルの間隔 かんかく border-collapse
- 23.4 枠線の非表示 わくせん ひひやうじ について

Step **24** か けつごう **グループ化とセルの結合** ..... 130

- 24.1 行のグループ化 ぎやう か <thead>、<tbody>、<tfoot>
- 24.2 列のグループ化 れつ か <colgroup>
- 24.3 セルを横に連結 よこ れんけつ
- 24.4 セルを縦に連結 たて れんけつ

Step **25** さくせい かつよう **リストの作成と活用** ..... 136

- 25.1 リストの作成 さくせい <ul>、<li>
- 25.2 番号付きリストの作成 ばんごうつき さくせい <ol>、<li>
- 25.3 リストの階層化 かいそう か
- 25.4 マーカー指定のCSS しやう list-style-type
- 25.5 リストを活用したリンク かつよう

Step **26** さくせい **ページレイアウトの作成 - 1** ..... 143

- 26.1 ページ幅を固定してウィンドウ中央に表示する はば こてい ちゆうおう ひやうじ
- 26.2 レイアウトに用いる要素 もち ようそ <header>、<footer>、<nav>、<aside>
- 26.3 ヘッダーの作成 さくせい

Step **27** さくせい **ページレイアウトの作成 - 2** ..... 149

- 27.1 ナビゲーションメニューの作成 さくせい
- 27.2 メインコンテンツの作成 さくせい
- 27.3 フッターの作成 さくせい
- 27.4 リンク先ページの作成 さくせい

## Step 28 CSSファイルの活用 ..... 155

- 28.1 CSSファイルとは？
- 28.2 CSSファイルの作成手順
- 28.3 CSSファイルの読み込み
- 28.4 CSSファイルと<style>～</style>の併用

## Step 29 インラインフレームの作成 ..... 160

- 29.1 インラインフレームの作成
- 29.2 インラインフレームに指定できる属性
- 29.3 インラインフレーム内に外部サイトを表示
- 29.4 リンク先をインラインフレーム内に表示

## Step 30 フォームの作成 ..... 166

- 30.1 フォームとは？
- 30.2 ラベルによる関連付け <label>
- 30.3 テキストボックスとテキストエリア <input>、<textarea>
- 30.4 チェックボックスとラジオボタン <input>
- 30.5 セレクトメニュー <select>、<option>
- 30.6 ボタン <input>、<button>
- 30.7 フォーム領域の指定 <form>

演習問題の解答	177
索引 (Index)	223

### ◆ サンプルファイルと演習で使う画像のダウンロードURL

本書で紹介したサンプルファイル、ならびに演習で使用する画像は以下のURLからダウンロードできます。

http://-----

# HTML ファイルと Web サーバー

Web ページ（ホームページ）は HTML と呼ばれる言語で作成します。HTML 学習の第一歩となるステップ 01 では、HTML ファイルを作成する方法と Web の基礎知識について学習します。

## 1.1 HTML とは？

Web ページ（ホームページ）を作成するには、HTML と呼ばれるコンピュータ言語を学ぶ必要があります。コンピュータ言語と聞くと「難しそうだな……」と思うかもしれませんが、一般的なプログラミング言語と比べて HTML はかなり習得しやすい言語です。このため、初心者でも問題なく学習を進められると思います。

HTML は HyperText Markup Language を略したもので、その中身は単なるテキストファイルでしかありません。テキストファイルとは「文字だけで構成されるファイル」のことです。このため、Windows に用意されている「メモ帳」など、文字を編集できるアプリケーション（テキストエディタ）があれば、誰でも HTML ファイルを作成できます。



図 1-1 HTML ファイルと Web ページ

## 1.2 HTML ファイルの作成手順

それでは、HTML ファイルの作成手順を解説していきましょう。HTML ファイルは「メモ帳」などのテキストエディタで作成します。もちろん、「メモ帳」以外のテキストエディタを使用しても構いません。ただし、ファイルの拡張子に注意しなければなりません。



通常、テキストエディタで保存したファイルは、拡張子が「.txt」のテキストファイルになります。HTMLファイルを作成するときは、この拡張子を「.html」に変更する必要があります。このため、ファイルを保存するときに「ファイルの種類」を指定しなければいけません。

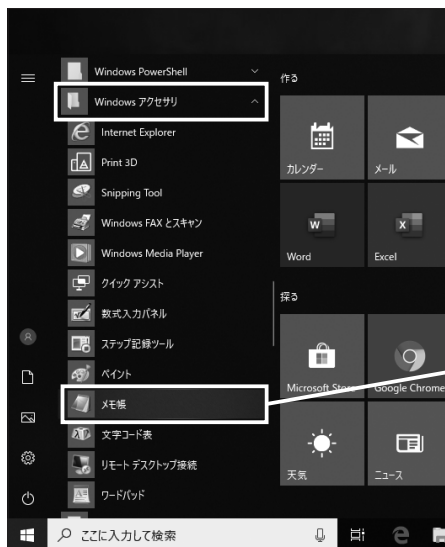


図 1-2

スタートメニューを開き、「アプリの一覧」から「Windows アクセサリ」→「メモ帳」を選択します。

これを選択

#### ◆漢字の読み

携帯電話、同じ、機能、追加、端末、以下、主な、特徴、紹介、利用、自由、追加、操作

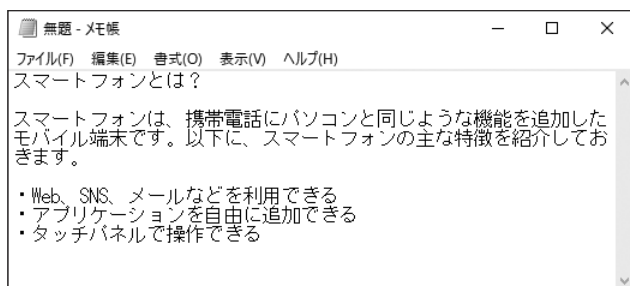


図 1-3

「メモ帳」が起動したら、キーボードを使ってテキスト(文章)を入力します。

ウィンドウの右端で文字を折り返すときは、[書式]メニューから「右端で折り返す」を選択します。



図 1-4

ファイルを保存するときは、「ファイルの種類」に「すべてのファイル」を選択し、ファイル名の最後に「.html」の拡張子を付けます。また、「文字コード」は「UTF-8」を指定するのが基本です。



文字コードに「UTF-8」を選択



図 1-5  
テキストがHTMLファイルとして保存されます。

## 1.3 HTML ファイルを Web ブラウザで表示する

続いては、作成したHTMLファイルをWebブラウザで閲覧するときの操作手順を解説します。この操作はとても簡単で、HTMLファイルのアイコンをダブルクリックするだけです。すると、Webブラウザ（Google Chrome など）が起動し、Webページが表示されます。



図 1-6 HTML ファイル

図 1-7 Web ブラウザに表示される

この結果（図 1-7）を見ると、「メモ帳」に記述した文章（図 1-3）がそのまま Web ブラウザに表示されていることを確認できます。このように、HTML ファイルに記述した文章は、そのままホームページとして表示させることが可能です。ただし、改行は反映されません。ホームページに表示される文章を改行するには、**<br>** というタグを記述する必要があります。

このように「タグの記述方法」を学ぶことが HTML の習得につながります。これについては、本書のステップ 02 以降で詳しく解説していきます。

## 1.4 HTML ファイルを再編集する

すでに保存されている HTML ファイルを編集するときも、「メモ帳」などのテキストエディタを使用します。ただし、普通に HTML ファイルをダブルクリックすると、Web ブラウザが起動してしまいます。

再編集を行うときは、HTML ファイルのアイコンをテキストエディタのウィンドウ内へドラッグ&ドロップします。すると、HTML ファイルの内容を編集できるようになります。

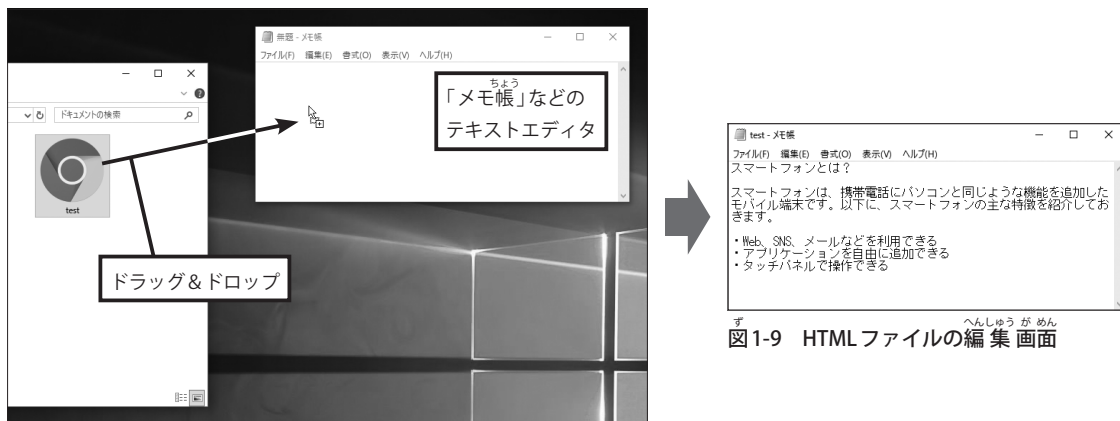


図1-8 ドラッグ&ドロップでファイルを開く

図1-9 HTMLファイルの編集画面

## 1.5 インターネットに公開するには？

作成したWebページをインターネットに公開するには、HTMLファイルをWebサーバーと呼ばれるコンピュータにコピーする必要があります。この作業をアップロードといいます。ただし、誰でもアップロードを実行できるわけではありません。アップロードを実行するには、Webサーバーを提供する業者（※1）と契約を交わし、自分専用のWebサーバーを確保しておく必要があります。

また、FTPクライアントと呼ばれるアプリケーションの使い方も習得しておかなければなりません。FTPクライアントは、「インターネット上にあるコンピュータ」（Webサーバー）と「自分のパソコン」の間でファイルのコピーを行うためのアプリケーションです。FTPクライアントの使い方や設定方法は、各自が契約したWebサーバーの利用手引などに記されているので、そちらを参照してください。

（※1）Webサーバーには、有料のものと無料のものがあります。

演習

- (1) 「メモ帳」などのテキストエディタに図1-3の文章を入力し、HTMLファイルとして保存してみましょう。
- (2) 演習(1)で保存したHTMLファイルをWebブラウザで閲覧してみましょう。
- (3) 演習(1)で保存したHTMLファイルを「メモ帳」などのテキストエディタで開いてみましょう。
- (4) 文章の一部を変更してから「上書き保存」を実行し、変更内容が反映されているかをWebブラウザで確認してみましょう。

## タグの基本と改行

ステップ01で解説したように、文章だけを記述したHTMLファイルをWebブラウザで閲覧すると、改行が無視されてしまいます。これを正しく改行するには<br>というタグを記述しなければなりません。ここでは「タグの基本」と「改行」について解説します。

### 2.1 改行を指定する <br>

HTMLファイルに記述した文章を改行して表示するには、その位置に<br>という文字を記述しなければいけません。たとえば、P9～10で紹介した例の場合、以下のように<br>を追加すると、正しい位置で改行して表示できるようになります。

#### ▼ sample02-1.html

- 1 スマートフォンとは？<br>
- 2 <br>
- 3 スマートフォンは、携帯電話にパソコンと同じような機能を追加したモバイル端末です。以下に、スマートフォンの主な特徴を紹介しておきます。<br>
- 4 <br>
- 5 ・Web、SNS、メールなどを利用できる<br>
- 6 ・アプリケーションを自由に追加できる<br>
- 7 ・タッチパネルで操作できる



図2-1 sample02-1.htmlをWebブラウザで閲覧した様子

このように、HTMLでは<>で囲まれた文字によりWebブラウザでの表示方法を指定します。この<>で囲まれた記述のことをタグと呼びます。

## 2.2 タグを記述するときのルール

<br>などのタグは、定められたルールに従って記述しなければなりません。ルールに違反している記述はタグと認識されず、「通常の文字」としてそのまま画面に表示されてしまいます。タグの記述に関するルールは、以下のとおりです。

- タグは半角文字で記述しなければいけません。
- <>内<sup>に</sup>記述する文字は、大文字でも小文字でも構いません。
- 「開始タグ」と「終了タグ」をペアにして記述します。

開始タグとは、<html>のように「前後に<>を記述したタグ」のことで、一方、終了タグは</html>のように「前後に</と>を記述したタグ」となります。タグを記述するときは、これらをペアにして<html>～</html>のような形にするのが基本です。

ただし、例外もあります。前ページで解説した<br>タグは、終了タグの記述が不要な少し特殊なタグとなります。

## 2.3 <html>、<head>、<body>の記述

続いては、「終了タグ」が必要になるタグの例として、html、head、bodyの3つのタグについて解説していきます。これらのタグは、HTMLファイルに必須のタグとなります。それぞれの役割を必ず覚えておいてください。

前ページでは、<br>タグだけを記述したHTMLを紹介しました。しかし、これは正しいHTMLとはいえません。というのも、HTMLでは<html>～</html>の中に内容を記述するというルールが定められているからです。また、<html>～</html>の中をheadとbodyの2つの領域に分ける必要もあります。これらを指定するタグが<head>～</head>と<body>～</body>です。

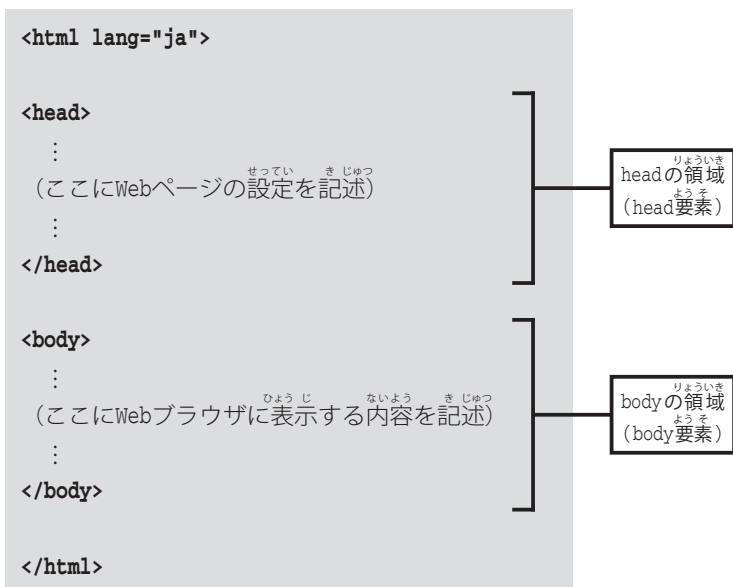
以上のルールをまとめると、HTMLは次ページに示したような構成になります。1行目のhtmlタグにある「lang="ja"」の記述は、言語が「日本語」であることを示しています。<html>だけを記述しても構いませんが、言語を明確にするためにも「lang="ja"」の記述を追加しておきましょう。



### ワンポイント

#### 要素について

「開始タグ」～「終了タグ」の範囲を要素と呼びます。たとえば、<body>～</body>の範囲は「body要素」となります。あわせて覚えておいてください。



<head> ~ </head>の領域には、Webページ全体に関わる設定などを記述します。一方、<body> ~ </body>の領域には、Webブラウザに表示する内容を記述します。

このルールに従うと、sample02-1.htmlに示したHTMLファイルは「<body> ~ </body>の領域内だけを記述した不完全なHTMLファイル」であると考えられます。

## 2.4 <head> ~ </head>に記述する内容

ここからは、<head> ~ </head>の中に記述する要素について解説していきます。まずはtitle要素について解説します。この要素はページタイトルを指定するもので、<title> ~ </title>の間に記述した文字がページタイトルとして扱われる仕組みになっています。Webブラウザでは、「タブ」の部分に<title> ~ </title>の間に記述した文字が表示されます。

<title>スマートフォンの紹介</title>

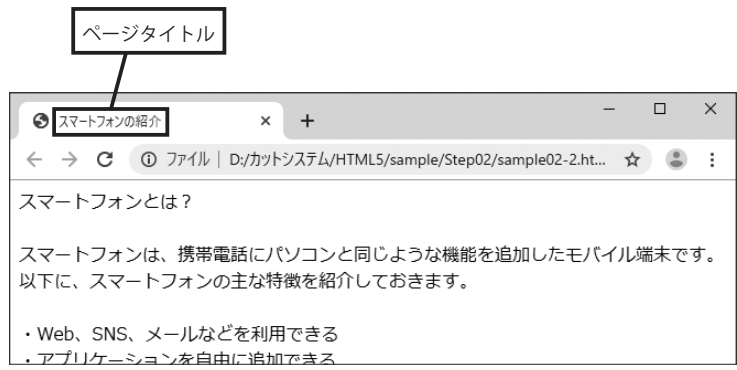


図2-2 ページタイトルの表示

また、<head>～</head>の領域には、文字コードを指定するmetaタグも記述するのが基本です。<br>と同様に、metaタグは「終了タグ」の記述が不要なタグとなります。

文字コードを指定するときは、以下のようにmetaタグを記述します。日本語の文字コードは「シフトJIS」や「UTF-8」などの種類があるため、正しく文字コードを指定しておかないとWebページが文字化けして表示される恐れがあります。

```
<meta charset="UTF-8">
```

上に示したmetaタグは、文字コードに「UTF-8」を指定した場合の記述です。他の文字コードを指定するときは、「UTF-8」の部分を以下の表のように書き換えてください。

### ■文字コードの指定

文字コード	metaタグの記述
UTF-8	UTF-8
シフトJIS	Shift_JIS

Windowsの「メモ帳」は、文字コードに「シフトJIS」(ANSI)が指定されている場合もあります。一方、Webの世界では「UTF-8」の文字コードを使用するのが一般的です。不要なトラブルを避けるためにも、文字コードに「UTF-8」を指定してから保存するようにしてください。

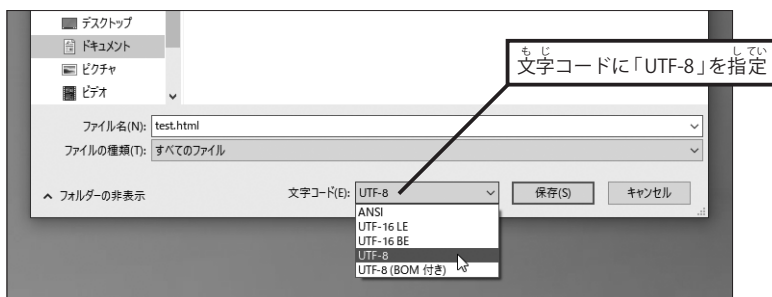


図2-3 「メモ帳」における文字コードの指定

## 2.5 DOCTYPEの記述

続いては、HTMLのバージョンを示す方法について解説します。HTMLを記述するときは、最初にDOCTYPE宣言を行っておく必要があります。DOCTYPE宣言は、文書の種類がHTMLであることを示すと同時に、HTMLのバージョンを示す記述となります。本書ではHTML5に従ってHTMLを作成していくので、DOCTYPE宣言は以下のように記述します。

```
<!DOCTYPE html>
```

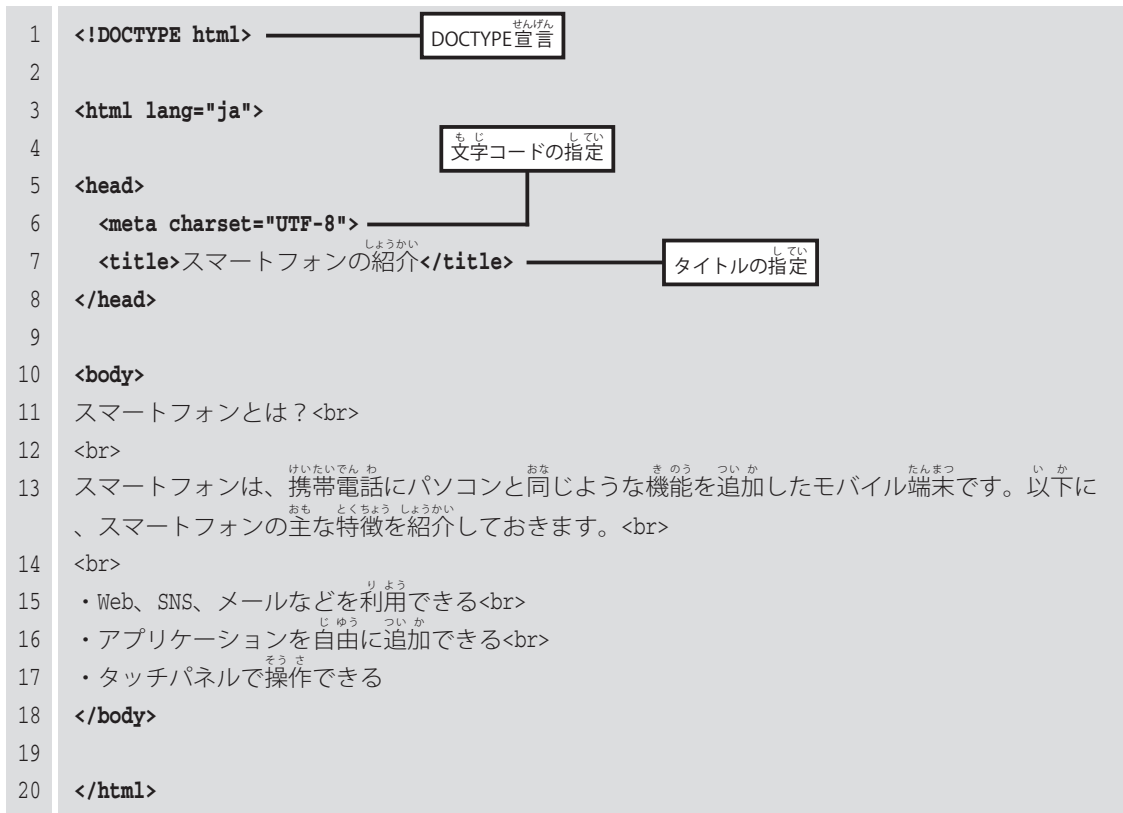
ただ正しいHTMLファイルを作成するには、このDOCTYPE宣言を文書の先頭に記述しておく必要があります。忘れないようにしてください。

## 2.6 正しいHTMLファイルの例

これまでに解説してきた話をまとめると、P12に示したHTMLファイルは以下のように記述するのが基本となります。HTMLファイルには、DOCTYPE宣言とhtml、head、bodyの3つの要素が必要になることを覚えておいてください。

### ▼ sample02-2.html

```
1 <!DOCTYPE html>
2
3 <html lang="ja">
4
5 <head>
6   <meta charset="UTF-8">
7   <title>スマートフォンの紹介</title>
8 </head>
9
10 <body>
11 スマートフォンとは？<br>
12 <br>
13 スマートフォンは、携帯電話にパソコンと同じような機能を追加したモバイル端末です。以下に
14 、スマートフォンの主な特徴を紹介しておきます。<br>
15 ・Web、SNS、メールなどを利用できる<br>
16 ・アプリケーションを自由に追加できる<br>
17 ・タッチパネルで操作できる
18 </body>
19
20 </html>
```



演習



- (1) P12のsample02-1.htmlのようにHTMLファイルを記述し、Webブラウザの表示が図2-1のようになることを確認してみましょう。
- (2) さらに、DOCTYPE宣言とhtml、head、bodyの要素を追加し、正しいHTMLファイルを作成してみましょう。headの領域では、「スマートフォンの紹介」というページタイトルを指定し、文字コードに「UTF-8」を指定します。