

HTML&CSSとWebデザインが 1冊できちんと身につく本

Chapter3 知っておきたいHTMLのきほんと書き方

3.1 HTMLのきほんの知識

1. タグのきほん的な付け方

次のようにタグを付けます。

接頭のタグを開始タグ、接尾のタグを終了タグと呼びます。

```
<p>テキスト</p>
```

タグの中には終了タグを省略できるタグもあります。

```
<li>, <dt>, <dd>, <tr>, <th>, <td> etc...
```

また単独で記述するタグもあります。

```
<img>, <br>, <hr>, <meta> etc..
```

タグの属性を書き換えることで要素の設定を変更します。

次の例だとsrcが属性名、"images/log.png"が属性値となります。

この形式で属性を記述するのがルールみたいです。

```

```

2.HTMLの構造は2つの大きな箱に分かれている

HTMLは次のような構造になっている

```
HTML - head
      L body
```

HTMLはhead要素とbody要素に分かれています。
各要素の役割を次に記載します。

要素	説明
head	主にコンピュータが参照する情報を記載する
body	Webブラウザに表示される内容を記載する

3. ページの基本情報をあらわすおもなタグ

headにはどのような言語、何という名前で、どのファイルとリンクしているかなど記述する。
そのためheadはブラウザや検索エンジンにとっては非常に重要な情報です。
headに記載できる要素として次のようなものがあります。

要素	説明
meta	文書に関するメタデータを指定する。 属性によって様々な役割をもたせることができる。
title	文書のタイトルを表す。ブラウザのタイトルバーに表示される。
link	文書を別の文書関連付ける。
style	文書のスタイル情報を記載する記述方法は、 開始タグと終了タグのあいだに通常どおりのCSSの文法で記述する。

4. ページの内容をあらわすおもなタグ

bodyに記載できる要素は非常に多くのものがあります。
ここでは代表的なもののみ抜粋し次に記載します。

p

文章の段落を表す。

```
<p>paragraph1</p>
<p>paragraph2</p>
<p>paragraph3</p>
<p>paragraph4</p>
<p>paragraph5</p>
```

paragraph1
paragraph2
paragraph3
paragraph4
paragraph5

h1～h6

hはheaddingの略で、文章内の見出しを指定する際に利用する。
文章の階層構造を意識し単純に大きい文字を使いたいからh1を使うというような、
レイアウト的な視点の使い方はしないようにしましょう。

```
<h1>見出し 1 </h1>
<h2>見出し 2 </h2>
<h3>見出し 3 </h3>
<h4>見出し 4 </h4>
<h5>見出し 5 </h5>
<h6>見出し 6 </h6>
```

見出し 1

見出し 2

見出し 3

見出し 4

見出し 5

見出し 6

section

見出しを伴う、意味的に関係のあるまとまりを表す。
必ず見出しを入れることが推奨されている。

```
<section>
  <h1>Paragraph</h1>
  <p>AAAAAAAAAAAAAAAAAAAA</p>
  <p>BBBBBBBBBBBBBBBBBBBB</p>
  <p>CCCCCCCCCCCCCCCCCCCC</p>
  <p>DDDDDDDDDDDDDDDDDDDD</p>
  <p>EEEEEEEEEEEEEEEEEEEE</p>
  <p>FFFFFFFFFFFFFFFFFFFF</p>
</section>
```

Paragraph

AAAAAAAAAAAAAAAAAAAA
BBBBBBBBBBBBBBBBBBBB
CCCCCCCCCCCCCCCCCCCC
DDDDDDDDDDDDDDDDDDDD
EEEEEEEEEEEEEEEEEEEE
FFFFFFFFFFFFFFFFFFFF

article

その内容だけを取り出したときに独立したコンテンツとして成り立つ場合に使用します。

```
<body>
  <article>
    <h1>見出し1</h1>
    <section>
      <h2>見出し2</h2>
      <p>AAAAAAAAAAAAAAAAAAAA</p>
      <p>BBBBBBBBBBBBBBBBBBBB</p>
      <p>CCCCCCCCCCCCCCCCCCCC</p>
    </section>

    <section>
      <h2>見出し2</h2>
      <p>DDDDDDDDDDDDDDDDDDDD</p>
      <p>EEEEEEEEEEEEEEEEEEEE</p>
      <p>FFFFFFFFFFFFFFFFFFFF</p>
    </section>
  </article>
</body>
```

見出し1

見出し2

AAAAAAAAAAAAAAAAAAAA
BBBBBBBBBBBBBBBBBBBB
CCCCCCCCCCCCCCCCCCCC

見出し2

DDDDDDDDDDDDDDDDDDDD
EEEEEEEEEEEEEEEEEEEE
FFFFFFFFFFFFFFFFFFFF

nav

その内容がページの主要なナビゲーションであることを表すときに使用する。

```
<nav>
  <ul>
    <li><a href="/">ROOT</a>
    <li><a href="/one">ONE</a>
    <li><a href="/two">TWO</a>
    <li><a href="/three">THREE</a>
  </ul>
</nav>
```

- [ROOT](#)
- [ONE](#)
- [TWO](#)
- [THREE](#)

aside

本筋とは関係しているものの、メインコンテンツから切り離すことが可能なセクションを表す。
例えばサイドバーなどによく使われるらしいが、だぶんこのタグを使わないければいけないという制約はないけどモラル的に使ったほうがよいはず。

```
<aside>
  <ul>
    <li><a href="/">ROOT</a>
    <li><a href="/one">ONE</a>
    <li><a href="/two">TWO</a>
    <li><a href="/three">THREE</a>
  </ul>
</aside>
```

- [ROOT](#)
- [ONE](#)
- [TWO](#)
- [THREE](#)

div

div要素そのものには特別な意味はない、
div要素で囲まれた範囲をグループとして扱えるので、
レイアウト目的の箱として使われる。

💡 すべてのHTML要素で使える属性 ”グローバル属性”

すべての要素で使えるグローバル属性というものがあります。
グローバル属性で代表的なものがid属性とclass属性です。
これらはCSSでスタイリングするうえで必ずといっていいほど使用します。

次のようにマークアップの際にdiv要素のid属性を決め、
それを目印にしてCSSでスタイリングするのが一般的です。

```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <meta charset="UTF-8">
    <title>Sample</title>
    <style>
      #div1 {
        background: #eeeeff
      }
      #div2 {
        background: #eeffee
      }
    </style>
  </head>

  <body>
    <div id="div1">
      <p>スタイル1を適用した場合</p>
    </div>
    <div id="div2">
      <p>スタイル2を適用した場合</p>
    </div>
  </body>
</html>
```

スタイル1を適用した場合

スタイル2を適用した場合

a

a要素はハイパーリンクを指定する要素です。
ちなみにaはanchorの略で、鎖が船を繋ぎ止めることから来ています。

```
<body>
  <a href="http://google.co.jp">Google</a>のWebサイトはこちらです。
</body>
```

GoogleのWebサイトはこちらです。

img

「image」つまり画像を表示する際に使用します。
次のようにsrc属性で画像ファイルの場所、
alt属性には画像が利用できない環境のために代替テキストを入力します。

```

```

そしてヘッダとかボディとか書くようになります。

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>サンプルページ</title>
  </head>

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

3. ページの基本構造を記述する

ヘッダーはheaderタグ、フッターはfooterタグ、
コンテンツエリアはdivタグで基本構造を記述します。

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>サンプルページ</title>
  </head>

  <body>
    <header></header>
    <div id="wrap">

      </div>
    <footer></footer>
  </body>
</html>
```

4. header要素内を記述する

headerにログとナビゲーションを追加します。

```
<!DOCTYPE html>
```



```

<html>
  <head>
    <meta charset="utf-8">
    <title>サンプルページ</title>
  </head>

  <body>
    <header>
      <div class="logo">
        <a href="index.html"></a>
      </div>
      <nav>
        <ul class="global-nav">
          <li><a href="portfolio.html">Portfolio</a></li>
          <li><a href="about.html">About</a></li>
          <li><a href="contact.html">Contact</a></li>
        </ul>
      </nav>
    </header>
    <div id="wrap">

    </div>
    <footer></footer>
  </body>
</html>

```

5.コンテンツエリアを記述する

共通ファイルの作成なのでとりあえずコンテンツエリアは記述しない。
5章以降に色々記述していきます。

6. footer要素内を記述する

smallタグを使ってfooterにコピーライトを記載します。

💡 smallタグって？

「SMALL」とは、テキストを一回り小さくするためのタグです。
<big>～</big>のタグで囲まれたテキストは1段階大きなサイズで表示され、
のと同じ意味を持ちます。
免責や警告などの注釈や細目などを表す際に活用されますが、
文字の表示はユーザーのブラウザの設定に左右されるため、
大きさを確定させることはできません。

<small>は非推奨タグではありませんが、
この指定内容はスタイルシートで代替できるので、
できるだけスタイルシートで指定しましょう。

スタイルシートで指定した場合は～などと記述します。
ちなみに、<small>の中にさらに<small>を入れ子にして入れることができ、
入れ子のタグ内の文字はより小さな文字で表示されます（ブラウザにより制限がある場合があります）。

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>サンプルページ</title>
  </head>

  <body>
    <header>
      <div class="logo">
        <a href="index.html"></a>
      </div>
      <nav>
        <ul class="global-nav">
          <li><a href="portfolio.html">Portfolio</a></li>
          <li><a href="about.html">About</a></li>
          <li><a href="contact.html">Contact</a></li>
        </ul>
      </nav>
    </header>
    <div id="wrap">

      </div>
    <footer>
      <small>(C)2019 Kaledot725.</small>
    </footer>
  </body>
</html>
```

3章でできたところ

3章でできたところは次のよう感じ！！

