

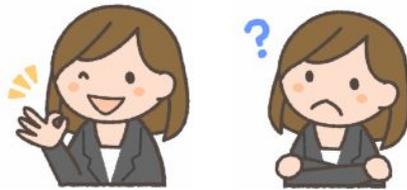
# ななちゃんのIT教室

## ウェブページを作ろうの巻

by nara.yasuhiro@gmail.com

ウェブページを見るだけだったななちゃんが  
自分のホームページを作れるようになるまでのお話

第 0.7 版 2017 年 5 月 7 日



フリー素材  
<http://freeillustration.net>



### もくじ

第1回 ウェブページの構造(見出し、改行、箇条書き)を指定 (HTML)

第2回 ウェブページの見たい目(中寄せ、色、フォント)を指定 (CSS)

第3回 ウェブページをアニメ化 (JavaScript)

まとめ (1) HTML

まとめ (2) HTML の リンク

まとめ (3) HTML/CSS/JavaScript の関係

## 第1回 ウェブページの構造(見出し、改行、箇条書き)を指定(HTML)

なな: これって、朝日新聞の「ののちゃんのDO科学」のパクリ?

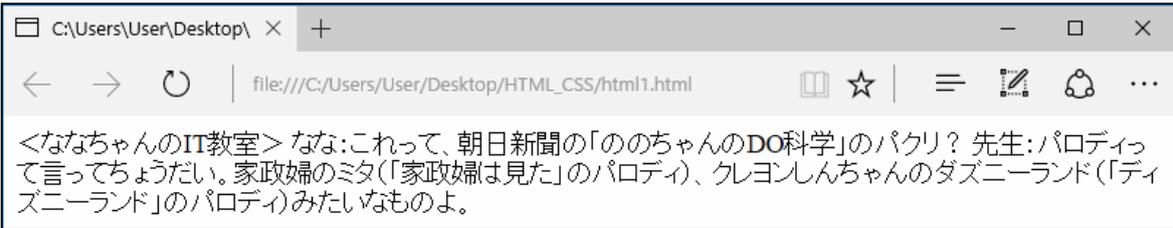
先生: パロディって言ってちょうだい。家政婦のミタ(「家政婦は見た」のパロディ)、クレヨンしんちゃんのダズニーランド(「ディズニーランド」のパロディ)みたいなものよ。

なな: 自分のホームページを持っている友達がいるんだけど、私にも作れるかしら?

先生: いきなり凝ったページを作ろうとせずに、まず簡単なページを作ってから、少しずつ機能追加してゆけば難しいわよ。まず、index.html という名前のファイルを作って、テキストを入力してみてね。出来上がったら、ファイルアイコンをダブルクリックしてみてね。

なな: これでどうかな? あれ、みんなくっついてしまったわ!

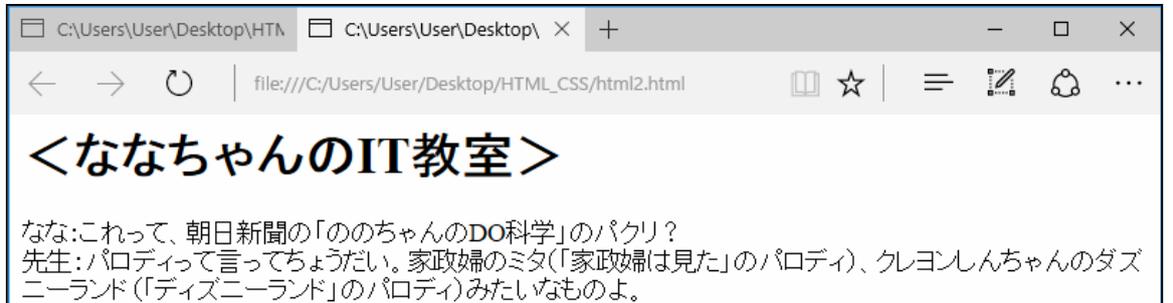
<p>&lt;ななちゃんの IT 教室&gt;</p>	index.html
<p>なな: これって、朝日新聞の「ののちゃんの DO 科学」のパクリ?</p> <p>先生: パロディって言ってちょうだい。家政婦のミタ(「家政婦は見た」のパロディ)、クレヨンしんちゃんのダズニーランド(「ディズニーランド」のパロディ)みたいなものよ。</p>	



A screenshot of a web browser window. The address bar shows the file path: file:///C:/Users/User/Desktop/HTML\_CSS/html1.html. The page content is the same as the previous block, but the text is now rendered in a standard font and size.

先生: 改行や空白は無視されるの。見出しを「<h1>」、「</h1>」で囲み、改行を入れたい部分に「<br>」を入れてみて。こういう「<」、「>」で囲まれた記号を「タグ」と呼びます。タグを入れることを「マークアップ」というの。

<p>&lt;h1&gt;&lt;ななちゃんの IT 教室&gt;&lt;/h1&gt;</p>	index.html
<p>なな: これって、朝日新聞の「ののちゃんの DO 科学」のパクリ? &lt;br&gt;</p> <p>先生: パロディって言ってちょうだい。家政婦のミタ(「家政婦は見た」のパロディ)、クレヨンしんちゃんのダズニーランド(「ディズニーランド」のパロディ)みたいなものよ。</p>	



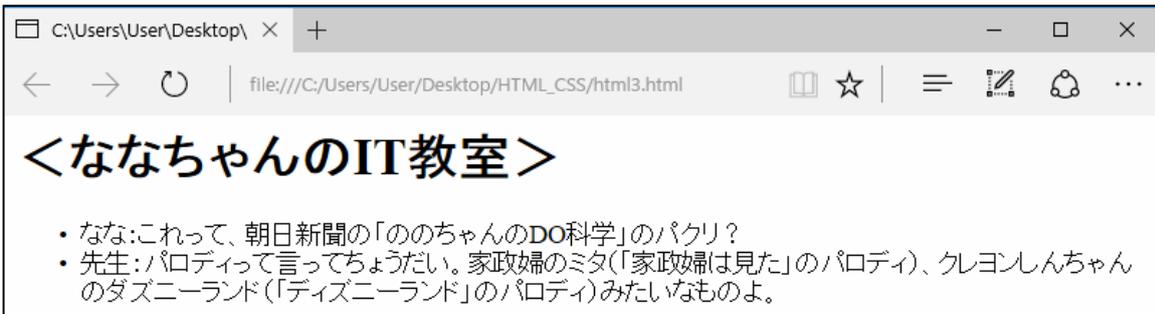
A screenshot of a web browser window. The address bar shows the file path: file:///C:/Users/User/Desktop/HTML\_CSS/html2.html. The page content shows the heading '<h1><ななちゃんの IT 教室></h1>' rendered in a large, bold font. Below it, the text from the previous block is rendered, with a line break (<br>) visible between the two paragraphs.

なな: 見出しが強調され、改行も入ったわ!



先生： ななちゃんと先生のセリフが箇条書きになっていると考えると、次のようにマークアップできるわ。  
「ul」は、「unordered list」(順序関係のない箇条書き) というような意味です。

```
<h1><ななちゃんの IT 教室></h1>
<ul>
<li>なな:これって、朝日新聞の「ののちゃんの DO 科学」のパクリ?
<li>先生:パロディって言ってちょうだい。家政婦のミタ(「家政婦は見た」のパロディ)、クレヨンしんちゃんのダズニーランド(「ディズニーランド」のパロディ)みたいなものよ。
</ul>
```



なな： じゃあ、「ordered list」、「<ol>」もあるの？

先生： あわるわよ。先頭が「・」でなくて、「1.」、「2.」のようになるの。html の書き方の目的は、ウェブに美しく表示するというより、文書の構造を表現して、コンピュータ処理をやりやすくすることにあるのよ。「ここが見出し」「ここが箇条書きになっていて、ことごとく対等に並んでいる」というようなことを表現しているの。さらに、いろいろなブラウザで文字化けなどが発生しないように制御情報を加えることが推奨されます。変化しない定型部分なので、ファイルに書き込んでおいて、毎回コピーして使うと便利です。

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title><ななちゃんの IT 教室></title>
  </head>
  <body>
    <h1><ななちゃんの IT 教室></h1>
    <ul>
      <li>なな:これって、朝日新聞の「ののちゃんの DO 科学」のパクリ? <br>
      <li>先生:パロディって言ってちょうだい。家政婦のミタ(「家政婦は見た」のパロディ)、クレヨンしんちゃんのダズニーランド(「ディズニーランド」のパロディ)みたいなものよ。
    </ul>
  </body>
</html>
```

制御情報  
(ファイルに書き込んでおいて、毎回コピーすると簡単)

タイトルが表示された！

ここだけ、毎回書き換える



## 第2回 ウェブページの見たい目(中寄せ、色、フォント)を指定(CSS)

なな: 見出しを中央にしたいわ。それから、先頭の「・」を無くして、セリフの 2 行目を字下げしたいわ。

先生: そういう、「見たい目」の指定を「スタイル」というの。このスタイルの指定方法を「CSS」というの。

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title><ななちゃんの IT 教室></title>
  </head>
  <body>
    <h1 style="text-align:center;"><ななちゃんの IT 教室></h1>
    <ul style="list-style-type: none;text-indent:-2.5em;padding-left:2.5em;">
      <li>なな:これって、朝日新聞の「ののちゃんのDO科学」のパクリ? <br>
      <li>先生:パロディって言ってちょうだい。家政婦のミタ(「家政婦は見た」のパロディ)、ク
      レヨンしんちゃんのダズニーランド(「ディズニーランド」のパロディ)みたいなものよ。
    </ul>
  </body>
</html>

```



なな: やった!

先生: スタイル情報がちらばっていると見にくいので、<head> のところに集結させることもできるわ。

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title><ななちゃんの IT 教室></title>
    <style>
      h1 { text-align:center; }
      ul { list-style-type: none;text-indent:-2.5em;padding-left:2.5em;}
    </style>
  </head>
  <body>
    <h1><ななちゃんの IT 教室></h1>
    <ul>
      <li>なな:これって、朝日新聞の「ののちゃんのDO科学」のパクリ? <br>
      <li>先生:パロディって言ってちょうだい。家政婦のミタ(「家政婦は見た」のパロディ)、ク
      レヨンしんちゃんのダズニーランド(「ディズニーランド」のパロディ)みたいなものよ。
    </ul>
  </body>
</html>

```

なな: ちょっとすっきりした感じね。

先生: HTML は内容の構造を指定して、CSS は見たい目を指定するという作業分担になっているのよ。

なな: 「ののちゃんとDO科学」の部分の文字を赤くしたいんだけど。

先生: 1箇所だけだったら、このようにすれば良いわ。id=here で「ここよ」と、場所に名前をつけて、「color: red」で色を指定しているの。「here」は、自分で考えた名前で、2か所以上あれば、それぞれに別々の名前をつけます。

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title><ななちゃんの IT 教室></title>
    <style>
      h1 { text-align:center; }
      ul { list-style-type: none;text-indent:-2.5em;padding-left:2.5em;}
      #here { color: red; }
    </style>
  </head>
  <body>
    <h1><ななちゃんの IT 教室></h1>
    <ul>
      <li>なな:これって、朝日新聞の<span id=here>「ののちゃんの DO 科学」</span>のパクリ? <br>
      <li>先生:パロディって言ってちょうだい。家政婦のミタ(「家政婦は見た」のパロディ)、クレヨンしんちゃんのダズニーランド(「ディズニーランド」のパロディ)みたいなものよ。
    </ul>
  </body>
</html>
        
```

なな: 「家政婦は見た」、「ディズニーランド」も赤くするには？

先生: id は、ひとつの場所を示す時にしか使えないけど、class を使えばグループ化できます。

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title><ななちゃんの IT 教室></title>
    <style>
      h1 { text-align:center; }
      ul { list-style-type: none;text-indent:-2.5em;padding-left:2.5em;}
      .original { color: red; }
    </style>
  </head>
  <body>
    <h1><ななちゃんの IT 教室></h1>
    <ul>
      <li>なな:これって、朝日新聞の<span class=original>「ののちゃんの DO 科学」</span>のパクリ?
      <br>
      <li>先生:パロディって言ってちょうだい。家政婦のミタ(<span class=original>「家政婦は見た」</span>のパロディ)、クレヨンしんちゃんのダズニーランド(<span class=original>「ディズニーランド」</span>のパロディ)みたいなものよ。
    </ul>
  </body>
        
```

なな: id は「#」で受けて、class は「.」で受けるのね。

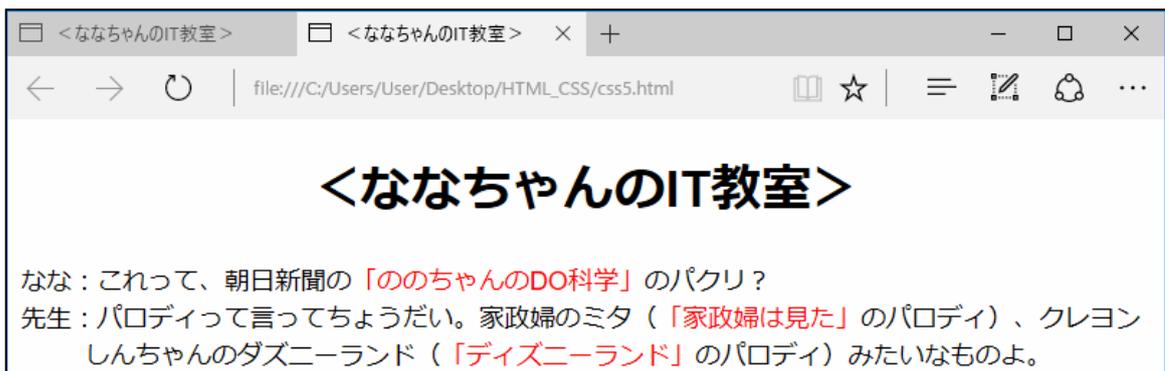
なな：文字が固いんだけど、もっと柔らかい字体は使えないかしら。

先生：字体のことを、「フォント(font)」と呼びます。明朝体、ゴシック体というような種類は「font-family」で「sans-serif」を指定します。「serif」は「飾り文字」、「sans-serif」は「飾り無し文字」ということ。太字にするとか、斜体字にするとかは、インターネットや書籍を調べてね。

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title><ななちゃんの IT 教室></title>
    <style>
      h1 { text-align:center; }
      ul { list-style-type: none;text-indent:-2.5em;padding-left:2.5em;}
      .red { color: red; }
      body { font-family:sans-serif; }
    </style>
  </head>
  <body>
    <h1><ななちゃんの IT 教室></h1>
    <ul>
      <li>なな:これって、朝日新聞の<span class=red>「ののちゃんの DO 科学」</span>のパクリ? <br>
      <li>先生:パロディって言ってちょうだい。家政婦のミタ(<span class=red>「家政婦は見た」</span>のパロディ)、クレヨンしんちゃんのダズニーランド(<span class=red>「ディズニーランド」</span>のパロディ)みたいなものよ。
    </ul>
  </body>

```



### 第3回 ウェブページをアニメ化(JavaScript)

なな: ウェブページに動きをつけることはできないかしら? 文字の大きさが伸縮したりとか、電光掲示板みたいに横に流れるとか、カーソルを動かすと畳み込まれていたメニューが出てくるとか。

先生: CSS の最新規格の CSS3 には、アニメーションの機能があるけど、できることに限界があるし、古いブラウザでは使えなかったりするので、ここでは JavaScript を使った方法を説明します。JavaScript を使った方法なら、多彩な使い方に発展できるし、古いブラウザでも使えるので。

なな: JavaScript ? ということ?

先生: JavaScript というプログラミング言語を使うと、HTML や、CSS を書き換えることができるの。

```

<span>12345</span>
<script>
areap = document.getElementsByTagName("span")[0];
areap.innerHTML = "ABCDE";
areap.style.backgroundColor = "pink";
</script>
    
```

span の場所を調べる

最初の span

HTML を書き換える

CSS を書き換える

JavaScript のプログラムは、<span> と </span> で囲みます。そして、対象部分を <span> と </span> で囲みます。「document.getElementsByTagName("span")」で、span の場所を調べます。「[0]」は、span が 2 箇所以上ある場合に、最初の span であることを示します。その文字列を書き換えるのが「innerHTML」です。HTML の命令を書き込めるので、「"ABCDE"」以外にも、「"ABC<br>DE"」みたいなのも書き込めます。スタイルを書き込むのが「style」です。JavaScript のキーワードでは「-」を使えないので、「background-color」は「backgroundColor」のように、「-」を取り去って、直後の「c」を大文字に変えます。

なな: 画面上の「12345」という部分が、「ABCDE」に変わって、背景色がピンクになるのね。分かったような気もするけど、アニメーションというより、はじめからそういう出力だったのと同じような。

先生: これを拡張して、たとえば、0.1 秒ごとに書き換えを行えば、アニメになるというわけ。

```

<span id=sp> +
var area = document.getElementById("sp") でも OK
<span>12345</span>
<script>
var area = document.getElementsByTagName("span")[0];
var a = 0;

function go() {
    area.style.fontSize = (Math.sin(a+=0.1)+1) * 25 + "pt";
}
setInterval(go, 20);
</script>
    
```

文字サイズを書き換える

正弦波で振動させる

0.02 秒ごとに go を呼び出す

なな: わお! アコーディオンみたいに、びよ〜んと、大きくなったり、小さくなったりするのね!

12345



12345



先生: JavaScript を使うと、じつにさまざまなことができるのよ。詳しくは、副読本の他の巻にたくさん書いてあるから、興味があったら読んでみてください。

なな: 電光掲示板だったら、「12345」→「2345 1」→「345 12」→「45 123」みたいな順番で書き換えてゆけば良いのね。分かった!

## まとめ (1) HTML

Webで情報を公開するには、HTMLと呼ばれる言語を用いてWebページを作成します(～.html または ～.htm という名前のファイル)。<> や </> で囲まれたものをタグと呼びます。HTMLでは文章の構造(見出し、段落等)や、修飾(文字の大きさ、色等)を指定するためにタグを用います。<>が開始タグ、</>が終了タグです。文章の構造をタグで指定していくことを「マークアップ」といい、このような目的のための言語をマークアップ言語と呼びます。HTMLはマークアップ言語のひとつです。

下記のようなタグは:

```
<h1>大文字</h1>
<h3>中文字</h3>
<h5>小文字</h5>
1行目
2行目<br>
3行目
<hr>
4行目
```

<br> は改行。

<hr> は横線(区切り線)

ブラウザで表示すると:

1行目と2行目の間には <br> がないのでつながってしまう

箇条書きは

```
<ul>
<li> 項目1
<li> 項目2
<li> 項目3
</ul>
```

```
・項目1
・項目2
・項目3
```

と表現する。ブラウザ画面上では:

「ul」(Un-ordered List)のかわりに「ol」(Ordered List)とすれば

```
<ol>
<li> 項目1
<li> 項目2
<li> 項目3
</ol>
```

```
1. 項目1
2. 項目2
3. 項目3
```

このような表示になる

### まとめ (2) HTML の リンク

下線つきの(通常は青い)文字列をクリックすると他のページにジャンプするリンクは HTML 文書に次の行を追加することで実現できる。「リンク」のことを「ハイパーリンク」とも呼ぶ。

```
<a href="リンク先のURL">ここをクリックするとジャンプします</a>
```

他の人のページにジャンプするにはURL欄に http://... の形を書く(=絶対アドレス)。元の HTML と同じフォルダにあるページにジャンプするならファイル名を書く(=相対アドレス)。

test1.html の中身は:↓

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>My Test Home Page</title>
  </head>
  <body>
    <p>
      これは HTML のテストです.
    </p>
    <a href="testsub.html">testsub.html にジャンプします</a>
  </body>
</html>
```

testsub.html  
の中身は:↓

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>My Test Home Page(sub)</title>
  </head>
  <body>
    <p>
      testsub.html にジャンプしました!
    </p>
  </body>
</html>
```

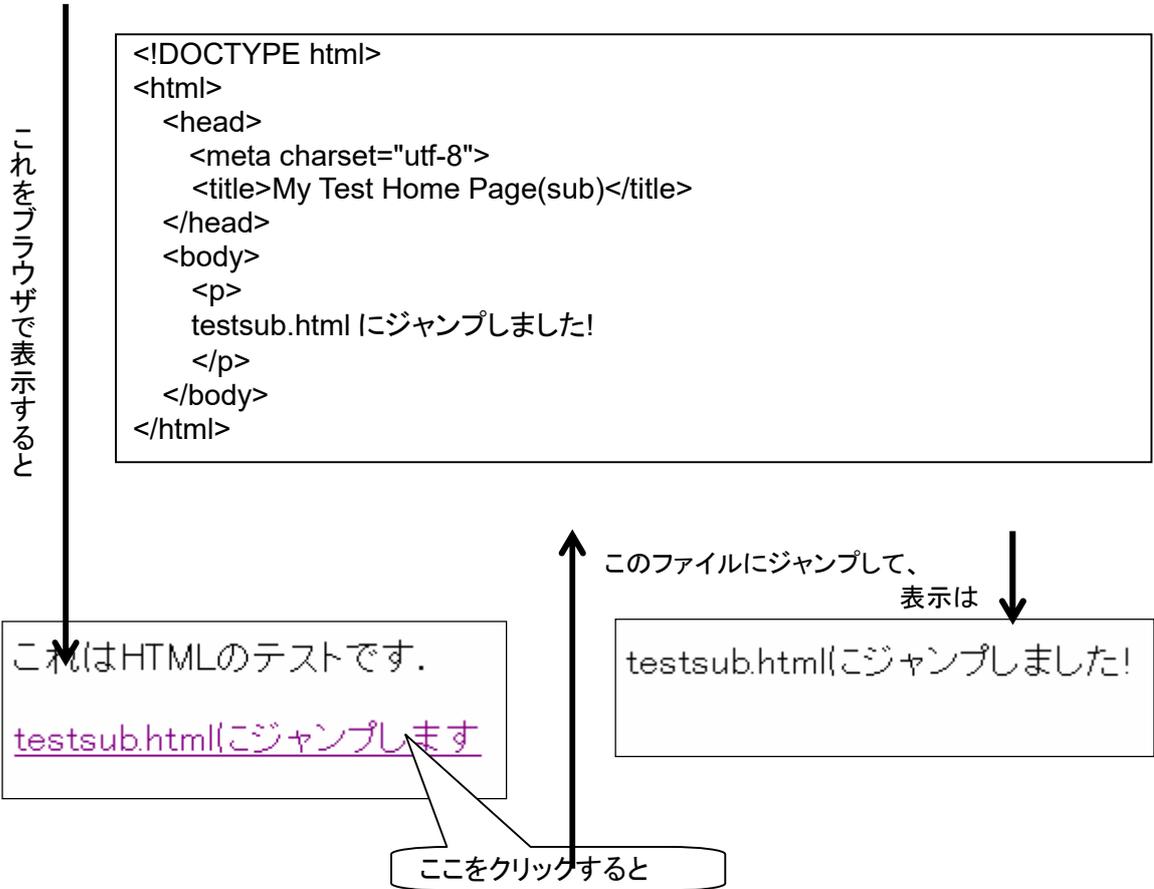
これをブラウザで表示すると

これはHTMLのテストです。  
[testsub.htmlにジャンプします](#)

testsub.htmlにジャンプしました!

ここをクリックすると

このファイルにジャンプして、  
表示は ↓



## まとめ (3) HTML/CSS/JavaScript の関係

### ☆ html (HyperText Markup Language)

(静的な)ウェブページの「内容」を記述する。普通の文章と、`<~>` の形のタグ。

例: `<html>~</html> <head>~</head> <body>~</body> <h1>~</h1> <title>~</title>  
<link /> <script>~</script> <img />  
src="..." などの付加情報も記述する場合がある。`

### ☆ css (Cascading Style Sheets)

「`<p>` タグはこう、でも `class` 指定があればこう、さらに `id` 指定があればこう」というような「段階的」「直列的」な使い方ができるので `cascade` (直列)をいう。原義は「滝」の意味。滝の下流にもうひとつの滝があるというイメージ。

ウェブページのスタイル(色、大きさ、位置)を指定する。

例:

```
body {
  background-color: white;
  color: black;
}
```

セレクタ:html の、どの部分を対象にするか

セレクタの例:

```
h1
p
.caption
#subnav
```

規則:どのような性質を  
どのような設定値にするか

html の h1 タグすべてが対象  
html の p タグ "

タグ中に「`class=caption`」と書いた部分が対象。先頭に「`.`」(ドット)をつける  
タグ中に「`id=subnav`」と書いた部分が対象。先頭に「`#`」をつける。

クラス:html 文書中で同じ  
クラスが複数個所にあっ  
てよい。

規則の例:

```
color: pink;
background-color: pink;
margin: 10px;
padding: 25px;
font-size: 12px;
line-height: 14px;
```

ID:html 文書中に同じ ID  
は1箇所だけ。複数個所  
あってはいけない。

色を表す値の表記:

名前付き色	<b>orange</b>
16 進値	<b>#3286af</b> または <b>#38a</b> ( <b>#3388aa</b> と同じ)
RGB 値	<b>rgb(10, 150, 20)</b>
アルファ値(透明度)指定 RGB 値	<b>rgba(10, 150, 20, 0.5)</b>



### ☆ JavaScript (スクリプト)

動的なウェブページの記述。

動的=毎回異なった内容(例:日時、~人目など)になったり、動きがあったり、クリックすると表示が変化したり、枠に入力した内容に対応した動作をしたりする。

`<body id=main>` または `<h1 id=main>本文</h1>` としておいて、

```
function go1 ( ) {
  var name = document.getElementById('main');
  name.style.backgroundColor="pink";
}
```

とすることも可能!

css の `background-color` は、  
`backgroundColor` に変える

```

index.html
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>自己紹介</title>
    <link rel="stylesheet" href="mystyle.css" />
  </head>
  <body>
    <h1>本文</h1>
    
    <input type="button" value="ここをクリック" onclick="sayhello()">
    <script src="ex05-3.js"></script>
  </body>
</html>
    
```

```

mystyle.css
h1 { color: blue; }
    
```

```

ex05-3.js
function sayhello() {
  alert('Hello, world!');
}
    
```



ファイル分割型 (推奨)

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>練習問題 05-3</title>
    <style>
      h1 { color: blue; }
    </style>
  </head>
  <body>
    <h1>関数の練習</h1>
    
    <input type="button" value="ここをクリック" onclick="sayhello()">
    <script>
      function sayhello() {
        alert('Hello, world!');
      }
    </script>
  </body>
</html>
    
```

ファイル一体型 (非推奨)

