

HTML/CSS実習

08. 文字のスタイル

株式会社ジーードライブ

今回学ぶこと

- 文字のスタイル
 - 文字の種類(書体／フォント)
 - 色
 - 大きさ
 - リストマーカーなど
- 変数(カスタムプロパティ)

文字の種類 font-family

- 使用する書体はfont-familyプロパティで指定する
 - **font-family: 値, 値, 値...**; のようにカンマ区切りで複数の値を指定できる
 - ユーザーの環境(PCやスマホ内にあるフォント)を使用して表示するので、指定したフォントで表示されるとは限らない
 - 複数のフォントを指定すると、先に記述したものが優先される

```
body {  
    font-family: 'Hiragino Kaku Gothic Pro', 'ヒラギノ角ゴ Pro W3',  
               'Meiryo', 'メイリオ', 'Osaka', 'MS PGothic', arial,  
               helvetica, sans-serif;  
}
```

→ 総称フォントファミリーは最後に指定

※全角文字や半角スペースを含むものはフォーテーションで囲む

※日本語表記のフォント名はMacで認識されないことがある

※半角英数と日本語でフォントを分けたい場合は、英語フォントを先に指定する

font-familyの値

代表的な値	
sans-serif	ゴシック体（総称フォントファミリー）
serif	明朝体（総称フォントファミリー）
MS Pゴシック	Win95以降対応。アンチエイリアス非対応
メイリオ	WinVista以降対応の日本語ゴシック体フォント
ヒラギノ角ゴ	MacOS X以降対応の日本語ゴシック体フォント
游ゴシック	Win8.1以降、MacOS10.9以降対応の日本語フォント
Segoe UI	マイクロソフト社の製品で使用されているフォント 日本語には対応していない
Roboto	AndroidやChrome OSのシステムフォント
Noto-Sans	NotoはGoogleが提供するグローバルなフォント

font-familyの設定例

Google

```
font-family: Roboto, 'Segoe UI', Arial, Meiryo, sans-serif;
```

Facebook

```
font-family: Segoe UI Historic, Segoe UI, Helvetica, Arial,  
"hiragino kaku gothic pro", meiryo,  
"ms pgothic", sans-serif;
```

Yahoo Japan

```
font-family: YJTop Ellipsis, ヒラギノ角ゴ ProN,  
Hiragino Kaku Gothic ProN, メイリオ, Meiryo, Osaka,  
MS Pゴシック, MS PGothic, sans-serif;
```

Webフォントの利用

- 通常、ブラウザは端末に存在するフォントで表示を行う
 - 制作者の意図しない表示になってしまう可能性がある
- サーバー上にフォントを配置しておくことで、制作者の意図するフォントでの表示を実現することができる
 - 自身でサーバー上にフォントを配置することも可能だが、Google Fontsというサービスを利用することで、手軽にWebフォントを利用することができる
 - GoogleFonts: <https://fonts.googleapis.com/>
 - GoogleFontsの使い方: <https://design.webclips.jp/google-font/>

文字の色 color

- 文字の色は**color**プロパティで指定する

HTML

```
<h2>関東地方</h2>
<p>明日は雨でしょう</p>
```

CSS

```
h2 {
  color: #1BE70C;
}
p {
  color: rgba(41, 12, 218, 0.75);
}
```



色の指定

- モニター上の色は、RGB(Red, Green, Blue)
それぞれの輝度の組み合わせで表現される
 - RGB各0～255の256段階
 $\Rightarrow 256 \times 256 \times 256 =$ 約1,600万色の指定が可能
 - $\Rightarrow 0 \sim 255$ の階調を16進数(00～FF)で表すことが多い

16進数による色指定の場合、頭に#をつける

```
color: #FF0000;
```

The diagram shows the color code `#FF0000` enclosed in a box. A vertical line above the first two digits (`FF`) is labeled with a slash (/). Three colored lines point from below to the digits: a red line points to `FF` and is labeled "Redの輝度"; a green line points to `00` and is labeled "Greenの輝度"; a blue line points to `00` and is labeled "Blueの輝度".

色の指定

- CSSでは以下のような値で色を指定することができる

値の種類	指定方法	記述例
色名	色の名前で指定	red
RGB	Red, Green, Blueの光の三原色の割合を16進数で2桁ずつ指定。2桁ずつが同じ値の場合は、3桁に省略可能。大文字でも小文字でも、どちらでもよい	#FF0000 #F00 #ff0000 #f00
RGB	RGBを10進数で指定(最大255, または100%)	rgb(255, 0, 0) rgb(100%, 0%, 0%)
RGBA	RGBに加え、0~1でalpha (不透明度)を指定。 0が透過で、1が不透過	rgba(255, 0, 0, 0.5) rgba(100%, 0%, 0%, 0.5)
HSL	Hue, Saturation, Lightness(色相, 彩度, 明度)の組み合わせで色を指定。Hueは色相環における角度の値を指定する。不透明度の指定も可能	hsl(120, 50%, 50%) hsl(120, 50%, 50%, 0.5)

文字の大きさ font-size

- 文字の大きさはfont-sizeプロパティで指定する

HTML

```
<body>
<h2>関東地方</h2>
<p>明日は雨でしょう</p>
</body>
```

CSS

```
h2 {
  font-size: 48px;
}
p {
  font-size: 1.5em; /* 16px×1.5=24pxに相当 */
}
```



文字の大きさ font-size

- 文字の大きさの単位
 - これらの単位は文字以外のサイズ指定にも使用できる

単位		説明
px	ピクセル	画面上の解像度を基準とした固定単位
%	パーセント	親要素の文字サイズを100%とした相対単位
em	エム	親要素の文字サイズを1emとした相対単位 1emは100%に相当する
rem	レム	最上位の指定(html)を基準とした相対単位

- ※ 単位指定は必須
- ※ 印刷を前提としたCSSでは、pt(ポイント)やmm(ミリメートル)を指定する場合もある

em と rem の比較

- ・ % と em は親要素の文字サイズが基準になる

CSS

```
html {  
    font-size: 24px;  
}  
div {  
    font-size: 1.5em;  
}  
.rainy {  
    font-size: 1.5em;  
}  
.sunny {  
    font-size: 1.5rem;  
}
```

rem : root + em
html の設定が基準になる

HTML

```
<body>  
<div>  
    <p class="rainy">明日は雨です</p>  
    <p class="sunny">明後日は晴れます</p>  
</div>  
</body>
```

$$24\text{px} \times 1.5 \times 1.5 = 54\text{px}$$

(html) (div) (.rainy)

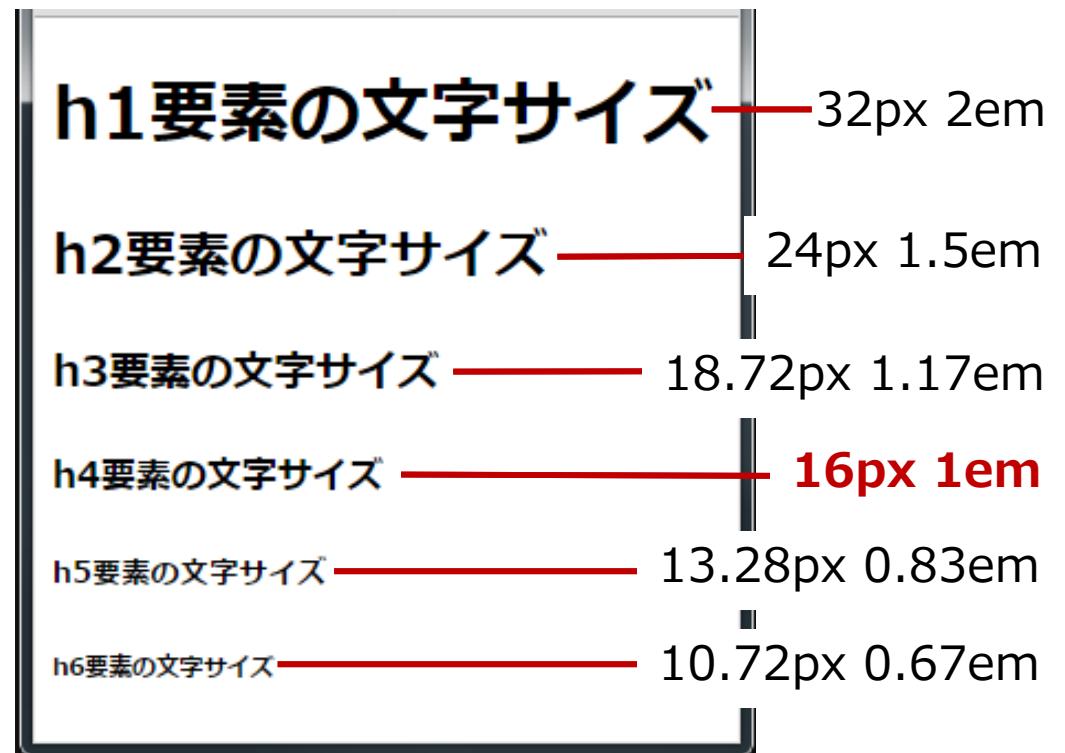
$$24\text{px} \times 1.5 = 36\text{px}$$

(html) (.sunny)



デフォルトの文字サイズ

- 任意の文字サイズを設定しない場合、各要素に対しては、ブラウザに実装されているデフォルトのCSS（ユーザーエージェントCSS）が反映される
- html要素のデフォルトの文字サイズは16pxなので $1\text{rem} = 16\text{px}$ となる
 - 同様にbody要素、p要素のデフォルトの文字サイズも16pxとなっている



見出し要素の初期サイズ

太字にする font-weight

- 以下の値を伴って、文字の太さを変更する

値	説明
normal	標準の太さにする (400に相当)
bold	太字にする (700に相当)
100～900	100きざみで、9種類の太さを指定できる。 ただし、太さが9種類も用意されているフォントは少ない

```
<h1>天気予報</h1>
<p>明日は<span>強い雨</span>が降るでしょう</p>
```

```
h1 {
  font-weight: normal;
}
span {
  font-weight: bold;
}
```



下線や取り消し線 text-decoration

- 以下の値を伴って、文字に下線や取り消し線をつける

値	説明
none	標準の文字にする
underline	下線をつける
line-through	取り消し線をつける

```
<p class="finished">デザイナー、募集中</p>
<p class="current">プログラマー、募集中</p>
<p>詳細は、<a href="">求人ページ</a>をご覧ください</p>
```

```
.finished {
    text-decoration: line-through;
}
.current {
    text-decoration: underline;
}
a {
    text-decoration: none;
}
```



※リンク(a要素)について下線を取るときに使うことが多い
※line-throughは、意味を持つdel要素の使用が望ましい

練習

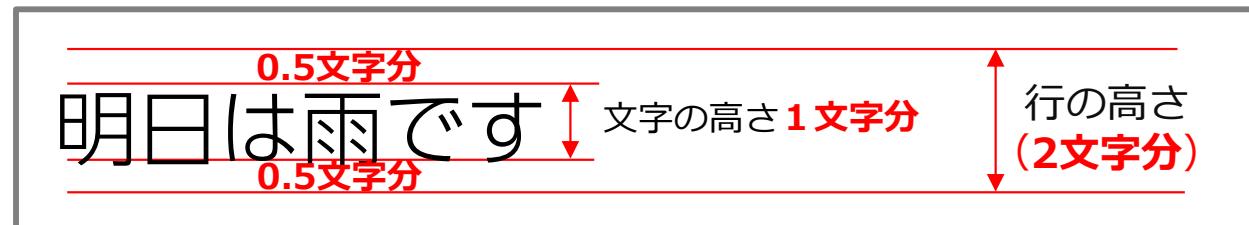
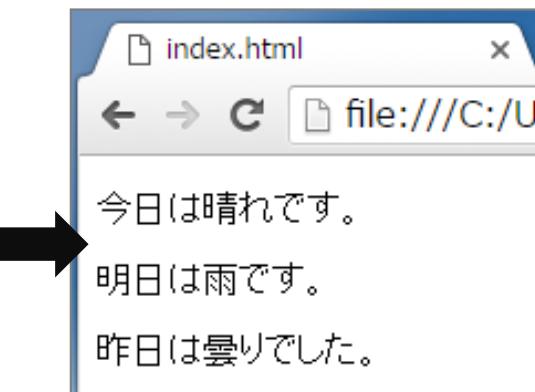
- 練習08-1

行の高さ line-height

- 1行分の高さを指定するためのプロパティで、行間の調整に利用される
- 単位なしで指定することが多い

```
<p>  
今日は晴れです。<br>  
明日は雨です。<br>  
昨日は曇りでした。  
</p>
```

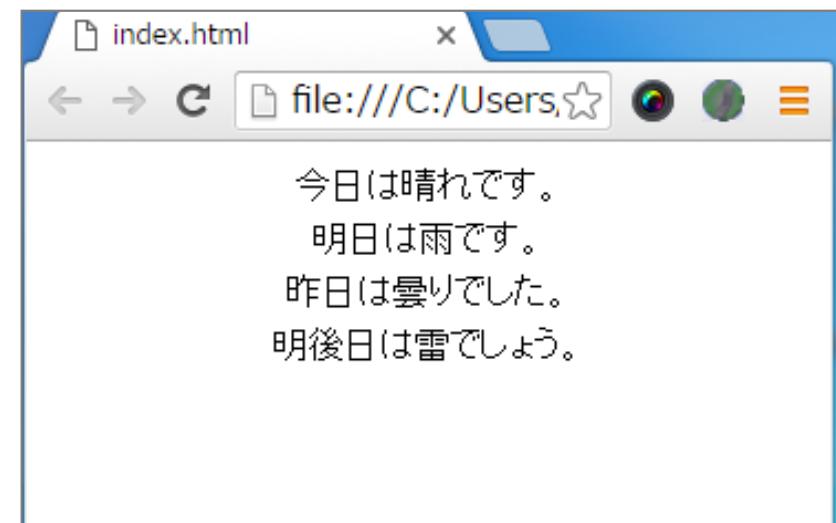
```
p {  
    line-height: 2;  
}
```



行揃え text-align

- 文章を揃える基準を指定する
 - left(左揃え:初期値),center(中央揃え),right(右揃え) のいずれかを値として指定する

```
<p>  
今日は晴れです。<br>  
明日は雨です。<br>  
昨日は曇りでした。<br>  
明後日は雷でしょう。  
</p>
```



```
p {  
    text-align: center;  
    line-height: 1.5;  
}
```

リストの種類 **list-style-type**

- リストの種類としては、以下のような値を指定可能

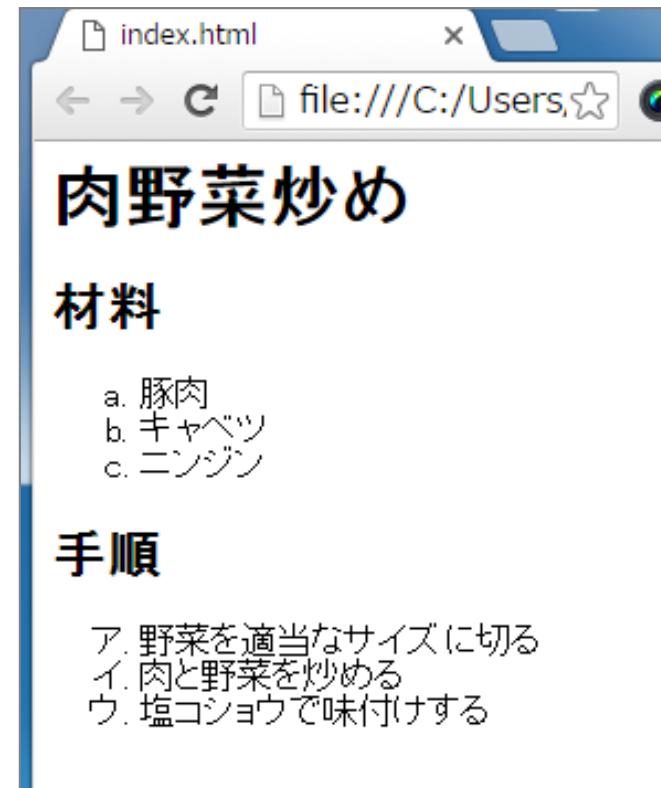
値	説明
none	マークなし
disk	黒丸 • (ul の初期値)
square	黒四角 ■
decimal	算用数字 1. 2. 3. (ol の初期値)
lower-roman	小文字のローマ数字 i . ii . iii .
lower-alpha	小文字のアルファベット a. b. c.
katakana	カタカナのアイウエオ ア. イ. ウ.

基本的にはul, li / ol, li に対して設定可能
list-style-imageプロパティで画像を設定することも可能

リストの種類 list-style-type

- 記述例

```
<h1>肉野菜炒め</h1>
<h2>材料</h2>
<ul>
  <li>豚肉</li>
  <li>キャベツ</li>
  <li>ニンジン</li>
</ul>
<h2>手順</h2>
<ol>
  <li>野菜を適当なサイズに切る</li>
  <li>肉と野菜を炒める</li>
  <li>塩コショウで味付けする</li>
</ol>
```



セレクタはli
でも構わない

```
ul {
  list-style-type: lower-alpha;
}
ol {
  list-style-type: katakana;
}
```

ドロップシャドウ text-shadow

- 文字に対してドロップシャドウを設定する
書式

text-shadow: 横方向の位置 縦方向の位置 ぼかし 影の色;

プラスの値で右、マイナスの値で左

プラスの値で下、マイナスの値で上

記述例

```
<h1>Blog Title</h1>
```

```
h1{  
    text-shadow: 2px 2px 5px rgba(0, 0, 0, 0.5);  
}
```

右下にずれた位置に影ができる

半透明の黒を影の色として指定

影の輪郭に対し、5pxのぼかしをかける



Blog Title

その他の文字系プロパティ

プロパティ	説明
font-style	斜体の設定 日本語フォントは対応していないことが多い
letter-spacing word-spacing	文字と文字の間隔、単語同士の間隔の設定
text-indent	インデント(行頭のスペース)の設定
vertical-align	垂直方向の位置揃え テーブル内の文字配置や上付き文字の設定で使用可能
overflow-wrap word-break hyphens	表示領域をはみ出してしまう長い単語の改行設定

練習

- 練習08-2

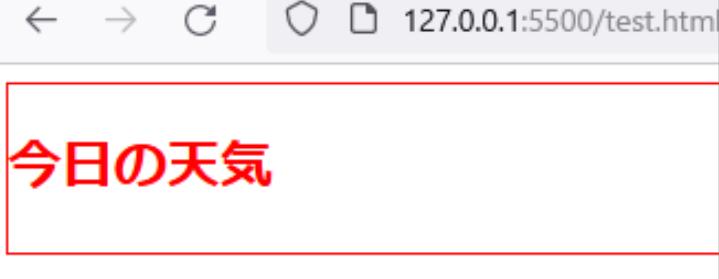
変数: カスタムプロパティ

- 独自のプロパティは、`--` で始まる形で定義できる
 - カスタムプロパティと呼ばれ、`var()`関数で呼び出すことが可能
 - 「変数」として利用することができる

```
<article>
```

```
  <h1>今日の天気</h1>
```

```
</article>
```



```
127.0.0.1:5500/test.html
```

今日の天気

```
article {
```

```
  --mycolor: #f00;
```

```
  border: 1px solid var(--mycolor);}
```

カスタムプロパティの定義

```
h1 {
```

```
  color: var(--mycolor);}
```

カスタムプロパティの呼び出し

変数: カスタムプロパティ

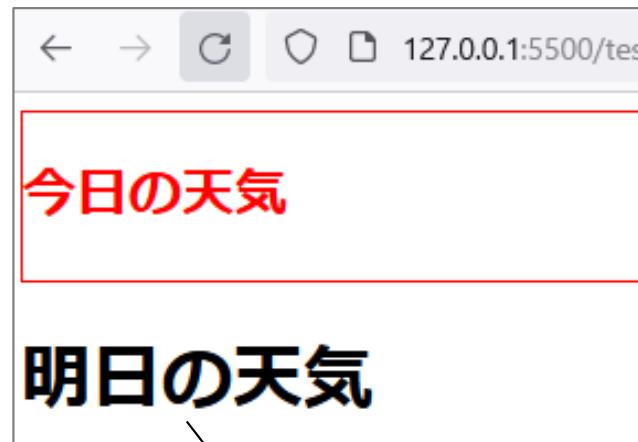
- カスタムプロパティは、宣言されたセレクタの子孫要素内でも利用可能(適用される)

```
<article>
  <h1>今日の天気</h1>
</article>

<h1>明日の天気</h1>
```

```
article {
  --mycolor: #f00;
  border: 1px solid var(--mycolor);
}

h1 {
  color: var(--mycolor);
```



articleの子孫要素でないので、
適用されていない

articleで定義されているプロパティなので、
articleの子孫要素であれば適用される

変数: カスタムプロパティ

- ページ全体で利用できるグローバルな変数を定義したい場合は、疑似クラス :root を利用する
 - :rootはhtmlセレクタと同等のセレクタ（詳細度だけが異なる）

```
:root {  
  --mycolor: #f00;  
}  
  
article {  
  border: 1px solid var(--mycolor);  
}  
  
h1 {  
  color: var(--mycolor);  
}
```

