

HTML & CSS

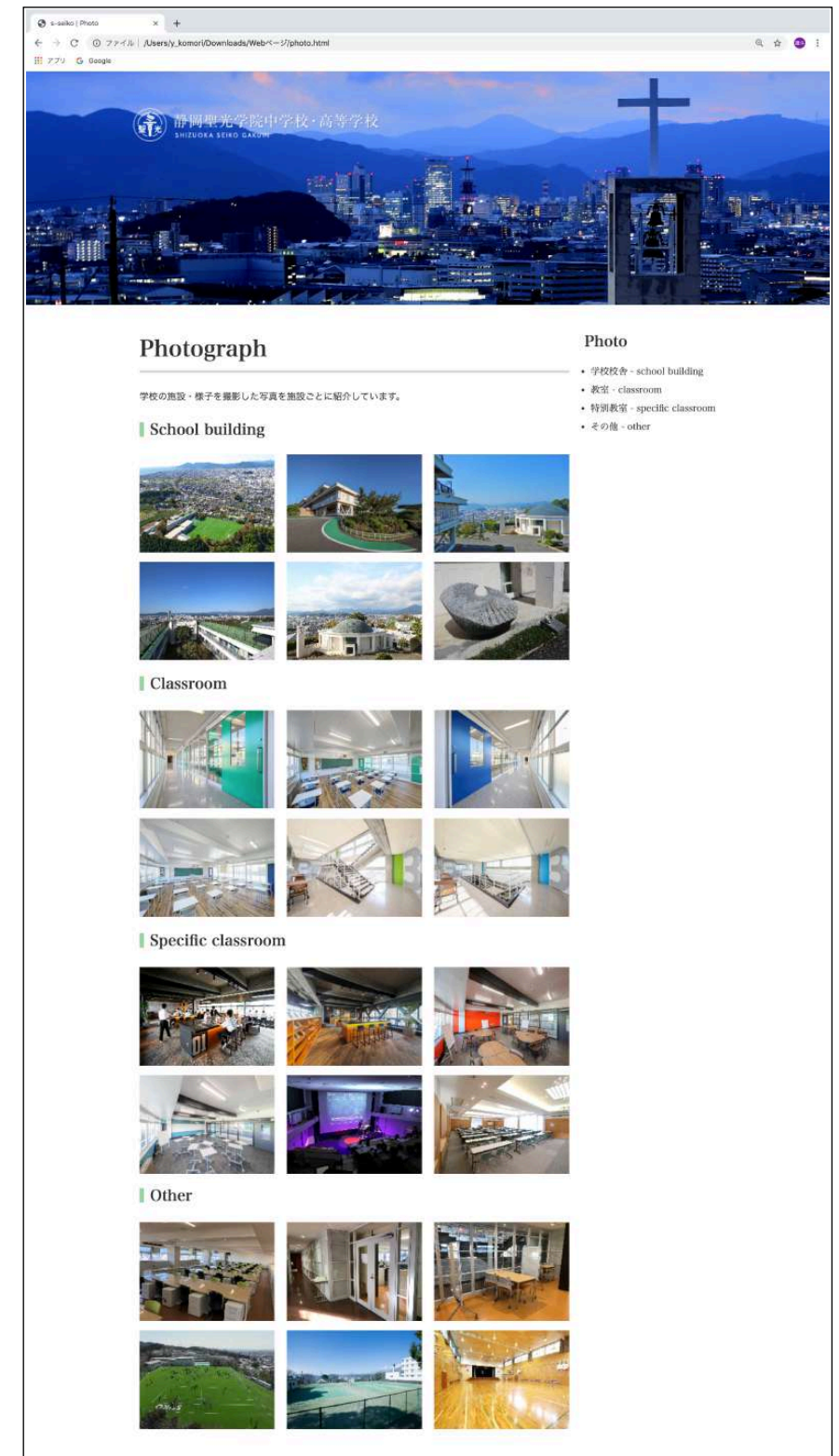
～自分だけのWebページを作成しよう！～



トップページ



2カラムページ



今回は、こんなWebページをコーディングで作成していきます！
この完成イメージを「デザインカンプ」や「モックアップ」と呼びます



HTMLの基本と書き方

- 1 HTML文書の基本構造を記述する
- 2 ページの基本構造を記述する
- 3 ヘッダー要素内を記述する
- 4 コンテンツエリアを記述する

1-1. HTML文書の基本構造を記述する

まずは共通部分を作成するために「common.html」というファイルを作成します。

● Brackets起動 Brackets

- メニューの「ファイル」
- 「新規作成」をクリック
- 「ファイル」
- 「名前をつけて保存」
- 作業フォルダ「Webページ」に「common.html」という名前で保存
(事前にフォルダを作っておくとgood 無い場合はdesktopへ保存)



1-1. HTML文書の基本構造を記述する

さっそくコーディングしていこう！

文章を作成する際に、まず1行目に記述するのがDOCTYPE宣言。

これは、その文書がHTMLのどのバージョンで作成されているのかを宣言するためのもので、必ず1行目に書きます。

```
1 <!DOCTYPE html>
2 <html></html>
```

DOCTYPE宣言に続いて、一番外側の大きな箱であるhtml要素の開始タグと終了タグを入力します。

```
1 <!DOCTYPE html>
2 ▼ <html>
3   <head></head>
4   <body></body>
5   </html>
```

そしてHTML要素の中に、head要素とbody要素を入力

1-1. HTML文書の基本構造を記述する

- head要素の中に文字コードとタイトルを入力

タイトルはtitle要素を使用し「サンプルページ」としておきましょう。

```
1  <!DOCTYPE html>
2  ▼ <html>
3  ▼ <head>
4    <meta charset="UTF-8">
5    <title>サンプルページ</title>
6  </head>
7  ▼ <body>
8    <header></header>
9    <footer></footer>
10 </body>
11 </html>
```

これで、ページの情報を伝えるhead要素に最低限の情報が入りました。

1-2. ページの基本構造を記述する

●body要素の中身の入力

まずは、ヘッダーとフッターを配置します。

```
1  <!DOCTYPE html>
2  ▼ <html>
3  ▼ <head>
4    <meta charset="UTF-8">
5    <title>サンプルページ</title>
6  </head>
7  ▼ <body>
8    <header></header>
9    <footer></footer>
10 </body>
11 </html>
```

ヘッダーとフッターには、
それぞれ専用のheader要素
とfooter要素を使用します

1-2. ページの基本構造を記述する

●コンテンツエリアの指定

コンテンツエリアにはdiv要素を使用し、id属性で「wrap」という名前をつけておきます。

```
1  <!DOCTYPE html>
2  ▼ <html>
3  ▼ <head>
4    <meta charset="UTF-8">
5    <title>サンプルページ</title>
6  </head>
7  ▼ <body>
8    <header></header>
9    <div id="wrap"></div>
10   <footer></footer>
11 </body>
12 </html>
```

id属性はCSSを記述する際の
目印となります

これで、ページの骨組みについての指定は終了です。

続いてヘッダー、コンテンツエリア、フッターの要素の中身を入力します。

■ 1-3. ヘッダー要素内を記述する

● ロゴの指定

今回は画像ファイルを使用します。

img要素を使ってロゴ画像を配置してみましょう。

```
7 ▼ <body>
8 ▼ <header>
9   
10 </header>
11 <div id="wrap"></div>
12 <footer></footer>
13 </body>
```

画像ファイルのある場所を指定するのはimg要素の「src要素」を使います。相対パスを使用して「images」フォルダ内の「logo.png」を指定します。「alt属性」には代替テキストを入力します。ここでは「seiko」と入力。

1-3. ヘッダー要素内を記述する

- ロゴにトップページのリンクを張る

a要素を使い、index.htmlへのリンクを張ります。

```
7 ▼ <body>
8 ▼ <header>
9   <a href="index.html"></a>
10 </header>
11 <div id="wrap"></div>
12 <footer></footer>
13 </body>
```

画像ファイルにリンクを張る場合は、img要素を<a>～タグで囲む。

■ 1-3. ヘッダー要素内を記述する

- ロゴにCSSでのスタイリング用の目印をつける

CSSでロゴを装飾する際の目印をつけておきます。

div要素で囲ったらclass属性で「logo」という名前をつけます。

```
7 ▼ <body>
8 ▼ <header>
9 ▼ <div class="logo">
10   <a href="index.html"></a>
11 </div>
12 </header>
```


■ 1-4. コンテンツエリアを記述する

- wrapの中にコンテンツエリアを作成

class属性で「content」という名前をつけたdiv要素を作成します。

```
13 ▼ <div id="wrap">  
14   <div class="content">  
15   </div>  
16 </div>
```

3章以降で制作するページコンテンツは
<div> ~ </div>の中に記述していきます。



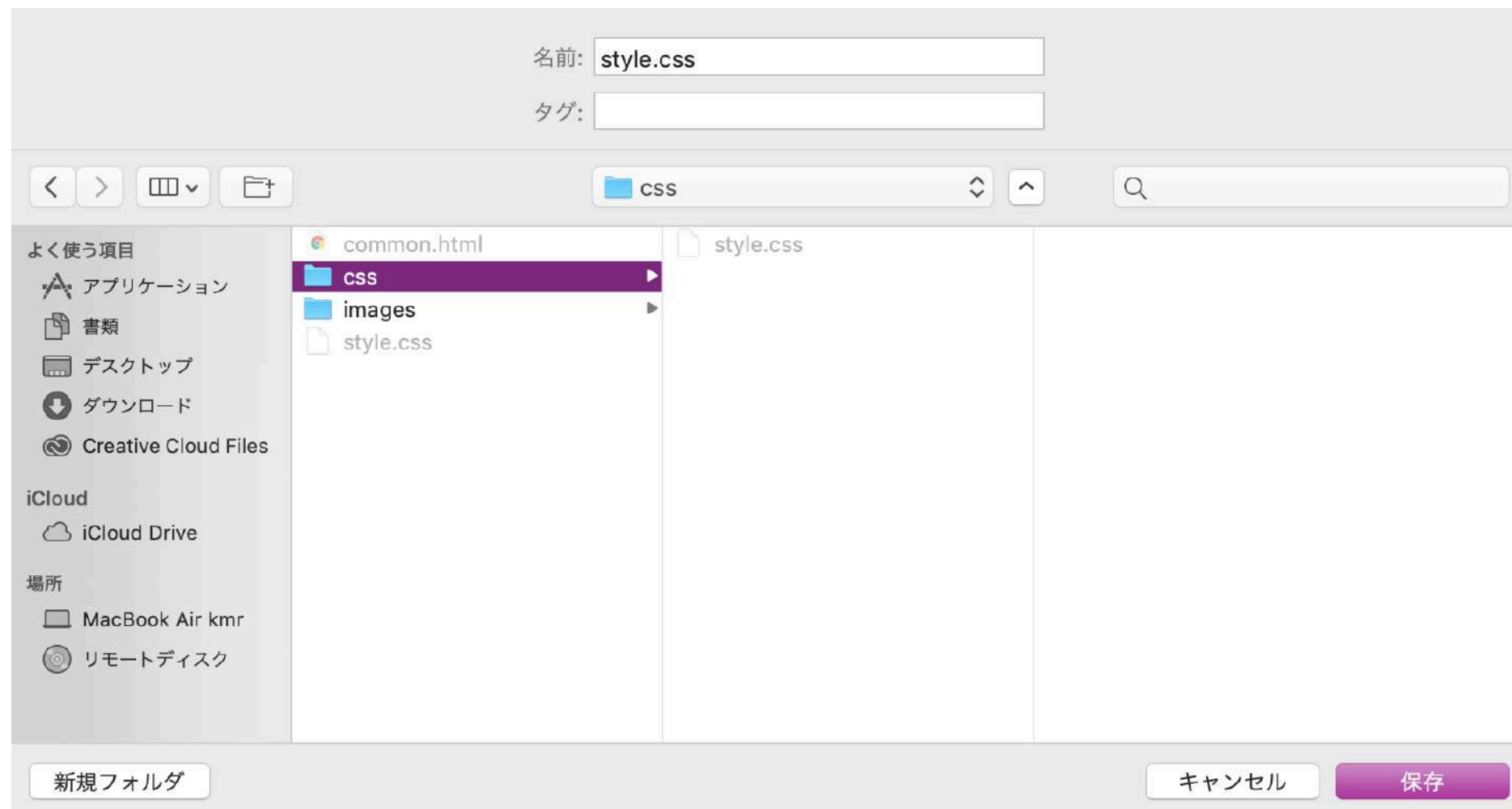
CSSの基本と書き方

- 1 CSSファイルを作成
- 2 ベースになるスタイルの設定
- 3 ブラウザの規定スタイルを解除する
- 4 リンクの書式と装飾の設定
- 5 ヘッダーのスタイルの指定
- 6 コンテンツエリアのレイアウトを整える

2-1. cssファイルを作成

● Brackets起動 Brackets

- メニューの「ファイル」
- 「新規作成」をクリック
- 「ファイル」
- 「名前をつけて保存」
- 「Webページ」フォルダ内の「CSS」フォルダへ「style.css」の名前で保存



2-1. cssファイルを作成

次にHTMLファイルとリンクさせます。common.htmlを開き、head要素内のtitle要素の下にlink要素を使いCSSとリンクさせるために記述を入力。

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>サンプルページ</title>
6   <link href="style.css" rel="stylesheet">
7 </head>
```



style.cssの編集作業に入ります。まず、「CSSファイルの1行目には文字コードを書く」というルールがあるため、それを記述していきます。

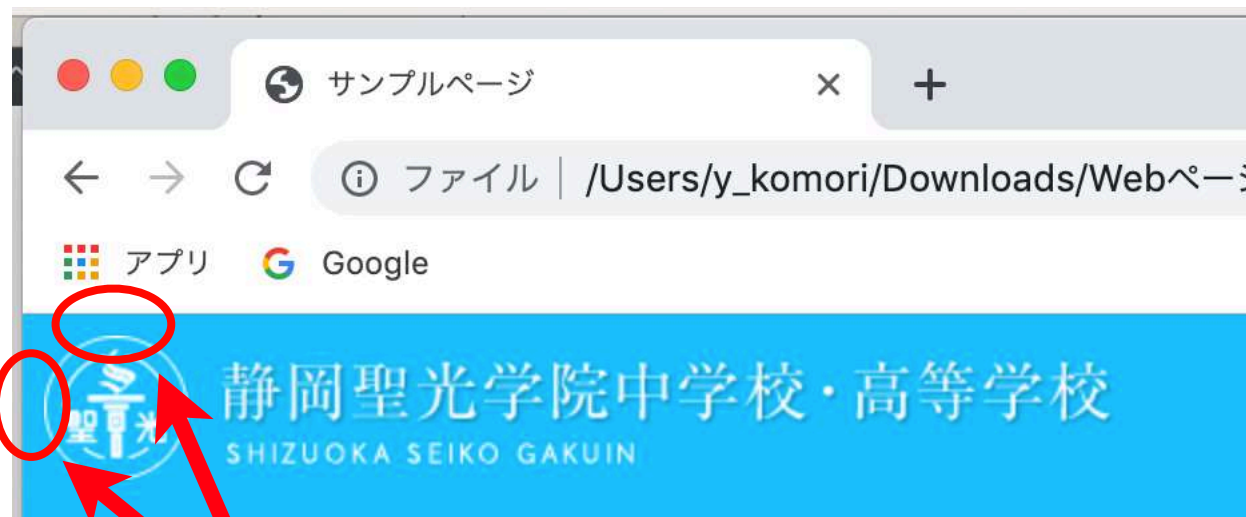
これで、CSSを記述するための準備が整いました。

2-2. ベースになるスタイルの設定

body要素の余白サイズはWebブラウザの種類によって初期値が異なることがある。それを解除するためにbody要素のmarginとpaddingを「0」に指定。

```
1  @charset "utf-8";
2
3  ▼ body {
4      margin: 0;
5      padding: 0;
6  }
```

CSS記述前



上下左右に余白あり

CSS記述後



余白が0になっていればOK

※わかりやすいよう背景色を変更しています

2-2. ベースになるスタイルの設定

●背景色の指定

初期値は白ですが、薄いグレーにします。

背景色の指定には、background-colorプロパティを使用し、値は16進数のカラーコード「#cccccc」で指定します。

●全体の基本となる文字色の指定

colorプロパティを使用して濃いグレー「#333333」にします。

```
3 ▼ body {
4     margin: 0;
5     padding: 0;
6     background-color: #cccccc;
7     color: #333333;
8     font-size: 15px;
9     line-height: 2;
10 }
```

```
3 ▼ body {
4     margin: 0;
5     padding: 0;
6     background-color: #cccccc;
7     color: #333333;
8 }
```



さらにfont-sizeプロパティで文字サイズを
line-heightプロパティで行間を指定。
ここでは特に画面上での変化はありません。

■ 2-3. ブラウザの規定スタイルを解除する

リセットcss

ブラウザのデフォルトスタイルを解除するためのCSSを「リセットcss」と呼びます。今回は、サイト作成に影響のある部分だけをリセットしていきます。

```
12 ▼ p, h1, h2, h3, h4, h5, h6 {  
13     margin-top: 0;  
14 }
```

- 見出しと段落の上のmarginを消す
h要素とp要素にはデフォルトで上部にmarginが設定されています。このmarginをmargin-topプロパティに「0」を設定して解除し、スタイリングをしやすくします。

2-3. ブラウザの規定スタイルを解除する

● 画像の下部の余白をなくす

タグに対してvertical-alignプロパティをbottom (下部揃え)に指定し、画像の縦位置を調整します。

```
16 ▼ img {  
17     vertical-align: bottom;  
18 }
```

● リスト項目の余白をなくす

今後の章で登場するul要素にもデフォルトで余白が設定されています。
margin、paddingともに「0」に設定しておきます。

```
20 ▼ ul {  
21     margin: 0;  
22     padding: 0;  
23 }
```

これでデフォルトスタイルの解除作業は完了です

2-4. リンクの書式と装飾の設定

リンクには「通常のリンク」「訪問済みのリンク」「マウスポインタを置いたとき」など様々な状態があります。各a要素の書式の設定をしていきましょう。

●訪問前/後のリンク文字の色を指定する

通常のリンクの文字色を「#3583aa」

訪問済みのリンクの文字色を「#788d98」

●リンク文字の下線の表示/非表示

テキストにリンクを張ると、初期設定では文字に下線が引かれます。text-decoration プロパティを使用し、通常のリンクには「下線なし」、マウスポインタを置いたときだけ「下線が現れる」ようにします。

Yahoo  訪問済みのリンク
<https://www.yahoo.com/> ▼ このページを訳す

Yahoo! JAPAN - ウィキペディア  通常のリンク
https://ja.wikipedia.org/wiki/Yahoo!_JAPAN ▼

Yahoo! - ウィキペディア  マウスポインタを置いたときのリンク
<https://ja.wikipedia.org/wiki/Yahoo!> ▼

```
25 ▼ a {  
26     color: #3583aa;  
27 }  
28  
29 ▼ a:visited {  
30     color: #788d98;  
31 }
```

```
25 ▼ a {  
26     color: #3583aa;  
27     text-decoration: none;  
28 }  
29  
30 ▼ a:visited {  
31     color: #788d98;  
32 }  
33  
34 ▼ a:hover {  
35     text-decoration: underline;  
36 }
```


2-5. ヘッダーのスタイルの指定

●ヘッダーのサイズを指定

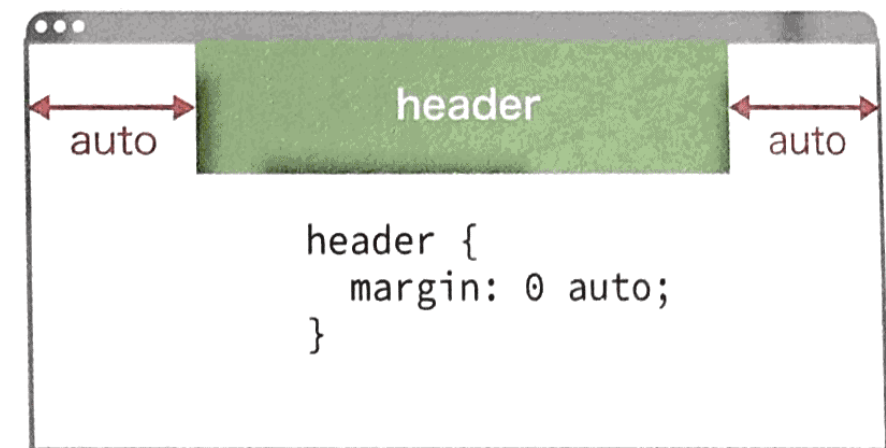
要素の大きさの指定にはwidthとheightプロパティを使用します。
今回は横幅960pxにします。

```
38 ▼ header {  
39     width: 960px;  
40     height: 100px;  
41 }
```

●エリアの位置を指定

ヘッダーエリアをWebブラウザの中央に表示を指定します。中央揃えにはmarginプロパティの上下の余白を「0」に、左右に「auto」という値を設定します。

この指定は必ずwidthで要素の指定をした後に記述します。



```
38 ▼ header {  
39     width: 960px;  
40     height: 100px;  
41     margin: 0 auto;  
42 }
```

2-5. ヘッダーのスタイルの指定

●ロゴの位置を調整

margin-topプロパティで上部の余白を作り、掲載位置を調整します。

```
44 ▼ .logo {  
45     margin-top: 60px;  
46 }
```

class属性のCSS記述

CSSでclass属性をセレクタに指定する際には、ピリオド(.)の後ろにクラス名を記述します。



2-6. コンテンツエリアのレイアウトを整える

●コンテンツエリアを中央に揃える

コンテンツエリア全体がWebブラウザの左端に寄っているため、クラス名

「.content」のdiv要素に対して中央揃えの記述をします。

やることは、headerを中央揃えにしたのと同じ要領で入力していきましょう。

```
48 ▼ .content {  
49     width: 960px;  
50     margin: 0 auto;  
51 }
```



まだコンテンツエリアは空なので
プレビュー表示に変化はなし



フルスクリーンページのCSS

- 1 index.htmlの基本部分を作成する
- 2 見出しを入力する
- 3 本文を入力する
- 4 ボタンを設定する
- 5 見出しを装飾する
- 6 見出しの位置を調整する
- 7 本文を装飾する
- 8 ボタンを装飾する
- 9 フルスクリーンページのための背景画像の設定方法

3-1. index.htmlの基本部分を作成する

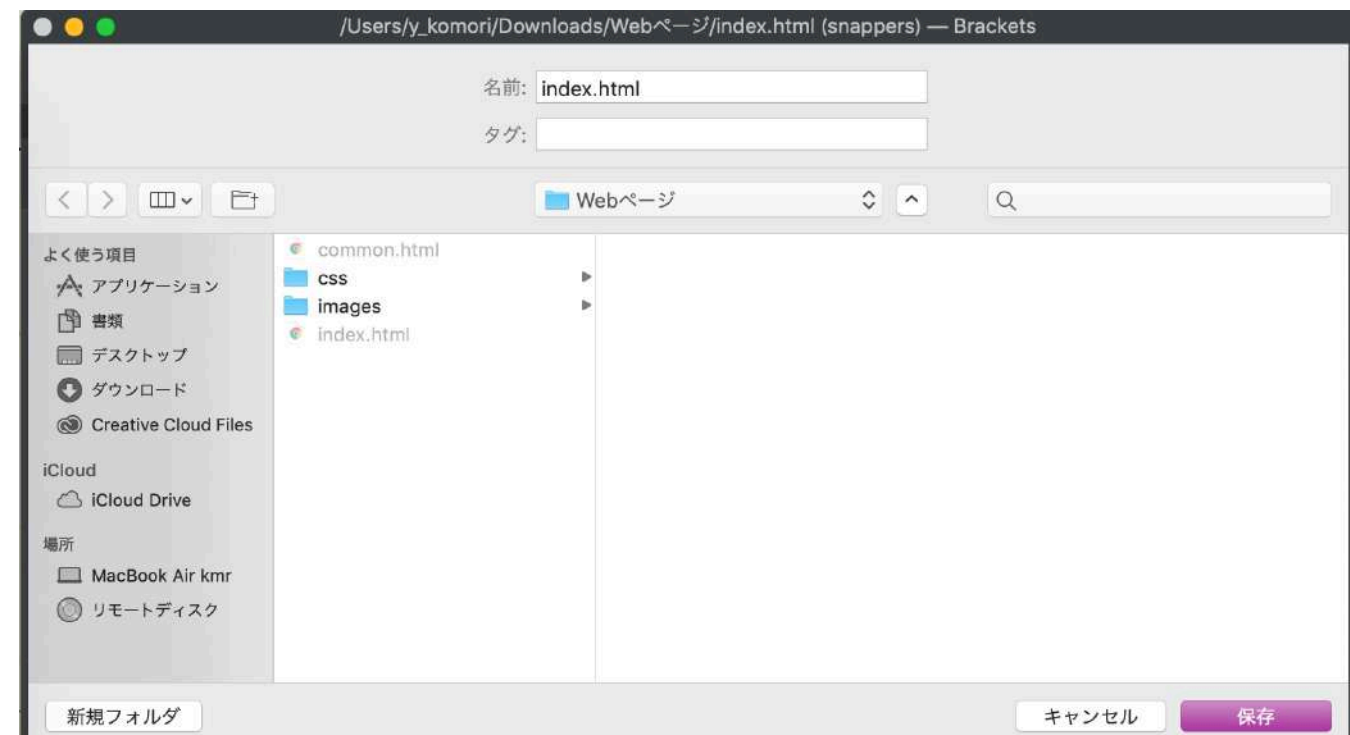
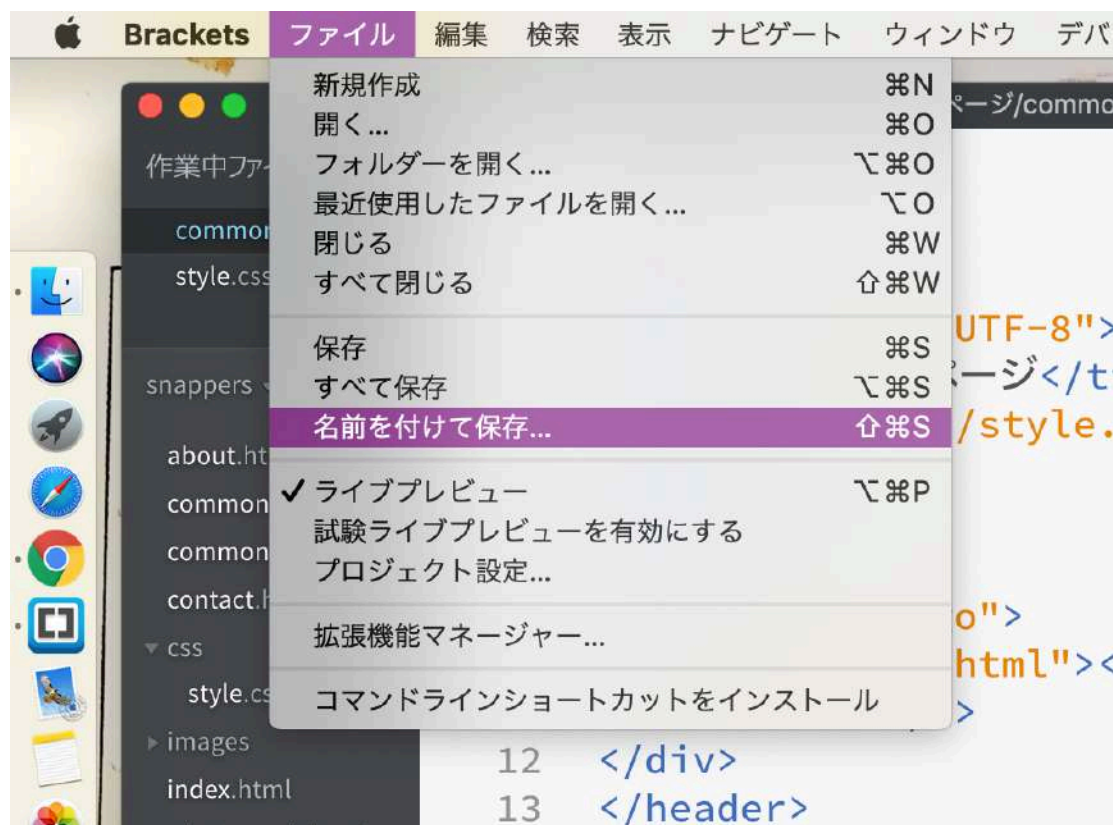
●common.htmlを複製する

「common.html」をBracketsで開く

→「ファイル」

→「名前をつけて保存」

→「index.html」という名前で保存



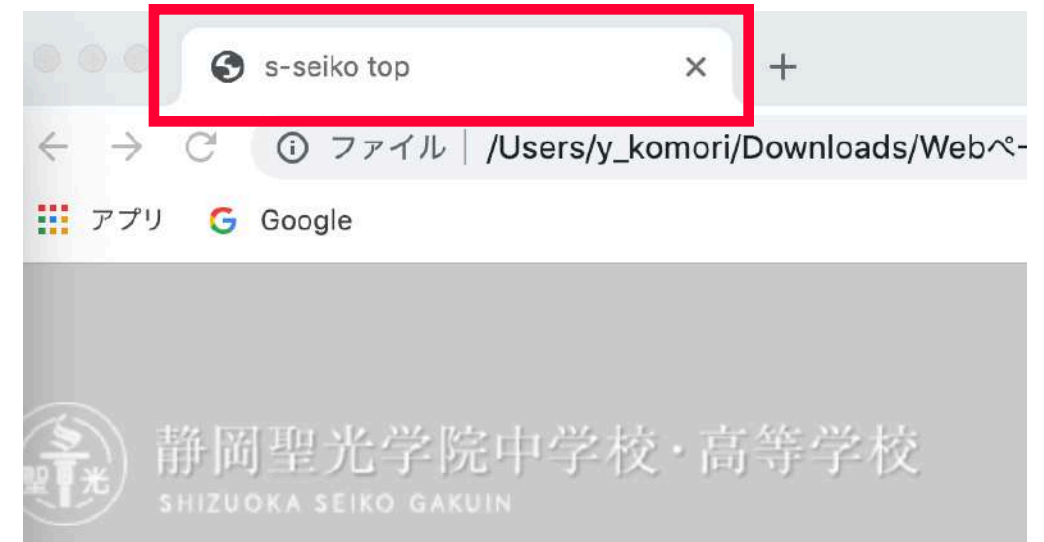
3-1. index.htmlの基本部分を作成する

●ページのタイトルを変更する

common.htmlでは、タイトルが「サンプルページ」となっています。

これを「s-seiko top」に変更します。

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>s-seiko top</title>
6   <link href="css/style.css" re
```



●body要素にid属性を設定する

これはトップページを他のページと区別するための目印として機能します。

こうすることで、CSSで装飾をする際に「トップページの見出しだけ大きくする」など、ページごとに異なる指定ができるようになります。

```
7 </head>
8 <body id="index">
9 <header>
10 <div class="logo">
```


3-2. 見出しを入力する

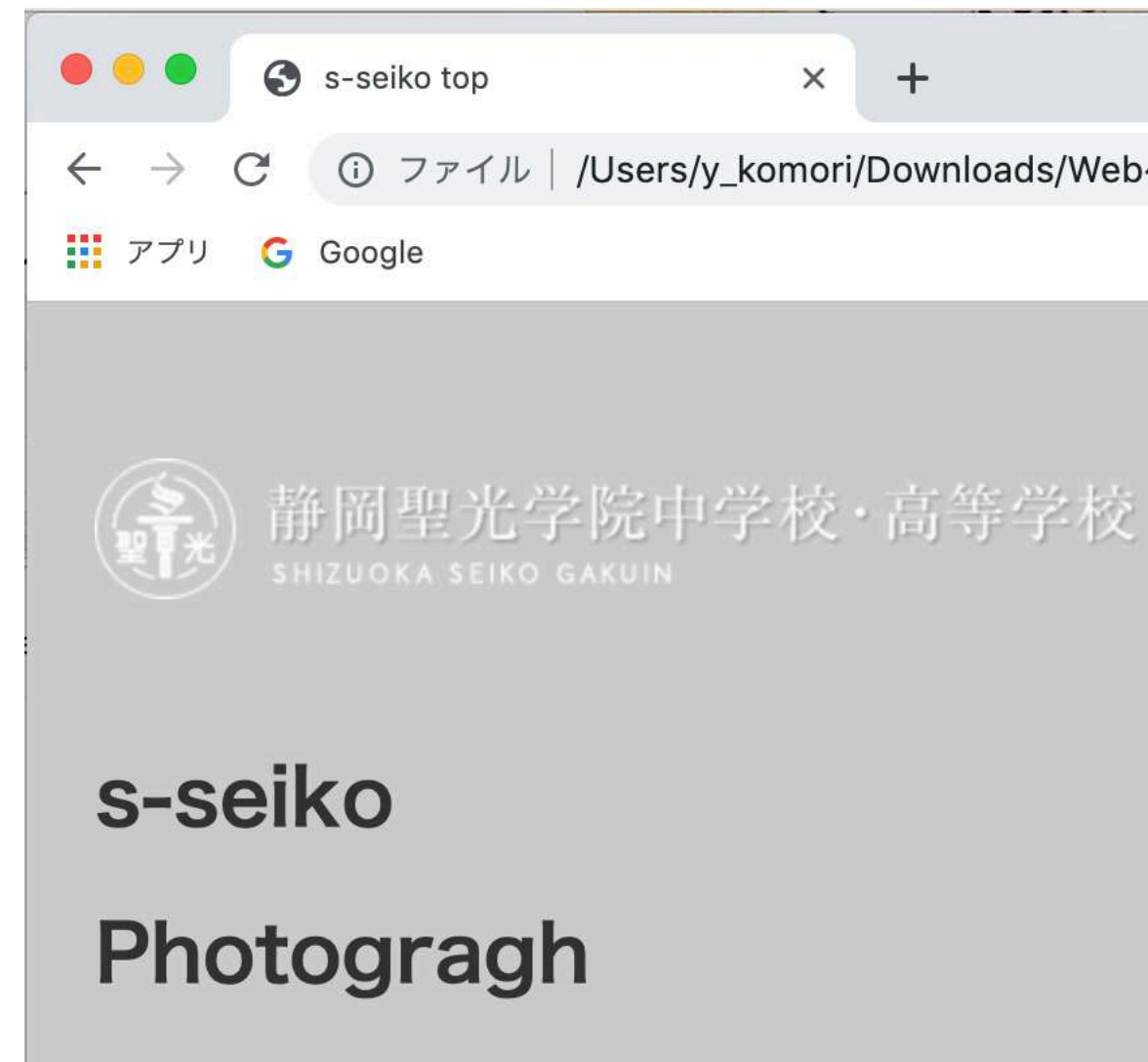
●h1要素で見出し入力

見出しは<h1>から<h6>まであります。
今回はトップページの大見出しを作成したいので、h1要素を使用します。

<div class="content">の中に、見出しの文字を入力していこう。

```
14 ▼ <div id="wrap">
15 ▼   <div class="content">
16     <h1>s-seiko<br>Photograph</h1>
17   </div>
18 </div>
```


タグ：改行



3-3. 本文を入力する

- p要素で本文を入力

見出しの下にp要素を使用して、
本文を記入していきます。

```
15 ▼    <div class="content">
16        <h1>s-seiko<br>Photograph</h1>
17        <p>小高き丘に佇む学び舎、静岡聖光学院の紹介ページです。<br>
          静岡の街や駿河湾を一望。そんな景色や、学校の施設紹介のWebペ
          ージです。</p>
18    </div>
19 </div>
```

3-4. ボタンを設定する

●p要素でボタンの文字列を作成する

まずはp要素でボタンに表示させたい文字列「Photo」を記述します。

この際、p要素にはclass属性で「btn」と指定しておきます。これがCSSでボタンを装飾するための目印になります。

次に、ボタンにリンクを設定します。

「Photo」を<a>タグで囲み、リンク先は「Photo.html」を指定しましょう。

```
16      <h1>s-seiko<br>Photograph</h1>
17      <p>小高き丘に佇む学び舎、静岡聖光学
        の街や駿河湾を一望。そんな景色や、学
        ず。</p>
18      <p class="btn">Photograph</p>
19    </div>
20  </div>
```

```
div class="content">
  <h1>S-SEIKO<br>Photograph</h1>
  <p>小高き丘に佇む学び舎、静岡聖光学院の紹介ページです。<br>静岡の
  街や駿河湾を一望。そんな景色や、学校の施設紹介のWebページで
  す。</p>
  <p class="btn"><a href="photo.html">Photograph</a></p>
</div>
```

s-seiko

Photograph

小高き丘に佇む学び舎、静岡聖光学院の紹介ページです。
静岡の街や駿河湾を一望。そんな景色や、学校の施設紹介

Photograph

3-5. 見出しを装飾する

●文字サイズと行間を指定

2章で作成したstyle.cssにCSSを記述します。

font-size : 見出しをダイナミックにするために、サイズを100pxにします。

line-height : 行間は1にします。

```
53 ▼ h1 {  
54     font-size: 100px;  
55     line-height: 1;  
56 }
```

S-SEIKO
Photograph

小高き丘に佇む学び舎、静岡聖光学院の紹介ページです。
静岡の街や駿河湾を一望。そんな景色や、学校の施設紹介のWebページです。
Photograph

●文字色と書体を指定

文字色 : 白 (# f f f f f f)

書体 : fantasy

```
53 ▼ h1 {  
54     font-size: 100px;  
55     line-height: 1;  
56     color: #ffffff;  
57     font-family: fantasy;  
58 }
```

S-SEIKO
Photograph

小高き丘に佇む学び舎、静岡聖光学院の紹介ページです。
静岡の街や駿河湾を一望。そんな景色や、学校の施設紹介のWebページです。
Photograph

3-5. 見出しを装飾する

- indexページのh1だけに装飾を設定
#indexを追加して、トップページの見出しに限定した記述に変更します。

```
53 ▼ h1 {  
54     font-size: 100px;  
55     line-height: 1;  
56     color: #ffffff;  
57     font-family: fantasy;  
58 }
```



```
53 ▼ #index h1 {  
54     font-size: 100px;  
55     line-height: 1;  
56     color: #ffffff;  
57     font-family: fantasy;  
58 }
```

3-6. 見出しの位置を調整する

●見出しの上の余白を設定

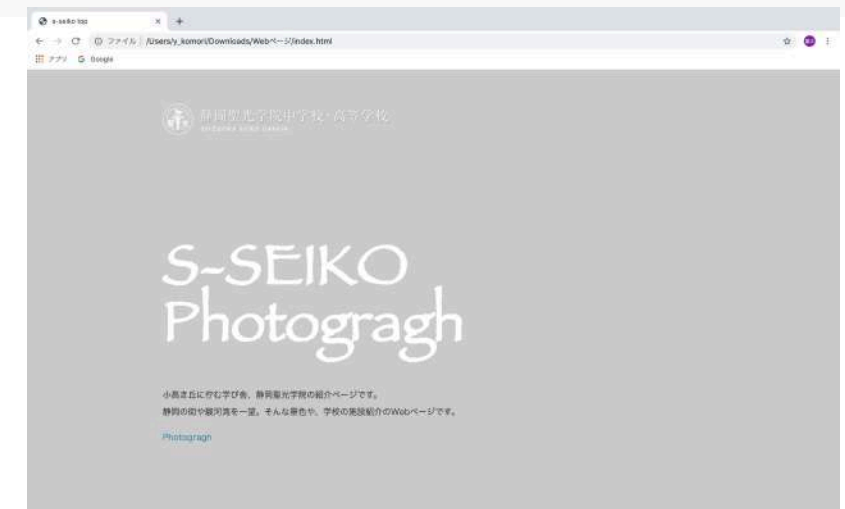
ヘッダーとコンテンツエリアの間に余白を挿入します。class属性「.content」を設定したdiv要素にmargin-topプロパティで余白のサイズを指定しましょう。

```
60 ▼ #index .content {  
61     margin-top: 150px;  
62 }
```

●見出しの下余白を設定

<h1>などの見出し要素には、Webブラウザのデフォルト設定による上下の余白が設けられています。
これはmargin-bottomを使用します。

```
53 ▼ #index h1 {  
54     font-size: 100px;  
55     line-height: 1;  
56     color: #ffffff;  
57     font-family: fantasy;  
58     margin-bottom: 50px;  
59 }
```



3-7. 本文を装飾する

●文字サイズと色の指定

p要素の装飾をします。

ここでもトップページのp要素だけに反映されるように設定します。

文字サイズ：16px

文字色：cyan

```
65 ▼ #index p {  
66     font-size: 16px;  
67     color: cyan;  
68 }
```



S-SEIKO
Photograph

小高き丘に佇む学び舎、静岡聖光学院の紹介ページです。

静岡の街や駿河湾を一望。そんな景色や、学校の施設紹介のWebページです。

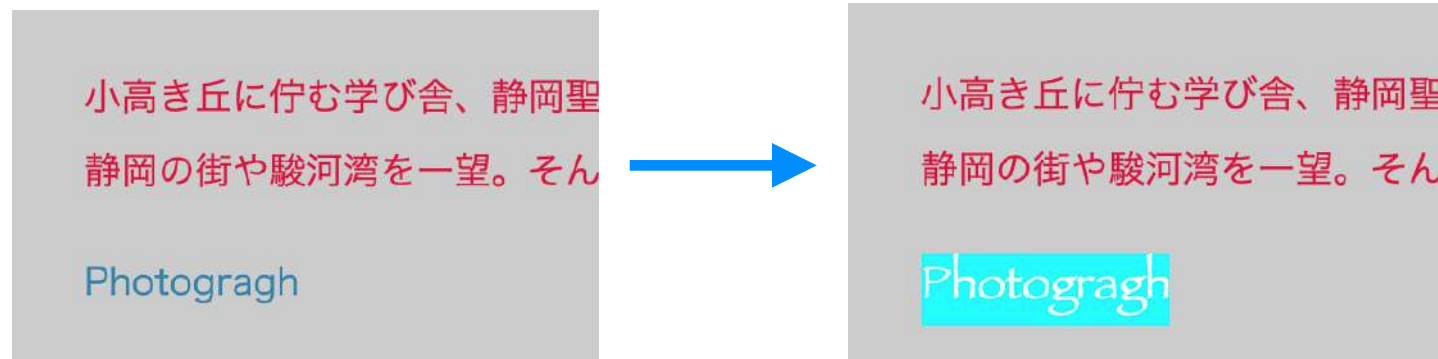
Photograph

3-8. ボタンを装飾する

●ボタンの背景色と文字の装飾

ボタンの色(背景色)と文字の色、サイズ、書体を指定します。

```
70 ▼ .btn a {  
71     background-color: aqua;  
72     color: #ffffff;  
73     font-size: 20px;  
74     font-family: fantasy;  
75 }
```



●ボタンのサイズの調整

ボタンに横幅をつけたいので、widthプロパティで調整します。

しかし、このままでは反映されないの
で、displayプロパティを使用し、値を
「block」と入力。すると、ブロック要素
として扱われるため、高さや幅を指定
できるようになります。

```
70 ▼ .btn a {  
71     background-color: aqua;  
72     color: #ffffff;  
73     font-size: 20px;  
74     font-family: fantasy;  
75     width: 185px;  
76     display: block;  
77 }
```



3-8. ボタンを装飾する

● ボタンの文字位置を調整

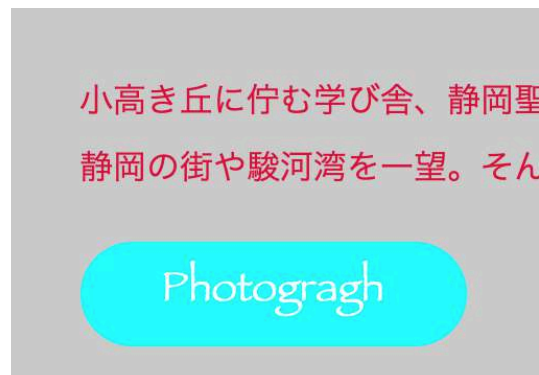
左右の位置を調整したいので、text-align プロパティで「center」を指定します。



```
70 ▼ .btn a {  
71     background-color: aqua;  
72     color: #ffffff;  
73     font-size: 20px;  
74     font-family: fantasy;  
75     width: 185px;  
76     display: block;  
77     text-align: center;  
78     line-height: 50px;  
79     margin-top: 20px;  
80 }
```

● ボタンの角を丸くする

要素の角を丸くする際には、border-radius プロパティを使用します。
スペースで区切ることで、左上・右上・右下・左下の順で、個々に指定することもできます。



```
70 ▼ .btn a {  
71     background-color: aqua;  
72     color: #ffffff;  
73     font-size: 20px;  
74     font-family: fantasy;  
75     width: 185px;  
76     display: block;  
77     text-align: center;  
78     line-height: 50px;  
79     margin-top: 20px;  
80     border-radius: 50px;  
81 }
```


3-8. ボタンを装飾する

● ボタンに枠線をつける

borderプロパティを使用して、背景色と同じ色で枠線をつけます。マウスを置いたときに、色を反転させる準備です。

```
77     text-align: center;
78     line-height: 50px;
79     margin-top: 20px;
80     border-radius: 50px;
81     border: 3px aqua;
82 }
```

● マウスを置いたときのデザインを作る

マウスポインタがボタンに乗ったときの表示を反転させて、より目立つようにしましょう。

.btn a:hoverに対して指定していきます

```
84 ▼ .btn a:hover {
85     text-decoration: none;
86     background-color: #ffffff;
87     color: aqua;
88 }
```

小高き丘に佇む学び舎、静岡聖
静岡の街や駿河湾を一望。そん

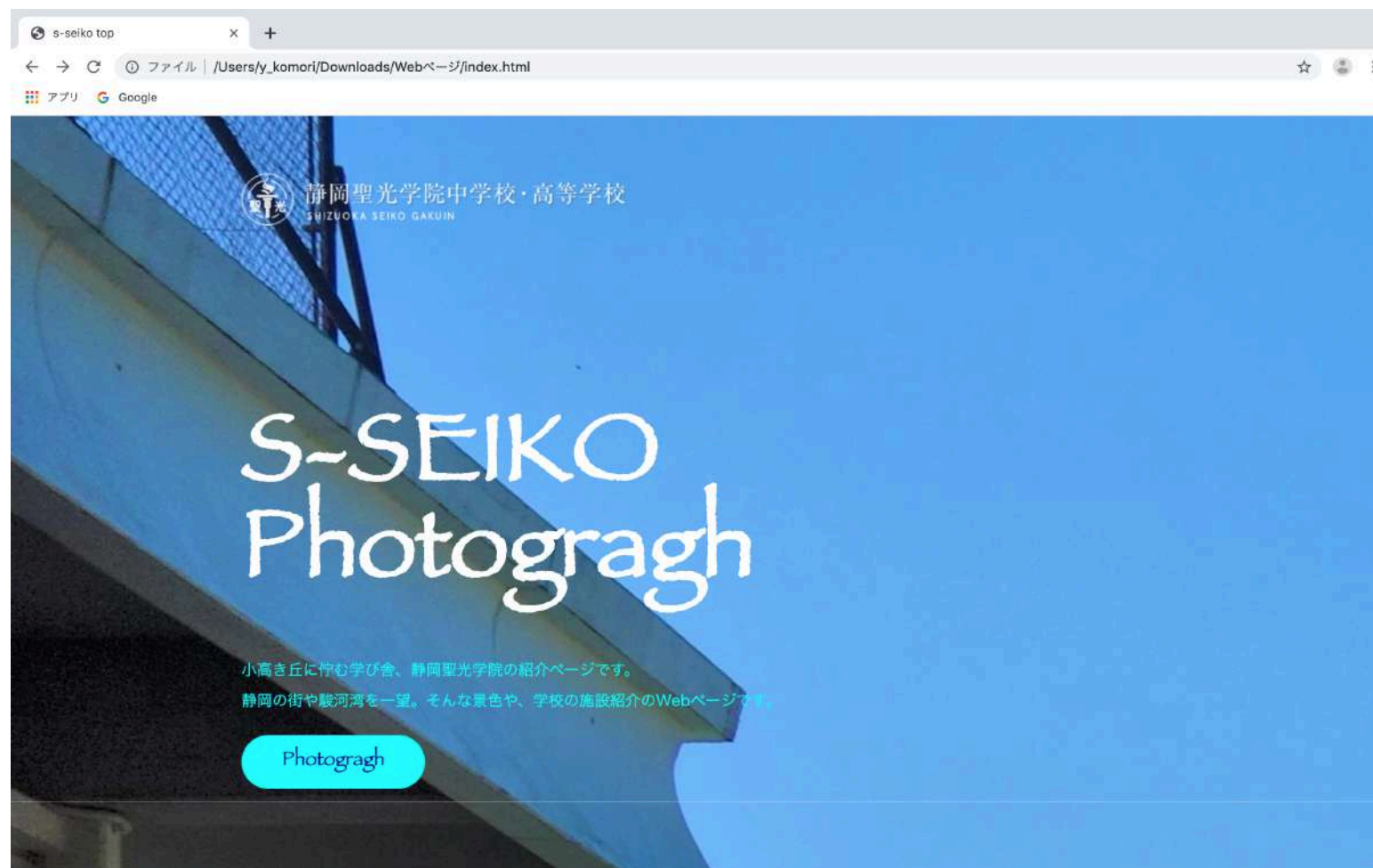
Photograph

3-9. フルスクリーンページのための背景画像の設定方法

● 背景画像のパスを指定する

背景画像を配置するにはbackground-imageプロパティを使用して、画像へのパスを指定していきます。

```
90 ▼ #index {  
91     background-image:  
92     url(../images/seiko.jpg);  
93 }
```



表示はできましたが、画面の大きさに対して画像が大きすぎるので、
上手く表示できていません。微調整が必要です。

3-9. フルスクリーンページのための背景画像の設定方法

●背景画像を微調整する

表示はできましたが、微調整がいくつか必要です。個々に設定していきます

```
90 ▼ #index {  
91     background-image:  
        url(../images/seiko.jpg);  
92     background-repeat: no-repeat;  
93     background-position: center center;  
94     background-attachment: fixed;  
95     background-size: cover;  
96 }
```

background-repeat

背景画像の繰り返しパターンを指定

デフォルトではタイル状に繰り返して配置される

background-position

背景画像の表示位置を指定 値を横位置、縦位置の順にスペースで区切って記述

デフォルトでは左上を基準にして配置される

background-attachment

背景画像のスクロールを指定

background-size

背景画像の大きさを指定

3-9. フルスクリーンページのための背景画像の設定方法



これで、フルスクリーンでデザインしたトップページの完成です。



section

2カラムページのHTML

- 1 photo.htmlの基本部分を作成する
- 2 コンテンツエリアを作成する
- 3 サイドバーのメニューを作成する
- 4 メインエリアのコンテンツを作成する

4-1. photo.htmlの基本部分を作成する

●common.htmlを複製する

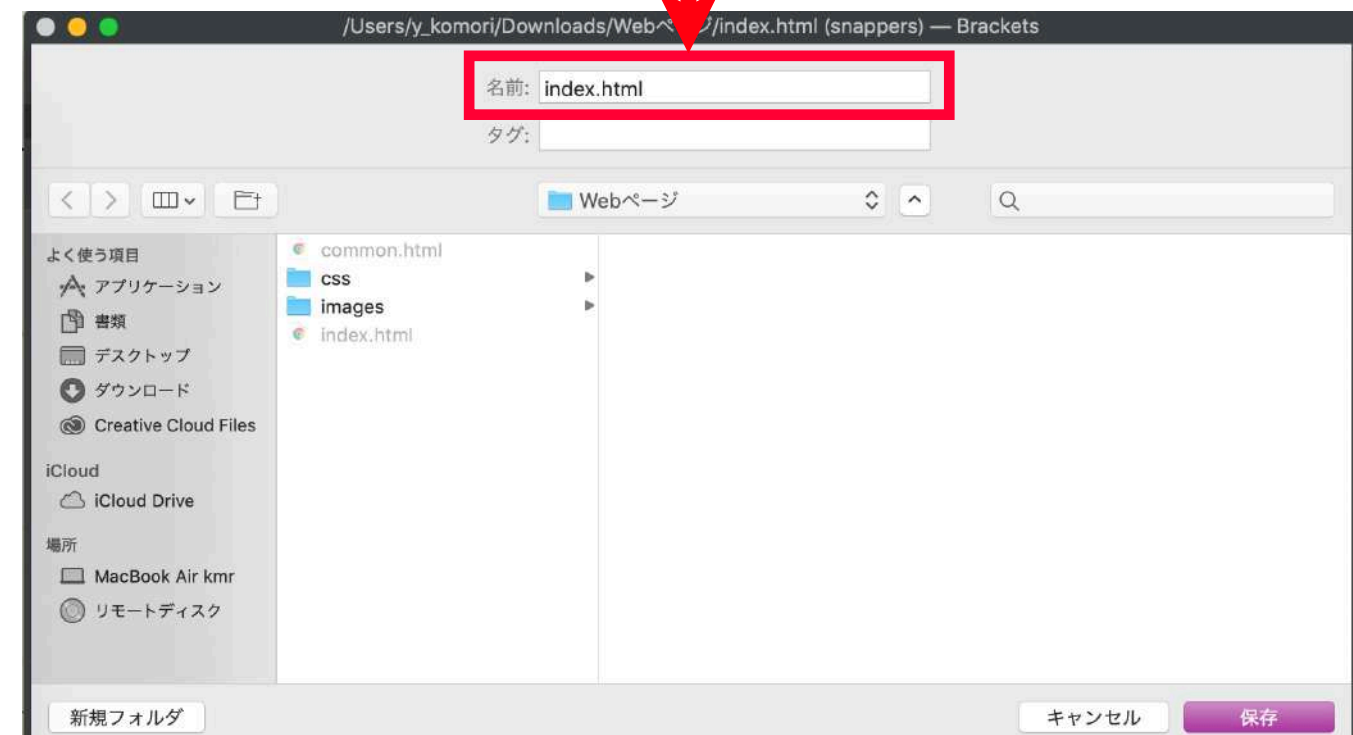
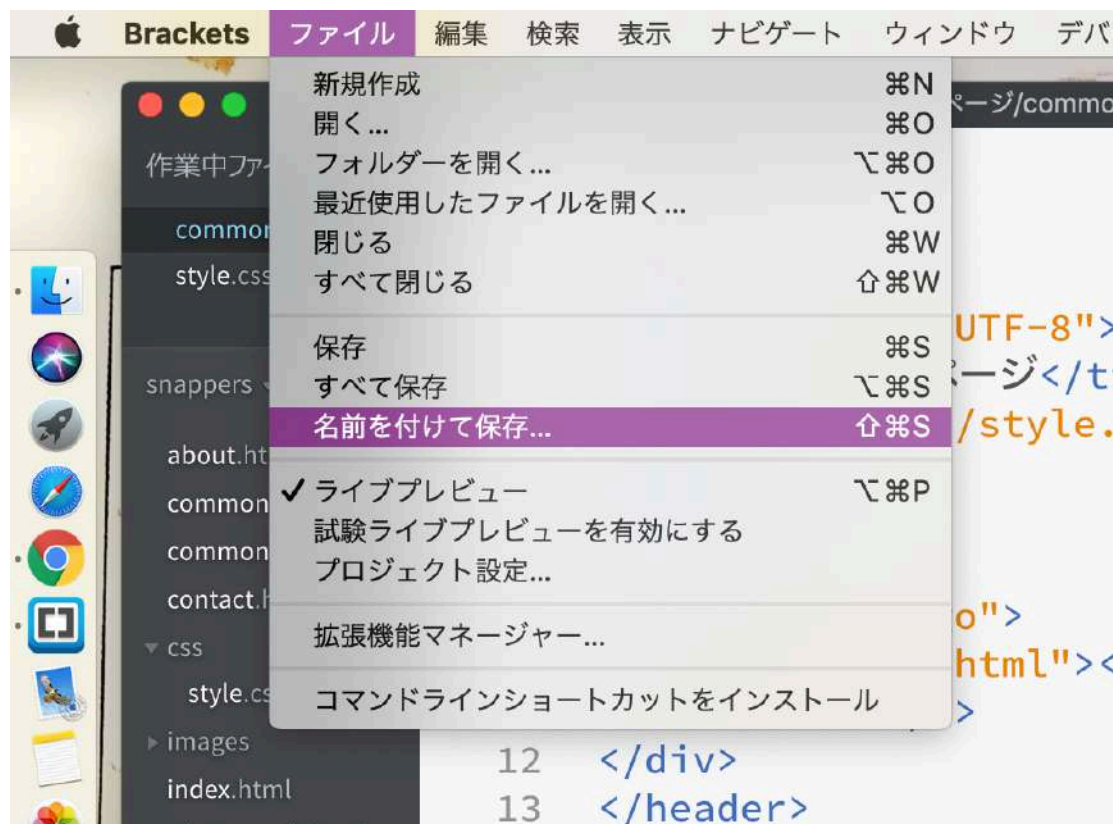
「common.html」をBracketsで開く

→「ファイル」

→「名前をつけて保存」

→「photo.html」という名前で保存

「photo.html」に変更



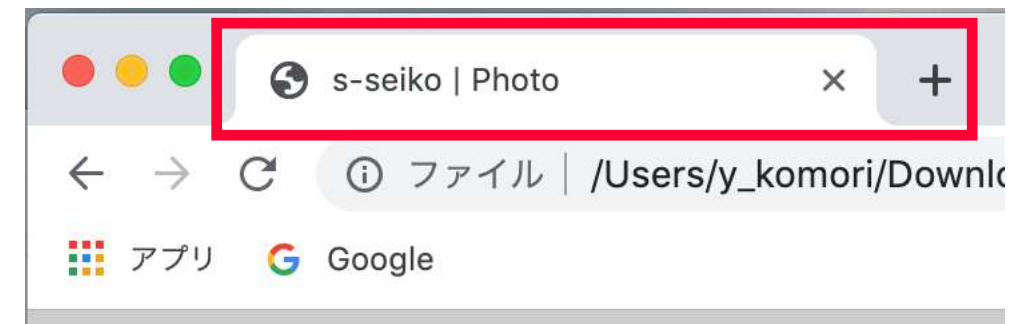
4-1. photo.htmlの基本部分を作成する

●ページのタイトルを変更する

common.htmlでは、タイトルが「サンプルページ」となっています。

これを「s-seiko | Photo」に変更します。

```
4 <meta charset="UTF-8">
5 <title>s-seiko | Photo</title>
6 <link href="css/style.css" rel="sty
7 </head>
```



●body要素にid属性を設定する

3章と同様に、body要素にid属性「photo」を設定します。

```
7 </head>
8 ▼ <body id="photo">
9 ▼ <header>
10 ▼ <div class="logo">
```

●14行目にclass属性を設定する

次の作業で作成するコンテンツエリアの、レイアウト崩れを防ぐために、目印をつけます。

```
13 </header>
14 ▼ <div id="wrap" class="clearfix">
15 <div class="content">
16 </div>
17 </div>
```

4-2. コンテンツエリアを作成する

●コンテンツエリアを2つに分ける

コンテンツエリアをメインエリアとサイドバーの2つに分けます。

メインエリア：div要素 class属性「main」

サイドバー：aside要素 class属性「sidebar」

```
<div id="wrap" class="clearfix">
  <div class="content">
    <div class="main">
    </div>
    <aside class="sidebar">
    </aside>
  </div>
</div>
```

エリアを2つに分けたら、サイドバーから作成していきましょう。

4-3. サイドバーのメニューを作成する

●見出しとメニューを入力する

「sidebar」エリアの中に、section要素を使いセクションを作成し、見出しとメニューを作成します。

見出しはh2要素を使用し、メニューはul要素とli要素でリスト化します

```
18 ▼ <div class="sidebar">
19 ▼   <section>
20     <h2>Photo</h2>
21 ▼   <ul>
22     <li>学校校舎 - school building</li>
23     <li>教室 - classroom</li>
24     <li>特別教室 - specific
      classroom</li>
25     <li>その他 - other</li>
26   </ul>
27 </section>
28 </div>
```



4-3. サイドバーのメニューを作成する

●メニューにリンクを設定する

サイドバーのメニューは下層ページの各項目への場所へ直接リンクする「ページ内リンク」という方法を使用します。

a要素、href属性を使用し「.html」の後に「#〇〇」と記述しましょう。

```
<ul>
  <li><a href="photo.html#school">学校校舎 - school building</a></li>
  <li><a href="photo.html#class">教室 - classroom</a></li>
  <li><a href="photo.html#specific">特別教室 - specific classroom</a></li>
  <li><a href="photo.html#other">その他 - other</a></li>
</ul>
```



次にメインエリアの記述をしていきましょう

4-4. メインエリアのコンテンツを作成する

●見出しとリード文を入力する

見出し：h1要素

リード文：p要素

```
16 <div class="main">
17     <h1>Photograph</h1>
18     <p>学校の施設・様子を撮影した写真を施設ごとに紹
19     介しています。</p>
20 </div>
```



●1つ目のセクションと見出しを作成する

リード文のp要素に続いて、section要素でセクションを作成します。このsection要素には、id属性で「school」という名前をつけておきます。これはページ内リンクの設定の際の目印になります。

また、レイアウト崩れを避けるためにclass属性で「clearfix」と記述することもない。

```
<div class="main">
  <h1>Photograph</h1>
  <p>学校の施設・様子を撮影した写真を施設ご
  介しています。</p>
  <section id="school" class="clearfix">
  </section>
</div>
<div class="sidebar">
```


4-4. メインエリアのコンテンツを作成する

●小見出しと装飾の目印をつける

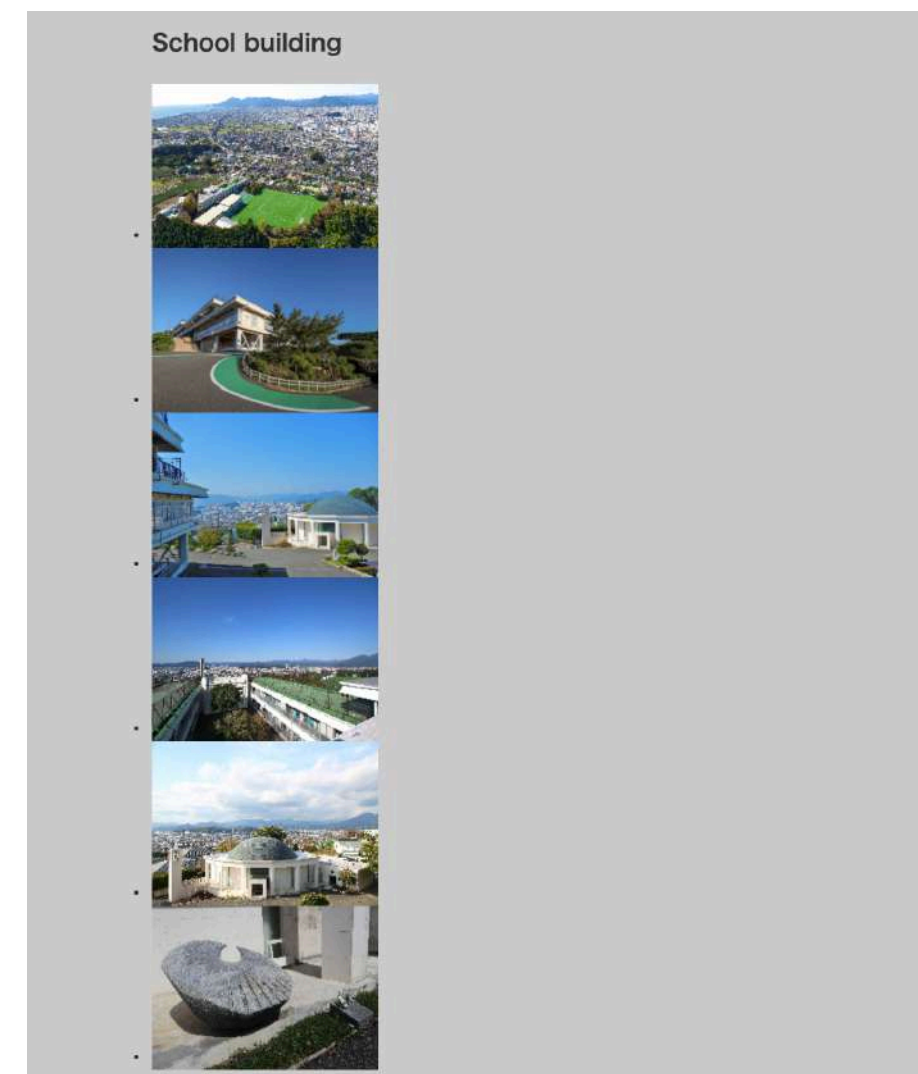
h2要素を使用して、小見出しを作ります。
この見出しは左側にアイコンをつけたいので、class属性で「icon」と指定します。

```
<h1>Photograph</h1>
<p>学校の施設・様子を撮影した写真を施設ごとに紹介しています。</p>
<section id="school" class="clearfix">
  <h2 class="icon">School building</h2>
</section>
</div>
```

●画像ギャラリーを作成する

画像ギャラリーにはul要素、li要素を使用します。画像1枚ずつをli要素でマークアップし、imagesフォルダ内の「0001.png」～「0006.png」を順に指定します。

```
19 <section id="school" class="clearfix">
20   <h2 class="icon">School building</h2>
21   <ul>
22     <li></li>
23     <li></li>
24     <li></li>
25     <li></li>
26     <li></li>
27     <li></li>
28   </ul>
29 </section>
```



4-4. メインエリアのコンテンツを作成する

●alt属性で画像の内容を記述する

img要素にはalt属性が必須なので、alt属性を追加し、各画像の内容を記述します。

また、ul要素にclass属性で「photograph-list」という名前をつけます。

これで、一つ目のセクションは完成です。

```
<ul class="photograph-list">
  <li></li>
  <li></li>
  <li></li>
  <li></li>
  <li></li>
  <li></li>
</ul>
```

●2つ目のセクションを作成する

1つ目のセクションを丸ごとコピーして作成していきましょう。

コピーしたら、section要素のid属性を「class」に書き換え、h2要素の見出しも「Classroom」に変更します。

```
<section id="class" class="clearfix">
  <h2 class="icon">Classroom</h2>
```

コピー

```
16 <div class="main">
17   <h1>Photograph</h1>
18   <p>学校の施設・様子を撮影した写真を施設ごとに紹介しています。</p>
19   <section id="school" class="clearfix">
20     <h2 class="icon">School building</h2>
21     <ul class="photograph-list">
22       <li></li>
23       <li></li>
24       <li></li>
25       <li></li>
26       <li></li>
27       <li></li>
28     </ul>
29   </section>
30   <section id="class" class="clearfix">
31     <h2 class="icon">School building</h2>
32     <ul class="photograph-list">
33       <li></li>
34       <li></li>
35       <li></li>
36       <li></li>
37       <li></li>
38       <li></li>
39     </ul>
40   </section>
41 </div>
```

ペースト

4-4. メインエリアのコンテンツを作成する

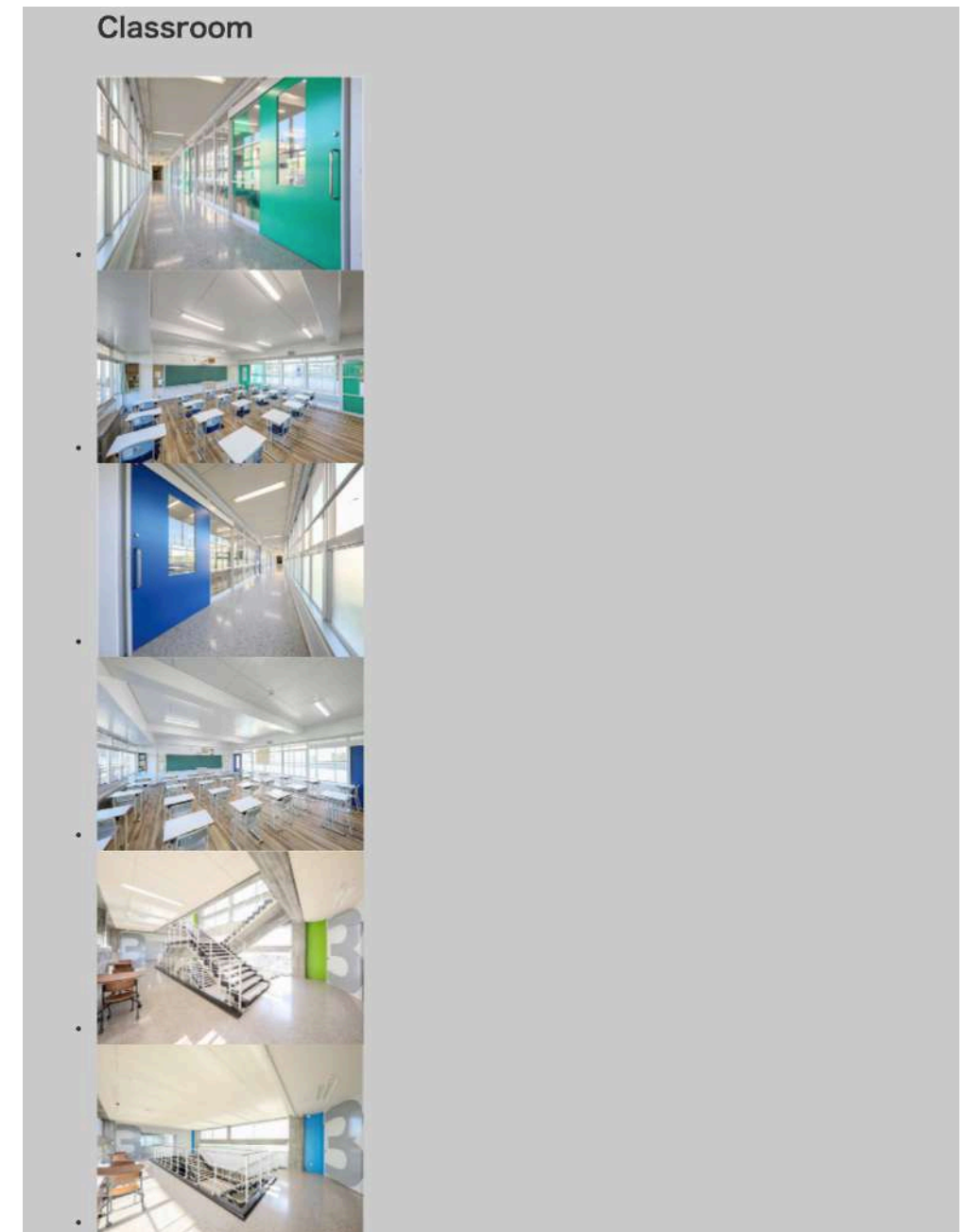
●画像のパスと内容を記述する

各img要素の画像ファイルへのパスとalt属性を、2つ目のセクション用に書き換えます

imageフォルダ内の「c01.png」～
「c06.png」を順に指定します。

同時にalt属性も書き換えていきましょう

```
<section id="class" class="clearfix">
  <h2 class="icon">Classroom</h2>
  <ul class="photograph-list">
    <li></li>
    <li></li>
    <li></li>
    <li></li>
    <li></li>
    <li></li>
  </ul>
</section>
```



4-4. メインエリアのコンテンツを作成する

●3・4つ目のセクションを作成する

先ほどと同じ要領で3つ目と4つ目のセクションを作成します。

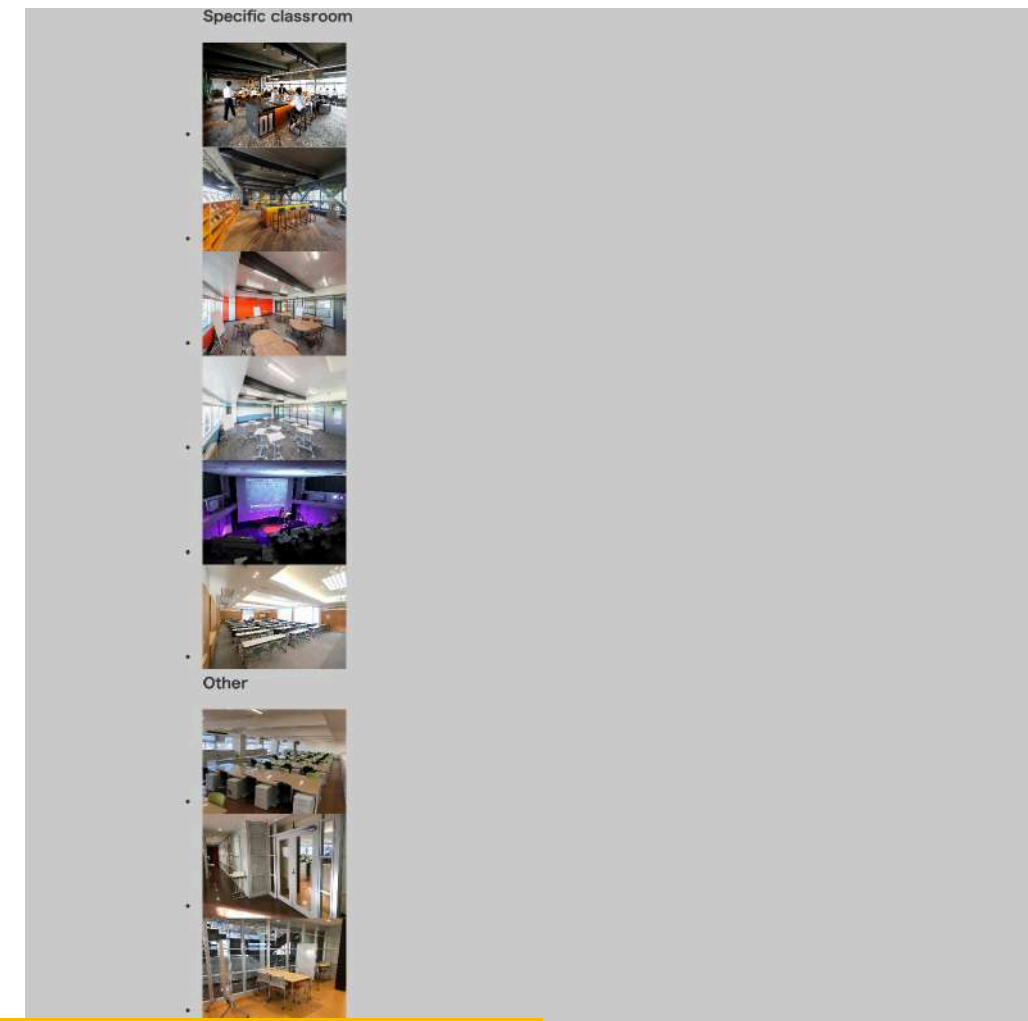
section要素のid属性は「specific」と「other」

h2要素の見出しは「specific classroom」と「other」

imageは、「s01.png ～ s06.png」と「o01.png ～ o06.png」

alt属性も同時に書き換えていきます

```
<section id="specific" class="clearfix">
  <h2 class="icon">Specific classroom</h2>
  <ul class="photograph-list">
    <li></li>
    <li></li>
    <li></li>
    <li></li>
    <li></li>
    <li></li>
  </ul>
</section>
<section id="other" class="clearfix">
  <h2 class="icon">Other</h2>
  <ul class="photograph-list">
    <li></li>
    <li></li>
    <li></li>
    <li></li>
    <li></li>
    <li></li>
  </ul>
</section>
```



これでHTMLの記述は完了です。CSSで仕上げをしていきましょう！



2カラムページのCSS

- 1 メインビジュアルを設定する
- 2 コンテンツエリアのスタイルを設定する
- 3 見出しを装飾する
- 4 2カラムのレイアウトを調整する
- 5 index.htmlに影響を与えている箇所を修正する
- 6 2カラムページのCSSを記述する
- 7 サイドバーのメニューを装飾する
- 8 画像ギャラリーを作成する

5-1. メインビジュアルを設定する

● 背景画像を設定する

body要素のid属性「#photo」に対して
トップページの#indexに指定した内容と
同じ記述をします。

背景画像を指定するbackground-image
のパスを書き換えます。

```
98 ▼ #photo {  
99     background-image: url(../images/photo-bg.jpg);  
100     background-repeat: no-repeat;  
101     background-position: center center;  
102     background-attachment: fixed;  
103     background-size: cover;  
104 }
```

```
90 ▼ #index {  
91     background-image: url(../images/seiko.jpg);  
92     background-repeat: no-repeat;  
93     background-position: center center;  
94     background-attachment: fixed;  
95     background-size: cover;  
96 }  
97  
98 ▼ #index {  
99     background-image: url(../images/seiko.jpg);  
100     background-repeat: no-repeat;  
101     background-position: center center;  
102     background-attachment: fixed;  
103     background-size: cover;  
104 }
```

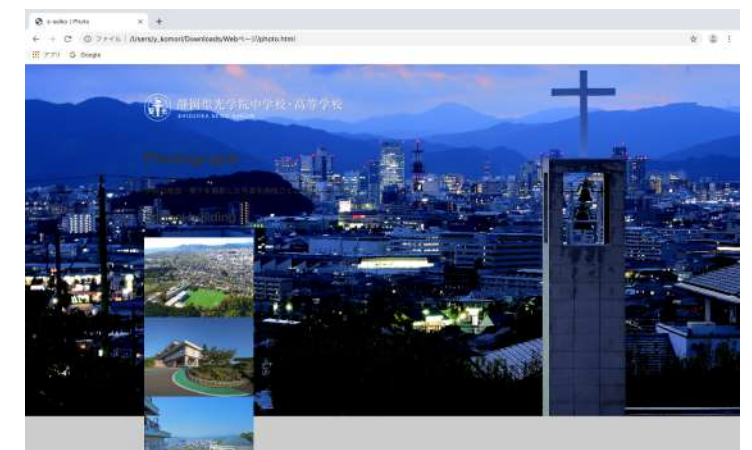


● 画像の大きさ・位置を調整する

背景画像が掲載されましたが、領域に収まるように表示したいので、サイズを変更します。

サイズを変更しただけだと、上下に余白が空いてしまいます。そのため、同時に位置も整えます。

```
101     background-position: center top;  
102     background-attachment: fixed;  
103     background-size: 100% auto;
```



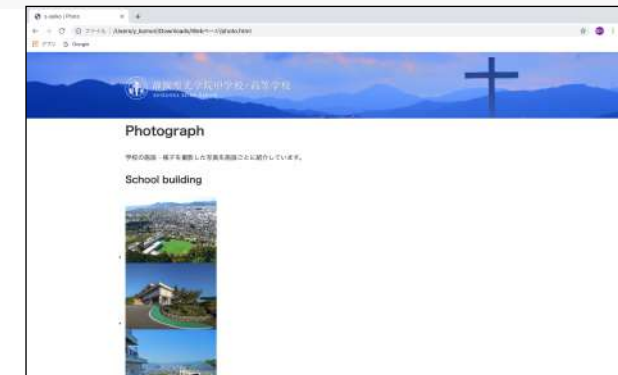
5-2. コンテンツエリアのスタイルを設定する

●背景を白くする

コンテンツエリアの背景は白色にしたいので、`<div id="wrap">`要素に対して背景色を白(`#ffffff`)に指定します。

また、`float`プロパティで左寄せに指定した要素への回り込みを解除します。これには`clear`プロパティを使用します。

```
106 ▼ #wrap {  
107     clear: both;  
108     background-color: #ffffff;  
109 }
```



●上部に余白を設定する

背景が白くなったことで、メインビジュアルが隠れてしまいました。そのため、上部に余白を入れてメインビジュアルが見えるようにしましょう。

```
106 ▼ #wrap {  
107     clear: both;  
108     background-color: #ffffff;  
109     margin-top: 220px;  
110 }
```

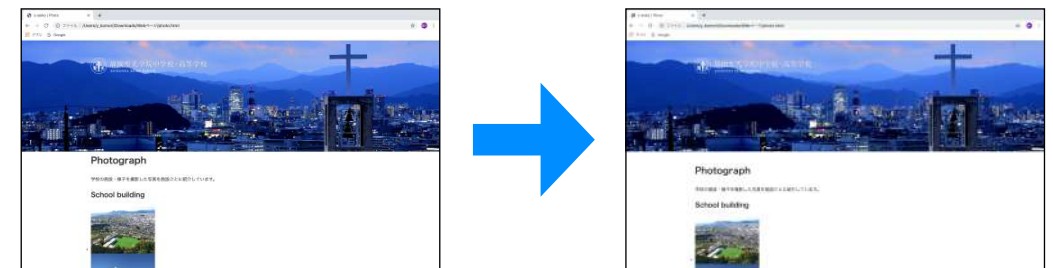


5-2. コンテンツエリアのスタイルを設定する

●エリアの内側に余白を設定する

コンテンツエリアの上下が詰まっていて窮屈なので、paddingプロパティで内側に余白を入れます。

```
106 ▼ #wrap {  
107     clear: both;  
108     background-color: #ffffff;  
109     margin-top: 220px;  
110     padding: 35px 0;  
111 }
```



●メインエリアの幅を指定する

親要素の「.content」に設定した横幅960pxから左右に10pxずつの余白を取り、余白分の20pxを引いた940pxをメインエリアの横幅にします。

「.main-center」セレクトタに940pxの幅を指定し、中央に揃えます。

```
113 ▼ .main-center {  
114     width: 940px;  
115     margin: 0 auto;  
116 }
```

これでレイアウトの大枠は完成です！
次はメインエリア内の装飾を行います

5-3. 見出しを装飾する

●大見出しのスタイルを指定する

シンプルなデザインで装飾は下線のみです。文字サイズを文字フォントを指定して、border-bottomプロパティで見出しの下部に線を引きます。

```
118 ▼ h1 {  
119     font-family: 'Bitter', serif;  
120     font-size: 36px;  
121     border-bottom: solid #cccccc;  
122 }
```



●中見出しのスタイルを指定する

次にh2要素の装飾をします。大見出しと同じフォントを使用して、文字サイズも変更します。

```
124 ▼ h2 {  
125     font-family: 'Bitter', serif;  
126     font-size: 24px;  
127 }
```



5-3. 見出しを装飾する

●アイコンの設定をする

class名「icon」をセレクトにします。
アイコンの作成にはborder-leftプロパティを使用します。

class名「icon」の「の前」にアイコンを追加したいので、類似要素「:before」を使用します。

```
129 .icon:before {  
130 }
```

●アイコンを設定する②

続いて「content」というプロパティを使用し、追加するテキストを「" "」で囲んで記述します。

今回はテキストを追加しませんが、

contentプロパティは:beforeや:afterとセットになっているので、必ず記述します。今回はテキストを追加しないので、空欄のままで構いません。

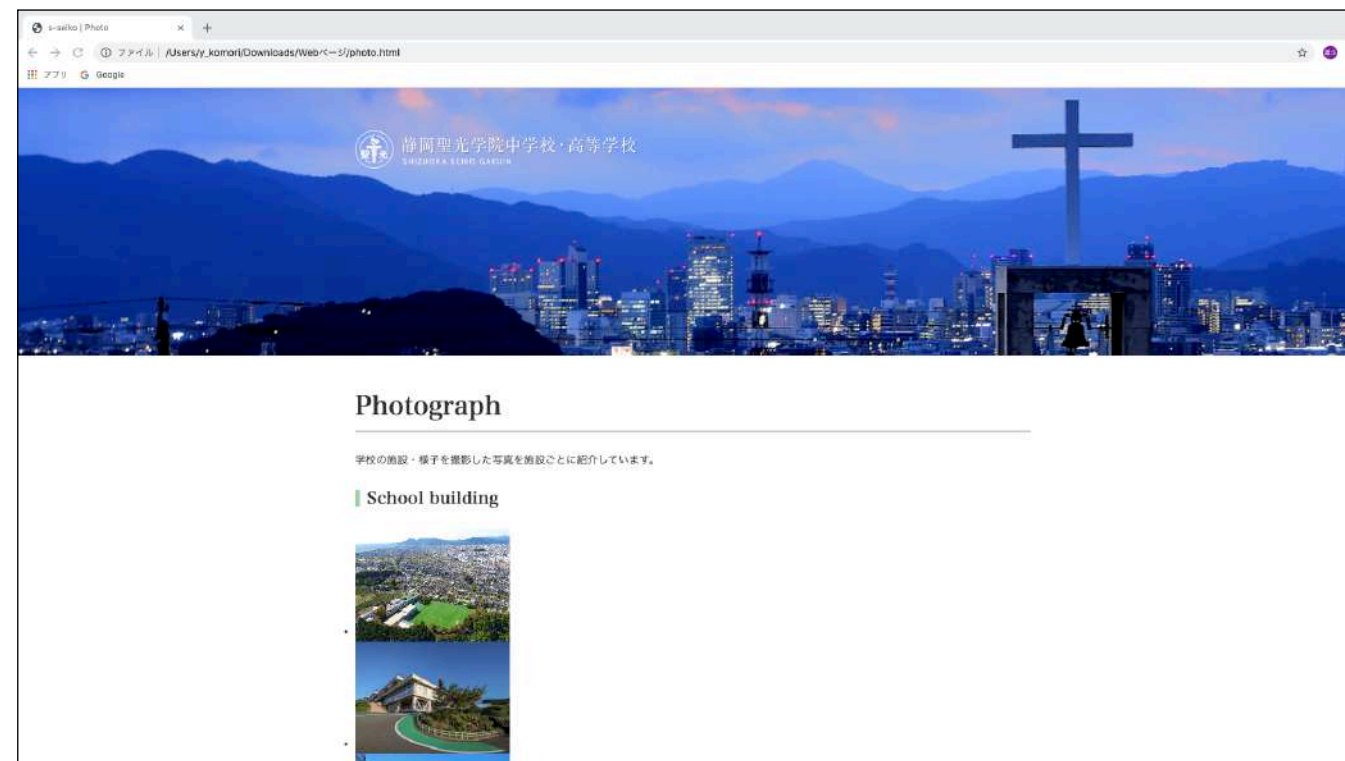
```
129 ▼ .icon:before {  
130     content: "";  
131 }
```


5-3. 見出しを装飾する

●アイコンの設定をする③

次に、見出しとのあいだに隙間を開けた
いので、padding-rightを使用します。
そして、border-leftプロパティを使用
し、左側に7pxの幅のボーダーを引いま
す。これに色を施してアイコンとします

```
129 ▼ .icon:before {  
130     content: "";  
131     padding-right: 10px;  
132     border-left: 7px solid #9cd4a4;  
133 }
```



これで中見出しの左側にアイコンがつけました

5-4. 2カラムのレイアウトを調整する

●clearfixで回り込みを解除する

class名「clearfix」をセレクトタにして記述していきます。

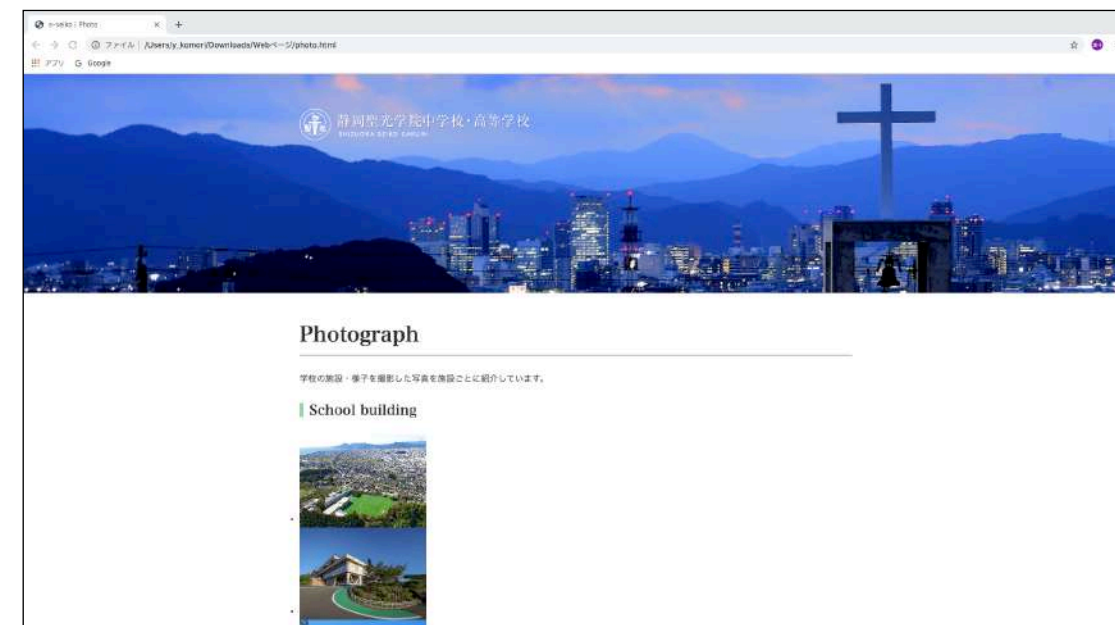
類似要素「:after」を使用してcontentプロパティに空の値を指定します。

displayプロパティを使用し、ボックスをブロックレベル要素として扱うように「block」と指定します。

最後にclearプロパティを使用してfloatを解除します。clearプロパティは、インライン要素に対して胃は無効になりますが、ブロックレベル要素として扱うことにより有効になります。

```
135 ▼ .clearfix:after {  
136     content: "";  
137 }
```

```
135 ▼ .clearfix:after {  
136     content: "";  
137     display: block;  
138     clear: both;  
139 }
```



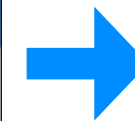
ブラウザ上は特に変化はありません

5-5. index.htmlに影響を与えている箇所を修正する

●背景色の設定を解除する

トップページでは、背景はフルスクリーンの画像となっているため、白い背景は必要ありません。background-colorプロパティに透明を表す「transparent」と入力します。

```
141 ▼ #index #wrap {  
142     background-color: transparent;  
143 }
```



●余白の設定を解除する

上部に指定したmarginと要素内部のpaddingを解除します。

```
141 ▼ #index #wrap {  
142     background-color: transparent;  
143     margin-top: 0;  
144     padding: 0;  
145 }
```



●見出しの飾りを解除する

h1要素の下線を削除します。

「#index h1」のセレクト
(53行目～) に右のように入
力します。

```
53 ▼ #index h1 {  
54     font-size: 100px;  
55     line-height: 1;  
56     color: #ffffff;  
57     font-family: fantasy;  
58     margin-bottom: 50px;  
59     border: none;  
60 }
```



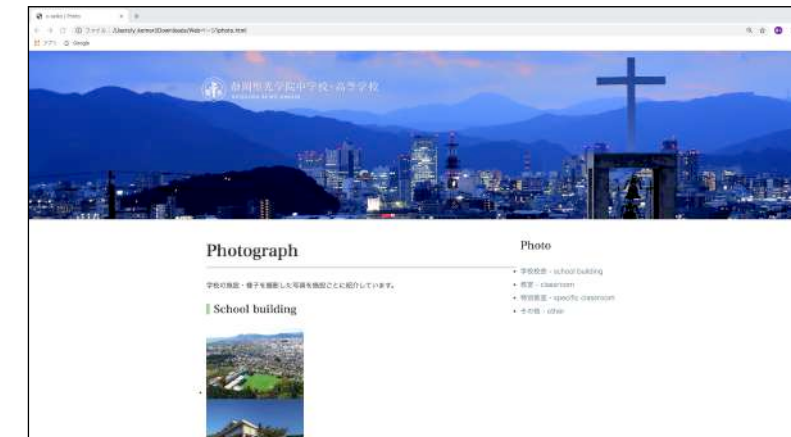
5-6. 2カラムページのCSSを記述する

●メインエリアを左寄せにする

まずはメインエリアの幅と余白を設定します。`.main`の横幅を700pxとし、左右に10pxずつ余白が入るようにmarginを設定します。

次に、`float`プロパティでメインエリアを左寄せにします。

```
148 ▼ .main {  
149     width: 700px;  
150     margin: 0 10px;  
151     float: left;  
152 }
```



●サイドバーを右寄せにする

サイドバーの横幅は215pxとします。そして右寄せにするため、`float`プロパティを「right」と指定します。`margin`の設定値については細かく指定していきましょう。

```
154 ▼ .sidebar {  
155     width: 215px;  
156     margin: 0 10px 0 15px;  
157     float: right;  
158 }
```



5-7. サイドバーのメニューを装飾する

●メニューのフォントを指定する

サイドバー全体にフォントの指定をします。`.sidebar`セレクトタにフォントの指定を追加します。

```
154 ▼ .sidebar {  
155     width: 215px;  
156     margin: 0 10px 0 15px;  
157     float: right;  
158     font-family: serif;  
159 }
```

●見出しの余白を設定する

`margin-bottom`プロパティで余白を調整しましょう。`.sidebar`の中にある

要素に対して余白を指定します。

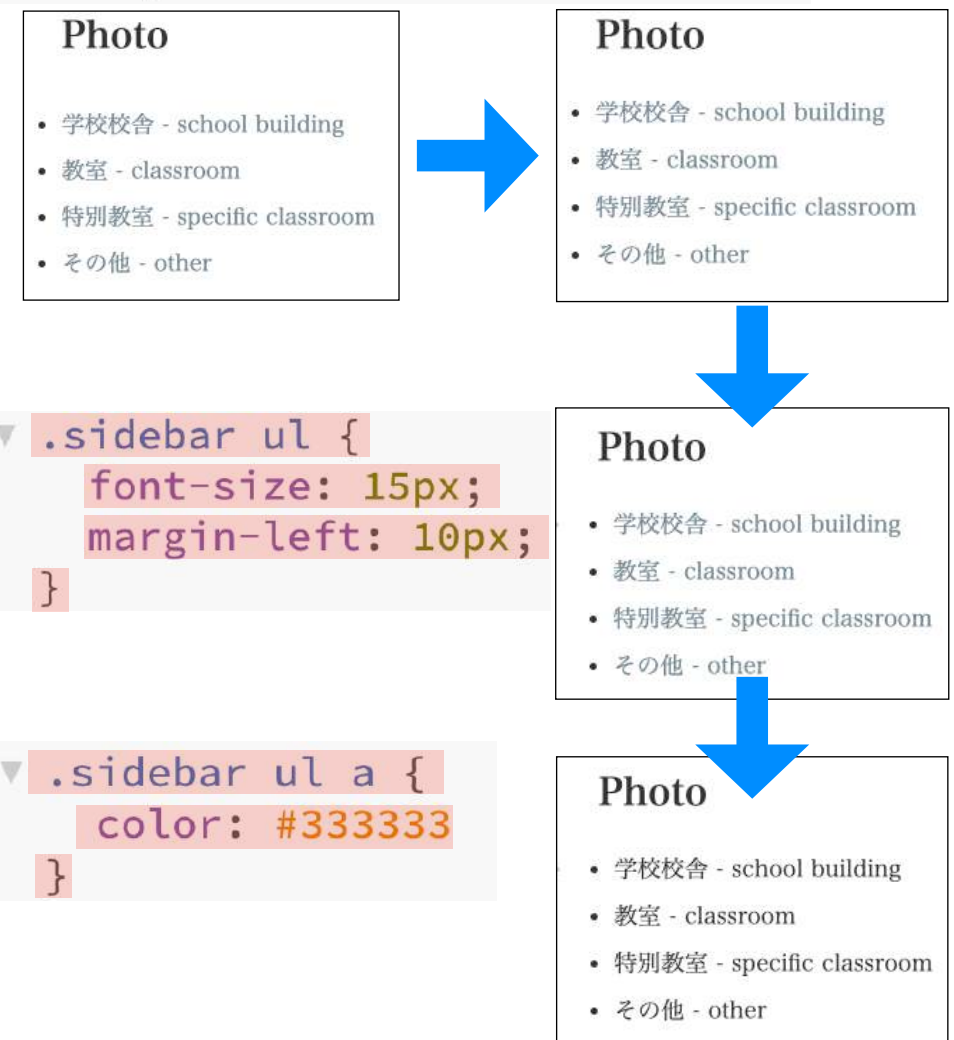
```
161 ▼ .sidebar h2 {  
162     margin-bottom: 10px;  
163 }
```

●リストの見た目を変更する

まずは文字サイズを変更します。そして、行頭アイコンが左にはみ出しているので、`margin-left`で左に余白を作ります。最後にメニュー文字の色を調整します。メニューでは文字色をリンクカラーにしたくないので、`a`要素に対して文字色を指定します。

```
165 ▼ .sidebar ul {  
166     font-size: 15px;  
167     margin-left: 10px;  
168 }
```

```
170 ▼ .sidebar ul a {  
171     color: #333333  
172 }
```



5-8. 画像ギャラリーを作成する

● 画像を左寄せでレイアウトする

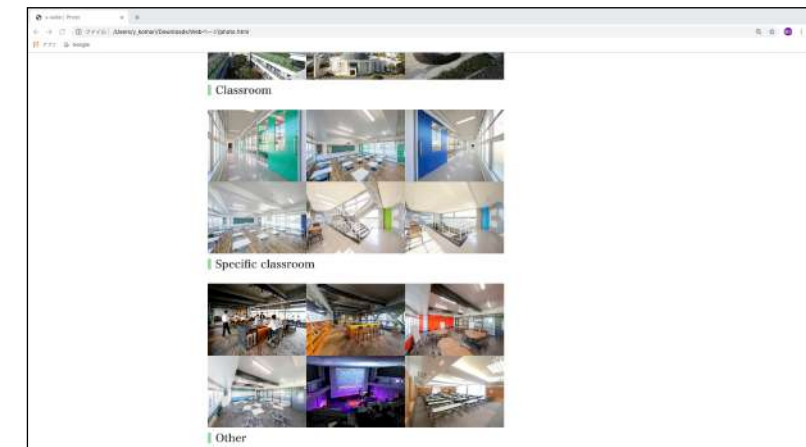
画像を横並びにするためにはfloatプロパティを使用します。class名「photograph-list」のul要素が内包するli要素に「float: left」を指定し画像を左寄せ3×2で表示します。

またli要素の行頭に表示されるアイコンも消しておきましょう。

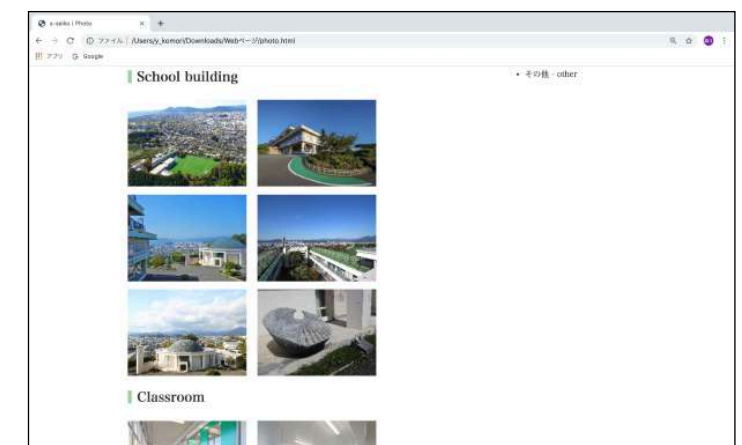
● 画像に余白を設定する

このままでは画像同士がくっついてしまっているので、各画像の間に余白を作ります。marginプロパティを使用して、右と下に余白を設定します。

```
174 ▼ #photo .photograph-list li {  
175     float: left;  
176     list-style: none;  
177 }
```



```
174 ▼ #photo .photograph-list li {  
175     float: left;  
176     list-style: none;  
177     margin: 0 20px 15px 0;  
178 }
```



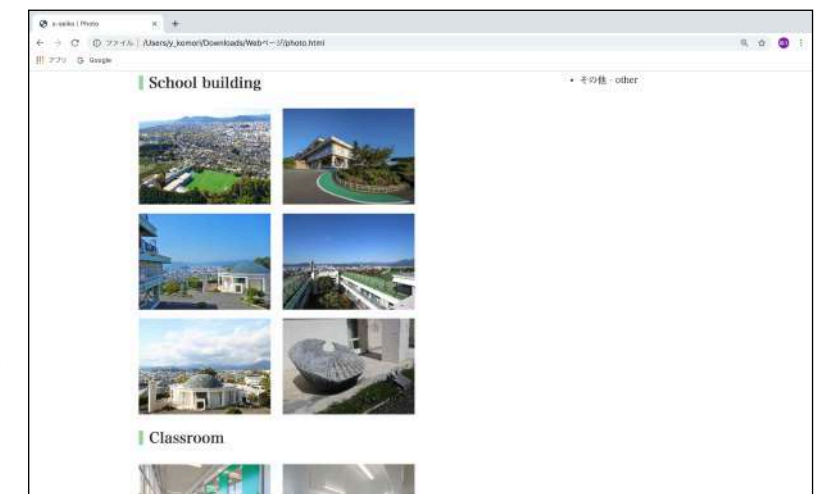
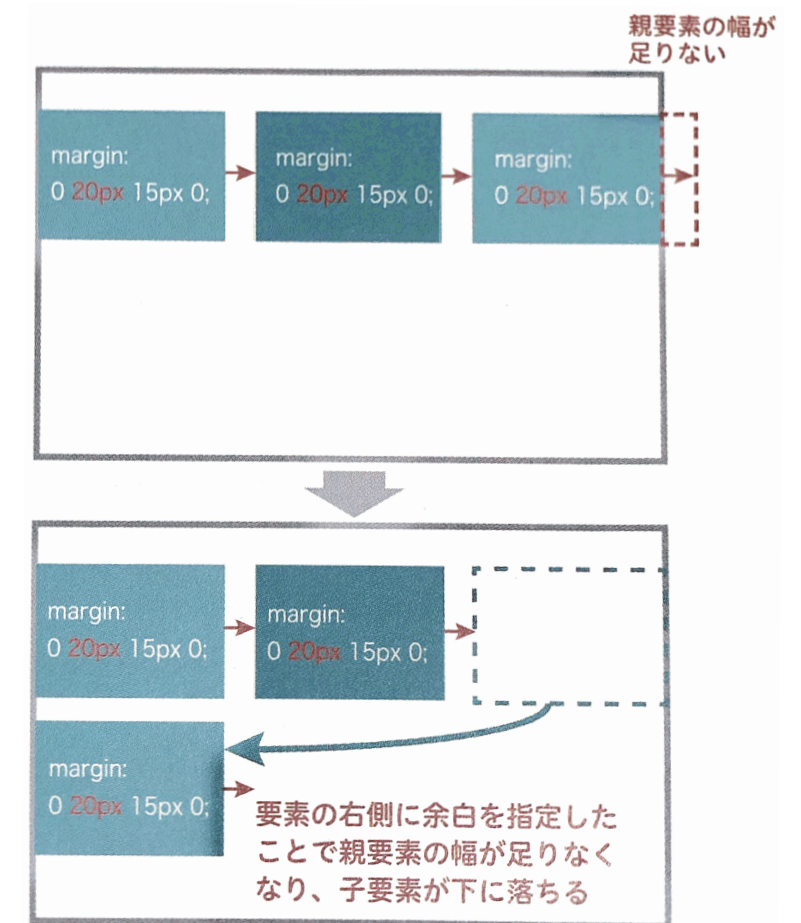
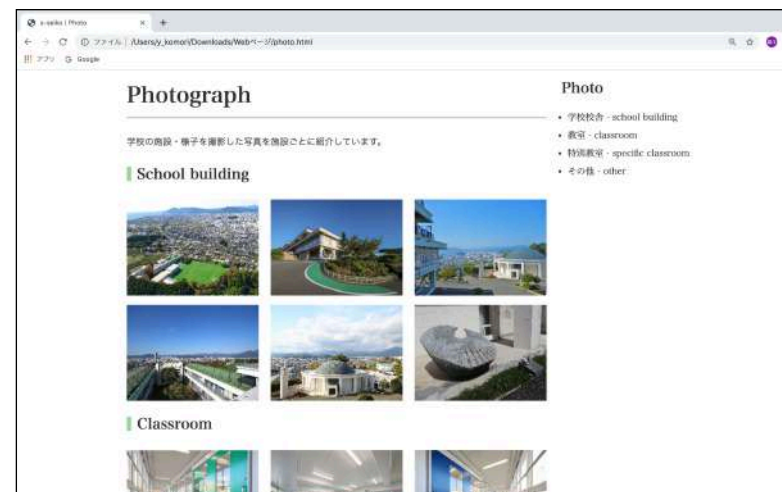
5-8. 画像ギャラリーを作成する

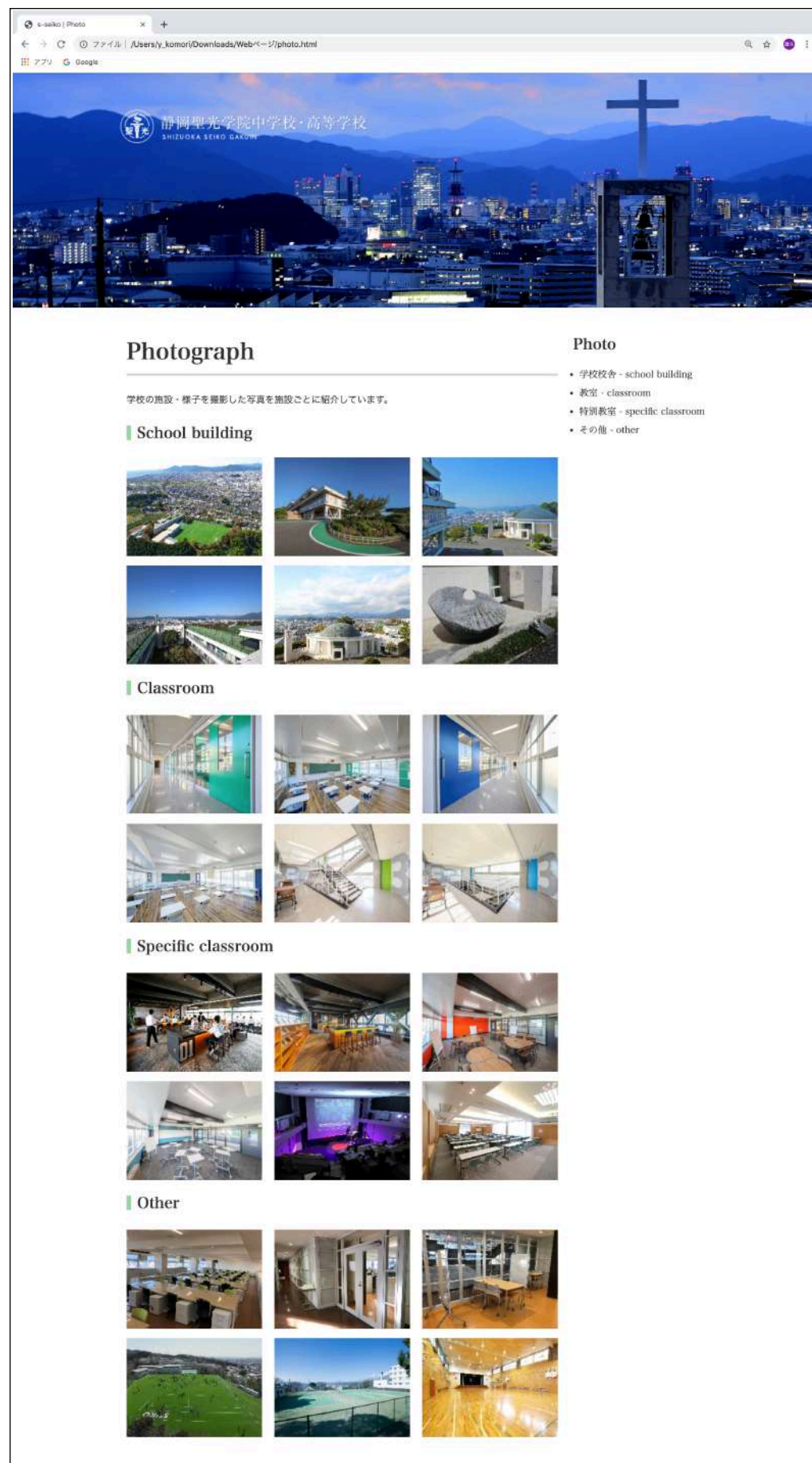
● 3枚目の画像の余白だけ設定を変える

余白を設定したことにより、レイアウトが崩れてしまいました。これを解決するためには、3つ目の画像だけ右の余白を「0」にしたい。

今回は3の倍数だけにマージンの指定をします。類似クラスの「:nth-child(3n)」を使用し、margin-rightプロパティで右側の余白を「0」にします。

```
180 ▼ #photo .photograph-list li:nth-child(3n) {  
181     margin-right: 0;  
182 }
```





これで完成です！
あとは個人の好みで改良していこう