

自作ホームページ  
“Oga’s Homepage”について

Oga

目次	頁
1. はじめに	1
2. ホームページ作成に関する概要	2
2.1. HTML について	
2.2. CSS について	
3. 自作ホームページのデザイン (HTML)	3
3.1. 全ページ共通事項	
3.2. トップページ	6
3.3. 自己紹介ページ	7
3.4. 論文・報告書ページ	8
3.5. プログラミングページ	9
3.6. 更新情報ページ	
3.7. リンク集ページ	10
4. 自作ホームページのデザイン (CSS)	11
4.1. 全ページ共通事項	
4.2. トップページ	12
4.3. 自己紹介ページ	
4.4. 論文・報告書ページ	13
4.5. プログラミングページ	
4.6. 更新情報ページ	
4.7. リンク集ページ	
5. 終わりに	14
6. 参考文献	15

# 1. はじめに

今、日本も含め世界全体で情報化社会化が加速している。自動車の自動運転技術やスマートアシスト家電など、身近なところに情報技術が用いられるようになってきている。そんな様々な情報技術の中で多くの人々が最も多く使用していると共に、最も身近に感じているものはインターネットだろう。かつては、パソコンからしかアクセスすることの出来なかった Web サイトだが、今では手に収まるサイズのスマートフォンからもアクセスすることが可能となった。インターネットにアクセスする人数は増えていく一途を辿っているだろう。

インターネットの利用者が増えていく中で、Web サイトの仕組み(どのように作られているのか)を知っている人は少ないのでは無いだろうか。自分もホームページを自作する前は知らない人の1人であった。ホームページの作り方や仕組みについては数多くの書籍やサイトで記述されているため、このレポート内で説明することはほとんどしない。

このレポートの目的は自作ホームページの紹介であるため、自作ホームページの構造をページごとに解説しながら、HTML と CSS のコードについて説明していく。HTML 文書の方は各ページ異なった構造をしているため、より多くの説明を行っている。一方、CSS ファイルはホームページのデザインの統一を図るために、ページごとにあまり大きく変えていないため、説明が少し簡素になっている。説明の中で写真などを用いて、文書の中身をお見せしていくが、全てを完全に載せることは難しい。もし完全な CSS ファイルや HTML 文書を確認したい場合は、Oga's HomePage を実際に訪れて、自身の目で確認していただきたい。

では、次のページから HTML 文書と CSS ファイルについて軽く概要を述べて、その後に本題である自作ホームページの紹介に入ろうと思う。

## 2. ホームページ作成に関する概要

### 2.1 HTML について

Web ページの内容(中身)を表示させるために用いるものが HTML 文書である。HTML とは”Hyper Text Markup Language”の略称であり、印をつけるための言語という意味である。HTML 文書は head 要素や main 要素など様々な要素に分けて記述されており、それらの要素を示すために記述するものを HTML タグと呼ぶ。HTML タグは要素の始まりではタグを<>のカッコで囲み記述し、要素の終わりはタグに”/“をつけて<>で囲んだものを記述する。

(例)

```
<head> ~~~~~ </head>
```

HTML タグには様々なものがあり、Web ページの内容に適したタグを用いることでより見やすい Web ページを作成することも可能となる。今回の自作のホームページでも様々なタグを用いたが、それについては 3 章以降で適宜説明していくこととする。

### 2.2 CSS について

HTML 文書が Web ページの内容を表示させる物だったのに対し、CSS ファイルは Web ページのデザイン・装飾を記すための文書である。CSS ファイルを記述する際には、HTML 文書で指定した要素をセレクタ(装飾の適用対象)として用いる。記述の際には、指定したいセレクタの後ろに{}を記し、括弧中に装飾の内容を示すプロパティやスタイルを入れていくことになる。プロパティの後ろには:を情報(スタイル)の後ろには;を記述する。

(例)

```
main{background-color:black;}
```

CSS ファイルは記述していくうちに非常に多くの内容を記すことになる。そのため、こまめに反映内容を確認していくことが大切になる。

## 3. 自作ホームページのデザイン(HTML)

### 3.1. 全ページの共通事項

全ページに共通している HTML 文書について説明をしていく。共通部分は文書のバージョンを記す DOCTYPE 宣言、HTML 文書の必要情報が記された”head”部分、そして Web ページ上部(図 1)の内容を記す”header”部分、最後にページ下部(図 2)の情報を記した”footer”部分の大きく分けて4箇所になる。



図 1 ページ上部 (筆者作成)



図 2 ページ下部 (筆者作成)

まずは DOCTYPE 宣言と”head”部分について解説していく。”head”部分に記される主な情報としては文字コードの指定、作者情報、概要文がある。それに加え、ページのタイトルや関連付ける CSS ファイルがある場合もここに記す。

今回、DOCTYPE 宣言では<!DOCTYPE html>と記すことで今回の HTML 文書が HTML5 の文書であることを記述した。

次に、”head”部分である。まずは文字コードの指定は<meta charset=“utf-8”>と記述することで utf-8 という文字コードで記述されることを指定した。次に<meta name=“author” content=“Oga”>という記述でこのページの作者が Oga であることを示した。さらに、ページの概要文として<meta name=“description” content=“Oga’s HomePage”>と記述することで、このページの概要文として”Oga’s Homepage”を追加した。今回はタイトルと、関連 CSS ファイルも存在するためその記述も行った。タイトルの設定として<title>ページ名:Oga’s HP</title>のように<title>で囲って記述した。このタイトルは、Web サイトを表示した際のタブ部分(図 3)に表示される。最後に関連づけるファイルの指定として<link

href="TopPage.css" media="all" rel="stylesheet">と記述した。TopPage の部分を変更することで、各ページでそれぞれの CSS ファイルを対応させることができる。

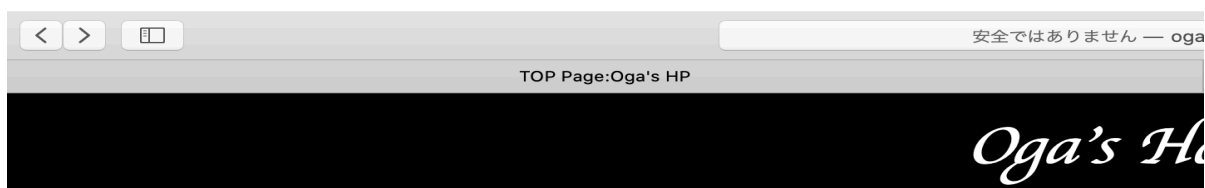


図3 タブ部分のタイトルの表示 (筆者作成)

次に”header”部分である。ここの内容は主に、ページの名前”Oga’s HomePage”を見出しとして表示することと、各ページに飛ぶことの出来る様にリンク付きのタブを表示することである。まず Web ページの名前を大きく表示するために見出しを作るタグである<h1>タグを用いて<h1 name=“title”><a href=“index.html” class=“title”> Oga’s HomePage</a></h1>のように記述した。リンクをつける<a>タグにトップページのリンクを設定することで、ページのタイトルをクリックすることでトップページを表示できるように設定した。次に各ページに飛ぶためのタブの記述だ。今回の Web ページでは、リストを作成する<ul>タグと、表を作成する<table>タグを用いて以下のように記述した。なお、ページに対する表部分のバランスの関係上、表の横幅を”width”を用いて 97%に指定している。

```
<ul>
<table width=97% border=0 >
<tr>
<th><li><a href=“index.html” class=“index”>トップページ</a></li></th>
<th><li><a href=“Introduction.html” class=“index”>自己紹介</a></li></th>
<th><li><a href=“Papers.html” class=“index”>論文・報告書</a></th>
<th><li><a href=“Programing.html” class=“index”>プログラミング</a></li> </th>
<th><li><a href=“Updating Information.html” class=“index”>更新情報</a></li></th>
<th><li><a href=“Link.html” class=“index”>リンク集</a></li></th>
</tr>
</table>
</ul>
```

最後に”footer”部分である。ここには主に著作権情報を記載する機会が多くなる。 ”Copyright 2020 Oga.”と記すこともできるが、今回の Web ページでは”©2020 Oga.”と記したかった為、`<p>&copy; 2020 Oga.</p>` というように記述した。”&copy;”によって©を表示させることができる。

ここまで説明してきた、全ページに共通する部分の HTML 文書は(図 4)のようになる。タイトルなどはページごとに異なる。

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <meta name="author" content="Oga">
6 <meta name="description" content="Oga's HomePage">
7 <title>TOP Page:Oga's HP</title>
8 <link href="TopPage.css" media="all" rel="stylesheet">
9 </head>
10 <body>
11 <header>
12 <h1 name="title"><a href="index.html" class="title"> Oga's HomePage</a></h1>
13 <div>
14 <nav>
15 <ul>
16 <table width=97% border=0 >
17 <tr>
18 <th><li><a href="index.html" class="index">トップページ</a></li></th>
19 <th><li><a href="Introduction.html" class="index">自己紹介</a></li></th>
20 <th><li><a href="Papers.html" class="index">論文・報告書</a></th>
21 <th><li><a href="Programing.html" class="index">プログラミング</a></li></th>
22 <th><li><a href="Updating Information.html" class="index">更新情報</a></li></th>
23 <th><li><a href="Link.html" class="index">リンク集</a></li></th>
24 </tr>
25 </table>
26 </ul>
27 </nav>
28 </header>
29 <footer>
30 <p>&copy; 2020 Oga.</p>
31 </footer>
32 </body>
33 </html>
```

図 4 ここまでの HTML 文書 (筆者作成)

## 3.2. トップページ

トップページの主な機能は簡単な挨拶文とページを飾るための、画像を表示することにした。画像を表示するために使用する `img` タグを用いて画像を表示した。その際に、高さや幅をそれぞれ指定することによって画面の幅に対して、画像が極端に小さくなったり大きくなったりしないように調整した。この下には、簡単な挨拶文を入れるために `section` タグで区切った。挨拶文の初めの部分は大きく表示したかったため `h2` タグを用いて少し大きく表示できるようにした。挨拶文の内容部分は `p` タグで囲むことで段落と指定するとともに、`br` タグを用いて改行をすることで読みやすくした。トップページの HTML 文書は以下の図 5 のようになる。

これに加え、トップページには footer の部分にアクセス数のカウンターを挿入した。これ

```
29 <main>
30 <br>
31 <section>
32 <h2>こんにちは 😊</h2>
33 <p>
34 このページを訪れていただき、ありがとうございます。<br>
35 これは私の初めての自作のホームページです。<br>
36 論文やプログラミングなどをアップしていく予定です。<br>
37 </section>
38 </main>
```

図 5 トップページの HTML 文書（筆者作成）

は、ホームページを訪れた人の数を自動で数えてくれるものである。いくつか無料サービスはあるが、その中で「無料アクセスカウンター」というサイト(<http://www.rays-counter.com>)で作成した。カウンターの記述は自動的に作成してくれるものであり、とても複雑なものであるが、参考程度にこちらに内容を記述しておく。

```
<table border="0" cellspacing="0" cellpadding="0">
<tr>
<td align="center"><a href="http://www.rays-counter.com/">
</a>
</td>
</tr>
```



```
<tr>
<td align="center">





</td><
</tr>
</table>
```

### 3.3. 自己紹介ページ

自己紹介ページではプロフィールを書く部分を table タグで表にすることで、整理されて見やすい画面にすることを心がけた。この表が Web ページ上部リンク部分の表と同じまたは大きいサイズだとバランスが悪いため、作成する際には、width プロパティを用いて、大きさを少し小さめ(60%)にした。また、align プロパティを用いて表が左右に偏ったりしないように設定した。それに加え、見出し部分と内容部分で th タグと td タグを使い分けることで、表示をはっきりさせた。ページの中身が表上の自己紹介のみにしてしまうととても簡素で寂しいイメージになっていたため、画像を挿入した。トップページと同じように img タグを用いるとともに、幅と高さの指定も行い表よりも少し大きいサイズに指定した。自己紹介ページの HTML 文書は次のページの図6のようになる。

```

29 <main>
30 <h2>自己紹介</h2>
31 <section>
32 
33 <table width=60% border="1" align="center">
34 <tr>
35 <th>名前</th><td>Oga</td>
36 </tr>
37 <tr>
38 <th>所属</th><td>某大学経営学部</td>
39 </tr>
40 <tr>
41 <th>年齢</th><td>ミレニアムベイビー</td>
42 </tr>
43 <tr>
44 <th>趣味</th>
45 <td>写真を撮ること</td>
46 </tr>
47 <tr>
48 <th>休日の過ごし方</th><td>ディズニーリゾートへ行くこと</td>
49 </tr>
50 </table>
51 </section>
52 </main>

```

図 6 自己紹介ページの HTML 文書(筆者作成)

### 3.4. 論文・報告書ページ

このページの目的は、学習していく中で書いた論文や報告書などをアップロードし、自由に他人が閲覧やダウンロードをすることができるようにすることである。そのため、装飾はあまり入れずにメインとなるアップロード場所のみを表示するようにした。今後、複数回論文や報告書を書く機会があると考えられることから、アップロード日と題名を軸にした表形式で表示することにした。

アップロードの方法だが、リンクをつけた際と同じく a タグを用いて記述することができる。今回私は、それぞれの報告書ごとに Word 形式と PDF 形式の両方のファイルをアップロードすることとした。これは、閲覧者の環境によっては Word 形式のファイルに対応していない可能性も考えられるからだ。今後アップロードする、論文の容量にもよるかもしれ

ないが、当面は2種類のファイル形式でのアップロードを予定している。ここまでのHTML 文書は以下の図7のようになる。

```
28 <main>
29 <section>
30 <h2>論文・報告書</h2>
31 <table width=60% border="1" align="center">
32 <tr>
33 <th>日付</th>
34 <th>题名(ファイル形式を選択すると表示・ダウンロードされます) </th>
35 </tr>
36 <tr>
37 <td>2020.7.17</td>
38 <td>試験用ファイル<a href="Test.pdf" class="index">(PDF)</a><a href="Test.docx" class="index">(Word)</a></td>
39 </tr>
40 <tr>
41 <td>:</td>
42 <td>今後随時掲載予定</td>
43 </tr>
44 </table>
45 </section>
46 </main>
```

図7 論文・報告書ページのHTML 文書(筆者作成)

### 3.5.プログラミングページ

このページの内容は、論文・報告書ページと極めて類似しているために詳しくは記述しない。具体的なアップロード方法も、現時点ではプログラムの作成も行っていないため未定である。プログラミング作成し次第、こちらのページに何かしらの形でアップロードしていく予定である。HTML 文書の画像も大差ないため割愛する

### 3.6.更新情報ページ

このページでは Web ページの更新情報を掲載していくページである。既に各ページの作成日等を記載しているが、これからの論文やプログラミングのアップロードの際にも適宜更新していく予定のページである。

論文・報告書やプログラミングページと同様に table タグを用いて表の形で表示するように設定している。しかし、前の 2 ページの横幅<table width=60%>と異なり<table width=80%>として横幅を大きめにしている。これは、場合によっては更新内容の部分が長くなってしまふことがあるからである。このページの HTML 文書は次ページの図8のようになる。

```

28 <main>
29 <section>
30 <h2>更新情報</h2>
31 <table width=80% border="1" align="center">
32 <tr>
33 <th>日付</th>
34 <th>更新内容</th>
35 </tr>
36 <tr>
37 <td>2020.5.26</td>
38 <td>「トップ」ページをアップロード</td>
39 </tr>
40 <tr>
41 <td>2020.5.29</td>
42 <td>「更新情報」ページをアップロード</td>
43 </tr>
44 <td>2020.5.30</td>
45 <td>「リンク集」ページをアップロード</td>
46 </tr>
47 <tr>
48 <td>2020.5.31</td>
49 <td>「自己紹介」ページをアップロード</td>
50 </tr>

```

図 8 更新情報ページの HTML 文書(筆者作成)

### 3.7. リンク集ページ

このページでは、自身の Web ページを作るにあたってお世話になった大学やサーバーのリンクを記載した。その際に、a タグを用いてリンクを紐づけることで、リンク集のページから直接それぞれのホームページに飛ぶことができるように設定した。このページも table タグで表形式にすることで、整理された見やすいレイアウトを意識した。横幅は論文・報告書ページと同様に<table width= 60%>で指定をした。このページの HTML 文書は以下の図9のようになる。

```

30 <main>
31 <section>
32 <h2>リンク集</h2>
33 <table width=60% border="1" align="center">
34 <tr>
35 <th colspan="2">関連リンク集(それぞれ選択するとサイトに飛びます) </th>
36 </tr>
37 <tr>
38 <td class="name"><a class="link" href=https://www.toyo.ac.jp/en/ target="_blank"> 東洋大学</a></td>
39 </tr>
40 <tr>
41 <td class="name"><a class="link" href=http://www2.toyo.ac.jp/~asahi/ target="_blank"> 旭ゼミ</a>
42 </td>
43 <tr>
44 <td class="name"><a class="link" href=https://www.ninja.co.jp/ target="_blank"> 忍者ツールズ</a>
45 </td>
46 </tr>
47 </table>
48 </section>
49 </main>

```

図 9 リンク集ページの HTML 文書(筆者作成)

## 4. 自作ホームページのデザイン(CSS)

### 4.1.全ページの共通事項

全ページ共通の CSS ファイルについて説明をしていく。まず、全体として文字や写真の配置は中央揃えで表示をするために、h1,h2,li,ul,main などの様々な要素をまとめて {text-align:center;}と記述した。次に全体の背景色を黒で統一するために body 部分に {background-color:black;}と指定を行った。ページの上部の”Oga’s HomePage”の部分を除いて全ての部分において、デフォルトの 100%の表示かつ、白色での文字の表記にしたため main,footer{color:white;}と記述した。この color を含め、今回の CSS ファイルでは全て色の名前の表記で記述している。カラーコードについては記述しないで進める。

ここまでは、特別細かい指定を行っていない部分について説明をしてきたが、次に少し細かい指定を行っている上部のリンク部分と、ホームページのタイトル部分の記述について説明を行っていく。まず、タイトル部分の説明から行う。タイトル部分は他の部分と異なり、筆記体のフォントを用い、さらに斜体にして、字のサイズを大きくすることで印象付けるようにデザインした。タイトル部分には”title”というクラスをつけているため、次のような記述を行っている。 .title{color:white; font-family:cursive; font-style:italic; font-size:125%;} この際、font-family がフォントの種類を示すプロパティであり、cursive が筆記体を表示するための値となっている。また、font-style が字の表記のままフォントのスタイルを表すプロパティであり、italic が車体を表示する値となっている。

次に上部にあるリンク部分のスタイルの記述について説明をしていく。この部分は通常時のスタイルの表記と、カーソルが重なった状態のスタイルの表記の 2 通りに分かれている。まずは通常状態でのスタイルについて解説を行う。まず、li タグを用いて箇条書きにしているため、箇条書きの先頭の記号が存在してしまう。横並びかつ、表のような形にするにあたって邪魔なので list-style:none;を記述して、記号を表示しないように設定している。また、table タグを用いて区別した部分の上下を黄色い線で囲むことで見やすくしている。そのために、{border-bottom: solid 2px yellow; border-top: solid 2px yellow;}と記述することで上下の線を太さ2px の黄色い線にすることを指定している。それに加え通常時のスタイルの記述にはカーソルが重なった状態のスタイルへの変化の所要時間や、字

の部分だけでなく、ブロック全体としてリンクとして機能するようにする設定が行われている。

まず、スタイルの変化が起こるまでの時間を指定するプロパティは”transition”というものを用いて、`{transition:1s;}`というふうに記述することで、1秒間かけて変化していくことを示している。次にリンク部分の範囲だが、通常何も指定をしない場合、文字のある部分のみがリンクとして機能して、それ以外の部分は反応しなくなっている。しかし `display` プロパティを用いて`{display:block;}`と表記することによって、表のブロック全体としてリンクの役割を果たすようにすることができる。この記述は次に記述する、カーソルが重なっている時に変化する範囲にも影響する部分である。

次にカーソルが重なった時のスタイルの記述について説明する。カーソルが重なった場合のスタイルを記述するにはセレクタの後ろに`:hover` を記述することで可能になる。カーソルが重なった後は字の色を黄色にし、背景を濃い青色に変化させたかった為、`{color:yellow;background-color:mediumblue}`と記述を行った。この記述を行うことでカーソルが重なった時、文字色と背景色に変化するようになる。(図 10 参照)ここまでが全ページに共通して記したスタイルの説明になる。次から各ページのスタイルについて説明する。



図 10 カーソルが重なった時の表示 (筆者作成)

## 4.2. トップページ

トップページ特有のスタイルは、挨拶文の周りを黄色い線で囲んで強調しているところだ。`.main{margin-left:auto;margin-right:auto;width:60%;border:solid 2px;border-color:yellow}`と記述することで、上下左右を太さ2pxの黄色い線で囲むことを指定している。

## 4.3. 自己紹介ページ

このページでは表形式にして自己紹介の部分を書いた為、文字を表の中心に寄せることができるように `td{text-align:center;}`と記述をして、各項目の中央揃えで文字が表示されるようにした。

#### 4.4.論文・報告書ページ

このページも自己紹介ページと同様に表の項目の中心に文字がくるように `td{text-align:center;}` を記述して表を整えた。

#### 4.5.プログラミングページ

論文・報告書ページと完全に同じ CSS ファイルのため省略する。

#### 4.6.更新情報ページ

プログラミングページ同様、論文・報告書ページと同じ CSS ファイルのため省略する。

#### 4.7 リンク集ページ

このページは他のページの表と少し異なったスタイルにしている部分がある。1つは枠線の色、もう1つは表内の文字のリンクに `hover` の場合のスタイルがあることだ。前のページと同様に表内の文字を中央揃えにすることに加え、`table{border-color:yellow;}` を記述することで、表の枠線の色を黄色に変更している。

次に表内の文字が各サイトにリンクしているため、選択しているものがわかりやすいように、ページ上部のリンク同様にカーソルを重ねた際に変化が起きるようにした。表内のリンクした文字に `.name` というクラス名をつけたので以下のように記述している。`.name: hover {color:yellow;background-color:mediumblue; transition:1s; display:block;}` このように記述することで、ページ上部のリンク同様、表内の文字に重ねた時に1秒かけて文字色と洗濯している部分の背景色が変わるようになる。

## 5. 終わりに

今回の自作ホームページ作成の経験を通して様々なことを知ることができた。大きく分けて2つあるが、1つ目は Web サイトを構成するには様々なファイルを組み合わせる必要があることを知ることができた。自分の中で Web ページを作成するというとブログサービスのような、テンプレートがあり、そこに入力するようなものしか知らなかった。しかし実際多くの Web ページは多くのタグ要素やプロパティを記述した HTML ファイルや CSS ファイルの組み合わせによって構成されていた。Web ページというものを作成するのにも大変な過程が必要だということを知ることができた。

知ることができたことの2つ目はデジタル情報の精密さだ。HTML 文書や CSS 文書を作成していく中で、自分が表示したい画面と実際に表示される画面が異なることが幾度となくあった。その際に、「どこの部分が間違っているのか、逆にどこの部分は正しくて変更すべきでないのか」このように考えていた。なぜなら間違っている方は直さなければ正しく表示されないのは勿論のこと、正しいものを変えてしまっはうまく表示されないこともあるからだ。このように全ての記述がそれぞれ関係しあって、一つのページが表示されることを実感し、デジタル情報の繊細さを感じた。

今回の自作ホームページの作成を通して学ぶことのできた、日々関わっているデジタルというものの複雑さ、作成の大変さ。そして、デジタル情報は繊細が故に何かがあった時にうまくいかなくなることもあるということ。この2つはこれからの情報化社会を生きていく上でとても大切なことだと思う。今回学び得たことを無駄にせずこれから生きていくとともに、さらに深く情報システムについて学んでいこうと思う。



## 6.参考文献

千貫りこ (2017) 『デザインの学校 これから始める HTML&CSS の本 [Windows 10 & macOS 対応版]』ロクナナワークショップ監修, 株式会社技術評論社.

小川裕子・加藤善規&できるシリーズ編集部 (2015) 『できるポケット HTML5&CSS3/2.1 全事典』株式会社インプレス

© 2009 - 2020 KiRAMEX CORPORATION 「TechAcademy マガジン」(2020-07-10 最終アクセス)

<https://techacademy.jp/magazine/>