

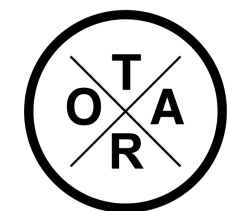
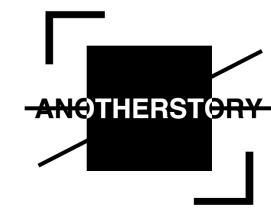
STORYTELLER

STORYTELLER

STORYTELLER

HTML / CSS 基礎講座

STORYTELLER



STORYTELLER

今回取り扱う勉強範囲について

Target



HTML/CSS

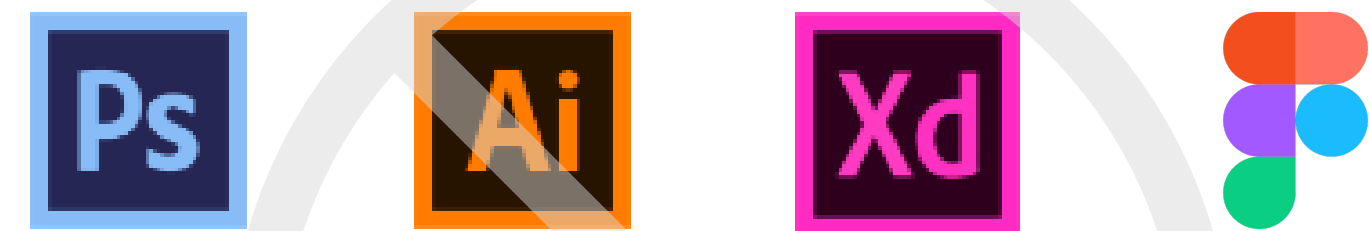
WEBサイトを作成するために必要な、マークアップ言語です。基本的にWEB上のページは、この二つの言語が使われて作られています。慣れてくると楽しいですが、最初の理解が難しいため、今回はこの二つに絞ってレクチャーします。

対象外



WordPress

WEBサイト製作を直感的に行うことができるCMSツールです。テンプレートに沿った基本的なレイアウトのサイトが作れますが、今回のテーマとは異なるため、対象外とします。



デザインツール

写真を加工したり、サイト上に配置するイラスト素材を作ったりできるソフト達ですが、こちらも対象外とします。

HTML / CSS とは？ HTML編

```
<div class="group-1">
  <p>imgタグは、画像を表示するためのタグです。</p>
  
</div>

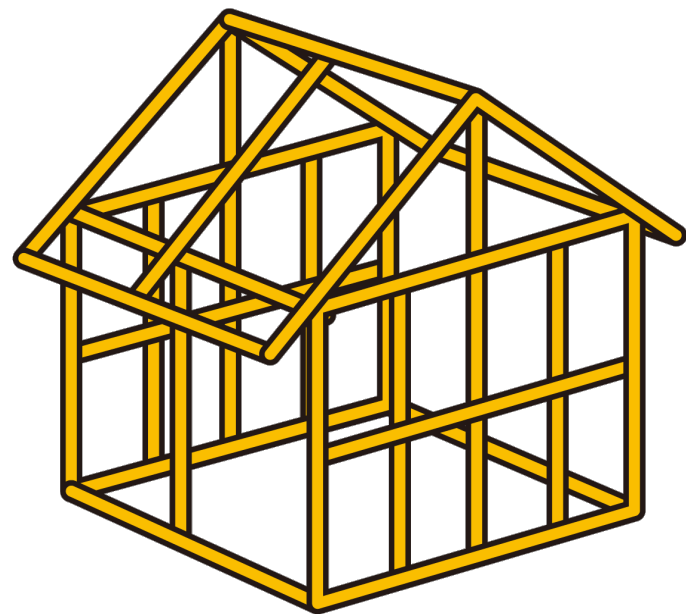
<div class="group-2">
  <p>imgタグは、画像を表示するためのタグです。</p>
  
</div>
```



PCやスマホの画面に、「何が表示されるか」を指示するための言語で、ページに表示したい文字や画像を、<と>で挟まれたタグと呼ばれる、コンピュータへ命令する文字列を書いています。



HTMLのソースコードのみだと、文字や画像がそのままブラウザに表示されます。



HTMLはWEBサイトの骨組みの役割があります。

HTML / CSS とは？ CSS編

```
p {
  background-color: #f00;
  color: #000;
}

img {
  width: 350px;
  height: 400px;
}

.box1 {
  margin-bottom: 50px;
  padding: 50px 20px 50px 20px;
}
```



CSS

HTMLで作ったWEBサイトの骨組みに対して、高さや幅、色やレイアウトなど、見た目のデザインを整えていくのがCSSです。



HTMLファイルにCSSが読み込まれることによって、WEBサイトに装飾がされるようになりました。



CSSは骨組みのCSSにデザインをする役割があります。

静的サイト、WordPress、ノーコードツールの違い

静的サイト



HTMLとCSSのみで作成し、サーバーにアップロードされたものが表示されるサイトのこと。ソースコードが書き換えられない限り、表示は同じままである。

CMSツール



HTML/CSSの書き換えをWeb上で、データベースを通じてボタン操作やテンプレート選択機能からサイトの作成、更新ができるようにしてあるツールのこと。HTMLとCSSを書き換えればデザインのカスタマイズが可能。WordPressが代表例。

ノーコードツール

Nocode

ソースコードを書かなくとも、WEB上で1からサイト製作ができるサービスのこと。CMSと似ているが、あくまでWEBサービスで、HTML/CSSの書き換え等ができず、デザインのカスタマイズが出来ないケースが多い。

HTML



HTMLを理解しよう！

```
<div class="group-1">  
  <p>imgタグは、画像を表示するためのタグです。</p>  
    
</div>  
  
<div class="group-2">  
  <p>imgタグは、画像を表示するためのタグです。</p>  
    
</div>
```

HTMLの書き方 開始タグ、閉じタグ、クラス名を理解しよう！

```
<h1>開始タグと閉じタグで表示したいテキストを囲います。</h1>  
<h2 class="section-title">クラス名は、HTMLタグに付けるあだ名のようなものです</h2>
```

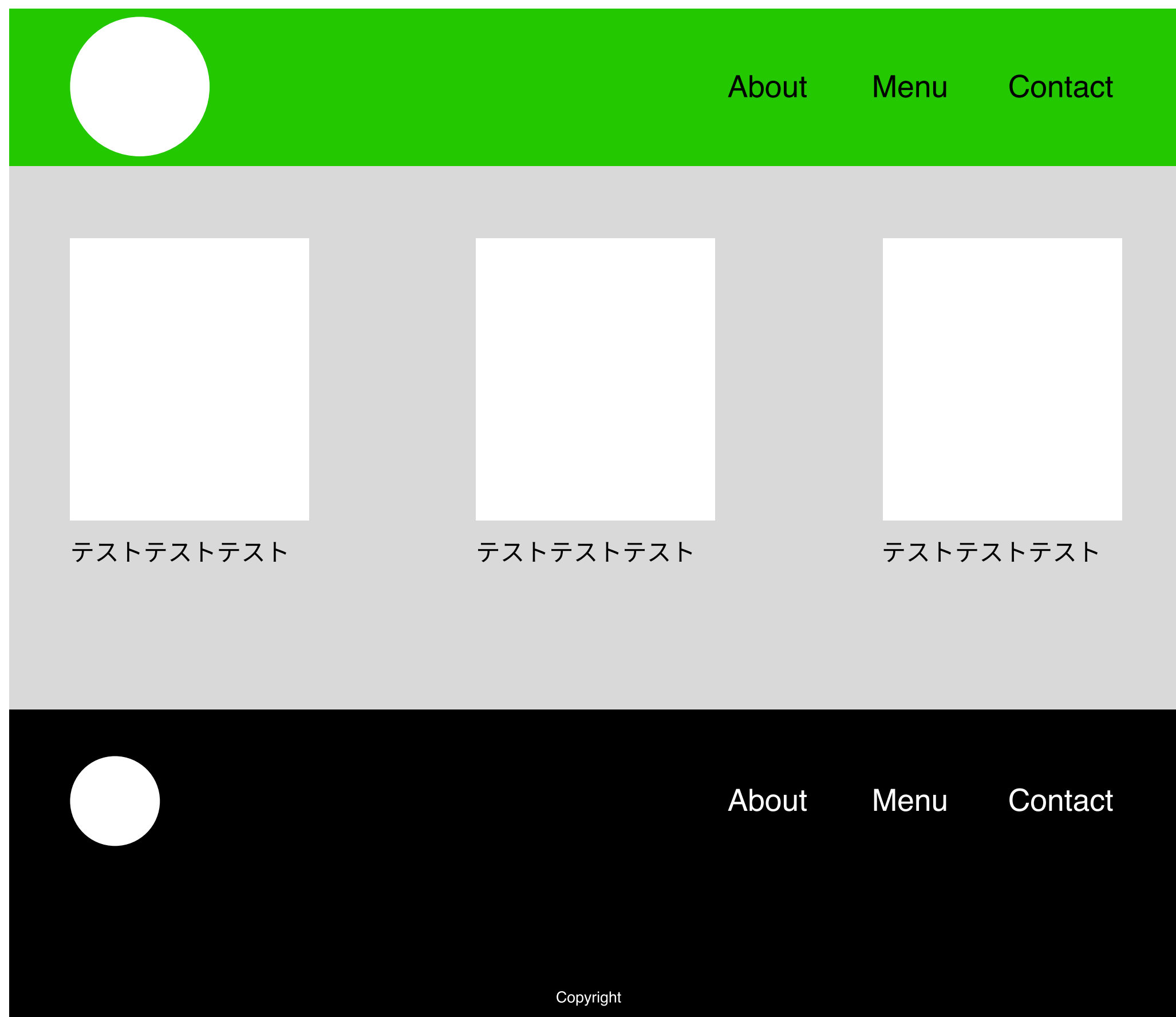
HTMLの書き方

上記が、HTMLの基本の書き方で、テキストの左側に<h1>、右側に</h1>という構成になっています。この記号が”タグ”というもので、左が開始タグ、右が閉じタグです。意味は名前の通りですね。ちなみにh1、h2は”見出しタグ”という名前で、1~6まで存在し、WEBサイトのタイトルやページ内項目のタイトルに使います。

class=”クラス名”

h2の開始タグに、class=”section-title”と書かれていますね。これは、クラス名といい、HTMLタグに付けるあだ名です。なぜ、あだ名をつける必要があるかというと、CSSでデザインする際に、たくさんあるHTMLタグの中から特定のものにだけデザイン指定したい場合にクラス名が役立ちます。

HTMLの書き方 header,main,footerなど、サイト全体を俯瞰しよう！



ヘッダーセクション

<header>と</header>で囲います。サイトのページメニューやロゴを配置します。

メインセクション

<main>と</main>で囲います。サイトのメインのコンテンツ（ほぼヘッダーとフッター以外の内容）はこのメインセクションの中に来ます。

フッターセクション

<footer>と</footer>で囲います。ここにもサイトのメニューやロゴ、それに加えて、会社の住所なども載せることが多いです。


HTMLの書き方 <p></p> で、文字を表示させよう！

pタグとは？

```
<p>pタグは、文字を出力するためのタグです。</p>  
<p>pタグは、文字を出力するためのタグです。</p>
```

pタグとは、英語で「段落」を意味する”paragraph”（パラグラフ）の頭文字を取った、HTMLのタグのことです。

ブラウザ上に文字や数字を出力させる方法は、数えられないほどありますが、pタグが「段落」を意味するように、基本的にはpタグを使って、サイト内の文章を書いていきましょう。



```
pタグは、文字を出力させるためのタグです。  
pタグは、文字を出力させるためのタグです。
```

HTMLの書き方 で、画像を表示させよう！

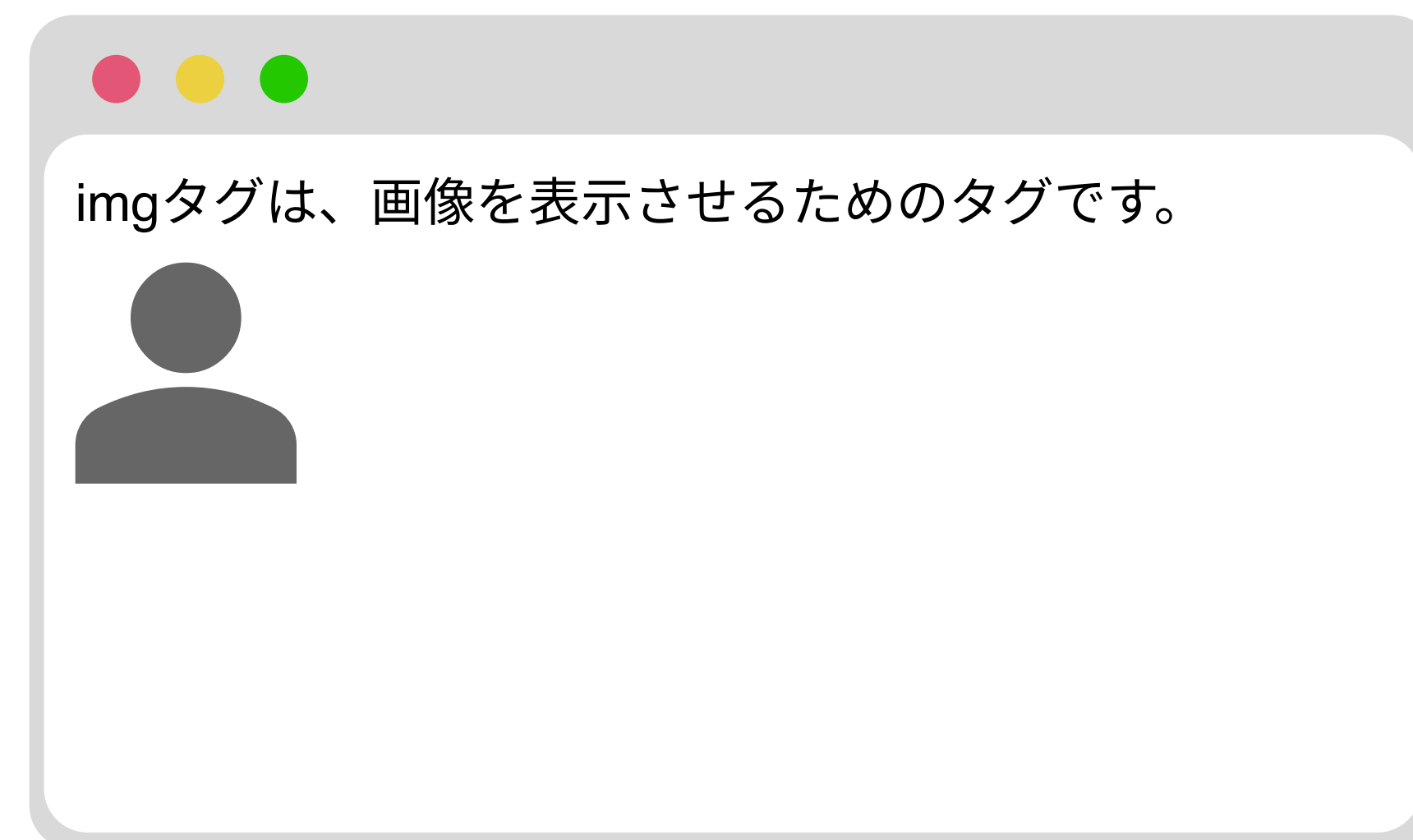
imgタグとは？

```
<p>imgタグは、画像を表示するためのタグです。</p>  

```

imgタグとは、ページ上に画像を表示させるためのタグです。よく見ると、pタグと違い、</p>のような閉じタグがありませんね！imgタグは閉じタグがいない空要素という特徴があります。

また、src、altといった”属性”というものが付いています。どちらもimgタグに必ず付いてくるものですので、属性の存在含めてしっかり理解しましょう！



HTMLの書き方 の属性を理解しよう！

```

```

src属性は、”画像が格納されているファイルから、htmlファイルまでの道筋”を指します。

上記だと、imagesというのが画像が格納されているファイル名で、「/」は、”ファイル階層が変わったよ”ということを表示デバイスに伝えます。

HTMLの書き方 <div></div>で、作ったタグをグループ分けしよう！

divタグとは？

```
<div class="group-1">  
  <p>imgタグは、画像を表示するためのタグです。</p>  
    
</div>
```

divタグは、あらゆるHTMLタグの中で一番登場回数が多くなり、class名とCSSありきの前提で使います。主な役割は、沢山あるHTMLタグをdivタグで囲んであげることにより、グループ化することができます。と、言われてもイメージは難しいと思いますが、このグループの背景の色を変えたい、このグループだけ移動させたい、そんなときにdivが役立ちます。

HTMLの書き方 ブロック要素、インライン要素の違い

今回は、このタグがこういった役割を持ちますという、タグ紹介ではなく、

”全てのHTMLタグは”「ブロック要素」と「インライン要素」の2種類に分けられます

という、”概念”のお話です。最初は難しいですが、避けては通れない道ですのでしっかり理解しましょう！

HTMLの書き方 ブロック要素の深掘り

ブロック要素とは？

コンテンツをまとめるブロック（箱）のような使い方をする要素で、このブロック要素によってWebサイトの大枠を形作られます。ブロック要素のHTMLタグは、記述すると、“画面いっぱいに広がる”という性質を持ちます。

```
<head>
  <style>
    p{
      background-color: #f00;
      color: #000;
      border: 1px solid #000
    }
  </style>
</head>

<body>
  <p>pタグは、文字を出力するためのタグです。</p>
  <p>pタグは、文字を出力するためのタグです。</p>
</body>
```

主なブロック要素

header

main

footer

div

section

ul

li

p

body

dl

dt

dd

article

h1~h6

and more...

pタグは、文字を出力させるためのタグです。
pタグは、文字を出力させるためのタグです。

HTMLの書き方 インライン要素の深掘り

インライン要素とは？

文章などの一部に意味や機能、修飾を付け加えるために用いられるもので、それ自体は段落のように意味のあるまとまりを構成することができません。また、”要素に高さや幅が調整できない”ので、CSSのレイアウト時に注意が必要です。

```
<head>
  <style>
    a{
      background-color: #f00;
      color: #000;
    }
  </style>
</head>

<body>
  <a href="#">aタグはリンクタグ</p>
  <a href="#">aタグはリンクタグ</p>
</body>
```

主なインライン要素

a**img****br****strong****span****small****and few more...****aタグはリンク用****aタグはリンク用**

HTMLの書き方 インデント構造を理解しておこう！

インデントとは？

```
<div class="group-1">  
  <p>imgタグは、画像を表示するためのタグです。</p>  
    
</div>
```

囲っているdivタグに対して、囲われているpタグとimgタグは半角スペース分隙間が空いています。これはインデント構造という文章の先頭を字下げする記法です。こうすることによって、どのタグがどのタグの中に囲われているかがパッと見でわかりやすいですね。

インデント構造をつけなくてもコードは書けますが、後から見返しにくくなったり、記述ミスの原因の一つである、“閉じタグのつけ忘れ”が起きやすくなりますので、適宜綺麗にインデントをつけてあげましょう！

```
<div class="group-1">  
<p>imgタグは、画像を表示するためのタグです。</p>  
  
</div>  
<div class="group-2">  
<p>imgタグは、画像を表示するためのタグです。</p>  
  
</div>
```



```
<div class="group-1">  
  <p>imgタグは、画像を表示するためのタグです。</p>  
    
</div>  
<div class="group-2">  
  <p>imgタグは、画像を表示するためのタグです。</p>  
    
</div>
```


CSS



CSSを理解しよう！

```
p {  
  background-color: #f00;  
  color: #000;  
}  
  
img {  
  width: 350px;  
  height: 400px;  
}  
  
.box1 {  
  margin-bottom: 50px;  
  padding: 50px 20px 50px 20px;  
}
```

CSSの書き方 セレクタ、プロパティ、値を理解しよう

CSSは、HTMLのソースコードがあって初めて役割を果たす言語です。そのため、CSSの文法もHTMLとかなり連動しています。

セレクタ

セレクタは、「どのHTMLタグに命令を出すか」をセレクトする箇所のことです。右図のように、タグ名を書いたり、クラス名に命令を出すときは「. + クラス名」を記載します。

プロパティ

プロパティは、どのような命令の内容を出すかを指示する、いわばCSSの必殺技名のような役割です。必殺技名の後ろには、「:」を必ずつけましょう。

値

値は、CSSの必殺技名であるプロパティを「どのくらいの強さか」「どのくらい長く攻撃するか」などを指定する部分です。値の後ろには必ず「;」をつけましょう。

セレクタ

```
p {  
  background-color: #f00;  
  color: #000;  
}
```

```
.box1 {  
  margin-bottom: 50px;  
  padding: 50px 20px 50px 20px;  
}
```

CSSの書き方 要素に色をつけてみよう！

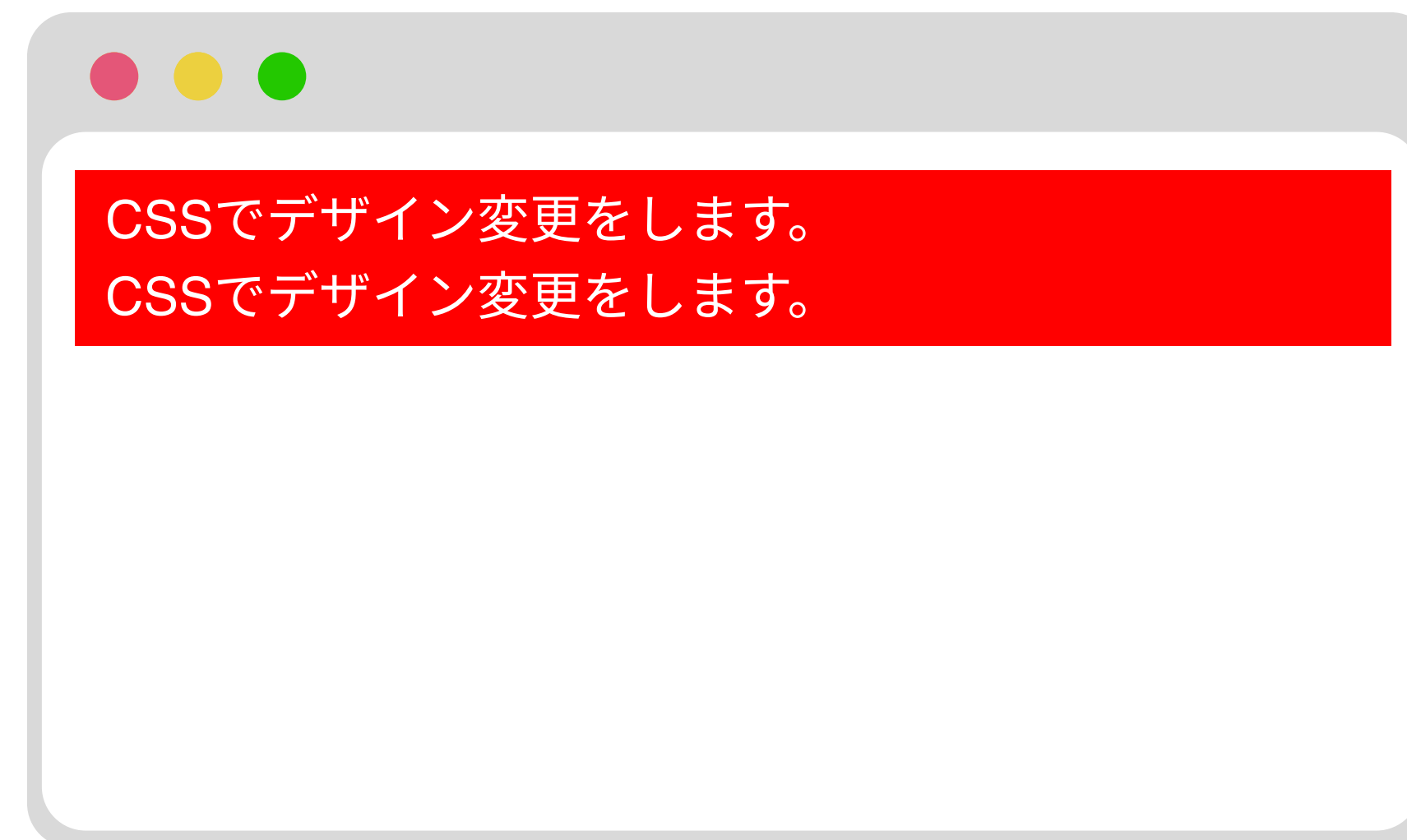
CSSの色変更



```
<div>
  <p class="text"> CSSでデザイン変更をします。 </p>
  <p class="text"> CSSでデザイン変更をします。 </p>
</div>
```



```
.text {
  background-color: red;
  color: white;
}
```



HTML側に、「text」というクラス名のpタグがあり、今回はこのタグに色を変える命令を行います。

色変更には、「文字色」と「背景色」を指定することができ、文字色には単に「color: カラーコード;」背景色には、「background-color: カラーコード;」のように記述します。カラーコードについては別の勉強が必要ですが、原色であればローマ字でも表記可能です。

CSSの書き方 横幅や高さを調整してみよう

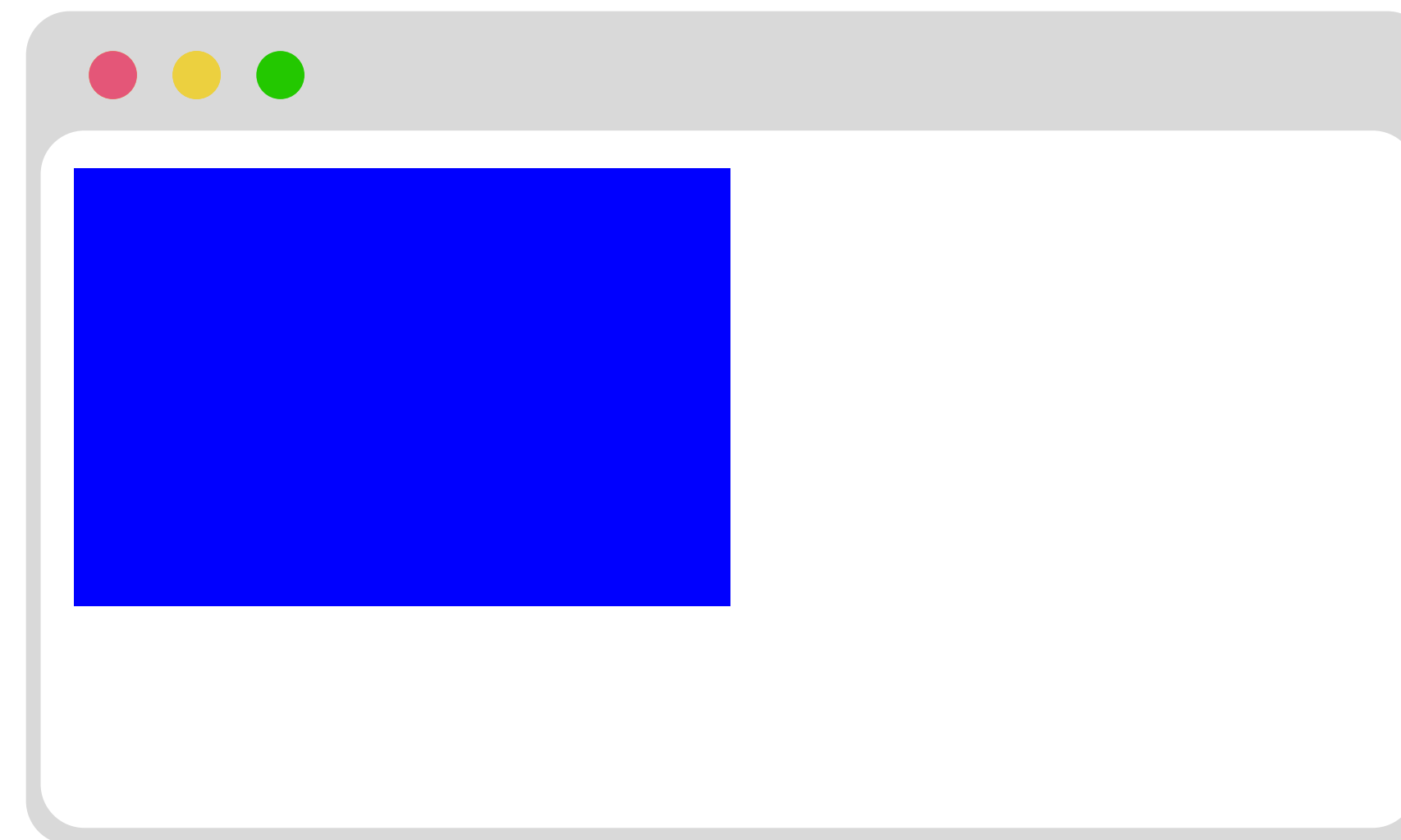
width,height



```
<div>  
<p></p>  
</div>
```



```
p {  
  width: 300px;  
  height: 200px;  
  background-color: blue;  
}
```



幅や高さといった調整も CSSの”プロパティ”部分で行います。横幅の広さを調整したい時には、”width”を、要素の縦の高さを調整したい場合は、”height”を使い、それぞれどのくらいの数値で調整するのかは、CSSの値を使って指定しましょう。ちなみに、インライン要素にはこういった指定は直接できないので注意しましょう。

CSSの書き方 marginとpaddingの違いを理解しよう

margin,padding

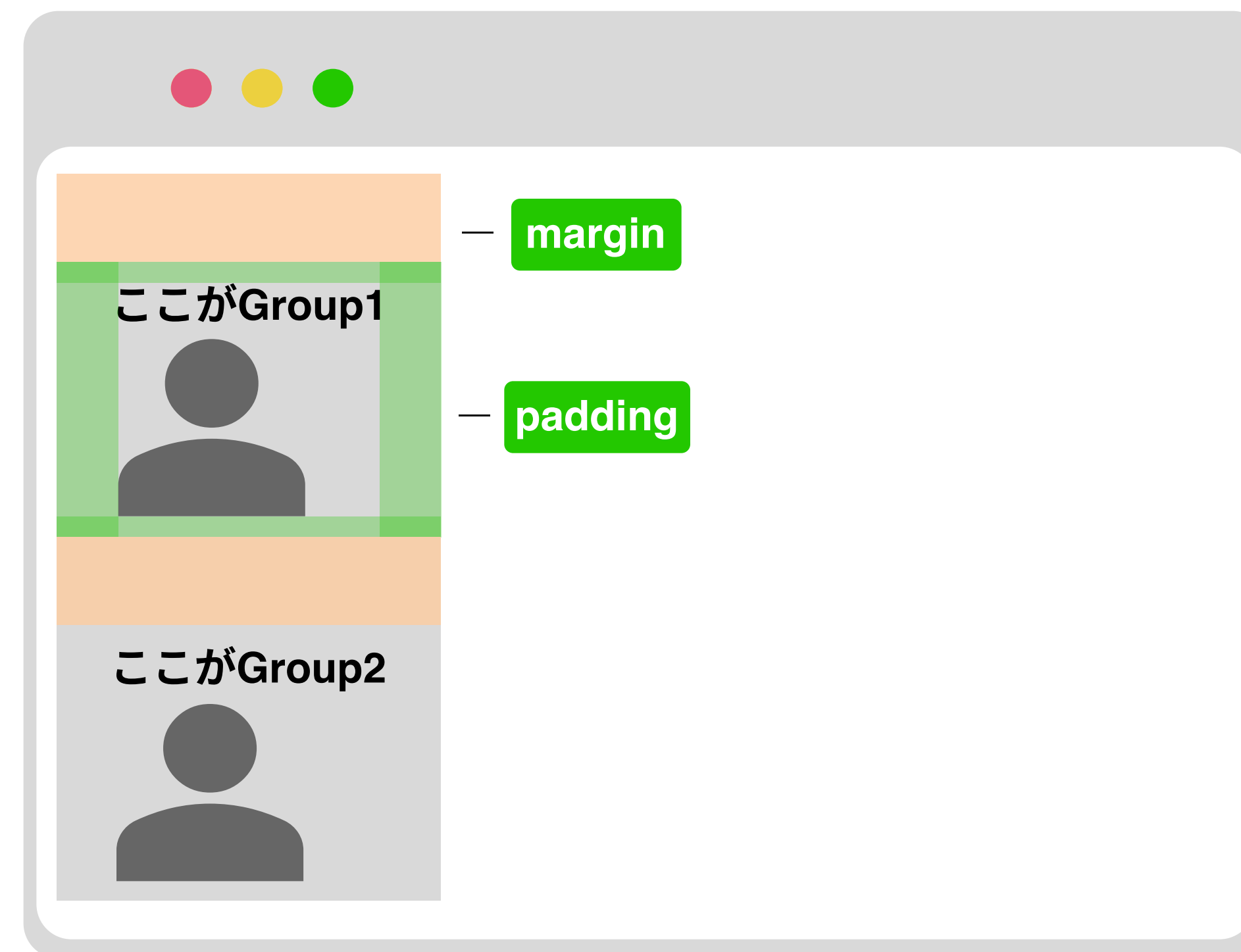


```
<div class="group">
  <p>ここがGroup1</p>
  
</div>

<div class="group">
  <p>ここがGroup2</p>
  
</div>
```



```
.group {
  padding: 10px 30px;
  margin-top: 50px;
  background-color:#D9D9D9;
}
```



marginとpaddingというプロパティは、WEBデザインでのレイアウト調整で使われます。

paddingは、要素の内側の余白を調整し、marginは、要素の外側の余白幅を調整します。

と、言われても最初は使いわけが難しいと思いますので、コードを書いてどんどん慣れていきましょう。

CSSの書き方 Flex-boxレイアウトを理解しよう！



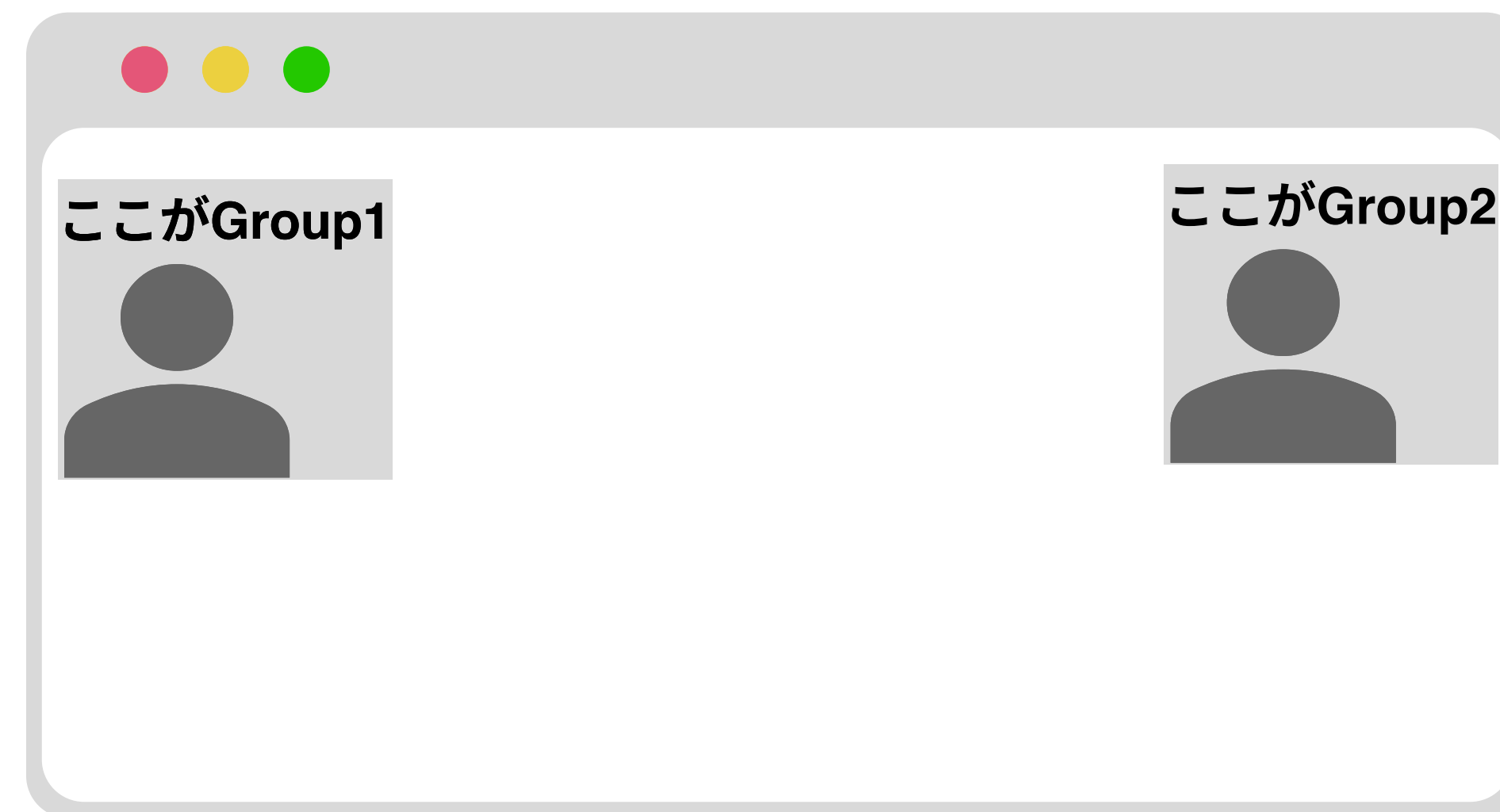
```
<div class="flex-box">
  <div class="group">
    <p>ここがGroup1</p>
    
  </div>
  <div class="group">
    <p>ここがGroup2</p>
    
  </div>
</div>
```

子

親



```
.flex-box {
  display: flex;
  justify-content: space-between;
  background-color: #D9D9D9;
}
```



横並びレイアウトの作成はflex-boxレイアウトという手法を活用しますが、インデントや要素の親子関係の理解が重要になってきます。

HTML側では、flex-boxというクラス名の親要素が、groupという子要素を囲っていますが、flex-boxのプロパティと値に「display: flex;」とありますね。この命令を出すと、「display: flex;」命令がされているセレクタのHTML子要素が複数ある場合、横並びになるのです。

CSSの書き方 Flex-boxとjustify-content 早見Map

justify-contentはdisplay flexで横並びになった子要素たちを「どう横並びさせるか」を制御するプロパティです。justify-contentは値が6種類あり、今回一覧にしました。

center

横並びの要素を中央寄せで配置します。

space-between

子要素を両端揃えで均等に配置します。

space-evenly

子要素が余白含めて完全に均等に並びます。

space-around

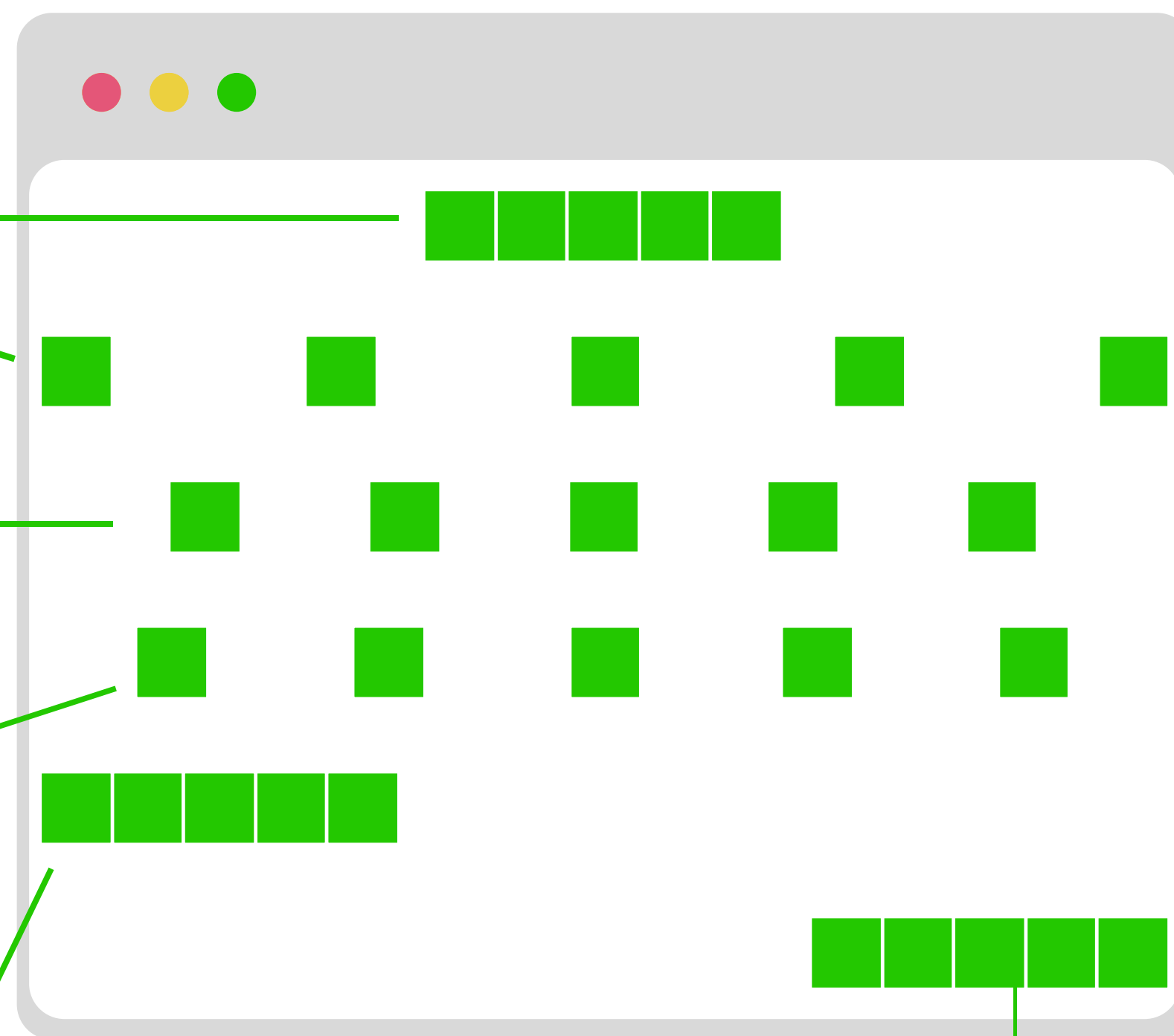
子要素の両側に均等な余白をつけます。

flex-start

全ての子要素が一番左に寄ります。

flex-end

全ての子要素が一番右に寄ります。



```
<div class="flex-box">
  <p class="flex-child"></p>
  <p class="flex-child"></p>
  <p class="flex-child"></p>
  <p class="flex-child"></p>
  <p class="flex-child"></p>
</div>
```



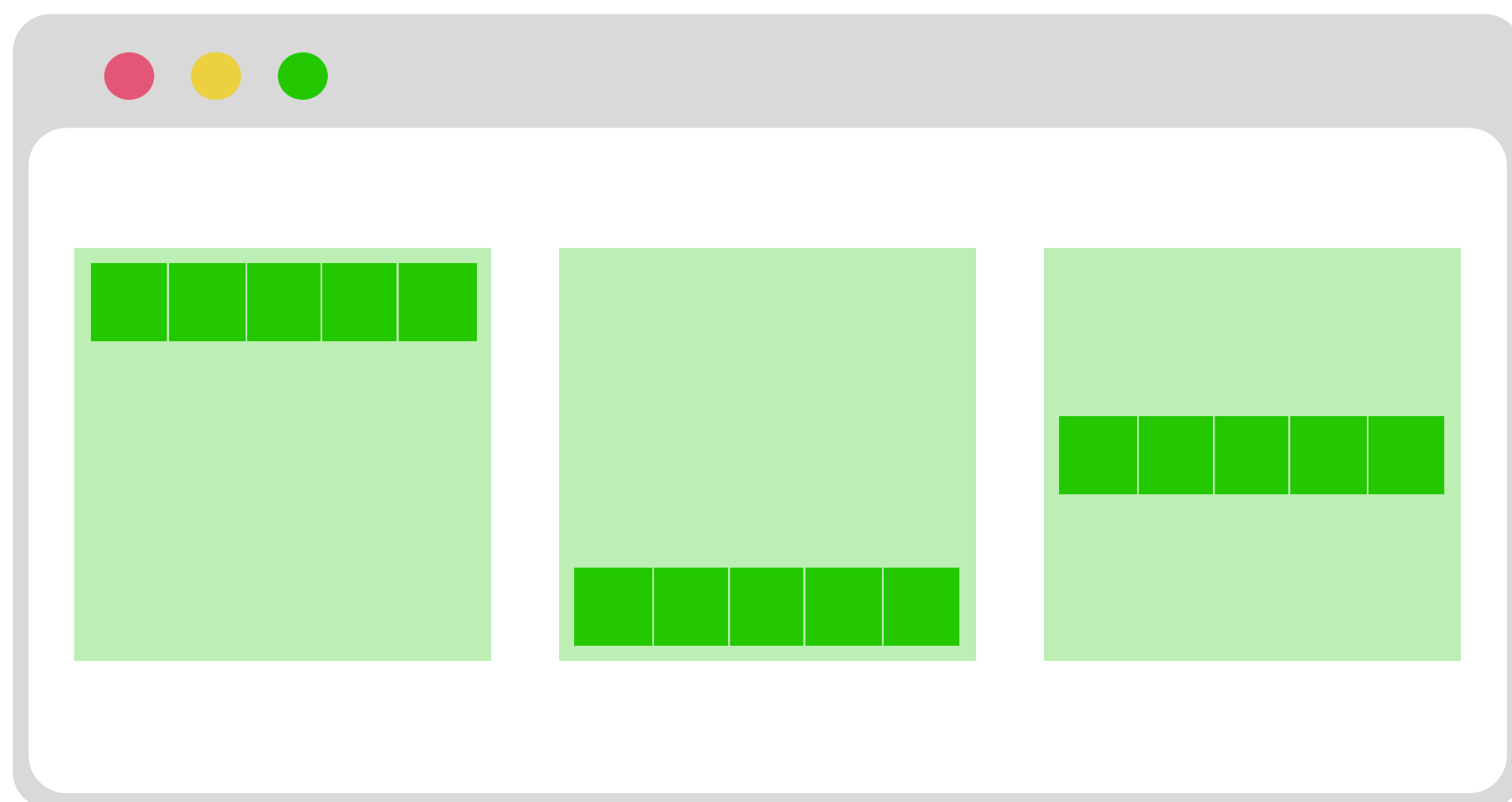
```
.flex-box {
  display: flex;
  justify-content: center;
  align-items: center;
}

.flex-child {
  background-color: #23C800;
  border: #fff;
  width: 20px;
  height: 20px;
}
```

CSSの書き方 Flex-boxとalign-items 早見Map

align-itemsはdisplay flexで横並びになった子要素たちを「垂直方向の整列」を制御するプロパティです。

align-itemsは値が5種類ありますが、3種類覚えておけばいいでしょう。



flex-start

親要素の開始位置から上揃えで配置します。

flex-end

要素を下揃えで配置します。

center

横並びの要素を中央寄せで配置します。



```
<div class="wrapper">
  <div class="flex-box flex-start">
    <p class="flex-child"></p>
    <p class="flex-child"></p>
    <p class="flex-child"></p>
    <p class="flex-child"></p>
    <p class="flex-child"></p>
  </div>

  <div class="flex-box flex-end">
    <p class="flex-child"></p>
    <p class="flex-child"></p>
    <p class="flex-child"></p>
    <p class="flex-child"></p>
    <p class="flex-child"></p>
  </div>

  <div class="flex-box center">
    <p class="flex-child"></p>
    <p class="flex-child"></p>
    <p class="flex-child"></p>
    <p class="flex-child"></p>
    <p class="flex-child"></p>
  </div>
</div>
```



```
.flex-box {
  display: flex;
  justify-content: center;
}

.flex-start {
  align-items: flex-start;
}

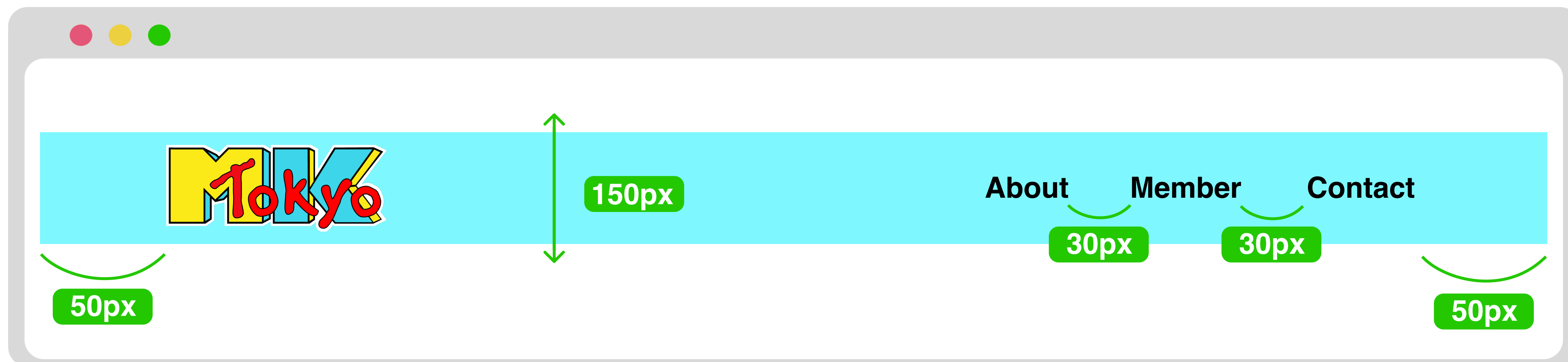
.flex-end {
  align-items: flex-end;
}

.center {
  align-items: center;
}

.flex-child {
  background-color: #23C800;
  border: #fff;
  width: 20px;
  height: 20px;
}

.wrapper {
  background-color: #C2F0B8;
}
```


CSSの書き方 実践編 WEBサイトのヘッダーを作ってみよう



最終課題

今回の最終課題として、WEBページのヘッダーを作ってみましょう。今までHTMLの基本的な書き方とタグの使い分け方、CSSを使った命令方法を習いましたが、上記のようなヘッダーは、今まで習ったことの全てが詰まっています。

作り方は1つではないので、まずは自分の作りやすい作り方で試してみましょう。

ルール

事前に配布したZipファイルを展開し、開いたファイルをVisual Studio Codeで開きます。

開いたファイルのHTML、CSSファイルに記述していきましょう。