

## 4-1-3.HTML・CSS課題

[提出する](#) [評定を受ける](#)

### divパズル

ここでは、実際にテキストエディタを使ってコードを書いてもらいます。  
課題は **divパズル** です。

#### Step1 : HTML・CSSの基本的な記述

今回はサンプルコードを用意しました。それを見ながら学んで行きましょう！

ファイルは「4-1-3」配下の「div\_puzzle」の中になります。

その中にある、 `index.html` 、 `index.css` を使っていきます。

まず `index.html` を開いてみましょう。下記のようになっているかと思います。

```
<!-- ← ここでは、HTML5を使うと宣言-->
<!DOCTYPE html>
<!-- 「lang="ja"」は日本語を使用するという宣言 -->
<html lang="ja">
<!--headエリアに文字コードやHTMLのタイトルを記述します。-->
<head>
    <!-- 文字コードの形式を指定 -->
    <meta charset="UTF-8">
    <!--HTMLのタイトルを記入。ブラウザのタブに表示したい文字を記述-->
    <title>divパズル</title>
    <!-- reset.cssの呼び出し。基本的にはCSSの一番最初に呼び出す。 -->
```

```
<link rel="stylesheet" type="text/css" href=".//css/reset.css">
<!-- ここでindex.cssを呼び出し。このindex.html専用のCSS。 -->
<link rel="stylesheet" type="text/css" href=".//css/index.css">
</head>
<!-- ← いよいよここからHTMLを記載！ -->
<body>
    <!-- class属性の値は任意だが、わかりやすい名前にする。今回は以下。 -->
    <div class="wrapper">
        <div class="header"></div>
        <div class="main">
            <div class="left"></div>
            <div class="right"></div>
        </div>
        <div class="footer"></div>
    </div>
</body>
</html>
```

#### 解説

```
<div class="header"></div>
<div class="main">
    <div class="left"></div>
    <div class="right"></div>
</div>
<div class="footer"></div>
```

まず、HTMLには文章構造というものがあります。

ウェブサイトは、一般的な構造として

**header**

**main****footer**

で成り立っています。

**header**

主にウェブサイトではサイトのロゴが入っていたり、各ページへリンクするメニューが置かれてる部分です。

**footer**

主にウェブサイトでは各種のメニューや、コピーライトなどが書かれている場所です。

**main**

そして真ん中の水色と赤色の2つが並んでいる部分が「main」です。

正式な名称では無いですが、便宜上、class属性の値を「main」としています。

商品を紹介する文章や画像が入っていたり、会社の情報が記載されてたりと、  
そのウェブサイト、ページにおけるメインのコンテンツとなる部分です。

こうした構成から、更に細かに区切って行くことでウェブサイトは成り立っています。

**親と子の関係**

今回は単純に「left」と「right」と名前をつけ、「main」の子要素とします。

親要素、子要素とは以下のようなものです。

```
<div class="main"><!--親要素-->
  <div class="left"></div><!--mainの子-->
  <div class="right"></div><!--mainの子-->
</div>
```

要素内に別の要素を入れ子にする状態を言います。

子要素は「インデント（字下げ）……」をしておくと見返した際にわかりやすいですね。

**wrapper**

そしてこれらとは別に「wrapper（ラッパー）」と呼ばれるものがありますが、  
その名の通り「wrapper」とは包むものを指します。

```
<div class="wrapper">
  <div class="header"></div>
  <div class="main">
    <div class="left"></div>
    <div class="right"></div>
  </div>
  <div class="footer"></div>
</div>
```

こうしておくことで、このページ全てを包む一番大きな要素が生まれたわけです。

このdivパズルであれば、「出来上がったものを画面中央に表示させる」際に、

このクラスにスタイルをあてることで、他に影響なく実現できます。

## **Step2 : CSS**

続いてCSSです。

CSSはHTMLで定義した内容を装飾する役割を持ちます。

テキストエディタで、`index.css`を開いて見ましょう。

現時点では以下のように記述されているだけだと思います。

```
@charset "utf-8";
/*=====
 index用css divパズル Lv.0
 =====*/
```

これだけではHTML側に何の影響もありません。

なので以下のように「header」と「footer」セレクタにスタイルを当てていきます。

```
/* セレクタ */
.header {
    /* プロパティ: 値; */
    width: 400px;
    height: 100px;
    background-color: #141f40;
}
/* セレクタ */
.footer {
    /* プロパティ: 値; */
    width: 400px;
    height: 100px;
    background-color: #d98d30;
}
```

### 補足

ちなみに「別々のセレクタに同じプロパティと値をあてる」場合には以下のように書くことで簡略化することができます。

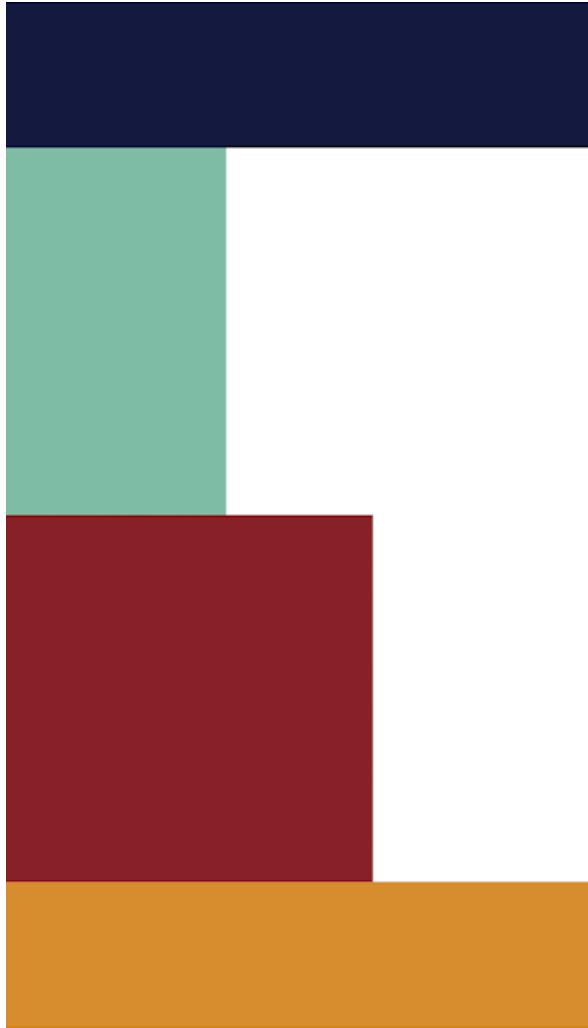
```
.header,
.footer {
    width: 400px;
    height: 100px;
}
.header {
    background-color: #141f40;
}
.footer {
    background-color: #d98d30;
}
```

仕様の変更があった際にも一箇所を直せば、  
一括で複数箇所を同時に治せるというメンテナンス性の高さもあるので覚えておきましょう。

では次に「main」の中の「right」と「left」にスタイルを当てて見ましょう！

```
.right {
    width: 250px;
    height: 250px;
    background-color: #8c2727;
}
.left {
    width: 150px;
    height: 250px;
    background-color: #80bfa8;
}
```

この状態で一度、 `index.html` を開いて見ましょう！



このままだと「main」の要素が縦並びになって画像のようになります。  
このdivパズルではこの「横並び」が重要なポイントになっています。

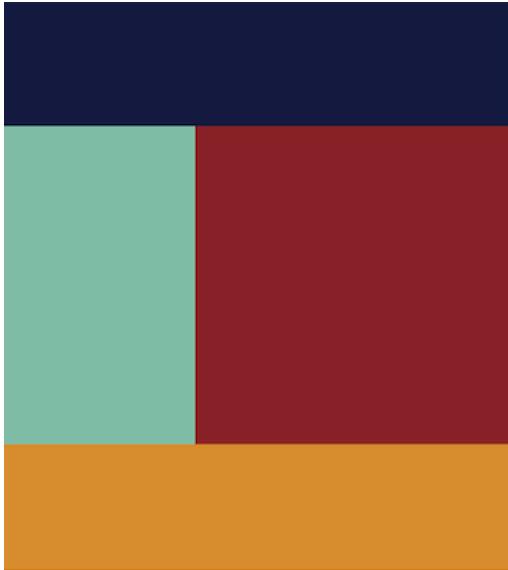
HTMLの要素とは、上から順番に表示されるようになっているため、  
CSSのプロパティを使って、表示する位置を制御する必要があります。

横並びにする方法はいくつかありますが、よく使われるのは **display:flex;** というプロパティです。

**横並びにしたい複数要素の、親要素に対してこちらを適用します。**

```
.main{  
  display: flex;  
}
```

これでもう一度、HTMLを更新して見ましょう。



「main」の中身が横並びになりました。

このようにして縦並びと横並びを駆使してWEBサイトを構築していきます。

最後に「wrapper」にスタイルをあてて、出来上がったものを画面の中央に寄せましょう。

通常のWebサイトは両端に余白があり、要素は中央に寄っていることが多いです。

全体の要素を囲むwrapperに「width」を指定し、「margin: 0 auto」で中央寄せができます。

```
.wrapper {  
    margin: 0 auto;  
    width: 400px;  
}
```

#### 補足 | reset.cssについて

Google ChromeやFireFox、Safari、InternetExplorer（IE）など各ブラウザにはデフォルトで決まっているスタイルがあります。

例えば、Google Chromeには `<li></li>` タグには必ず `margin: 30px;` であり、IEでは `margin: 10px;` というように、

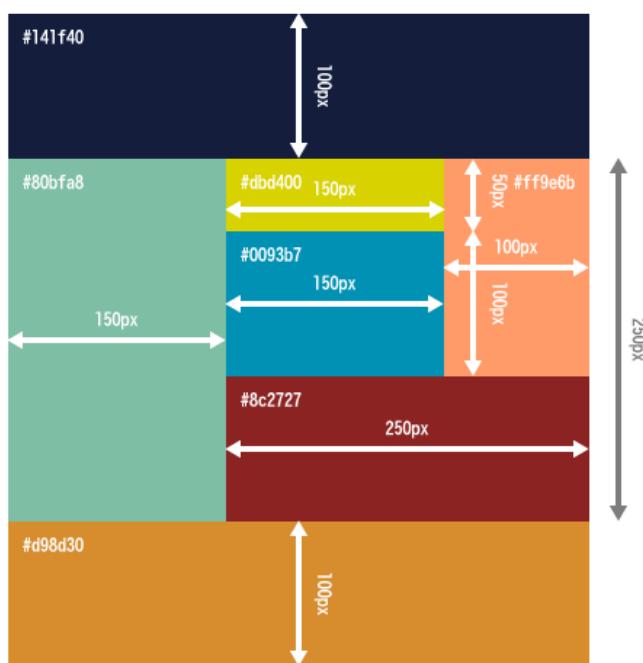
自分で指定していない `margin` や `padding`、`font-size` などが勝手についてしまいデザインがブラウザによって崩れてしまったりするわけです。

`reset.css` ファイル内の記述を見ると `margin: 0;` などブラウザがデフォルトで持つ、スタイルを打ち消す記述がされているので `reset.css` を読み込むことでデザイン崩れを防ぎます。

## 課題 : divパズル

div\_puzzle\_Check フォルダ内に課題がありますのでそちらに記述して下さい。

HTMLファイルとCSSファイルを編集して、下記の表示になるようにしましょう。



## 評定概要

学生から秘匿	No
参加者	65
提出	48
要評定	2