

# プログラミング

第3回

Webページの見栄えを整える (css)

久保田 匠

# [準備]授業資料にアクセス

- 久保田の授業ホームページに資料がアップロードされている。
- まずは「愛教大 数学」と検索してみよう。



The screenshot shows the homepage of Kuroda's teaching website. It includes two tables for the academic year: '授業用ホームページ (久保田)' (2025年度前期担当科目) and '授業用ホームページ (久保田)' (2025年度後期担当科目). In the second table, under the 4年ゼミ column, the word 'プログラミング' is highlighted with a red box. At the bottom, there is a link to other content and contact information: '数学教育講座 久保田匠 (自然科学棟 521 研究室) Email: skubota [at] auecc.aichi-edu.ac.jp'.

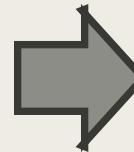
	内容	資料	コード
第1回	いろいろなプログラミング言語 VSCode のインストール	● <a href="#">Prog_01-1</a>	
第2回	Webページを構築する(HTML)	● <a href="#">Prog_02-1</a>	
第3回	Webページの見栄えを整える(CSS)	● <a href="#">Prog_03-1</a> ● <a href="#">Prog_03-2</a>	
第4回	JavaScriptに触れてみよう		<a href="#">Prog_04-1</a>
第5回	変数と演算		(なし)
第6回	条件文		(なし)
第7回	繰り返し(1)		(なし)
第8回	繰り返し(2)		<a href="#">Prog_08-1</a>
第9回	繰り返し(3)		(なし)

# [準備] コードの新規作成①

- 授業用ホームページからサンプルコードをコピーしよう。

## プログラミング

	内容	資料	コード
第1回	いろいろなプログラミング言語 VSCode のインストール	●	<a href="#">Prog_01-1</a>
第2回	Webページを構築する(HTML)	●	<a href="#">Prog_02-1</a>
第3回	Webページの見栄えを整える(CSS)	●	<a href="#">Prog_03-1</a>
第4回	JavaScriptに触れてみよう		<a href="#">Prog_04-1</a>
第5回	変数と演算		(なし)
第6回	条件文		(なし)
第7回	繰り返し(1)		(なし)
第8回	繰り返し(2)		<a href="#">Prog_08-1</a>
第9回	繰り返し(3)		(なし)



## Prog\_03-1

```
<!DOCTYPE html>
<html>

<head>
  <meta charset="UTF-8">
  <title>Prog_03-1</title>
  <!-- 今日はここも使います -->
</head>

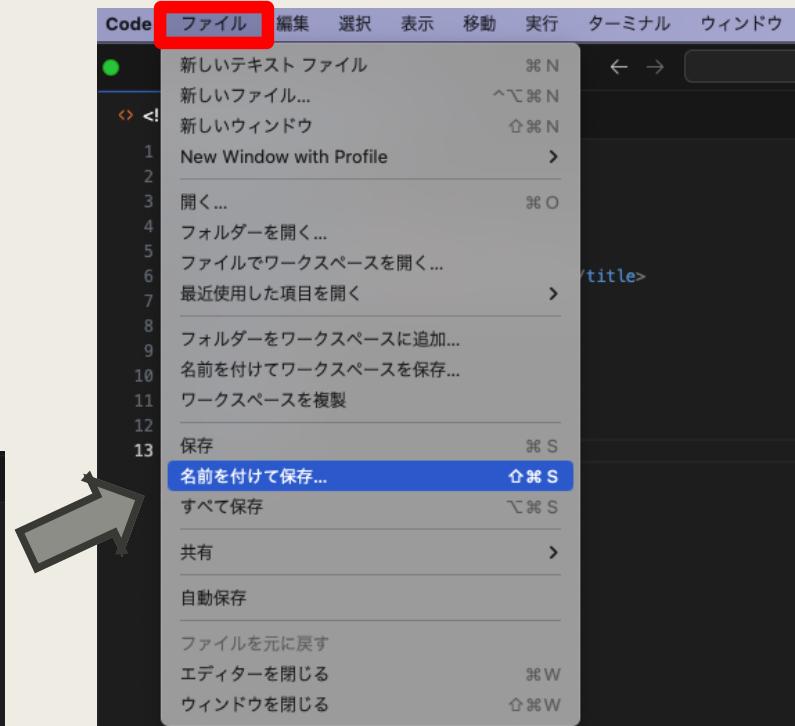
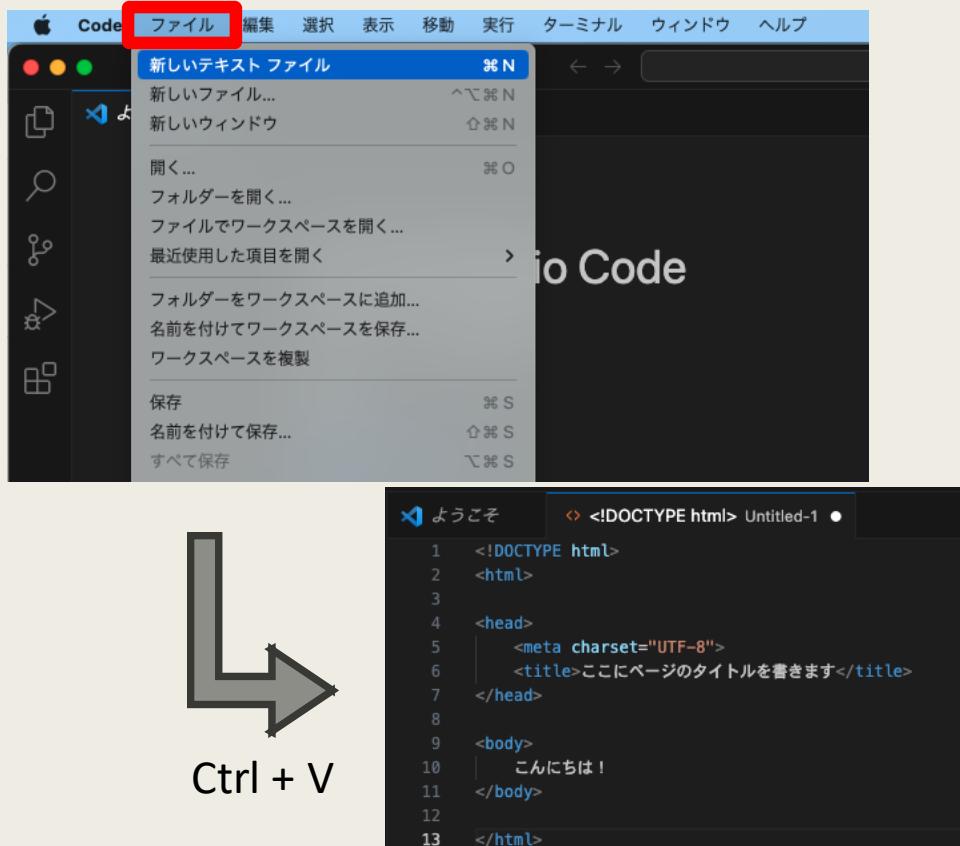
<body>
  ここに今日の授業内容を入力します。
</body>

</html>
```

コピー

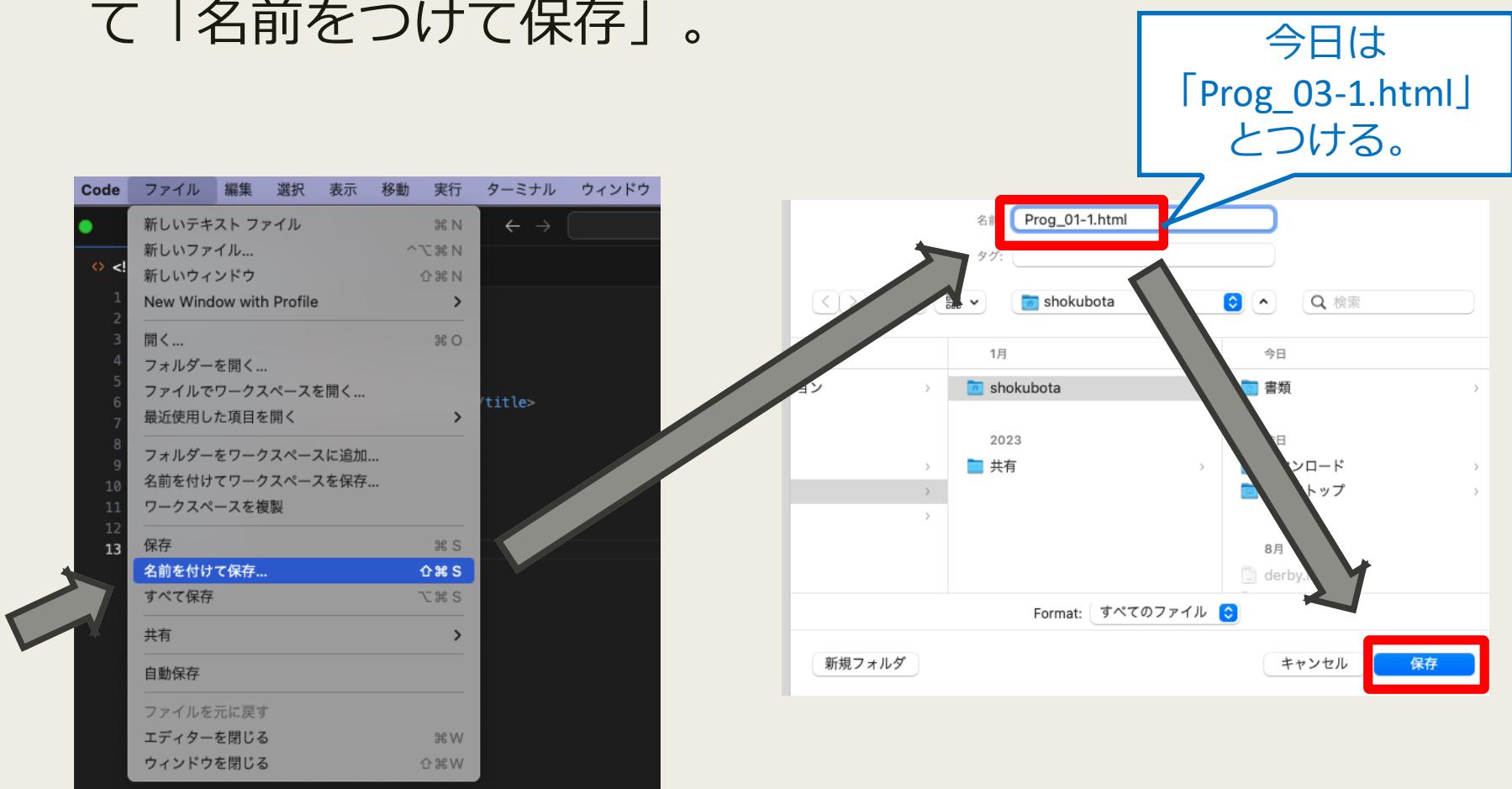
# [準備] コードの新規作成②

- VSCode を起動し「ファイル」から「新しいテキストファイル」を選択。
- その後、さきほどコピーした文書をペースト（Ctrl + V）して「名前をつけて保存」。



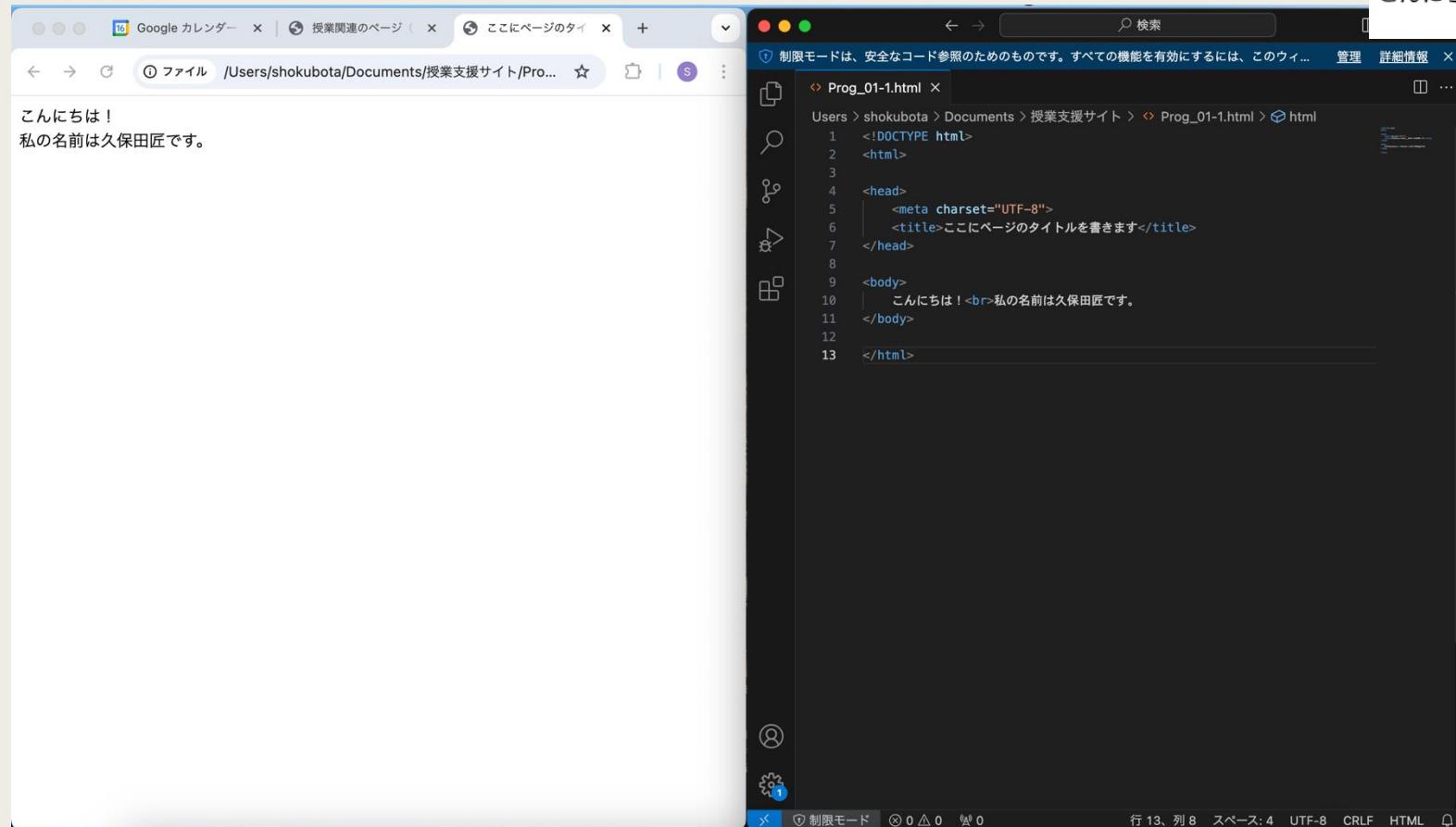
# [準備] コードの新規作成②

- VSCode を起動し「ファイル」から「新しいテキストファイル」を選択。
- その後、さきほどコピーした文書をペースト (Ctrl + V) して「名前をつけて保存」。



# [準備]作業環境を整える

- 保存したhtmlファイルをダブルクリックして開いておく。
- PCの画面をふたつに分け、片方はブラウザ、もう片方はvsCodeを開いておくと便利。

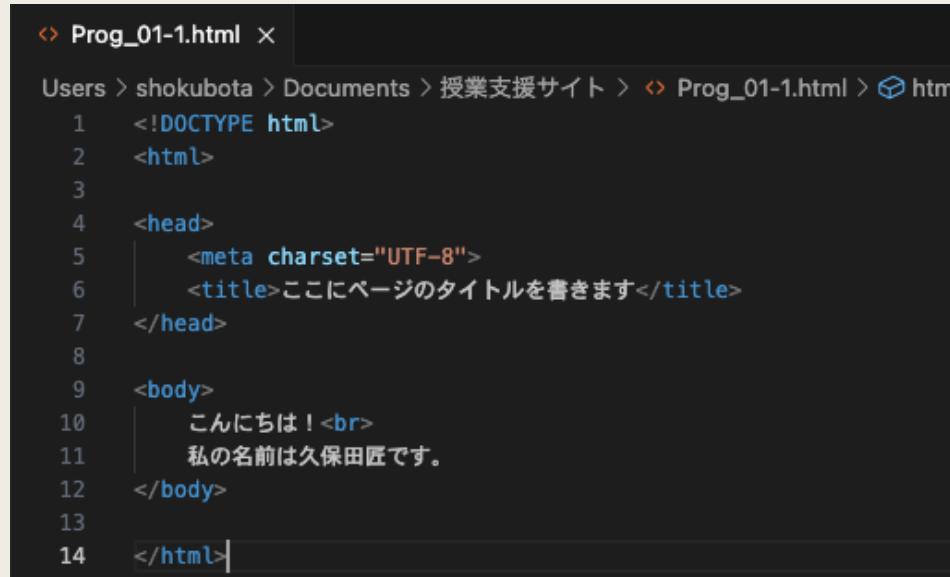


```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="UTF-8">
6   <title>ここにページのタイトルを書きます</title>
7 </head>
8
9 <body>
10  こんにちは！<br>私の名前は久保田匠です。
11 </body>
12
13 </html>
```



# [復習]HTML

- JavaScript は Web ページで動くプログラミング言語。
- JavaScript を使うためにはまず Web ページを構築する必要がある。
- Web ページを構築するために HTML を用いる。
- HTML は Web ページの土台を作るためのツール。
- 前回は HTML の構造を学んだ。



The screenshot shows a code editor window with the title "Prog\_01-1.html". The file path is "Users > shokubota > Documents > 授業支援サイト > Prog\_01-1.html". The code itself is:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>ここにページのタイトルを書きます</title>
</head>
<body>
    こんにちは！<br>
    私の名前は久保田匠です。
</body>
</html>
```

# [復習]タグ

- html ファイルは タグ と呼ばれる記号で構成されている。
- 多くのタグには 開始タグ と 終了タグ があり、その間に中身を記述する。
- 開始タグは「<」と「>」で囲み、終了タグは「</」と「>」で囲む。
- タグの書き方は <要素名> 要素の内容 </要素名>

開始タグ  
↓

終了タグ  
↓

<p>こんにちは！</p>

↑  
中身

```
↳ Prog_01-1.html ×

Users > shokubota > Documents > 授業支援サイト > ↳ Prog_01-1.html > ⏺ html

1  <!DOCTYPE html>
2  <html>
3
4  <head>
5      <meta charset="UTF-8">
6      <title>ここにページのタイトルを書きます</title>
7  </head>
8
9  <body>
10     こんにちは！<br>
11     私の名前は久保田匠です。
12  </body>
13
14 </html>
```

# [復習]htmlファイルの構造

- 一番外側は <html> \*\*\*\*\* </html>
- このタグは、htmlファイルがどこからどこまで書かれているかを示す。
- <html>タグの内側に <head>タグと <body>タグが配置される。
- <head>タグにはページの見栄えや動作に関する設定を書く。
- <body>タグには実際に画面に表示させる内容を書く。

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>ここにページのタイトルを書きます</title>
</head>
<body>
こんにちは！<br>
私の名前は久保田匠です。
</body>
</html>
```

裏方の仕事を担当

# [復習] 基本的なタグとその用途

- <h1>: 見出し。heading が由来。<h2>, <h3> なども同様。  
数字が小さいほど見出しのレベルが上がる。
- <p>: 段落。paragraph が由来。
- <span>: テキストの一部。  
文章の一部の色を変えたいときなどに使う。

■ <div>: グループ化。document division が由来。

■ <a>: 文書間のリンクを記述。

■ <img>: 画像。

この3つは覚えて

■ 必要なときに「調べて使える」ようにしておけば問題ない。

# Webページの見栄えを整える

- HTML は Web ページの構造を記述する。
- しかし、見栄えの点では、物足りない部分や使い勝手の悪い点が少なくない。
- CSS は Web ページの見た目（色や配置、余白など）を自由に整えるための仕組み。
  - CSS は Cascading Style Sheets の略。

全部中央揃えにしたい

```
9  <body>
10   <table border="1">
11   <tr>
12     <td>□</td>
13     <td><center>大人</center></td>
14     <td><center>子ども</center></td>
15   </tr>
16   <tr>
17     <td><center>入場料</center></td>
18     <td><center>2000円</center></td>
19     <td><center>1000円</center></td>
20   </tr>
21   <tr>
22     <td><center>限定アトラクション</center></td>
23     <td><center>500円</center></td>
24     <td><center>300円</center></td>
25   </tr>
26 </body>
```

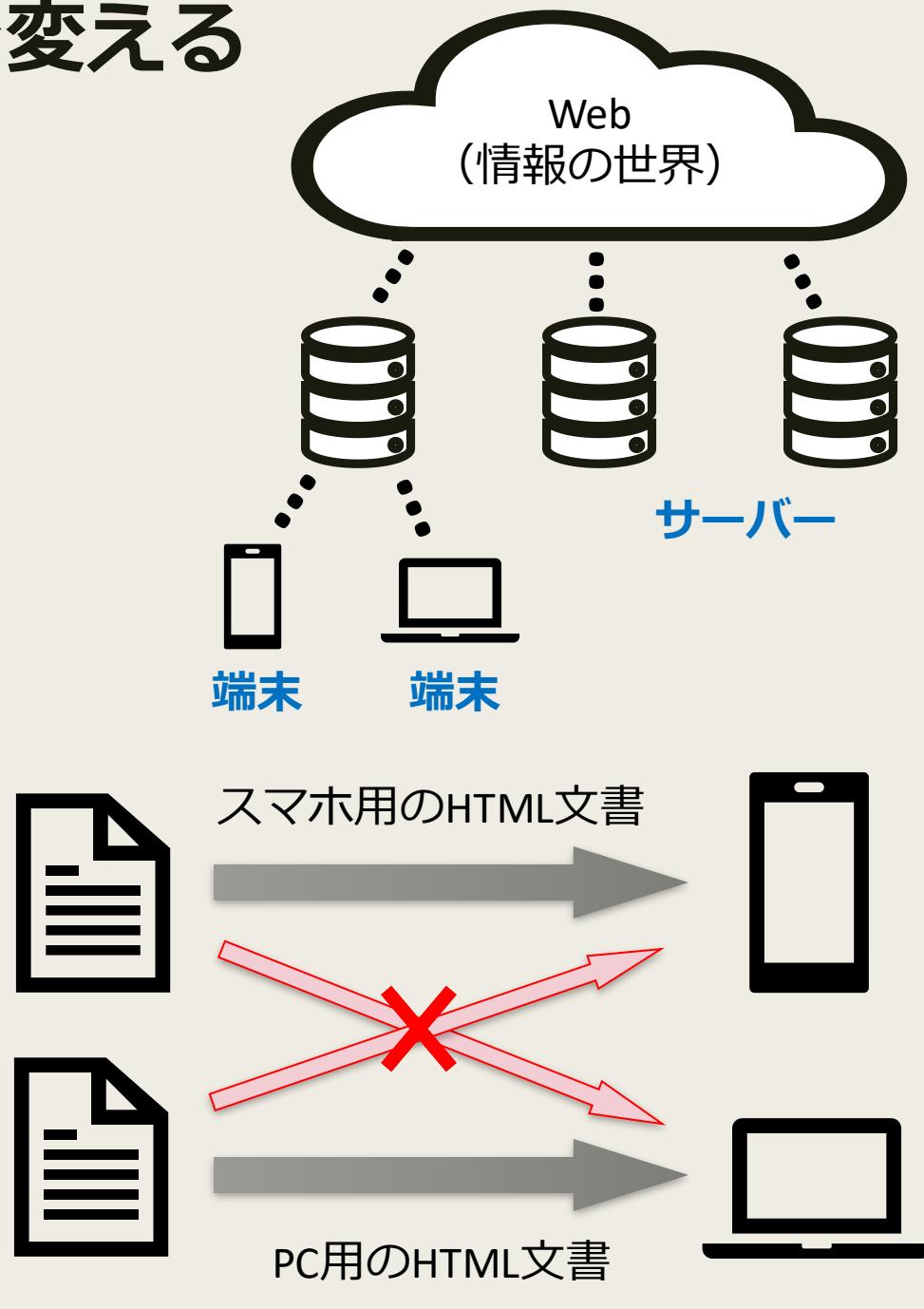
	大人	子ども
入場料	2000円	1000円
限定アトラクション	500円	300円

めんどくさい('A')

```
1  <!DOCTYPE html>
2  <html>
3
4  <head>
5    <meta charset="UTF-8">
6    <title>Prog 03-0</title>
7    <style>
8      td {
9        text-align: center;
10      }
11    </style>
12  </head>
13
14  <body>
15    <table border="1">
16      <tr>
17        <td>□</td>
18        <td>大人</td>
19        <td>子ども</td>
20      </tr>
21      <tr>
22        <td>入場料</td>
23        <td>2000円</td>
24        <td>1000円</td>
25      </tr>
26      <tr>
27        <td>限定アトラクション</td>
28        <td>500円</td>
29        <td>300円</td>
30      </tr>
31    </table>
32  </body>
33 </html>
```

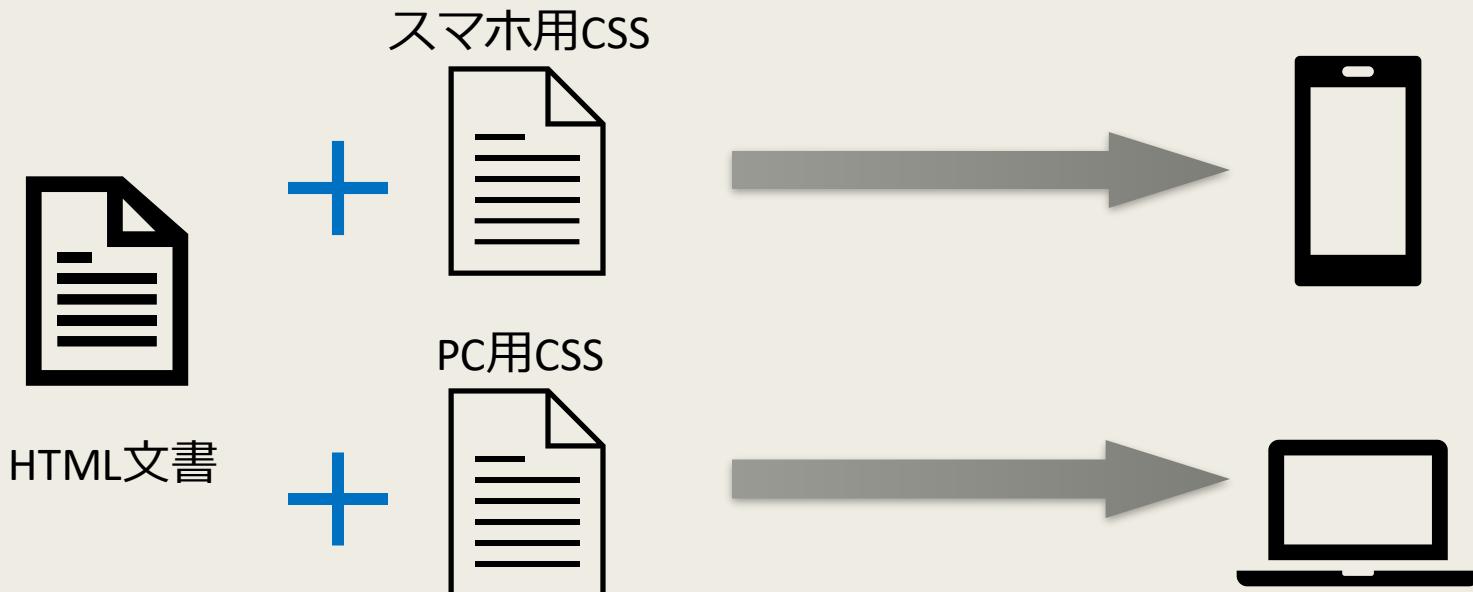
# 端末によって見せ方を変える

- 端末とはパソコンやスマートフォンのこと。
- パソコン用のページは大きな画面を想定して作成され、レイアウト（配置）の自由度も高い。
- 一方、スマートフォンの場合は、PCよりも画面が小さいものの、縦に長いページ構成であればユーザーにとっての負担は少ない。
- 端末によって見やすいページ構成は違う！
- 端末によってWebページの見せ方を変えたい場合、HTMLのタグのみで見栄えを整えるのは難しい。



# 文章の構造と見栄えの構造を切り離す

- もし文章の構造（章、段落、見出しなど）と、見栄え（色、文字の大きさなど）を分けて管理できれば、PC用かスマホ用かに応じて「見栄え」だけを調整することで同じ内容を適切に表示できる。
- この仕組みを可能にするのがcssである。
- 実際には、文章の構造をHTMLで、見た目やデザインをCSSでそれぞれ指定する。



# CSSを使ってみよう

- 最も簡単な方法は、HTMLのタグ内で **style 属性**を使う方法。
- ただし、このやり方は HTML のみで見栄えを整えるやり方とさほど変わらない。
- 次のコードを入力してみよう。

```
↳ Prog_03-2.html ×  
Users > shokubota > Documents > 授業支援サイト > ↳ Prog_03-2.html > html  
1  <!DOCTYPE html>  
2  <html>  
3  
4  <head>  
5    <meta charset="UTF-8">  
6    <title>Prog_03-2</title>  
7  </head>  
8  
9  <body style="text-align: center;">  
10   <p style="color: red">赤色のテキスト</p>  
11   <p style="color: green; font-size: 24px;">緑色、 24px</p>  
12   <p>  
13     テキストの一部にスタイルを適用する場合は、  
14     <span style="text-decoration: underline">span要素</span>を使う。  
15   </p>  
16 </body>  
17  
18 </html>
```

style 属性

- <タグ名 **style= “ \*\* ”**> で使うのが基本。
- <body>タグで使うと 文章全体に適用される。
- 11行目のように複数 指定した場合はセミ コロン（;）をつける。

# プロパティ

- <タグ名 style=“\*\*”> の \*\* の部分をプロパティ という。
- 主なプロパティを列挙するが覚える必要はなく「たしかこんな指定ができたなあ」とうっすら把握しておけばよい。
- 大事なことは「調べれば使える」という状態にしておくこと。

プロパティ名	用途	使用例
background-color	背景色の指定	background-color: blue;
font-size	字の大きさを指定。px, pt などの単位がある。small なども可	font-size: 12px; font-size: small;
opacity	半透明の度合いを0~1の範囲で指定	opacity: 0.6;
text-align	left(左寄せ)、right(右寄せ)など、文字の位置を指定	text-align: center;
text-decoration	underline(下線)、underline dotted(点線下線)など、テキストの装飾敵な線を表示	text-decoration: underline;

今日の内容はここまで理解できていればOK

# セレクタ

- 先ほどまでの方法では各タグごとに文字色や文字の大きさを指定していたが、これらを一括で指定する方法がある。
- セレクタは、どのタグにどのプロパティを適用するかを指定するパターンやルールを指す。
- HTML 文書の head 部において、`<style>`タグを使って次のように宣言する（例は次のスライドで）。

```
<head>
  <style>
    セレクタ{
      CSSプロパティ名: 値;
    }
  </style>
</head>

<body>
  <!-- ここに文章が書かれる -->
</body>
```

← h1 など  
← text-align: center; など

h1タグを使ったら  
text-align: center  
を適用してください、  
という意味になる

# セレクタの使用例

```
Prog_03-3.html
Users > shokubota > Documents > 授業支援サイト > Prog_03-3.html > html > body
1  <!DOCTYPE html>
2  <html>
3
4  <head>
5      <meta charset="UTF-8">
6      <title>Prog 03-3</title>
7      <style>
8          h1 {
9              text-align: center;
10         }
11         h2 {
12             text-decoration: underline;
13             color: blue;
14         }
15         p {
16             color: gray;
17             font-size: 14px;
18         }
19     </style>
20 </head>
21
22 <body>
23     <h1>ドラえもんの登場人物</h1>
24     <h2>ドラえもん</h2>
25     <p>
26         | 22世紀から来た猫型ロボット。9月3日生まれ。
27     </p>
28     <h2>野比のび太</h2>
29     <p>
30         | 勉強も運動も苦手なうえ、それらを克服する努力を怠るなど駄目な人間の典型とされるが、
31         実際は「やればできる」タイプの人間。8月7日生まれ。
32     </p>
33 </body>
34
35 </html>
```

- 7行目~19行目のように書くと、body 部で<h1>, <h2>, <p>タグを使用したら一括で指定したプロパティが適用される。
- あとから<p>タグ内のテキストをグレーから黄色に変えたくなったら16行目の color: gray を color: yellow に変えるだけでよい。
- 左のコードを実際に入力してみよう。まずは body 部から入力するといいかも。

前回取り組んだ「時間割の作成」でも、あらかじめ td タグに対するルールを指定しておくと一括で表内のセルをを中心揃えにできる。

# 全称セレクタなど

他の設定まで上書きするので  
意図しない動作を起こすことも...

- \*{ ... } を全称セレクタと言う。良くも悪くもすべての文章に適用される。使用例は右下の赤色で囲まれた部分。
- マージンをうまく設定することで PC とスマホ両方で読みやすいWebページになる。
  - マージンとは文書の余白のことである。
- [発展] 特定のタグではなく、自分で宣言した「クラス」に対して同じスタイルを一括で設定することもできる。
  - 興味のある人や余力のある人は クラスセレクタ で検索してみよう。
- 先のファイルに赤色で囲まれたコードを追加してみよう。

```
4 <head>
5   <meta charset="UTF-8">
6   <title>Prog_03-3</title>
7   <style>
8     *{
9       width: 700px;
10      margin-left: auto;
11      margin-right: auto;
12    }
13
14   h1 {
15     text-align: center;
16   }
17   h2 {
18     text-decoration: underline;
19     color: blue;
20   }
21   p {
22     color: gray;
23     font-size: 14px;
24   }
25 </style>
26 </head>
```

# ordered list で(1)(2)... と出力したい

- 前回、番号付きリストのための <ol> タグを紹介した。
- デフォルト（特別な設定を行っていない状態）では左下のように 1. 2. 3. ... と出力される。

## ジャーマンポテト

### 材料

- じゃがいも 2個
- 玉ねぎ 1個
- ペーコン 200g

### 作り方

1. じゃがいもと玉ねぎを薄切りにします
2. ペーコン、じゃがいも、玉ねぎの順で炒めます
3. 塩コショウで味を整えます

## 行列の積

$$A = \begin{bmatrix} -2 & -2 & -1 \\ 0 & 3 & 0 \end{bmatrix}, B = \begin{bmatrix} -3 & -5 \\ -5 & 4 \\ 1 & 4 \end{bmatrix} \text{ とする。}$$

(1) 積  $AB$  は

答え

(2) 積  $BA$  は

答え

- 一方、数学の問題では番号付きリスト（小問番号）は右上のように (1)(2) ... となっていることが多い。
- このような出力をするためにはどのようにすればよいか？

# 自分の頭で考える or 調べてパクる

- <ol>タグに一括で適用されるセレクタを自作すればよいが…
- 番号付きリストの出力 1. 2. 3. … を (1)(2)(3) … に変更するのは実はちょっと難しい。
- 今回のようなケースは「調べてコピペする」方がはやい。
- 初心者の段階では「自分の頭で考える」にこだわり過ぎると進まなくなるので、本質的に重要な箇所は既存のコードをネットで調べたり生成AIなどのツールを活用して補うといい。

## 教科書や人（先生）から学ぶ領域

- ・特に基礎的な知識
- ・よく整理された枠組み

## 自分の頭を使わない方が効率的な領域

- ・学習者にとって高度すぎる内容 ✓
- ・アイディアを必要とするもの
- ・定石
- ・モデルケース（成功例）
- ・重要ではないが時間がかかる作業 ✓

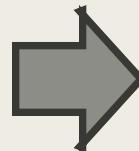
## 自分の頭で考えるべき領域

- ・知識の整理（確認、要約）
- ・試行錯誤で解決する問題

# 自分の頭で考える or 調べてパクる

- 授業用ホームページからサンプルコード Prog\_03-2 をコピーし、新しいテキストファイルにペーストしよう。
- 番号付きリストの出力が (1)(2)(3) ... に変わっているはず。

プログラミング			
	内容	資料	コード
第1回	いろいろなプログラミング言語 VSCode のインストール	●	<a href="#">Prog_01-1</a>
第2回	Webページを構築する(HTML)	●	<a href="#">Prog_02-1</a>
第3回	Webページの見栄えを整える(CSS)	●	<a href="#">Prog_03-1</a>
第4回	JavaScriptに触れてみよう	●	<a href="#">Prog_03-2</a>
第5回	変数と演算	●	
第6回	剩余演算、条件文(1)	●	
第7回	条件文(2)、繰り返し(1)	●	
第8回	繰り返し(2)	●	
第9回	オブジェクト	●	



## Prog\_03-2

```
<!DOCTYPE html>
<html>

<head>
  <meta charset="UTF-8">
  <title>Prog_03-2</title>
  <style>
    ol.list_parentheses li{
      list-style-type:none;
      counter-increment: cnt;
      text-indent: -1.7em;
      padding-left: 1em;
    }
    ol.list_parentheses li:before{
      content: "(" counter(cnt) ")";
    }
  </style>
</head>

<body>
```

[\[コピー\]](#)

## 朝の準備

- (1) 顔を洗う
- (2) 朝ごはんを食べる
- (3) 歯を磨く

# コードの使い方の解説

- <ol>タグを使うときに、  
`<ol class="list_parentheses">`  
と入力して使う。
- [発展] list\_parentheses は <ol> タグにつけられたクラス名で、  
<ol> タグにのみ有効なクラスセレクタ。
- 実際のところ CSS はそれだけでも奥が深いのが、細かい CSS の技術は本講義の範囲外。
- この授業では「タグに style 属性を使う」方法（本スライド p14 参照）を知っていれば最低限の内容としてはOK。

```
1  <!DOCTYPE html>
2  <html>
3
4  <head>
5      <meta charset="UTF-8">
6      <title>Prog_03-2</title>
7      <style>
8          ol.list_parentheses li{
9              list-style-type:none;
10             counter-increment: cnt;
11             text-indent: -1.7em;
12             padding-left: 1em;
13         }
14         ol.list_parentheses li:before{
15             content: "(" counter(cnt) ") ";
16         }
17     </style>
18 </head>
19
20 <body>
21     <h2>朝の準備</h2>
22     <ol class="list_parentheses">
23         <li>顔を洗う</li>
24         <li>朝ごはんを食べる</li>
25         <li>歯を磨く</li>
26     </ol>
27 </body>
28
29 </html>
```

# [演習]年表を作ろう

- 次を参考にし、年表を作成してみよう。
- 特定の人物（推しや自分自身を含む）に関する年表でも良いし、漫画・アニメに関する年表でも良い。
- その際、head部でいくつかのセレクタを宣言し、見やすい表になるよう工夫をしてみよう。

ドラえもん年表

年	月日	出来事
2112	9月3日	ドラえもん誕生
2114	12月2日	ドラミ誕生
2115	1月19日	ドラえもんがセワシの家へ
2122	8月30日	ドラえもんがネズミに耳をかじられる

- 必要な情報は適宜自分で調べてみよう。
- 上記のように、奇数行と偶数行で行の背景色を（自動的に）変更するにはどうすればよいか調べてみよう。

# [演習] いろいろな CSS プロパティを調べよう

- 例えば次のサイトを閲覧し、今日の授業で取り扱わなかった CSS のプロパティも見てみよう。

<https://www.ink.or.jp/~bigblock/css/index.html>

- 次回 JavaScript に入ります。
- CSS に飽きた人は JavaScript を先取りして教科書の p30 から読み進めていいってよい。