

プログラミング

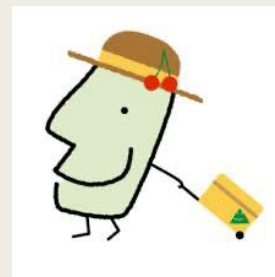
第1回

いろいろなプログラミング言語
VSCode のインストール

久保田 匠

自己紹介

名前 : 久保田 匠 (くぼた しょう)
研究室 : 自然科学棟 521 研究室
担当科目 : 確率統計II, 線形数学演習I, プログラミング
専門 : スペクトルグラフ理論 (線形代数+グラフ理論)
量子ウォーク (確率論+量子力学)
出身 : 山形県寒河江市 (さくらんぼが有名)



↑
←山形県公式HPより



<https://machico.mu/special/detail/1827>

授業資料について

- 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。



愛知教育大学 数学教育講座

	所属教員	時間割
教員と研究	愛知教育大学数学教育学会	イプシロン
	他の研究会	
リンク	愛知教育大学	MathSciNet
	数学第2サーバー	まなびネット
	ICT教育基盤センター	AUEリンク

専任講師	Watanabe, Yuta 渡邊 悠太	有限射影幾何学	自然科学棟 523	2336	ywatanabe	
専任講師	Kubota, Sho 久保田 匠	代数的確率論	自然科学棟 521	2323	skubota	●
助教	Ishikawa, Masaaki 石川 雅章	数学教育学	自然科学棟 535	2331	m-ishikawa	●

● Eメールアドレスは後に、@auecc.aichi-edu.ac.jp を付けて下さい。
● 電話番号は、内線番号です。外線からは、前に0566-26-を付けて下さい。

授業用ホームページ (久保田)

2025年度前期担当科目

	月曜	火曜	水曜	木曜	金曜
1限					
2限	確率統計II			確率統計II	
3限				線形数学演習I	確率統計II
4限	4年ゼミ				(オフィスアワー)
5限					

2025年度後期担当科目

	月曜	火曜	水曜	木曜	金曜
1限					
2限					
3限	科学リテラシー				プログラミング
4限	(オフィスアワー)	3年ゼミ		4年ゼミ	プログラミング
5限					

その他のコンテンツ → ● ●

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

授業資料について



授業用ホームページ (久保田)

2025年度前期担当科目

	月曜	火曜	水曜	木曜	金曜
1限					
2限	確率統計II			確率統計II	
3限				線形数学演習I	確率統計II
4限	4年ゼミ				(オフィスアワー)
5限					

2025年度後期担当科目


	月曜	火曜	水曜	木曜	金曜
1限					
2限					
3限	科学リテラシー				プログラミング
4限	(オフィスアワー)	3年ゼミ		4年ゼミ	プログラミング
5限					

その他のコンテンツ →  

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



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		Prog_01-1
第2回	Webページを構築する(HTML)		Prog_02-1
第3回	Webページの見栄えを整える(CSS)		Prog_03-1 Prog_03-2
第4回	JavaScriptに触れてみよう		Prog_04-1
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)		Prog_08-1
第9回	繰り返し(2)		(なし)

金曜3限も金曜4限も同じ

- 毎回の授業資料はこのページにアップロードするのでブックマーク（お気に入り登録）しておくとい。

本授業の目標

「数学の問題が自動で生成され、『答え』ボタンをクリックすると実際に答えが表示されるウェブページを作成すること」

そのようなウェブページの作成を通して、

- ① プログラミング技術の基礎
- ② ウェブページを構成するための技術
- ③ 数式を綺麗に表示するための TeX を学習する。

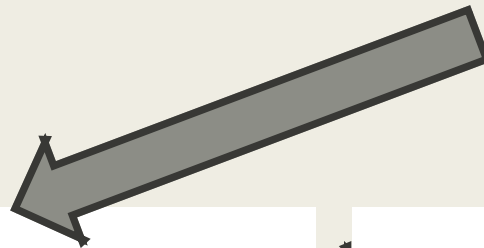
■ 課題の提出例を見てみよう。

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	●	Prog_01-1
第2回	Webページを構築する(HTML)		Prog_02-1
第3回	Webページの見栄えを整える(CSS)		Prog_03-1 Prog_03-2
第4回	JavaScriptに触れてみよう		Prog_04-1
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)		Prog_08-1
第9回	繰り返し(3)		(なし)
第10回	オブジェクト		(なし)
第11回	配列		Prog_11-1
第12回	ユーザー定義関数		Prog_12-1
第13回	イベントハンドラ		(なし)
第14回	数式の表示(TeXについて)		Prog_14-1
第15回	ウェブツールを開発してみよう		課題提出例

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	●	Prog_01-1
第2回	Webページを構築する(HTML)		Prog_02-1
第3回	Webページの見栄えを整える(CSS)		Prog_03-1 Prog_03-2
第4回	JavaScriptに触れてみよう		Prog_04-1
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)		Prog_08-1
第9回	繰り返し(3)		(なし)
第10回	オブジェクト		(なし)
第11回	配列		Prog_11-1
第12回	ユーザー定義関数		Prog_12-1
第13回	イベントハンドラ		(なし)
第14回	数式の表示(TeXについて)		Prog_14-1
第15回	ウェブツールを開発してみよう		課題提出例



逆行列

行列 $\begin{bmatrix} -5 & 0 & 1 \\ -4 & -1 & 1 \\ -2 & -2 & 1 \end{bmatrix}$ の逆行列は

である。

答え

逆行列

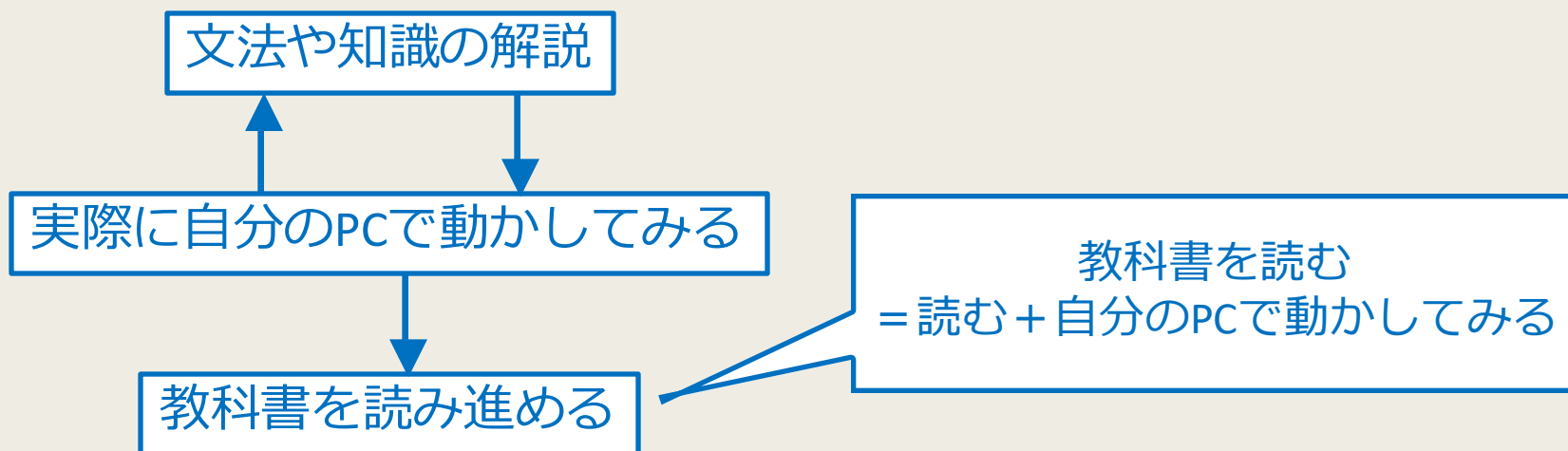
行列 $\begin{bmatrix} -5 & 0 & 1 \\ -4 & -1 & 1 \\ -2 & -2 & 1 \end{bmatrix}$ の逆行列は $\begin{bmatrix} 1 & -2 & 1 \\ 2 & -3 & 1 \\ 6 & -10 & 5 \end{bmatrix}$ である。

答え

↑このページで変な文字列が出てきたらページを再読み込みしてください。



授業の進め方と教科書



◆教科書◆

大津真『3ステップでしっかり学ぶ
JavaScript入門』技術評論社

第4回の授業までには**必ず購入**しておくこと



成績評価

◆最終成果物（100%）

逆行列

行列 $\begin{bmatrix} -5 & 0 & 1 \\ -4 & -1 & 1 \\ -2 & -2 & 1 \end{bmatrix}$ の逆行列は

である。

答え

逆行列

行列 $\begin{bmatrix} -5 & 0 & 1 \\ -4 & -1 & 1 \\ -2 & -2 & 1 \end{bmatrix}$ の逆行列は $\begin{bmatrix} 1 & -2 & 1 \\ 2 & -3 & 1 \\ 6 & -10 & 5 \end{bmatrix}$ である。

答え

↑こういうのを作ってください↑

- 問題の題材は数学であれば何でもよし。
 - 中学高校の教科書レベルの問題
 - 大学入試の典型的な問題、など。
- 先輩の作品を見てみよう（次のフニイド）

本資料では非公開。
授業で紹介します。

最初期のプログラミング言語

■ 最初は機械語（マシン語）

- コンピュータが直接理解して実行できる言語は**機械語（マシン語）**である。
- 機械語は2進数や16進数で表現された命令の集合。
- 最初期（1940年頃）のプログラミング言語は機械語で、プログラマは機械語を使ってプログラムを書いていた。

画面に「Hello world!」
と表示させるプログラム

1	00000000	7f	45	4c	46	02	01	01	00	00	00	00	00	00	00	00	00
2	00000010	01	00	3e	00	01	00	00	00	00	00	00	00	00	00	00	00
3	00000020	00	00	00	00	00	00	00	00	30	01	00	00	00	00	00	00
4	00000030	00	00	00	00	40	00	00	00	00	00	40	00	0d	00	0a	00
5	00000040	55	48	89	e5	bf	00	00	00	00	e8	00	00	00	00	b8	00
6	00000050	00	00	00	5d	c3	00	00	00	48	65	6c	6c	6f	20	77	6f
7	00000060	72	6c	64	21	00	00	47	43	43	3a	20	28	55	62	75	6e
8	00000070	74	75	2f	4c	69	6e	61	72	6f	20	34	2e	36	2e	33	2d
9	00000080	31	75	62	75	6e	74	75	35	29	20	34	2e	36	2e	33	00
10	00000090	14	00	00	00	00	00	00	00	01	7a	52	00	01	78	10	01
11	000000a0	1b	0c	07	08	90	01	00	00	1c	00	00	00	1c	00	00	00
12	000000b0	00	00	00	00	15	00	00	00	00	41	0e	10	86	02	43	0d
13	000000c0	06	50	0c	07	08	00	00	00	00	2e	73	79	6d	74	61	62
14	000000d0	00	2e	73	74	72	74	61	62	00	2e	73	68	73	74	72	74
15	000000e0	61	62	00	2e	72	65	6c	61	2e	74	65	78	74	00	2e	64
16	000000f0	61	74	61	00	2e	62	73	73	00	2e	72	6f	64	61	74	61
17	00000100	00	2e	63	6f	6d	6d	65	6e	74	00	2e	6e	6f	74	65	2e
18	00000110	47	4e	55	2d	73	74	61	63	6b	00	2e	72	65	6c	61	2e
19	00000120	65	68	5f	66	72	61	6d	65	00	00	00	00	00	00	00	00
20	00000130	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

アセンブリ言語の登場

■ アセンブリ言語（1950年前後）

- プログラマは機械語を直接扱うのは困難と認識。
- 人間がより理解しやすい形でプログラムを記述する方法を模索し、**アセンブリ言語**を開発した。
- アセンブリ言語は機械語と1対1に対応するプログラミング言語。

```
1 section .data
2     msg db 'Hello, world!',0xA
3     len equ $ - msg
4
5 section .text
6     global _start
7
8 _start:
9     mov eax, 4
10    mov ebx, 1
11    mov ecx, msg
12    mov edx, len
13    int 0x80
14
15    mov eax, 1
16    xor ebx, ebx
17    int 0x80
```

画面に「Hello world!」
と表示させるプログラム

$2 + 3 \times 4$ を計算
(計算結果の表示はしない)

```
section .text
global _start

_start:
    mov eax, 3
    imul eax, 4
    add eax, 2

    mov eax, 1
    xor ebx, ebx
    int 0x80
```

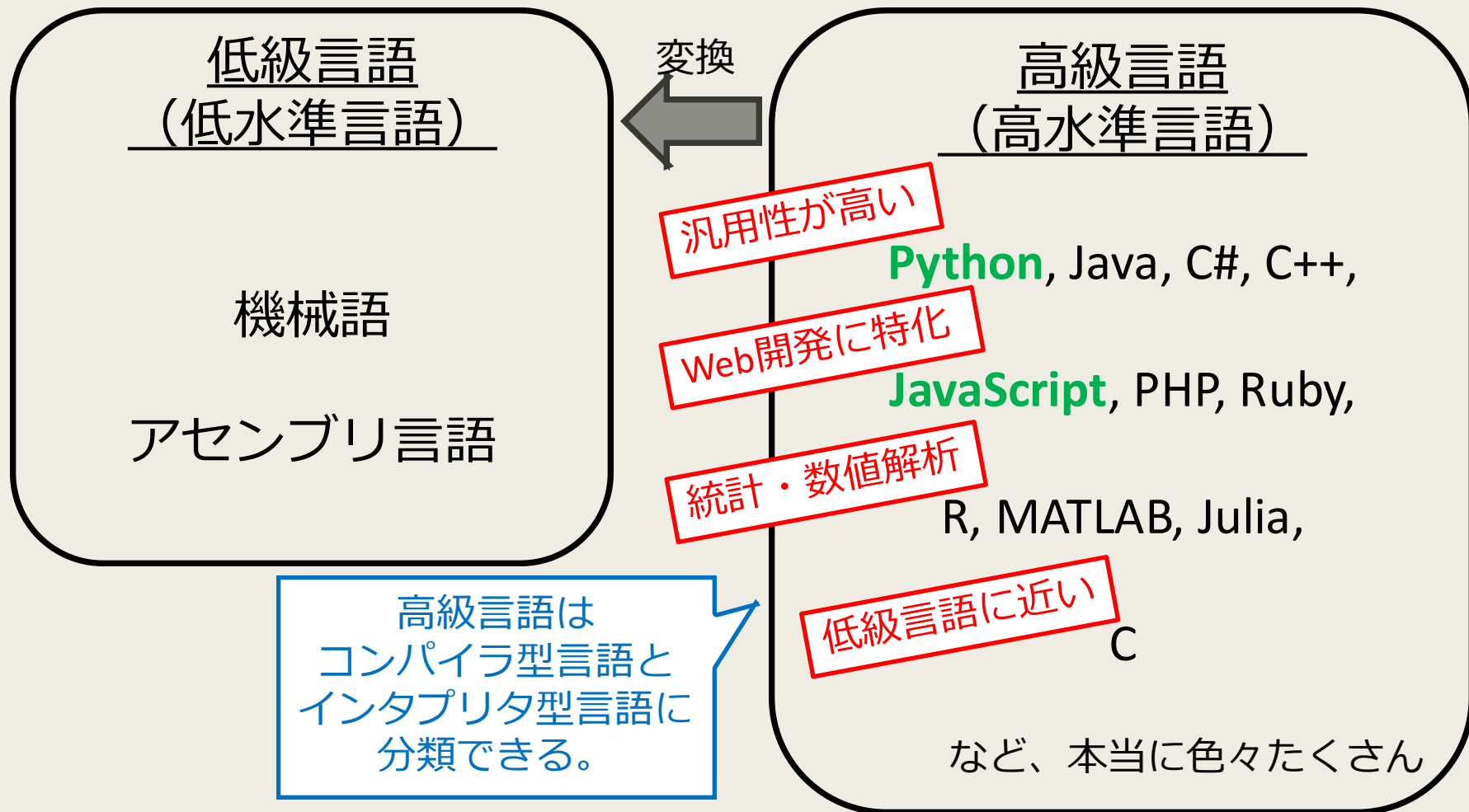
もっと人間が分かりやすいように...

■ Fortran (1957～)

- 機械語やアセンブリ言語によるプログラミングは手間がかかりすぎる。
- 数学の式を書けば、それを自動的に機械語のプログラムに変換してくれる仕組みが望まれた。
- Fortran は「 $2 + 3 \times 4$ 」とかけば、そのとおりに計算してくれるという、今では当たり前のことを最初に実現した言語。
- **Formula Translating System** の略。
- Fortran のように、人間にとって分かりやすい形式のプログラミング言語を **高級言語 (高水準言語)** という。
- Fortran は最初期の高級言語である。
- 現在、我々が「プログラミング言語」と呼んでいるものは高級言語を指す。

いろいろなプログラミング言語

※ Java と JavaScript は違う言語



- プログラミング言語によって得意なことは違う。

Python or JavaScript ??

- プログラミング言語によって得意なことは違う。
- どの言語が「良い」か？という問いは学習者の目的に依存して答えが変わる。
- 本授業は「数学の問題が自動生成されるウェブページの作成」を最終目標としているので JavaScript が適当。

	Python	JavaScript
主な用途	データ処理、AI開発など	Web開発
文法の特徴	非常にシンプルで見やすいがやや独特	比較的シンプルで学習しやすい
計算速度	比較的遅い	比較的速い（らしい）

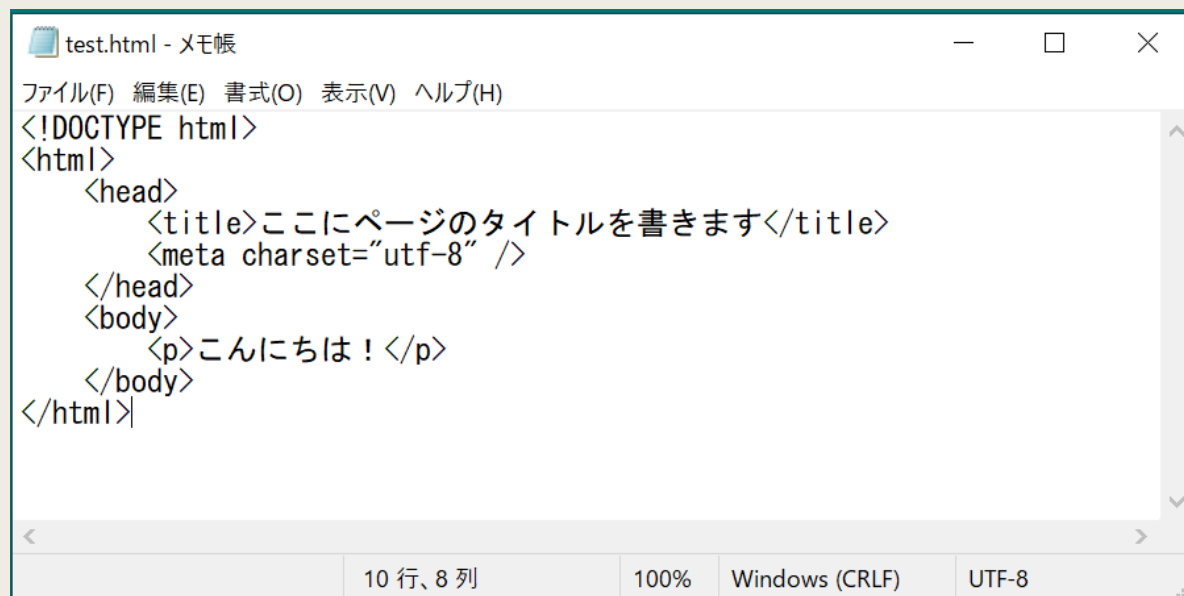
- ただし、初学者が学ぶ項目についてはどの言語も本質的な部分はだいたい同じ。

コンピュータリテラシーは人それぞれ

- 本授業の受講者は「大学に入ってからパソコンを使い始めた学生」から「自分でプログラムを書いたことがある（アプリ開発をしたことがある）学生」まで想定される。
- したがって、授業を難しいと感じる学生もいれば、簡単すぎてつまらないと感じる学生もいるだろう。
- 「難しい」と感じる学生は友達や先生に積極的に質問しよう。
- 「簡単」と感じる学生は躓いている友達に教えてあげよう。

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。
 - Hyper Text Markup Language の略。
- メモ帳でも作れる。

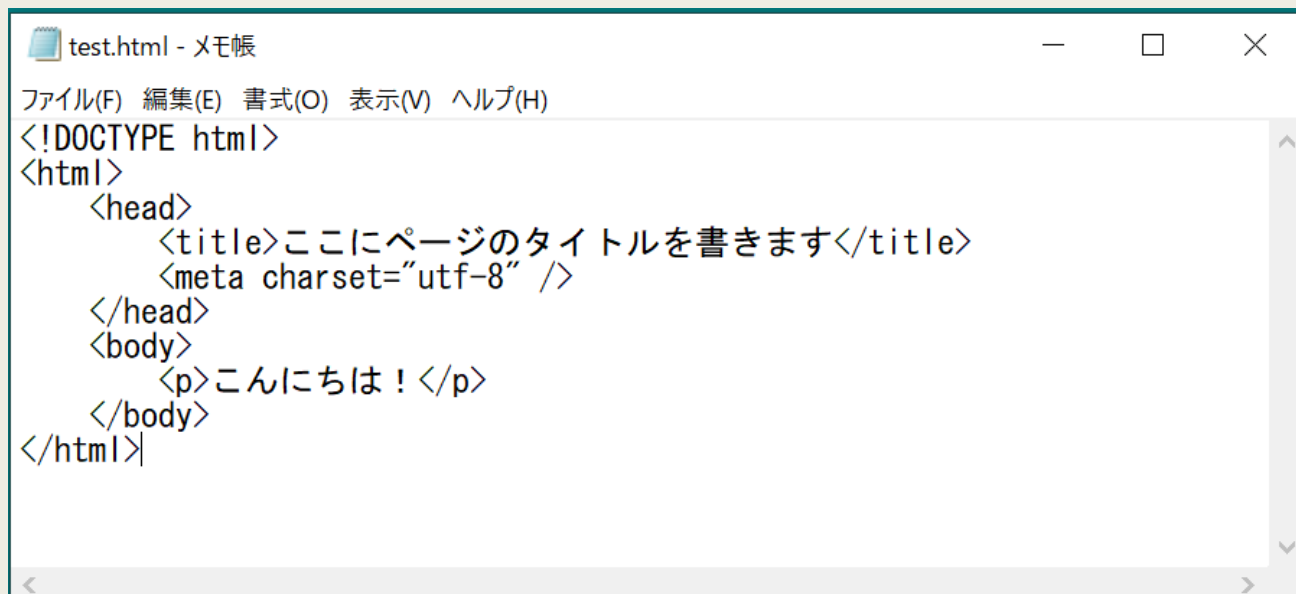


```
test.html - メモ帳
ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)
<!DOCTYPE html>
<html>
  <head>
    <title>ここにページのタイトルを書きます</title>
    <meta charset="utf-8" />
  </head>
  <body>
    <p>こんにちは！</p>
  </body>
</html>
```

10 行、8 列 100% Windows (CRLF) UTF-8

VSCode

- メモ帳ひとつでWebページを立ち上げることができるが、コードが複雑になるとプログラムが見づらくなる。
- メモ帳よりも高機能なプログラム作成ツール（**コードエディタ**）を使おう。
- コードエディタは **Visual Studio Code**（通称 **VSCode**）がおすすめ。
- [分かる人向け] もし自分でお気に入りのコードエディタがある人はそれを使ってもらって構わない。



The screenshot shows a window titled "test.html - メモ帳" (test.html - Notepad). The menu bar includes "ファイル(F)", "編集(E)", "書式(O)", "表示(V)", and "ヘルプ(H)". The text area contains the following HTML code:

```
<!DOCTYPE html>
<html>
  <head>
    <title>ここにページのタイトルを書きます</title>
    <meta charset="utf-8" />
  </head>
  <body>
    <p>こんにちは！</p>
  </body>
</html>
```


VSCode のインストール

- VSCode と、VSCode を日本語で使える拡張機能をダウンロードしてインストールしよう。
 - ダウンロードとは、PC にファイルを取り込む（保存する）こと。
 - インストールとは、そのファイルを使用できる状況にすること。



<https://www.naporitansushi.com/download-install/>

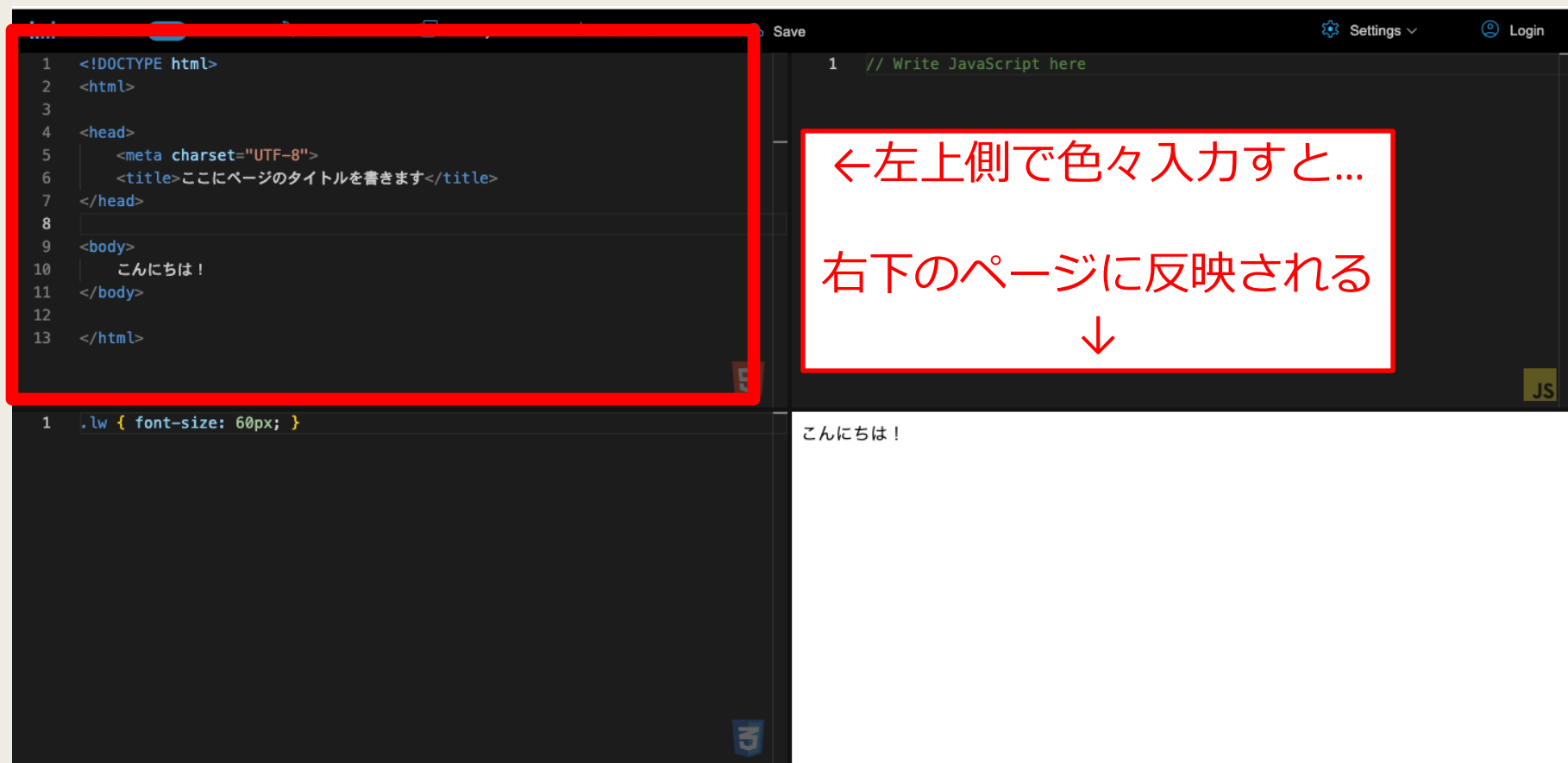
VSCode のインストール

- VSCode と、VSCode を日本語で使える拡張機能をダウンロードしてインストールしよう。
 - ダウンロードとは、PC にファイルを取り込む（保存する）こと。
 - インストールとは、そのファイルを使用できる状況にすること。
- 教科書を既にもっている人は p24, 25 にしたがってください。
- 教科書をまだ持っていない人は友達に見せてもらうか次のサイトを参考にしてください。
- Windows の PC を使っている人 ↓
<https://www602.math.ryukoku.ac.jp/Prog1/vscode-win.html>
- Mac の PC を使っている人 ↓
<https://www602.math.ryukoku.ac.jp/Prog1/vscode-mac.html>

インストールがうまくいかなかった人

- 次回までになんとかしておくこと。
- 今日は、次のサイトにアクセスしてしのごう。

<https://liveweave.com/>



サンプルコードのコピー

- 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。
- [コード](#)とは、コンピュータに命令を与えるためのデータ、あるいはプログラミング言語で記述された文書のこと。
- [サンプル](#)とは、ここでは「例」や「見本」のような意味。
- html はどちらといえばプログラミング言語ではない（マークアップ言語という）が html ファイルに対しても「コード」という言葉を使う。

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	●	Prog_01-1
第2回	Webページを構築する(HTML)		Prog_02-1
第3回	Webページの見栄えを整える(CSS)		Prog_03-1 Prog_03-2
第4回	JavaScriptに触れてみよう		Prog_04-1
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)		Prog_08-1
第9回	繰り返し(2)		(なし)



Prog_01-1

```
<!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！
</body>

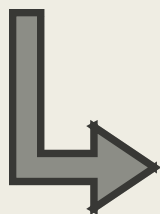
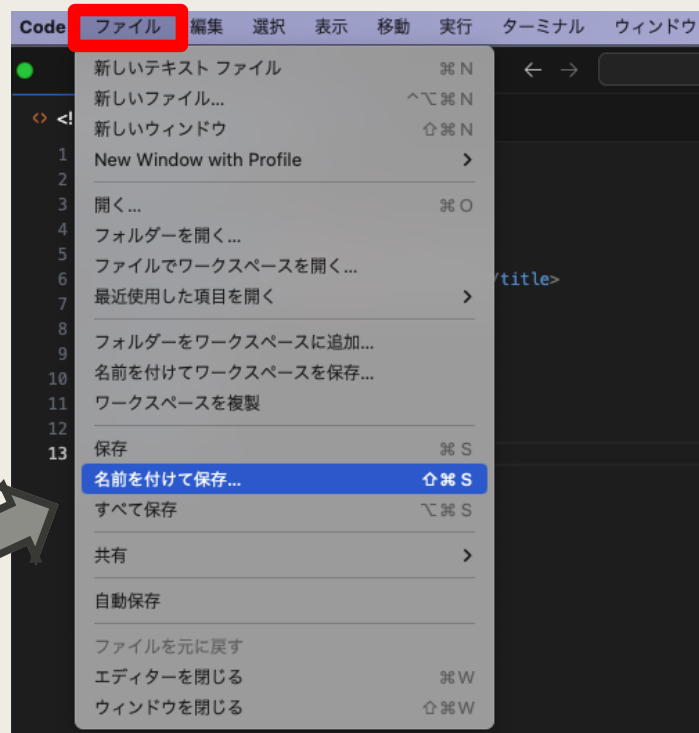
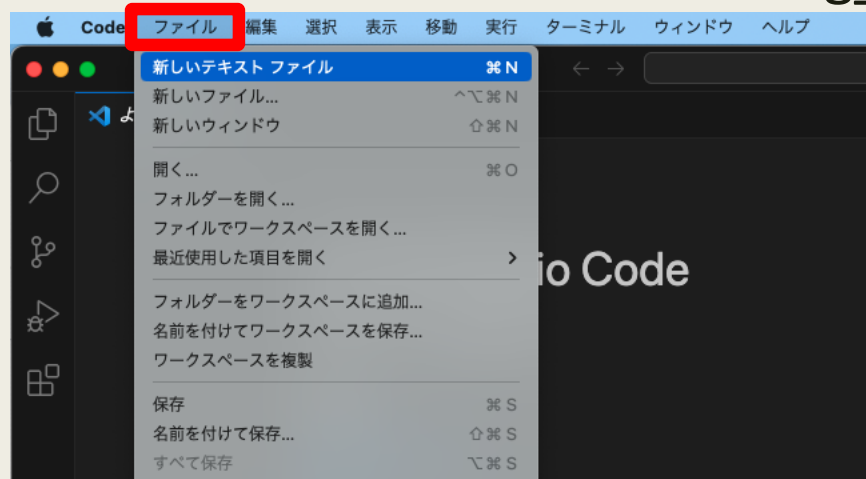
</html>
```

コピー

コピーボタンを押せばとりあえずOK

コードの新規作成（重要！）

- VSCode を起動し「ファイル」から「新しいテキストファイル」を選択しよう。
- そのあと、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」。
- ファイル名は例えば「Prog_01-1.html」としておく。



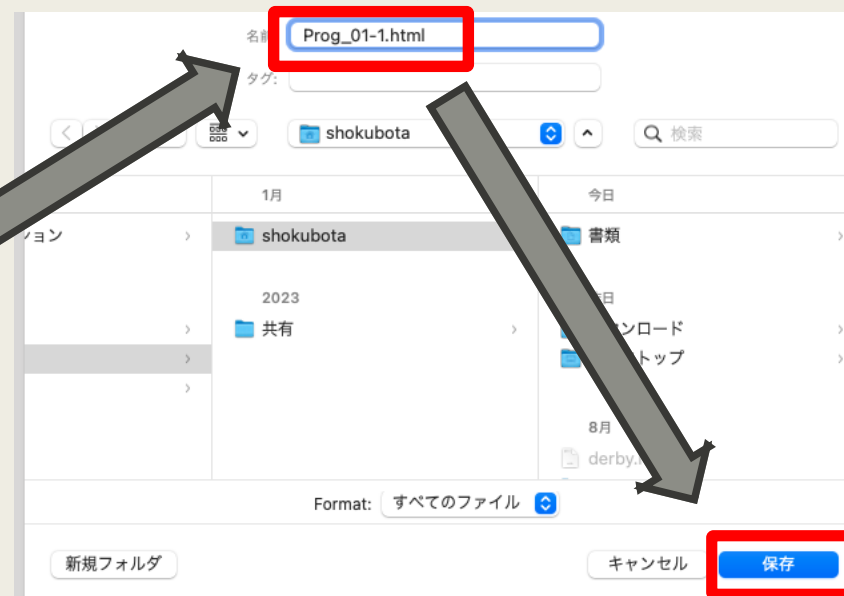
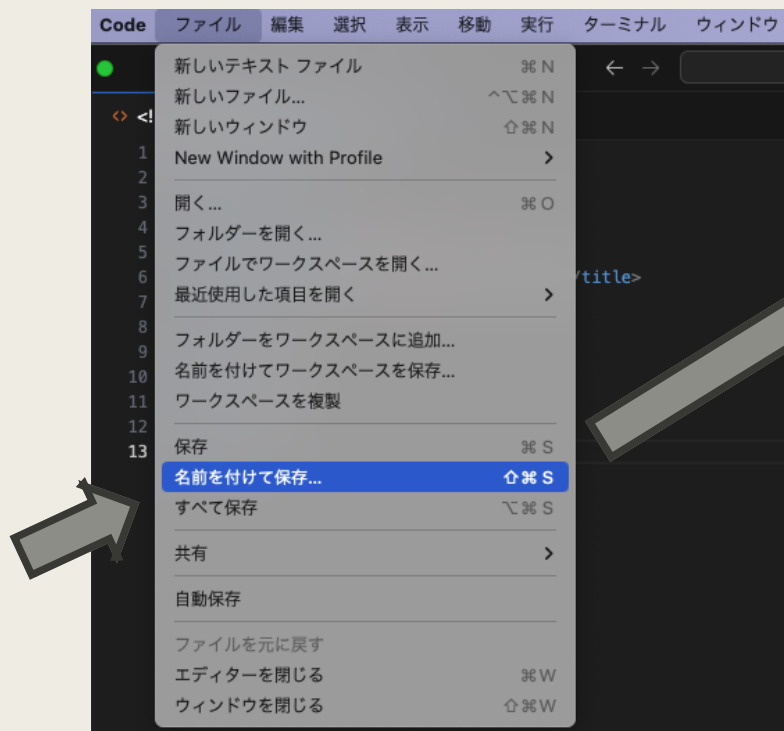
Ctrl + V



コードの新規作成（重要！）

- VSCode を起動し「ファイル」から「新しいテキストファイル」を選択しよう。
- そのあと、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」。
- ファイル名は例えば「Prog_01-1.html」として保存。

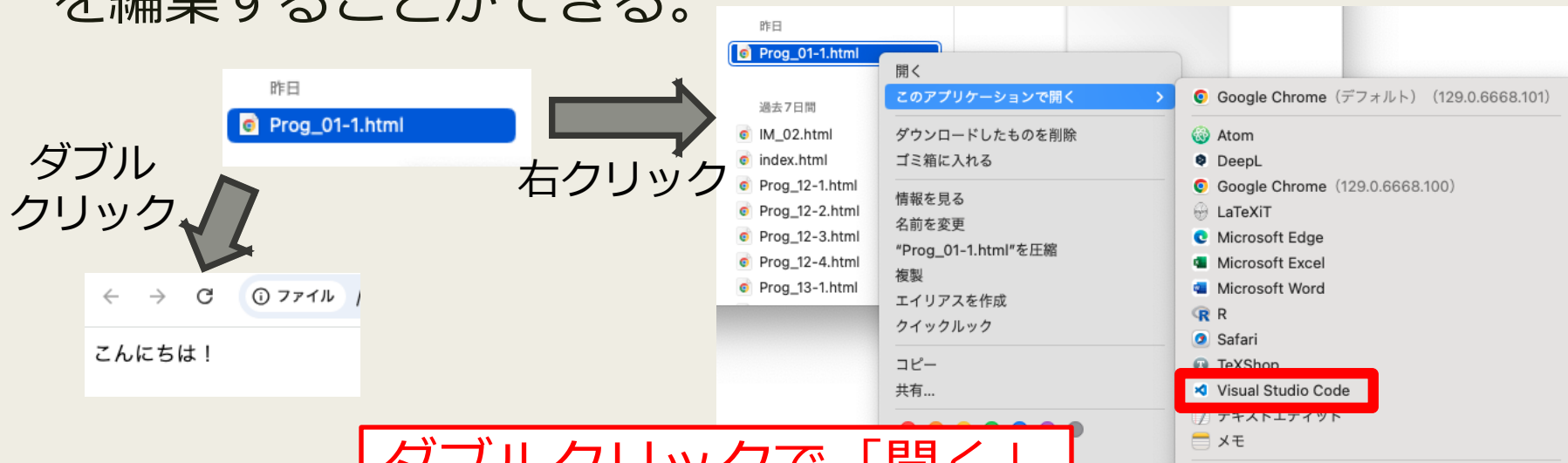
.html は必ずつける



保存が済んだら一度 VScode を閉じよう

htmlファイルを開く・編集する

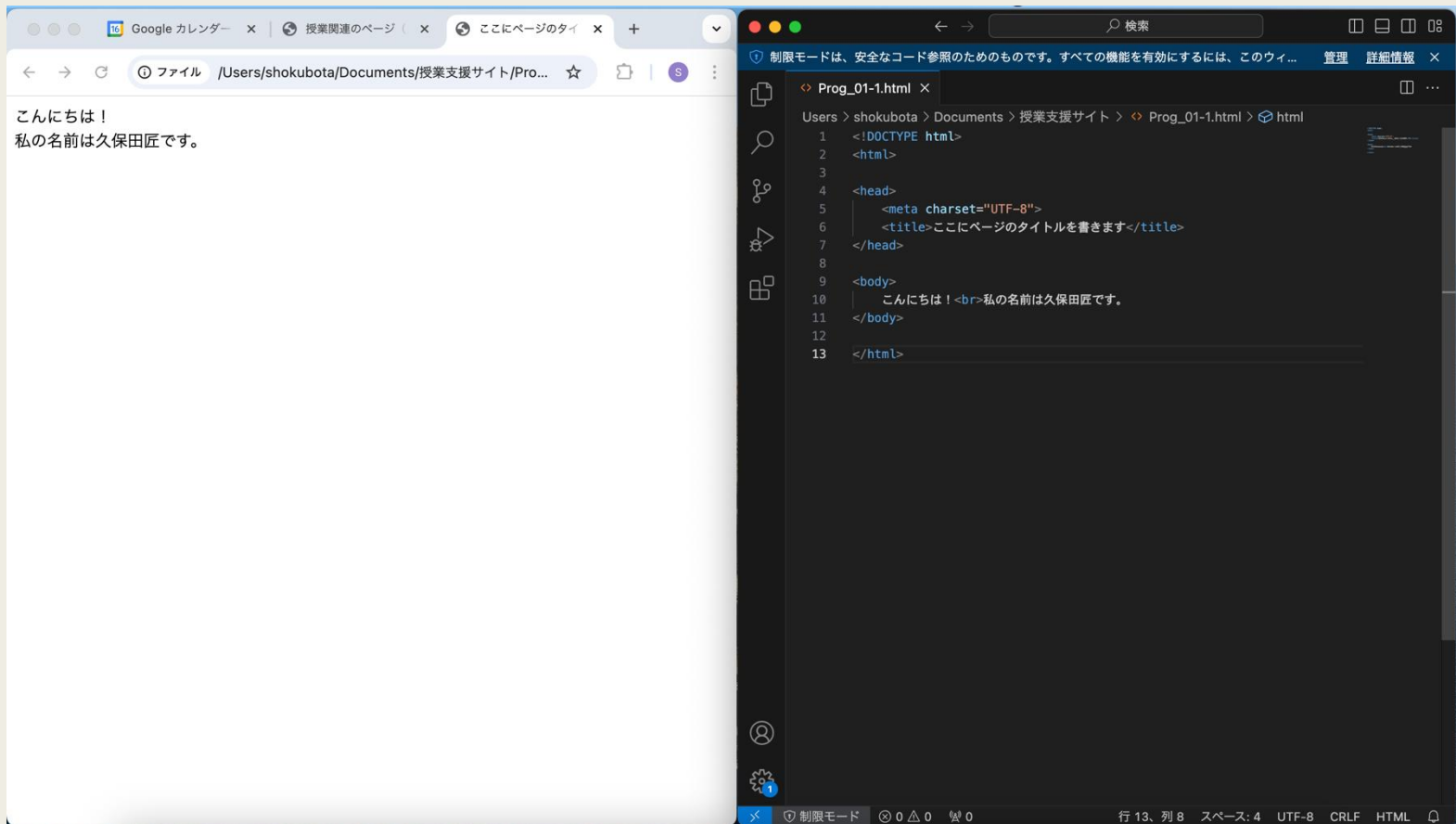
- 次に、ファイル管理ソフト（エクスプローラー、Finder）でさきほど保存したファイル「Prog_01-1.html」をダブルクリックしよう。
- ダブルクリックするとhtmlファイルが開かれ、「こんにちは！」とだけ書かれたwebページが立ち上がる。
- 一方、「Prog_01-1.html」を右クリックし「Visual Studio Code」で開くを選択すると、「Prog_01-1.html」内のコードを編集することができる。



ダブルクリックで「開く」
右クリックで「編集」

おすすめの作業環境

- 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。
- ブラウザとは、Webページを表示・閲覧するためのソフトウェア。Google Chrome や Microsoft Edge など。



サンプルコードをいじってみよう

- サンプルコードを見てみると10行目の「こんにちは！」が、ファイルを開いたときの画面に反映されている。
- 10行目を編集して「私の名前は〇〇〇〇です。」と入力してみよう。
- その後、編集したファイル（コード）を保存（Ctrl + S）してブラウザを再読み込みしてみよう。
- 再読み込みは「🔄」を押すことができる。

```
<> 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！
11  </body>
12
13 </html>
```

```
<> 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！私の名前は久保田匠です。
11  </body>
12
13 </html>
```

編集したら忘れずに保存する

改行してみよう

- 改行したいときは `
` と入力する。
- `br` は `break` の略で、改行(line break)を意味する。

↓このコードだと改行されない

```
Prog_01-1.html x
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！
11   私の名前は久保田匠です。
12 </body>
13
14 </html>
```

↓このコードだと改行される

```
Prog_01-1.html x
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>
```

編集したら忘れずに保存

```
Prog_01-1.html x
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>私の名前は久保田匠です。
11 </body>
12
13 </html>
```

[演習]サンプルコードをいじってみよう

- 残った時間でサンプルコードをいじってみよう。
- 6行目をいじるとページはどのように反映されるだろうか？

```
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！
11 </body>
12
13 </html>
```

- 次回、<>などの意味を紹介。