# **楽しくゲームを作りながら、覚える**

# **HTML/CSS/JS入門（5ステップ）**

子ども向け・初心者向けに、****実際に動くページ****を作りながらHTML/CSS/JavaScript（JS）を学ぶ教材です。テーマは「****実際のコーディングと問題を考えて楽しく解く****」。最終的には****簡単なポートフォリオ****として成果をまとめます。 ※RaspberryPi 4B や RaspberryPi 5、Ubuntu等は、Linuxとして一括りにまとめて説明します。操作方法は基本的に同じです。

## **進め方（保護者・先生向け）**

* 各ステップは ****1ファイル（index.html）**** を基本に作ります。
* ****書いて → 保存して → ブラウザで開く****の繰り返しでOK。難しいインストールは不要です。
* 1ステップは目安30〜60分。合計で3〜5時間程度。
* 各ステップに****チャレンジ問題****と****難易度ダイヤル****（簡単/むずかしい調整）を用意。

## **用意するもの**

* パソコン：Windows/Mac/Chromebook/Linux（RaspberryPi等）どれでもOK
* エディタ：メモ帳でも可。おすすめ：****VS Code****
* ブラウザ：Chrome/Safari/Edgeなど

## **はじめての人のためのHow To Use（開き方・コードの見方・直し方）**

まったくの初心者でも、この順にやれば“開ける・読める・直せる”を体験できます。

### **1) まずは動かしてみる**

1. 好きな場所にフォルダを作る（例：portfolio/step1/）。
2. フォルダの中に****新しいファイル****を作って、名前を\*\*index.html\*\*にする。
3. 教材のコードを****まるごとコピペ****してindex.htmlに貼り付け、****保存****（Windows/Chromebook/LinuxはCtrl + S、Macは⌘ + S）。
4. フォルダでindex.htmlを****ダブルクリック**** → ブラウザ（Chrome/Safari/Edgeなど）で開く。
5. 直したいときは再び保存して、ブラウザを****更新****（F5 / Ctrl + R（Windows/Chromebook/Linux）/ ⌘ + R（Mac））。

****ポイント****：作業は「エディタで書く → 保存 → ブラウザで見る → 更新」のくり返し。

### **2) エディタの選び方（メモ帳でもOK）**

* いちばん簡単：****メモ帳 / TextEdit****（Macは“標準テキスト”に設定、Linuxは標準の「テキストエディタ」でOK）
* おすすめ：****VS Code****（無料）。File > Open Folderでプロジェクトのフォルダごと開くと迷子になりにくい。
* どこでも使える：****nano**** sudo nano ~~~でファイルを指定して開く。Ctrl + X,Y,Enterの順で保存して閉じる。覚えておくと便利

### **3) OS別・新規ファイルの作り方ミニガイド**

* ****Windows****：エクスプローラーで右クリック → 新規作成 → テキストドキュメント → 名前をindex.htmlに。拡張子が見えないときは「表示 → 表示/非表示 → ****ファイル名拡張子****」にチェック。
* ****Mac****：TextEditを開く → 環境設定で「****標準テキスト****」に → 新規作成 → index.htmlとして保存。拡張子が.txtにならないよう注意。
* ****Chromebook****：ファイルアプリまたはテキストエディタで新規作成 → index.htmlとして保存 → Chromeで開く。
* ****Linux****：ファイルマネージャで右クリック → 新規作成 → 空のファイル → 名前をindex.htmlにして保存。テキストエディタで開いて編集。
* ****terminal****：作りたいディレクトリ（フォルダのある場所）に移動（cd ~/flder\_name）し、sudo nano index.htmlで新しいファイルに書き込み。既存のindex.htmlがある場合は、それが開かれるので注意。（コピーの方法：cp index.html index2.html cp [コピー元] [コピー後]）

### **4) コードの“見方・参照の仕方”超入門**

****3つの視点****で追いかけると迷いません。

1. ****HTML（なにがある？）**** … 画面の部品。例：<div id="score">0</div> は「スコア表示の箱」。
2. ****CSS（どう見える？）**** … 見た目。例：#score { font-size:48px; } はid="score"の文字を大きく。
3. ****JS（どう動く？）**** … 動き。例：document.getElementById('score').textContent = 1;で数字を書き替え。

****対応表（ステップ1の例）****

* HTML：<div id="score">0</div> ↔ CSS：#score { ... } ↔ JS：document.getElementById('score')
* HTML：<button id="btnAdd"> ↔ JS：btnAdd.addEventListener('click', ...)（ボタンを押したら〜をする）

****探し方のコツ****

* 画面で気になる部分の****id / class****をHTMLで探す → 同じ名前を****CSS****や****JS****でCtrl + F（検索）してたどる。
* 「数字が増える仕組み」を知りたい → scoreで検索 → HTML（箱）→ JS（足し算と表示）を見る。

### **5) よく使う記号・用語チートシート**

* ()：関数のかっこ / []：配列 / {}：まとまり（ブロック）
* ""や''：文字 / ;：文のおわり / //：メモ（コメント）
* ****タグ****：<h1>〜</h1> / ****属性****：id="score" / ****クラス****：class="card"
* ****イベント****：クリックなどのきっかけ（addEventListener('click', ...)）

### **6) つまずいたらチェック（子ども向け）**

* ぜんかく（全角）になってない？ "や'、(、)、; が****半角****になっているか確認。
* idの名前、****JSとHTMLで同じ？****（scoreとScoreは別物）
* </div>などの****閉じタグ****をわすれてない？
* 保存した？ → ブラウザを更新した？

### **7) つまずいたらチェック（おとな向けオプション）**

* ****開発者ツール****のコンソールを見る：F12（Windows/Chromebook/Linux）/ ⌥⌘I（Mac）。赤いエラー行の****行番号****をクリック → その行を直す。
* 文字化けは<meta charset="utf-8">を最上部の<head>内に。

### **8) 安全に試すコツ**

* まず****1行だけ****変える → 保存 → 画面で確認 → うまくいかなければCtrl + Z / ⌘ + Zで戻す。
* どこを変えたか残すには、HTMLは<!-- ここを変えた -->、JSは// ここを変えたと****コメント****を書く。

### **9) 用語ミニ辞典**

* ****ブラウザ****：ページを見るアプリ（Chromeなど）
* ****エディタ****：コードを書くアプリ（メモ帳/VS Codeなど）
* ****フォルダ****：ファイルを入れる箱 / ****拡張子****：.htmlのようにファイルの種類を表す部分
* ****HTML****：骨組み / ****CSS****：見た目 / ****JS****：動き

### **ライセンス**

この教材は個人学習・学校授業で自由に使ってOKです（クレジット表記推奨：YashubuStudio / "Made with ❤️ + HTML/CSS/JS" / 2025）。