

ステップ30

留学生のための

JavaScript

ワークブック



Step 01 JavaScriptの記述ルール 8

- 1.1 JavaScriptの概要
- 1.2 JavaScriptを記述する場所
- 1.3 JavaScriptの記述ルール

Step 02 イベントハンドラ 13

- 2.1 イベントハンドラとは？
- 2.2 クリック時にJavaScriptを実行
- 2.3 マウスの移動時にJavaScriptを実行
- 2.4 ページが読み込まれた直後にJavaScriptを実行
- 2.5 その他のイベントハンドラ

Step 03 コンソールの使い方 17

- 3.1 コンソールとは？
- 3.2 コンソールの表示
- 3.3 コンソールに文字を表示
- 3.4 エラーメッセージの確認
- 3.5 コンソールにJavaScriptを記述して実行

Step 04 関数の基本 22

- 4.1 関数とは？
- 4.2 関数の作成方法
- 4.3 関数名に指定できない文字
- 4.4 イベントハンドラから関数を呼び出す

Step 05 変数の利用 26

- 5.1 変数の宣言
- 5.2 変数に数値を代入
- 5.3 変数に文字を代入
- 5.4 変数の表示

Step 06 変数の演算 31

- 6.1 演算子の記述
- 6.2 プログラムならでの記述方法
- 6.3 インクリメントとデクリメント
- 6.4 文字の足し算
- 6.5 変数に文字を追加して表示

Step 07 配列の利用 35

- 7.1 配列とは？
- 7.2 配列の宣言とデータの代入
- 7.3 配列の読み込み
- 7.4 配列の長さ

Step 08 関数の引数 39

- 8.1 関数の引数とは？
- 8.2 引数の指定方法
- 8.3 引数を利用して関数を汎用化
- 8.4 複数の引数を指定する場合

Step 09 要素の取得と内容の変更 44

- 9.1 要素の取得
- 9.2 要素内の文字の変更
- 9.3 要素内のHTMLを書き換える

Step 10 複数の要素の取得 50

- 10.1 セレクタを使った要素の取得
- 10.2 取得する要素を絞り込む方法
- 10.3 要素内の文字の変更

Step 11 ようそ ついか さくじよ 要素の追加と削除 57

- 11.1 ようそ ついか 要素の追加
- 11.2 ようそ ついか ちゆういでん 要素を追加するときの注意点
- 11.3 ようそ さくじよ 要素の削除

Step 12 く かえ しよりに 繰り返し処理-1 63

- 12.1 く かえ しよりに 繰り返し処理とは？
- 12.2 ぶん きじゆつほう for文の記述方法
- 12.3 ひかくえんざんし 比較演算子
- 12.4 く かえ しよりに れい 繰り返し処理の例(1)
- 12.5 く かえ しよりに れい 繰り返し処理の例(2)

Step 13 く かえ しよりに 繰り返し処理-2 70

- 13.1 じゆう く かえ しよりに 2重ループの繰り返し処理
- 13.2 たじゆう つか ひよう さくせい 多重ループを使った表の作成

Step 14 じようけんぶん き 条件分岐-1 77

- 14.1 じようけんぶん き 条件分岐とは？
- 14.2 ぶん きじゆつほう if文の記述方法
- 14.3 しよりに ふた ぶん き if ~ elseで処理を2つに分岐
- 14.4 ぶん き if ~ else文のサンプルプログラム

Step 15 じようけんぶん き 条件分岐-2 82

- 15.1 しよりに みつ いじよう ぶん き else ifで処理を3つ以上に分岐
- 15.2 else ifのサンプルプログラム
- 15.3 ろんり えんざんし 論理演算子
- 15.4 ぶん き しよりに ぶん き switch文で処理を分岐

Step 16 break と continue 91

- 16.1 break の活用方法
- 16.2 continue の活用方法
- 16.3 while 文を使った繰り返し処理

Step 17 関数の戻り値 96

- 17.1 戻り値とは？
- 17.2 戻り値を受け取るには？
- 17.3 戻り値を返す return
- 17.4 true / false を返す return

Step 18 文字入力とエラー処理 103

- 18.1 プロンプトの使い方
- 18.2 プロンプトに数値を入力する場合
- 18.3 エラー対策用の処理

Step 19 オブジェクト、メソッド、プロパティ 110

- 19.1 オブジェクトとは？
- 19.2 オブジェクトツリー
- 19.3 メソッドとは？
- 19.4 プロパティとは？

Step 20 スタイルの変更 116

- 20.1 CSS の変更方法
- 20.2 文字サイズを変更する JavaScript
- 20.3 画像の書式を変更する JavaScript

Step **21** 属性値の変更 123

- 21.1 属性値の変更
- 21.2 属性値の取得

Step **22** イベントリスナー 129

- 22.1 イベントリスナーの使い方
- 22.2 無名関数を使った引数の受け渡し

Step **23** フォームの操作-1 135

- 23.1 テキストボックスに入力した文字
- 23.2 チェックボックスのON/OFF

Step **24** フォームの操作-2 140

- 24.1 ラジオボタンのON/OFF
- 24.2 セレクトボックスの処理

Step **25** 日付、時刻の操作 147

- 25.1 Dateオブジェクトの作成
- 25.2 Dateオブジェクトから年月日、時分秒を取り出す
- 25.3 Dateオブジェクトに年月日、時分秒を指定する

Step **26** Mathオブジェクト 153

- 26.1 Mathオブジェクトとは？
- 26.2 Mathオブジェクトのプロパティ
- 26.3 Mathオブジェクトのメソッド
- 26.4 乱数を利用したJavaScript

Step 27 文字の操作 158

- 27.1 文字変数のプロパティ
- 27.2 文字変数のメソッド
- 27.3 文字変数のメソッドを使用したファイル名の指定

Step 28 オブジェクト変数 163

- 28.1 オブジェクト変数の使い方
- 28.2 オブジェクト配列の使い方

Step 29 一定間隔で処理を繰り返す 169

- 29.1 setInterval() の使い方
- 29.2 setInterval() の中断
- 29.3 スライドショーの作成
- 29.4 setTimeout() の使い方
- 29.5 スライドショーのアニメーション効果

Step 30 jsファイルの分離とエラー対策 176

- 30.1 JavaScript ファイルの作成
- 30.2 JavaScript ファイルの読み込み
- 30.3 変数のスコープについて
- 30.4 スペルミスを防ぐ対策法
- 30.5 コンソール画面を使ったデバッグ処理

演習問題の解答	181
索引 (Index)	223

◆ サンプルファイルと演習で使う画像のダウンロード URL

本書で紹介したサンプルファイル、ならびに演習で使用する画像は以下の URL からダウンロードできます。

http://-----

JavaScriptの記述ルール

JavaScriptは、Webページ上でさまざまな動作を実現できるプログラミング言語です。ステップ01では、JavaScriptの概要と記述ルールについて学習します。

1.1 JavaScriptの概要

Webページを作成するときは、テキストエディタなどでHTMLを記述して、HTMLファイルを作成するのが一般的です。また、HTMLファイル内にCSSを記述し、各要素の書式を指定することも可能です。これらをまとめると、HTMLファイルは以下の2つの言語で構成されることになります。

- HTML 文章や画像など、Webページに掲載する内容を指定する
- CSS 色やサイズなど、各要素の書式を指定する

もちろん、これらの2つの言語だけでWebページを作成しても構いません。ただし、このようにして作成されたWebページは「動きのないWebページ」にしかなりません。一方、JavaScriptを利用してWebページを作成すると、ユーザーの操作に応じて「動きのあるWebページ」を作成できるようになります。

たとえば、「ボタンをクリックすると画像を変更する」などの動作をJavaScriptで実現することが可能です。



図 1-1 JavaScriptを利用した「動きのあるWebページ」

そのほか、「^{せんたく}選択した^{こうもく}項目の^{ごうけいしんがく}合計金額を算出する」など、^{じつようてき}実用的な^{しく}仕組みをJavaScriptで^{じつげん}実現することも^{かのう}可能です。

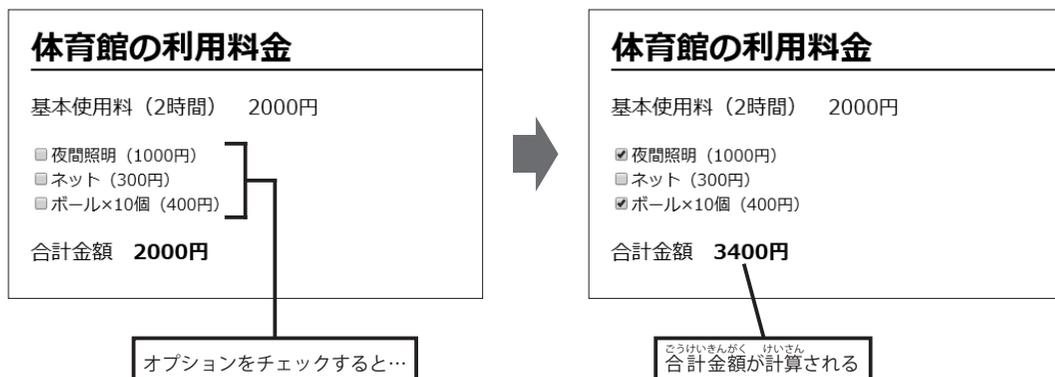


図1-2 JavaScriptを利用して「計算を行うWebページ」

このように、ユーザーの^{そうさ}操作に応じてWebページにさまざまな^{しより}処理をプログラミングできるのがJavaScriptです。

1.2 JavaScriptを記述する場所

続いては、JavaScriptの^{きじゆつ}記述ルールについて^{かいせつ}解説していきます。JavaScriptを利用したWebページを作成するとき、HTMLファイル内にJavaScriptを^{きじゆつ}記述することも可能です。たとえば、^{しめ}図1-2に示したWebページのHTMLファイルは以下のような^{こうせい}構成になっています。

※現時点では、JavaScriptの^{きじゆつ}記述内容を^{りかい}理解できなくても構いません。JavaScriptの^か書き方は、ステップ02以降で詳しく^{かいせつ}解説していきます。

▼ sample01-1.html

```
1 <!DOCTYPE html>
2
3 <html lang="ja">
4
5 <head>
6 <meta charset="UTF-8">
7 <title>体育館の利用料金</title>
8 <style>
9   body{
10     padding: 0px 15px;
11   }
```

CSS

```

12  h1{
13      border-bottom: solid 2px #000000;
14      line-height: 1.2;
15  }
16  p{
17      font-size: 20px;
18  }
19  #total{
20      font-weight: bold;
21  }
22 </style>
23 </head>
24
25 <body>
26 <h1>体育館の利用料金</h1>
27 <p>基本使用料 (2時間) 2000円</p>
28 <form onchange="checkSum();" >
29     <label><input type="checkbox" id="c1">夜間照明 (1000円) </label><br>
30     <label><input type="checkbox" id="c2">ネット (300円) </label><br>
31     <label><input type="checkbox" id="c3">ボール×10個 (400円) </label>
32 </form>
33 <p>合計金額 <span id="total">2000円</span></p>
34
35 <script>
36 function checkSum() {
37     let price = 2000;
38     if(document.getElementById('c1').checked == true){
39         price = price + 1000;
40     }
41     if(document.getElementById('c2').checked == true){
42         price = price + 300;
43     }
44     if(document.getElementById('c3').checked == true){
45         price = price + 400;
46     }
47     document.getElementById('total').textContent = price + '円';
48 }
49 </script>
50
51 </body>
52
53 </html>

```

CSS

JavaScript

このように、HTMLファイルには以下の3つの言語を含むことができます。

■ HTMLファイルに必須となる言語

- HTML 文章や画像など、Webページに掲載する内容を指定する

■ 必要に応じてHTMLに追記する言語

- CSS 色やサイズなど、各要素の書式を指定する
- JavaScript さまざまな処理をプログラミングする

1.3 JavaScriptの記述ルール

続いては、JavaScriptの記述ルールについて解説していきます。JavaScriptを記述するときは、大文字と小文字が区別されることに注意しなければなりません。以下にJavaScriptの主な記述ルールをまとめておきます。

■ <script> ~ </script>の中に記述する

HTMLファイル内にJavaScriptを記述するときは、<script> ~ </script>の中にJavaScriptを記述しなければなりません。

■ 半角文字で記述する

JavaScriptは半角文字で記述するのが基本です。命令文や関数、変数などを全角文字で記述すると、JavaScriptは正しく動作しません。

■ 大文字と小文字の区別

JavaScriptでは、大文字と小文字が別の文字として扱われます。たとえば、「ABC」「abc」「Abc」は、いずれも別の文字として扱われます。HTMLやCSSのように、大文字/小文字を区別なく記述することはできません。

■ 命令文の最後にセミコロン

JavaScriptでは、各文の最後に「;」（セミコロン）を記述し、文の区切りを明確に示す必要があります。なお、CSSの場合と同様に、改行は無視されます。

○ 正しい記述

```
let a = 1;
let b = 2;
let c = a + b;
```

× 間違った記述

```
let a = 1
let b = 2
let c = a + b
```

文末にセミコロンがない

■ 半角スペース、タブ文字の扱い

JavaScriptでは、**連続した半角スペース**や**タブ文字**は無視されます。このため、**行頭にインデント**を設けて、JavaScriptを見やすく記述しても構いません。

```
function tax() {  
  let c = a + b;  
  let taxIn = c * 1.10;  
  alert('料金は' + taxIn + '円です');  
}
```

■ コメント文の記述

JavaScriptを読みやすくするために**コメント**を記すことも可能です。スラッシュを2つ続けて「//」と入力すると、以降の文字がコメント文として無視されます。複数行にわたるコメント文を記述するときは、その範囲を「/* ~ */」で囲んで記述します。

```
// 税込み価格を計算する関数  
function tax() {  
  let c = a + b;  
  let taxIn = c * 1.10; // 税率10%の場合  
  alert('料金は' + taxIn + '円です');  
  /*  
  一時的な確認としてalert()を利用しています。  
  最終的にはWebページ内に表示します。  
  */  
}
```

コメント文

コメント文

コメント文

■ JavaScriptを記述する場所

JavaScriptは<head> ~ </head> または <body> ~ </body> の中に記述します。たいていの場合、</body>の直前に記述するのが一般的です。なお、Webページを表示する前に処理を実行させたいときは、<head> ~ </head> の中にJavaScriptを記述しなければなりません。

演習

演習

- (1) 「sample01-1.html」(P9 ~ 10) に示したHTMLファイルを作成してみましょう。
- (2) 演習(1) で作成したHTMLファイルをWebブラウザに表示し、チェックボックスの操作に応じて合計金額が変化することを確認してみましょう。