

# JavaScriptについて

## JavaScriptとは

JavaScriptは比較的コンパクトですが、一方でとても柔軟性があります。開発者はJavaScript言語のコアをベースに多種多様なツールを作成し、最小限の労力で膨大な様々な機能を利用できるようにしました。

JavaScriptは、Brendan Eich氏によって考案されました動的プログラミング言語であり、ブラウザ標準のスクリプト言語として主要なブラウザのほとんどで実装されています。

JavaScriptは汎用性が高く、初心者にもやさしいプログラミング言語です。ゲーム、2D/3Dのアニメーション、包括的なデータベース駆動型のWebアプリケーションなど、Webサイトに様々なインタラクティブな機能を実装することができます。

### スクリプト言語

スクリプト言語とは、簡易であることを目的として作られたプログラミング言語で、プログラミング初心者も取り組みやすい言語となっています。

### インタプリター型の言語

プログラムを先頭から逐一解析し、コンピュータに理解できる形式に翻訳しながら実行していく言語になります。

コンパイル言語に比べると動作が遅い反面、プログラムの実行にコンパイルする必要がないため手軽に開発を進めることができます。

### 様々な環境での利用

JavaScriptは、もともとブラウザ上で動作することを想定して作られた言語ですが、現在はブラウザ上での用途に留まらず、様々な環境で動作しています。

Node.js	サーバーサイドの用途を中心としたJavaScript実行環境
Windows Script Host Script	Script Host：Windows環境のスクリプト実行環境
Java Platform, Standard Edition	Java言語の実行環境
Android/iOS（WebView）	Webページを表示するための組み込みブラウザ

## JavaScriptの記述場所

JavaScriptをHTMLファイルに組み込むには、script要素を使います。

```
<script>

  console.log( 'Hello JS World' ); // コンソールに「Hello JS World」と表示する

</script>
```

script要素のtype属性には、スクリプトの種類を表す「text/javascript」を指定しますが、HTML5から「text/javascript」が初期値となっているため省略しても構いません。

## 1. HTML内部のJavaScript

### body要素の内部（</body>の直前）に記述

JavaScriptによる処理は、ブラウザがWebページのすべての準備ができてから行こなうべきなので、</body>の直前なら弊害がほぼない。

一般的なブラウザでは、JavaScriptの読み込みや実行が完了するまで、<script>要素以降の描画を行わないため、ページ描画の高速化が図れる。

### head要素の内部に記述

ライブラリや関数定義ファイルなどは、処理を記述したscript要素より先に記述しておく必要があるため、head要素内部で事前に読み込むことが多いです。



</body> 直前のscript要素の記述順序が守られておれば、head要素の内部に限らず記述することは可能です。

## 2. 外部のJavaScript

CSSファイルのように複数のHTMLファイルに同じJavaScriptを組み込みたい場合は、拡張子が「.js」のJavaScriptファイルを用意して、HTMLファイル内のscript要素でsrc属性を使って読み込みます。

```
<!DOCTYPE html>
<html lang="ja">
<head>
  <meta charset="UTF-8">
  <title>Web演習2 - サンプル1</title>
</head>
<body>
  <h1>JavaScript</h1>
  <section>
    <h2>JavaScriptとは</h2>
    <p>JavaScriptは...</p>
  </section>
  <script src="sample01.js"></script>
</body>
</html>
```

```
{
  console.log( 'Hello JS World' ); // コンソールにHello JS
}
```

src属性で外部JavaScriptに置き換えているため、script要素内に記述したJavaScriptのプログラムコードは上書きされてしまいます。

もし、外部のJavaScriptファイルのコードと内部のJavaScriptを併用する場合は、script要素を複数用意します。

```
<script src="https://cdn.jsdelivr.net/npm/vue@2.6.14/dist/vue.js"></script>
<script>
  console.log( 'Internal JS' );
</script>
```

### 3. インラインJavaScriptハンドラー（非推奨）

要素のイベント属性に直接JavaScriptを記述することもできますが、HTMLの汚染や・JavaScriptコードが分散して非効率となります。

## 開発ツールの活用

モダンブラウザとよばれる一般的に利用されているブラウザには、Webサイト開発者向けのデベロッパーツールが標準で実装されており、クライアントサイドの開発（HTML/CSS/JavaScript）には、必須となるツールになります。

ブラウザの種類によって、デベロッパーツールを表示するメニューは違うため、利用するブラウザごとのデベロッパーツールの表示方法を確認しておきましょう。



開発者ツール