

🏠 → JavaScript 言語 → 導入

📅 2022年1月11日

JavaScript 入門

JavaScript について、何が特別なのか、それを使ってできることや他のどの技術と上手くやるのか見てみましょう。

JavaScript とは？

JavaScript は当初 “Web ページを活かすため” に作られました。

この言語のプログラムは スクリプト と呼ばれます。それらはHTMLの中に書かれ、ページが読み込まれると自動的に実行されます。

スクリプトはプレーンテキストとして提供され、実行されます。特別な準備や、実行するためのコンパイルは必要ありません。

この点において、JavaScript は Java とは大きく異なります。

i なぜ JavaScript と言うのでしょうか？

JavaScript が作られたとき、当初は別の名前を持っていました: “LiveScript”。しかし当時 Java 言語が非常に人気であったため、Java の “弟” として新しい言語を位置づけるのが良いと判断されました。しかし、それ以降の進化により、JavaScript は完全に独立した言語になり、ECMAScript と呼ばれる独自の仕様を持ちました。現在 Java とは全く関係ありません。

現在、JavaScript はブラウザだけでなく、サーバ上でも実行することができます。実際には JavaScript エンジン と呼ばれるプログラムが存在するデバイスであれば実行することができます。

ブラウザにはエンジンが組み込まれており、“JavaScript 仮想マシン” と呼ばれる場合があります。

異なるエンジンは異なる “コードネーム” を持っています。例えば:

- V8 – Chrome と Opera.
- SpiderMonkey – Firefox.
- ...IEはバージョンによって “Trident” や “Chakra”, Microsoft Edge 用の “ChakraCore”, Safari 用の “Nitro” や “SquirrelFish” 等のように他のコードネームもあります。

これらの用語は、インターネット上の開発者の記事で使用されているため、覚えておくの良いでしょう。たとえば、“ある機能 X がV8でサポートされている” と言った場合、おそらく ChromeとOperaで動作します。

i エンジンはどうに動く？

エンジンは複雑ですが、基本は単純です。

1. エンジン (ブラウザの場合は組み込まれています) はスクリプトを読み("パース")ます。
2. その後、スクリプトを機械語に変換("コンパイル")します。
3. 機械語が実行されます。非常に早く動作します。

エンジンは処理の各ステップで最適化を行います。実行時にコンパイルされたスクリプトも見ており、そこを流れるデータを分析し、それに基づいて機械語を最適化します。最終的に、スクリプトはとても速く実行されます。

ブラウザ内のJavaScriptができることは？

モダンなJavaScriptは "安全な" プログラミング言語です。それは、メモリやCPUのような低レベルのアクセスは提供しません。なぜなら、当初はそれらを必要としないブラウザ用に作成されたものだからです。

JavaScript の機能は、実行される環境に大きく依存します。例えば、Node.js では、JavaScriptが任意のファイルを読み書きしたりできる機能をサポートしています。

ブラウザ内のJavaScriptは、Webページの操作、ユーザーやWebサーバとのやり取りに関する様々なことを実行できます。

たとえば、ブラウザ内のJavaScriptは次のようなことが可能です:

- 新たなHTMLをページに追加したり、既存のコンテンツの変更やスタイルの変更をする。
- ユーザーの操作(マウスのクリック、ポインタの動き、キーの押下など)に反応する。
- リモートサーバへネットワーク越しのリクエストを送り、ファイルのダウンロードやアップロードをする(AJAX や COMET と呼ばれる技術)。
- クッキーの取得と設定、訪問者への質問やメッセージを表示する。

- ・クライアント側でデータを記憶する("ローカルストレージ")。

ブラウザ内のJavaScriptで出来ないことは？

ブラウザでは、JavaScriptの機能はユーザの安全のために制限されています。その目的は、悪意のあるWebページがプライベートな情報へアクセスしたり、ユーザデータへ危害を加えることを防ぐことです。

制限の例として、次のようなものがあります：

- ・ Webページ上のJavaScriptは、ハードディスク上の任意のファイルの読み書きや、それらのコピー、プログラムの実行をすることができません。OSのシステム機能に直接アクセスすることはできません。

現代のブラウザは、ファイルを扱うことはできますがアクセスは制限されており、ブラウザウィンドウへのファイルの "ドロップ" や、`<input>` タグを経由したファイル選択と言ったユーザの特定の操作のみを提供しています。

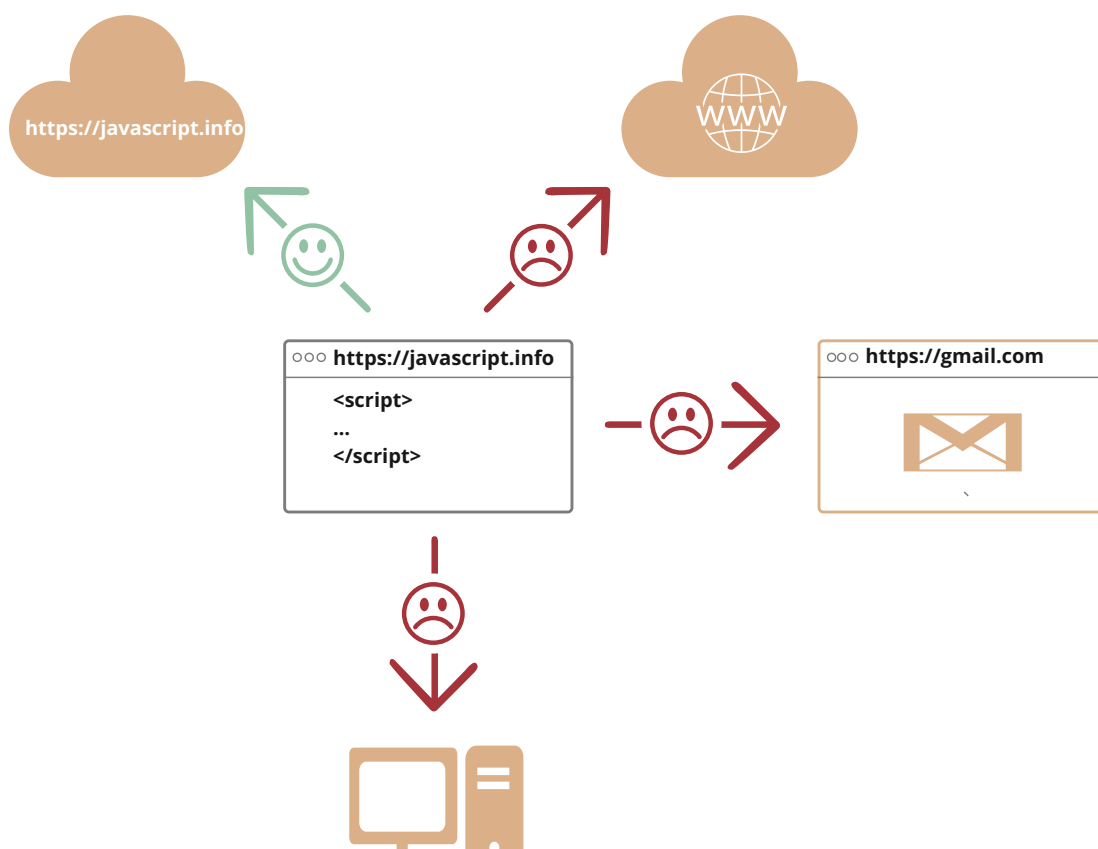
カメラ/マイクやその他デバイスとやり取りする方法はありますが、ユーザの明示的な許可が求められます。したがって、JavaScriptが有効なページがWebカメラを密かに有効にしたり、それを利用して利用者の周囲を観察したり、NSA に情報を送信すると言ったことはできません。

- 異なるタブやウィンドウは一般的にお互いについて知りません。時々、例えばあるウィンドウがJavaScriptを利用して別のウィンドウを開くケースがあります。しかし、この場合においても、あるページからのJavaScriptは、別のサイト（異なるドメイン、プロトコル、ポート）からのものである場合は、アクセスすることはできません。

これは“Same Origin Policy”と呼ばれています。回避するためには、*両方のページ*にデータ交換を行うための特別なJavaScriptコードを含める必要があります。

この制限もユーザの安全のためです。ユーザが開いた `http://anysite.com` というサイトのページは、URL `http://gmail.com` の別のブラウザのタブにアクセスし、そこから情報を盗むことはできません。

- JavaScriptはネットワークを介して、現在のページがきたサーバと簡単にやり取りすることができます。しかし、他のサイト/ドメインからデータを受信することは制限されています。可能ですが、リモート側からの明示的な同意(HTTPヘッダで表現)が必要になります。繰り返しますが、これらは安全上の制限です。





JavaScriptがブラウザ外で使われる場合はこのような制限は存在しません。たとえば、サーバ上です。また、現代のブラウザは、より拡張されたアクセス権限を必要とするプラグイン/拡張機能を利用することもできます。

なにがJavaScriptを特別なものに行っている？

JavaScriptには少なくとも3つの素晴らしいことがあります:

- HTML/CSSとの完全な統合
- シンプルなことはシンプルに
- すべてのメジャーブラウザでサポートされており、デフォルトで有効

JavaScriptは、これら3つのことを組み合わせた唯一のブラウザテクノロジーです。

それがJavaScriptをユニークなものにしています。だからこそ、ブラウザインターフェイスを作成するための最も普及しているツールとなっています。

とはいえ、JavaScriptを利用してサーバやモバイルアプリケーションなどを作成することもできます。

JavaScriptを“覆う”言語

JavaScriptの構文は、すべての人のニーズにマッチしている訳ではありません。人によって求める機能は異なります。

プロジェクトや要件はそれぞれ異なるため、それは自然なことです。

そのため、最近では新しい言語が数多く登場しています。これらはブラウザで実行する前にJavaScriptにトランスパイル(変換)されます。

最新のツールは非常に高速にトランスパイルでき、透過的です。開発者が別の言語でコードを作成し、それを自動変換することができます。

これは、そのような言語の例です:

- CoffeeScript はJavaScriptの "シンタックスシュガー"です。より短い構文を導入し、より簡潔でクリアなコードを書くことができます。たいてい、Ruby 開発者は好きです。
- TypeScript は "厳密なデータ型指定" の追加に焦点をあてています。それは複雑なシステムの開発とサポートを簡素化するためです。これは Microsoftにより開発されています。
- Flow もデータ型定義を追加しますが、その方法は異なります。Facebook により開発されました。
- Dart はブラウザ以外の環境（モバイルアプリのような）で動作する独自のエンジンを持ったスタンドアローンな言語ですが、JavaScript へトランスパイルすることもできます。Googleによって開発されました。
- Brython は JavaScript への Python トランスパイラで、JavaScript を使用することなく、純粋な Python

でアプリケーションを作成することができます。

他にもあります。もちろん、上記のような言語を利用する予定だとしても、実際に行われていることを本当に理解するためにJavaScriptは知っておくのがよいです。

サマリ

- JavaScriptは当初ブラウザ用の言語として作られました。しかし今はその他の多くの環境で利用されています。
- 現時点では、JavaScriptはHTML/CSSと完全に統合し、最も広く採用されたブラウザ言語として、独立した地位にいます。
- JavaScriptに“トランスパイル”し、特定の機能を提供する多くの言語があります。JavaScriptをマスターした後、少なくとも簡単にでも目を通しておくことをお勧めします。

前のレッスン

次のレッスン

シェア  

 チュートリアルマップ