**超絶初心者のためのフロント入門（HTML、CSS、JavaScript）**

[HTML](https://qiita.com/tags/html)[CSS](https://qiita.com/tags/css)[JavaScript](https://qiita.com/tags/javascript)[新人プログラマ応援](https://qiita.com/tags/%e6%96%b0%e4%ba%ba%e3%83%97%e3%83%ad%e3%82%b0%e3%83%a9%e3%83%9e%e5%bf%9c%e6%8f%b4)

More than 3 years have passed since last update.

**1. はじめに**

**1-1. 技術の概要**

エンジニアの技術は、かなりざっくり言うと、

* フロントエンド　→　見える部分。

　　（例）HTML、CSS、JavaScriptなど

* サーバサイド　→　アプリケーションを動かす中身の部分。

　　（例）Java、PHP、Ruby、C、C++、Perlなど

* インフラ　→　Webアプリケーションを動かすための環境構築など。

　　（例）Webサーバ構築、DBサーバ構築、DNSサーバ構築など

の3つに類別できる。

その中で、プログラミング未経験者が最も手をつけやすいのが、フロント（HTML、CSS、JavaScript）の技術である。

※　上記は、「Webアプリケーション開発」のお話です。  
※　「ネイティブ開発」ではObjective-C/Swift/AndroidJava/Cocos2d-x/Unityなどがよく使われる。

**1-2. フロントの概要**

フロントの技術を学ぶために準備すべきなのは、

* webブラウザ　→　IE、Google Chrome、Firefox、Safariなど
* エディタ　→　メモ帳、テキストエディット、サクラエディタ、mi、Sublime Text、vimなど

だけでよい。HTML、CSS、JavaScriptの住み分けは、

* HTML　→　「構造」を記述。マークアップ。Hyper Text Markup Language。
* CSS　→　「装飾」を記述。スタイルシート。Cascading Style Sheet。
* JavaScript　→　「動的な処理」を記述。JSとも言う。

のようになっており、プログレッシブ・エンハンスメントという思想に基づくと、HTML > CSS > JavaScriptの順に重要であるとされている。（HTMLが1番重要）

**2. HTML**

**2-1. 最小限の記述**

拡張子は.html。  
（ファイル名が、test.htmlのように最後が.htmlということ）

<!DOCTYPE html>

<html>

<head>

（headタグで囲まれた部分（head要素）には「付加情報」を記述）

</head>

<body>

（bodyタグで囲まれた部分（body要素）には「コンテンツ」を記述）

</body>

</html>

* 1行目　→　文書型宣言。おまじないだと思えばOK。

　　（例）<!DOCTYPE html>だと、HTML5を利用するよ！という意味。

* タグとは、<と>で囲まれたもののこと。

　　（例）headタグとは、<head>のこと。

* 要素とは、開始タグ<>と終了タグ</>で囲まれた全体を指す。タグと要素を混同しないように！

　　（例）head要素とは、<head>　・・・　</head>のこと。

* htmlタグ<html>　→　HTMLファイルの中身は、文書型宣言以外をhtmlタグで囲むべし。

1番大事なのは、HTMLファイルには必ずheadとbodyがあるよ！ってこと。

**2-2. HTML記述例**

<!DOCTYPE html>

<html>

<head>

<title>ここにページのタイトル書くよ！</title>

<meta charset="utf-8" />

</head>

<body>

<h1>1. h1タグの中身は、見出し1だよ！1番大きい見出しだよ！</h1>

<h2>1-1. h2タグの中身は、見出し2だよ！</h2>

<p>pタグの中には段落の塊を記述するよ！

<u>アンダーバーはuタグで囲むよ！</u>

<i>斜体はiタグで囲むよ！</i>

改行したいときは<br />って書くよ！

</p>

<h3>1-1-1. h3タグは見出し3だよ！</h3>

<p>見出しタグはh4,h5,h6とかもあるよ！h1タグが存在しないのにh2とかh3を使うのは良くないよ！</p>

<div>

<ul>

<li>りんご</li>

<li>バナナ</li>

<li>もも</li>

</ul>

</div>

<p>liタグはリストを書く時使う。liタグはulタグ（またはolタグ）で囲むことで使えるよ！</p>

</body>

</html>

　↑  
これをコピペしてtest.htmlみたいな感じで.htmlという拡張子で保存して、ブラウザで見てみるといいかも！

**3. CSS**

**3-1. CSSの書き方は3通り**

CSSには書き方が3通りある。

* 「style属性」として記述（pタグなど「タグの中」に記述）
* 「styleタグ」の中に記述（HTMLファイルの「headタグの中」に記述）
* 外部ファイルに記述（linkタグで呼び出す）

**3-1-1. 「style属性」として記述**

HTMLファイル内の特定の要素に直接CSSをあてる方法。  
記述例は以下の通り。

<!DOCTYPE html>

<html>

<head>

</head>

<body>

<p style="color: blue;">Hello World:D</p>

</body>

</html>

* style="color: blue;"をpタグの中に書くことで、Hello World:Dの部分が青文字になる。
* 確かめたかったら、コピペして保存して、ブラウザで開いてみよう！保存するときの拡張子は.htmlだよ！

**3-1-2. 「styleタグ」の中に記述**

HTMLファイル内のheadタグの中に、style要素としてCSSを記述する方法。  
記述例は以下の通り。

<!DOCTYPE html>

<html>

<head>

<style type="text/css">

p {

color: yellow;

background-color: black;

}

</style>

</head>

<body>

<p>Hello World:D</p>

</body>

</html>

* <style type="text/css">と</style>で囲まれた部分にCSSを記述。
* type="text/css"と書くことで、scriptタグの中身がCSSファイルであることを明記。

CSS部分の書き方は、

タグ名 {

プロパティ名: 値

}

となる。classやidは、

.class名 {

プロパティ名: 値

}

#id名 {

プロパティ名: 値

}

のように、classには.、idには#をつけることでCSSを記述する。

**3-1-3. 外部ファイルに記述**

上記2つの方法では、HTMLファイルの中にCSSを記述していたが、3つ目のこの方法では「外部ファイル」としてCSSを記述する。  
拡張子は.css。記述例は以下の通り。

<!DOCTYPE html>

<html>

<head>

<link href="test.css" rel="stylesheet" type="text/css">

</head>

<body>

<p>Hello World:D</p>

</body>

</html>

p {

color: yellow;

background-color: black;

}

* HTMLファイル内ではlinkタグで外部CSSファイルを呼び出す。
* href="test.css"と書くことで、呼び出すファイル名を記述。この場合、呼び出すCSSファイルの名前はtest.css。
* rel="stylesheet" type="text/css"は、CSSを呼び出す時のおまじないと思えばOK！

**4. JavaScript**

**4-1. JavaScriptの書き方は2通り**

JavaScriptには書き方が2通りある。

* 「scriptタグ」の中に記述
* 外部ファイルに記述（scriptタグで呼び出す）

**4-1-1. 「scriptタグ」の中に記述**

CSSの3-1-2. 「styleタグ」の中に記述と似た感じで、HTMLファイルの中にscript要素として記述する方法。  
記述例は以下の通り。

<!DOCTYPE html>

<html>

<head>

</head>

<body>

<p>Hello World:D</p>

<script type="text/javascript">

alert("Hello JavaScript:D");

</script>

</body>

</html>

* <script type="text/javascript">と</script>で囲まれた部分にJavaScriptを記述。
* type="text/javascript"と書くことで、scriptタグの中身がJavaScriptファイルであることを明記。
* alert("Hello JavaScript:D");により、Hello JavaScript:Dと書かれたアラートがポップアップで出てくる。
* script要素を書く場所は、headタグの中でもbodyタグの中でもどちらでも良い。

**4-1-2. 外部ファイルに記述**

「外部ファイル」としてJavaScriptを記述する。  
拡張子は.js。記述例は以下の通り。

<!DOCTYPE html>

<html>

<head>

<script type="text/javascript" src="test.js"></script>

</head>

<body>

<p>Hello World:D</p>

</body>

</html>

alert("Hello JavaScript:D");

* HTMLファイル内ではscriptタグで外部CSSファイルを呼び出す。
* src="test.js"と書くことで、呼び出すファイル名を記述。この場合、呼び出すJavaScriptファイルの名前はtest.js。
* type="text/javascript"と書くことで、scriptタグで呼び出すファイルがJavaScriptファイルであることを明記。

**4-2. JavaScript記述例**

<!DOCTYPE html>

<html>

<head>

<title>ここにページのタイトル書くよ！</title>

<meta charset="utf-8" />

<style type="text/css">

.foo-before {

background-color: white;

color: black;

}

.foo-after {

background-color: black;

color: yellow;

}

</style>

</head>

<body>

<p id="id\_1">1. JS側にonclick書いてクリックイベント発火。</p>

<p id="id\_2">2. JSでは無名関数も使えるよ！</p>

<p onclick="pClick()">3. HTMLでonclick属性書いた。クリックしてみて！</p>

<p id="id\_3" class="foo-before">4. クリックするたび色変わるよ！（クラスの入れ替えをjQueryで実装）</p>

<p id="id\_4">5. クリックするとCSS追加されるよ！</p>

<script type="text/javascript">

// 変数定義（「//」から始まる行はコメントアウト）

var id\_1 = document.getElementById("id\_1");

var id\_2 = document.getElementById("id\_2");

var id\_3 = document.getElementById("id\_3");

// id\_1（id\_2）をクリックしたら、functionの中身を実行

id\_1.onclick = function id1Click() {

alert("id\_1クリックしたよ！");

}

id\_2.onclick = function () {

alert("id\_2クリック！関数名は書いても書かなくてもOK！");

}

// HTML側にクリックイベント記述

function pClick() {

alert("pタグクリック！HTMLでonclick属性書いたよ！");

}

</script>

<script type="text/javascript">

var stylesheet = document.styleSheets.item(0);

// 上で定義したid\_4はここでも使えるよ！

id\_4.onclick = function () {

//第2引数は、「CSSの何番目のルールにするか」。0始まりの番号。

stylesheet.insertRule("#id\_4 {font-size: 20px;}", 0);

stylesheet.insertRule("#id\_4 {background-color: pink;}", stylesheet.cssRules.length);

}

</script>

<!-- これでjQuery使える。ただしネットつながってれば。 -->

<script type="text/javascript" src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.0/jquery.min.js"></script>

<script type="text/javascript">

// $関数はjQueryの定番！

$('#id\_3').click(function() {

this.className = (this.className === 'foo-before') ? 'foo-after' : 'foo-before';

});

document.writeln("<p>6. JSでHTML記述！（リロードするたび少し表示が遅れてるけど、わかるかな？）</p>");

document.writeln("<p>7. URLは、" + document.URL + "だよ！</p>");

document.writeln("<p>8. このページのtitleは、「" + document.title + "」だよ！</p>");

</script>

</body>

</html>

　↑  
これをコピペしてtest.htmlみたいな感じで.htmlという拡張子で保存して、ブラウザで見てみるといいかも！

※おまけ（jQuery）  
▼クリック時、非同期でHTTPリクエストを投げる方法  
<http://qiita.com/shuntaro_tamura/items/d1b52a3a35440daf84ea>  
▼便利なjQueryメソッドまとめ  
<http://qiita.com/shuntaro_tamura/items/94fe7e7a7dc797999287>