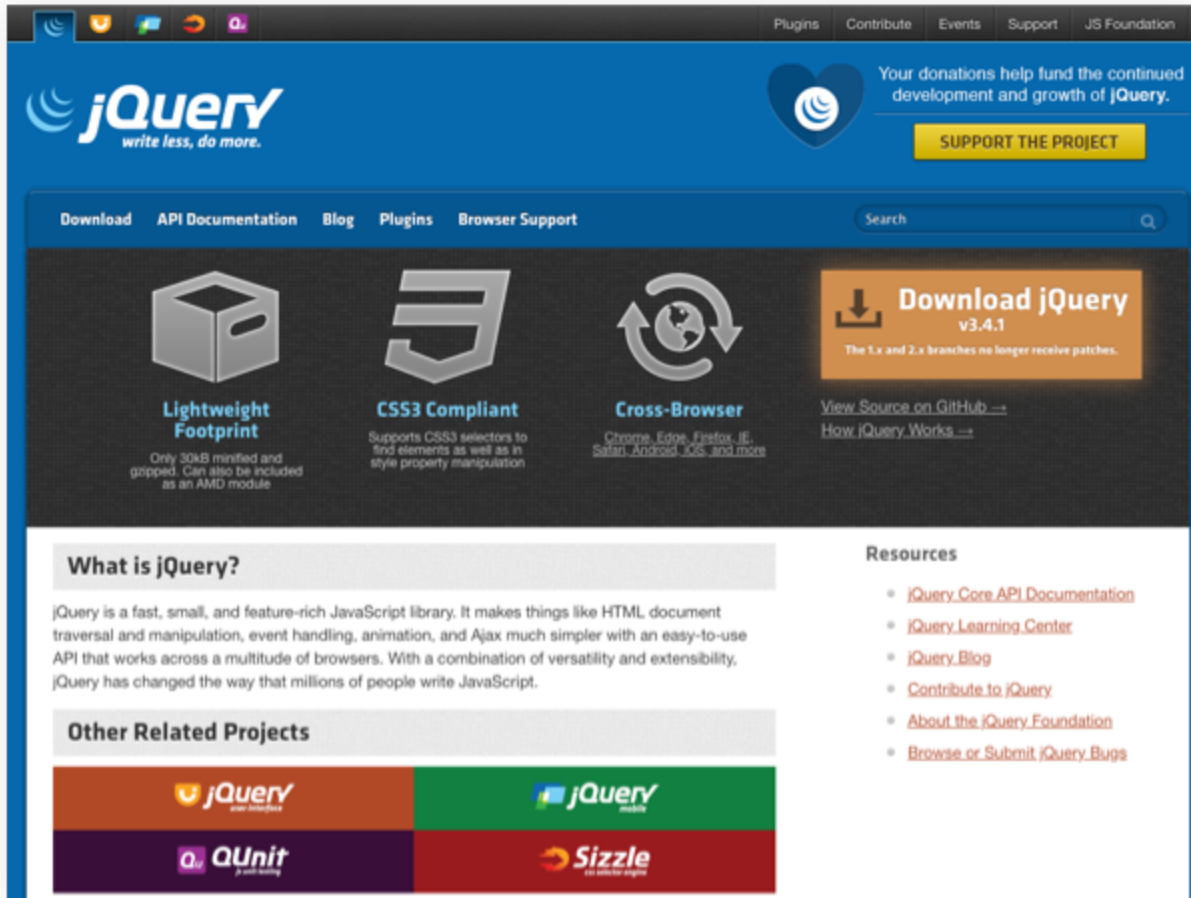


jQuery

jQueryとは

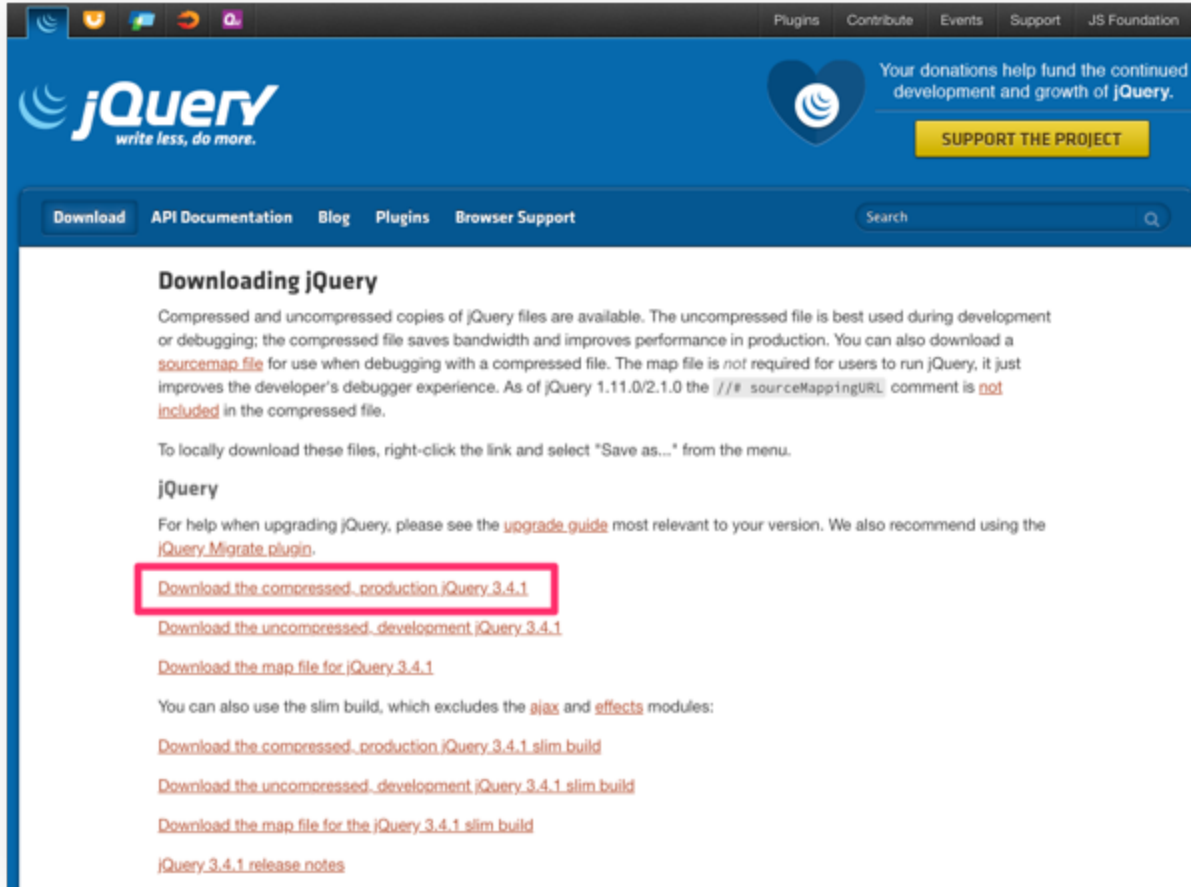
<https://jquery.com/>

JavaScriptの構文を簡略化して、CSS風に記述することができるライブラリ



jQueryのダウンロード方法1

<https://jquery.com/download/>



The screenshot shows the jQuery website's download page. At the top, there's a navigation bar with links for Plugins, Contribute, Events, Support, and JS Foundation. Below this is the jQuery logo with the tagline "write less, do more." and a heart icon with the text "Your donations help fund the continued development and growth of jQuery." and a "SUPPORT THE PROJECT" button. A secondary navigation bar contains links for Download, API Documentation, Blog, Plugins, and Browser Support, along with a search bar. The main content area is titled "Downloading jQuery" and explains the availability of compressed and uncompressed files. It provides instructions for downloading and mentions the sourceMappingURL comment. A red box highlights the link "Download the compressed, production jQuery 3.4.1". Other links include "Download the uncompressed, development jQuery 3.4.1", "Download the map file for jQuery 3.4.1", and links for the slim build and release notes.

Plugins Contribute Events Support JS Foundation

jQuery
write less, do more.

Your donations help fund the continued development and growth of jQuery.

SUPPORT THE PROJECT

Download API Documentation Blog Plugins Browser Support Search

Downloading jQuery

Compressed and uncompressed copies of jQuery files are available. The uncompressed file is best used during development or debugging; the compressed file saves bandwidth and improves performance in production. You can also download a [source map file](#) for use when debugging with a compressed file. The map file is *not* required for users to run jQuery, it just improves the developer's debugger experience. As of jQuery 1.11.0/2.1.0 the `/** sourceMappingURL` comment is *not included* in the compressed file.

To locally download these files, right-click the link and select "Save as..." from the menu.

jQuery

For help when upgrading jQuery, please see the [upgrade guide](#) most relevant to your version. We also recommend using the [jQuery Migrate plugin](#).

[Download the compressed, production jQuery 3.4.1](#)

[Download the uncompressed, development jQuery 3.4.1](#)

[Download the map file for jQuery 3.4.1](#)

You can also use the slim build, which excludes the [ajax](#) and [effects](#) modules:

[Download the compressed, production jQuery 3.4.1 slim build](#)

[Download the uncompressed, development jQuery 3.4.1 slim build](#)

[Download the map file for the jQuery 3.4.1 slim build](#)

[jQuery 3.4.1 release notes](#)

右クリックで「リンク先を別名で保存」



jsフォルダを作成してダウンロードしたファイルを保存する

HTMLにjQueryを読み込む

index.html

```
<footer/>  
  
<script src="js/jquery-3.4.1.min.js"></script>  
</body>  
</html>
```

body の閉じタグの前に script タグを作成して読み込む

jQueryのダウンロード方法2

HTMLにjquery.comからサーバー経由でjQueryを読み込む

index.html

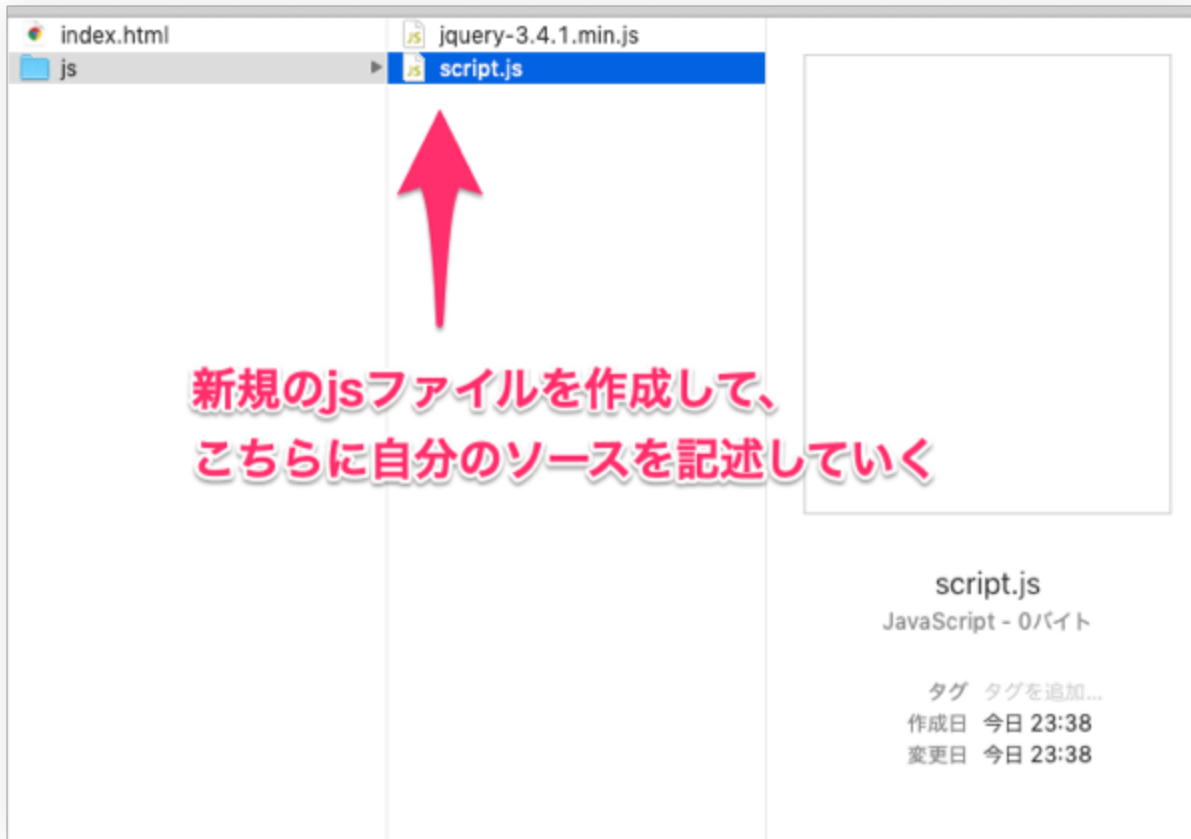
```
<footer/>  
  
<script src="https://code.jquery.com/jquery-3.4.1.min.js"></script>  
</body>  
</html>
```

body の閉じタグの前に script タグを作成して https:~ から読み込む

インターネットに接続していないと使用できない。

JavaScriptの記述場所

jsファイルを新規作成



新規jsファイルの読み込み方

```
<footer/>  
<script src="https://code.jquery.com/jquery-3.4.1.min.js"></script>  
<script src="js/script.js"></script>  
</body>  
</html>
```

ライブラリ（jQuery）の読み込みが先。自分のソースが後

jQueryの書き方

jQueryの記述パターンはこの3つ

1. HTMLの読み込み待ち read
2. HTMLの操作 selector
3. イベントの扱い on

1 HTMLの読み込み待ち ready

JavaScriptを実行する場合、ブラウザがHTMLタグを読み込み終わってから処理を始めないと、操作対象が見つからなくなる。

script.js

```
jQuery(document).ready(function(){  
    //HTMLの読み込みが終わるとこの中が実行される  
});
```


readyの短い書き方

script.js

```
$(function(){  
    //HTMLの読み込みが終わるとこの中が実行される  
});
```

'\$' は jQueryの略称

jQuery セレクタ

JavaScriptからHTMLにアクセスする

HTMLタグ / id / classの取得

index.html

```
<h1>ドリンクメニュー</h1>  
<h2 id="title">本日のコーヒー</h2>  
<p class="type">ホット</p>
```

script.js

```
$( 'h1' ); //h1タグを取得  
$( '#title' ); //h2タグを取得  
$( '.type' ); //pタグを取得
```

'\$' はjQueryの略

親要素から子要素の取得

index.html

```
<div class="container">  
  <p>本日のコーヒー</p>  
</div>
```

script.js

```
$('.container p');  // .containerの中の`p`タグ
```

半角スペースで子要素にアクセスする

find

script.js

```
$('.container').find('p'); // .containerの中の#title
```

.find() で子要素を探す

CSSの操作

`css()` の命令でCSSのプロパティを扱うことができる

index.html

```
<h1>ドリンクメニュー</h1>
<h2 id="title">本日のコーヒー</h2>
<p class="type">ホット</p>
```

script.js

```
$('#h1').css('color', '#FF0000'); //文字色の変更
$('#title').css('background-color', '#EFEFEF'); //背景色の変更
$('.type').css({
  'font-size': '15px',
  'color': '#00FF00',
  'background-color', '#EFEFEF'
});
```

liの適用

index.html

```
<ul id="menu">
  <li>ブレンドコーヒー</li>
  <li>カプチーノ</li>
  <li>カフェラテ</li>
  <li>キャラメル マキアート</li>
</ul>
```

script.js

```
// (0から数えて) 2番目の文字色を変更
$('#menu li').eq(2).css('color', '#00FF00');
```


イベント on

イベントとは

ユーザーの挙動などによって、何か要素を変化させる場合、イベントという処理の流れが必要になる

イベントの種類

- マウスでクリックする
- マウスでダブルクリックする
- マウスが動く
- スマホで画面をタップする
- スマホで画面上を動く
- スマホで画面上から指が離れる
- キーボードのキーが押される
- ウィンドウが閉じられる
- フォームに値が入力される
- データを読み込み始める
- データを読み込み終わる

on

`on` を要素に付与することで、イベントを待つ状態になる。

```
要素.on(イベントの種類, 呼び出される関数);
```

イベントの種類

- click クリック
- scroll スクロール
- mouseover マウスオーバー
- mouseout マウスアウト
- load ロード

click イベント

```
// #btnをクリックしたら
$('#btn').on('click', function(){
    //この中が実行される
    $('.menu').css('color', '#00FF00');
});
```

scroll

ユーザーがスクロールした際に呼ばれるイベント

```
$(window).on('scroll',function() {  
    // スクロール値  
    var dy = $(this).scrollTop(); // Y座標  
  
});
```

属性 attribute

属性とは

```

```

- src 属性
- alt 属性

| src 属性の値が main.jpg 、 alt 属性の値が メイン画像

attr 属性値の変更

```
attr('属性名','値');
```

attrの中に属性名と値を入れると属性値を変更できる

index.html

```
<div class="main">  
    
</div>
```

script.js

```
$('.main').find('img').attr('src','sub.jpg');  
// src属性のmain.jpgをsub.jpgに変更できる
```

HTMLタグのクラス変更

addClass

HTMLタグにクラスを付与する

```
addClass('クラス名');
```

html

```
<ul class="tab-menu">
  <li>Get Started</li>
  <li>Tutorial</li>
  <li>Guidelines</li>
</ul>
```

js

```
$('.tab-menu li').on('click', function () {
  $(this).addClass('active');
});
```

liタグをクリックすると

html

```
<ul class="tab-menu">  
  <li class="active">Get Started</li>  
  <li>Tutorial</li>  
  <li>Guidelines</li>  
</ul>
```

class="active" が追加される

removeClass

HTMLタグからクラスを削除する

```
removeClass('クラス名');
```

toggleClass

HTMLタグからクラスを付与と削除を交互に行う

```
toggleClass('クラス名');
```

