JavaScript

# 「JavaScript」とは

動的なWebページを作成する事のできるプログラミング言語です。

通常はブラウザ上で実行されます。

JavaScriptを使うと、ユーザーのアクションに応じたコンテンツの表示の他、ブラウザ上で表示される地図やグラフィックアニメーションなども表示する事ができます。

「メッセージボックスの表示」や「HTMLコンテンツの操作」、「カレンダー」、「地図」、「ゲーム」など、様々な動的コンテンツを作成できる言語です。

開発当初のJavaScriptはクライアントサイドで実行され動的Webページを作成していましたが、現在ではサーバーサイドで動作する「Node.js」があり、かなりいろんな動き方ができるようになってきました。

# JavaScriptを記述方法

HTMLページの中に JavaScript のコードを記述しておくと、ブラウザで HTMLページを開いた時に ページ内に記述されたコードを実行することができます。

# JavaScriptリファレンス

## windowまわり

ウィンドウ名.alert()・・・［OK］ボタンのみの警告ダイアログを表示する

ウィンドウ名.confirm()・・・［OK］と［キャンセル］ボタンを持つ確認ダイアログを表示する

ウィンドウ名.prompt()・・・文字入力ダイアログを表示する

ウィンドウ名.defaultStatus・・・ステータスバーにデフォルトで文字列を表示する

ウィンドウ名.status・・・ステータスバーに（一時的に）文字列を表示する

ウィンドウ名.length・・・ウィンドウ内のフレーム数を参照する

ウィンドウ名.open()・・・新しいウィンドウを開く

ウィンドウ名.close()・・・ウィンドウを閉じる

ウィンドウ名.closed・・・ウィンドウが閉じているかどうかを返す

ウィンドウ名.name・・・ウィンドウ名を参照する

ウィンドウ名.opener・・・オープン元の親ウィンドウを参照する

ウィンドウ名.resizeBy()・・・現在のウィンドウに指定した幅・高さを追加する

ウィンドウ名.resizeTo()・・・ウィンドウの幅・高さを指定する

ウィンドウ名.moveBy()・・・ウィンドウ表示位置を指定距離だけ移動する

ウィンドウ名.moveTo()・・・ウィンドウ表示位置を指定座標へ移動する

ウィンドウ名.scrollBy()・・・スクロール位置を指定距離だけ移動する

ウィンドウ名.scrollTo()・・・スクロール位置を指定座標へ移動する

ウィンドウ名.scroll()・・・スクロール位置を指定座標へ移動する

ウィンドウ名.focus()・・・ウィンドウにフォーカスをあてる（前面へ移動）

ウィンドウ名.blur()・・・ウィンドウからフォーカスをはずす（背面へ移動）

ウィンドウ名.outerWidth・・・ブラウザウィンドウの外側の幅を返す

ウィンドウ名.outerHeight・・・ブラウザウィンドウの外側の高さを返す

ウィンドウ名.innerWidth・・・ブラウザウィンドウのコンテンツ領域の幅を返す

ウィンドウ名.innerHeight・・・ブラウザウィンドウのコンテンツ領域の高さを返す

ウィンドウ名.setInterval()・・・一定時間が経過するごとに処理を実行する

ウィンドウ名.setTimeout()・・・一定時間経過後に処理を一回だけ実行する

ウィンドウ名.clearInterval()・・・clearInterval()でセットしたタイマーを解除する

ウィンドウ名.clearTimeout()・・・setTimeout()でセットしたタイマーを解除する

## documentまわり

document.URL・・・ドキュメントのURLを設定・参照する

document.location・・・ドキュメントのURLを設定・参照する

document.domain・・・ドメイン名を参照する

document.title・・・ドキュメントのタイトルを設定・参照する

document.referrer・・・リファラー（リンク元URL）を参照する

document.lastModified・・・最終更新日を参照する

document.cookie・・・クッキーを保存・参照する

document.open()・・・ドキュメントの出力を開始する

document.close()・・・ドキュメントの出力を終了する

document.write()・・・ドキュメントに文字列を書き出す

document.writeln()・・・ドキュメントに文字列を書き出して改行する

document.getSelection()・・・選択された文字列を返す

document.bgColor・・・背景色を設定・参照する

document.fgColor・・・文字色を設定・参照する

document.linkColor・・・リンクの文字色を設定・参照する

document.alinkColor・・・読み込み中リンクの文字色を設定・参照する

document.vlinkColor・・・読み込み済リンクの文字色を設定・参照する

## frameまわり

frames.length・・・フレーム数を返す

フレーム名.location・・・フレームのURLを設定・参照する

フレーム名.name・・・フレーム名を設定・参照する

top.フレーム名.操作内容・・・トップフレームを操作する

parent.フレーム名.操作内容・・・親フレームを操作する

self.操作内容・・・自フレームを操作する

フレーム名.alert()・・・警告ダイアログを表示する

フレーム名.blur()・・・フレームからフォーカスをはずす

フレーム名.confirm()・・・確認ダイアログを表示する

フレーム名.focus()・・・フレームにフォーカスをあてる

フレーム名.prompt()・・・文字入力ダイアログを表示する

## historyまわり

history.length・・・ブラウザで表示した履歴の数を返す

history.back()・・・一つ前のページへ戻る

history.forward()・・・一つ先のページへ進む

history.go()・・・指定数だけ前後のページへ移動する

## locationまわり

location.href・・・現在ページURLを参照する

location.protocol・・・現在ページURLのプロトコルを参照する

location.hostname・・・現在ページURLのホスト名を参照する

location.host・・・現在ページURLのホスト情報を参照する

location.port・・・現在ページURLのポート番号を参照する

location.pathname・・・現在ページURLのパス名を参照する

location.search・・・現在ページURLのサーチ情報を参照する

location.hash・・・現在ページURLのハッシュ部分を参照する

location.reload()・・・ページをリロード（再読み込み）する

location.replace()・・・指定したページへ移動する

## linkまわり

document.links.length・・・ドキュメント内にあるリンクの数を参照する

document.link.href・・・リンク先URLを参照する

document.link.target・・・リンクのターゲット属性を参照する

document.link.protocol・・・リンク先URLのプロトコルを参照する

document.link.hostname・・・リンク先URLのホスト名を参照する

document.link.host・・・リンク先URLのホスト情報を参照する

document.link.port・・・リンク先URLのポート番号を参照する

document.link.pathname・・・リンク先URLのパス名を参照する

document.link.search・・・リンク先URLのサーチ情報を参照する

document.link.hash・・・リンク先URLのハッシュ部分を参照する

## formまわり

document.フォーム名.action・・・フォームの送信先を設定する

document.フォーム名.method・・・フォームの送信方法を設定する

document.フォーム名.encoding・・・フォーム送信のエンコード方式を設定する

document.フォーム名.target・・・フォーム送信先のターゲットを設定する

document.フォーム名.reset()・・・フォームの入力内容をリセットする

document.フォーム名.submit()・・・フォームを送信する

document.forms.length・・・ドキュメント内にあるフォームの数を参照する

document.フォーム名.length・・・フォーム内のフォーム部品の数を参照する

document.フォーム名.name・・・フォームのname属性の値を参照する

## historyまわり

history.length・・・ブラウザで表示した履歴の数を返す

history.back()・・・一つ前のページへ戻る

history.forward()・・・一つ先のページへ進む

history.go()・・・指定数だけ前後のページへ移動する

## locationまわり

location.href・・・現在ページURLを参照する

location.protocol・・・現在ページURLのプロトコルを参照する

location.hostname・・・現在ページURLのホスト名を参照する

location.host・・・現在ページURLのホスト情報を参照する

location.port・・・現在ページURLのポート番号を参照する

location.pathname・・・現在ページURLのパス名を参照する

location.search・・・現在ページURLのサーチ情報を参照する

location.hash・・・現在ページURLのハッシュ部分を参照する

location.reload()・・・ページをリロード（再読み込み）する

location.replace()・・・指定したページへ移動する

## linkまわり

document.links.length・・・ドキュメント内にあるリンクの数を参照する

document.link.href・・・リンク先URLを参照する

document.link.target・・・リンクのターゲット属性を参照する

document.link.protocol・・・リンク先URLのプロトコルを参照する

document.link.hostname・・・リンク先URLのホスト名を参照する

document.link.host・・・リンク先URLのホスト情報を参照する

document.link.port・・・リンク先URLのポート番号を参照する

document.link.pathname・・・リンク先URLのパス名を参照する

document.link.search・・・リンク先URLのサーチ情報を参照する

document.link.hash・・・リンク先URLのハッシュ部分を参照する

formまわり

document.フォーム名.action・・・フォームの送信先を設定する

document.フォーム名.method・・・フォームの送信方法を設定する

document.フォーム名.encoding・・・フォーム送信のエンコード方式を設定する

document.フォーム名.target・・・フォーム送信先のターゲットを設定する

document.フォーム名.reset()・・・フォームの入力内容をリセットする

document.フォーム名.submit()・・・フォームを送信する

document.forms.length・・・ドキュメント内にあるフォームの数を参照する

document.フォーム名.length・・・フォーム内のフォーム部品の数を参照する

document.フォーム名.name・・・フォームのname属性の値を参照する

imageまわり

document.images.length・・・ドキュメント内にある画像の数を参照する

document.イメージ名.src・・・画像ファイルのURLを参照する

document.イメージ名.lowsrc・・・低解像度画像ファイルのURLを参照する

document.イメージ名.name・・・画像のname属性の値を参照する

document.イメージ名.complete・・・画像の読み込みが完了したかどうかを返す

document.イメージ名.width・・・画像の横幅を設定する

document.イメージ名.height・・・画像の高さを設定する

document.イメージ名.hspace・・・画像の水平方向の余白を設定する

document.イメージ名.vspace・・・画像の垂直方向の余白を設定する

document.イメージ名.border・・・画像のボーダーの太さを設定する

anchorまわり

document.anchors.length・・・ドキュメント内にあるアンカー数を参照する

Stringまわり

ストリング名.length・・・文字列の長さを返す

ストリング名.replace()・・・文字列内の指定文字列を置換する

ストリング名.substr()・・・指定位置から指定文字数分の文字列を抜き出す

ストリング名.substring()・・・指定位置から終了位置までの文字列を抜き出す（負の引数は0扱い）

ストリング名.slice()・・・指定位置から終了位置までの文字列を抜き出す（負の引数は後ろから）

ストリング名.charAt()・・・指定した位置の文字を抜き出す

ストリング名.charCodeAt()・・・指定した位置の文字のASCIIコードを返す

ストリング名.fromCharCode()・・・文字コードを文字に変換する

ストリング名.split()・・・文字列を区切り文字で分割して配列として返す

ストリング名.concat()・・・文字列を連結する

ストリング名.indexOf()・・・文字列が最初に現れる位置を検索する

ストリング名.lastIndexOf()・・・文字列が最後に現れる位置を検索する

ストリング名.match()・・・正規表現による検索を行い一致した文字列を返す

ストリング名.search()・・・正規表現による検索を行い一致した位置を返す

ストリング名.link()・・・文字列にリンクを設定する

ストリング名.anchor()・・・アンカーを動的に生成して文字列にアンカー名を付ける

ストリング名.toLowerCase()・・・アルファベットを小文字にする

ストリング名.toUpperCase()・・・アルファベットを大文字にする

ストリング名.bold()・・・文字列を太字にする

ストリング名.italics()・・・文字列をイタリック体にする

ストリング名.fixed()・・・文字列を等幅フォントにする

ストリング名.strike()・・・文字列に打ち消し線をつける

ストリング名.big()・・・文字列のサイズを大きくする

ストリング名.small()・・・文字列のサイズを小さくする

ストリング名.sup()・・・文字列を上付文字にする

ストリング名.sub()・・・文字列を下付文字にする

ストリング名.blink()・・・文字列を点滅させる

ストリング名.fontcolor()・・・文字列の色を設定する

ストリング名.fontsize()・・・文字列のサイズを設定する

正規表現

RegExp.global・・・グローバルマッチかどうかを返す

RegExp.ignoreCase・・・大文字と小文字の違いを無視するかどうかを返す

RegExp.multiline・・・改行コードを認識して複数行として扱うかどうかを返す

RegExp.source・・・正規表現のソース（指定内容）を返す

RegExp.input・・・検索対象の文字列を設定・参照する

RegExp.lastIndex・・・検索で一致した部分に続く次の文字の開始位置を返す

RegExp.lastParen・・・最後に ( ～ ) の条件にマッチした文字列を返す

RegExp.$1-$9・・・正規表現に含まれる括弧内のパターンにマッチした部分を返す

RegExp.lastMatch・・・最後にマッチした文字列を返す

RegExp.leftContext・・・直前にマッチした部分の左側の文字列を返す

RegExp.rightContext・・・直前にマッチした部分の右側の文字列を返す

RegExp.test()・・・指定した文字列が正規表現にマッチするかどうかを返す

RegExp.exec()・・・指定した文字列に対して検索を実行する

RegExp.compile()・・・正規表現の内容を設定する

配列

配列名.length・・・配列の要素数を返す

配列名.unshift()・・・配列の先頭に要素を追加する

配列名.push()・・・配列の末尾に要素を追加する

配列名.shift()・・・配列の先頭から要素を削除する

配列名.pop()・・・配列の末尾から要素を削除する

配列名.splice()・・・配列から要素を削除・追加して組み替える

配列名.reverse()・・・配列の要素を逆順にする

配列名.sort()・・・配列の要素をソートする

配列名.concat()・・・配列を結合する

配列名.join()・・・配列の要素をすべて連結して文字列にする

配列名.slice()・・・配列内の指定範囲の要素を新しい配列として取り出す

日付

日付オブジェクト名.getFullYear()・・・年を4桁の整数で取得する

日付オブジェクト名.getMonth()・・・月を0から11の整数で取得する

日付オブジェクト名.getDate()・・・日を1から31の整数で取得する

日付オブジェクト名.getHours()・・・時を0から23の整数で取得する

日付オブジェクト名.getMinutes()・・・分を0から59の整数で取得する

日付オブジェクト名.getSeconds()・・・秒を0から59の整数で取得する

日付オブジェクト名.getMilliseconds()・・・ミリ秒を0から999の整数で取得する

日付オブジェクト名.getDay()・・・曜日を0から6の整数で取得する

日付オブジェクト名.getUTCFullYear()・・・協定世界時の年を4桁の整数で取得する

日付オブジェクト名.getUTCMonth()・・・協定世界時の月を0から11の整数で取得する

日付オブジェクト名.getUTCDate()・・・協定世界時の日を1から31の整数で取得する

日付オブジェクト名.getUTCHours()・・・協定世界時のを0から23の整数で取得する

日付オブジェクト名.getUTCMinutes()・・・協定世界時の分を0から59の整数で取得する

日付オブジェクト名.getUTCSeconds()・・・協定世界時の秒を0から59の整数で取得する

日付オブジェクト名.getUTCMilliseconds()・・・協定世界時のミリ秒を0から999の整数で取得する

日付オブジェクト名.getUTCDay()・・・協定世界時の曜日を0から6の整数で取得する

日付オブジェクト名.setFullYear()・・・年を設定する

日付オブジェクト名.setMonth()・・・月を設定する

日付オブジェクト名.setDate()・・・日を設定する

日付オブジェクト名.setHours()・・・時を設定する

日付オブジェクト名.setMinutes()・・・分を設定する

日付オブジェクト名.setSeconds()・・・秒を設定する

日付オブジェクト名.setMilliseconds()・・・ミリ秒を設定する

日付オブジェクト名.setUTCFullYear()・・・協定世界時の年を設定する

日付オブジェクト名.setUTCMonth()・・・協定世界時の月を設定する

日付オブジェクト名.setUTCDate()・・・協定世界時の日を設定する

日付オブジェクト名.setUTCHours()・・・協定世界時の時を設定する

日付オブジェクト名.setUTCMinutes()・・・協定世界時の分を設定する

日付オブジェクト名.setUTCSeconds()・・・協定世界時の秒を設定する

日付オブジェクト名.setUTCMilliseconds()・・・協定世界時のミリ秒を設定する

日付オブジェクト名.getTimezoneOffset()・・・地方時と協定世界時の分単位の差を取得する

Date.UTC()・・・1970年からDateオブジェクトで指定した日時までの経過ミリ秒数を取得する

Date.parse()・・・1970年から日付を表す文字列で指定した日時までの経過ミリ秒数を取得する

日付オブジェクト名.getTime()・・・1970年からの経過ミリ秒数を取得する

日付オブジェクト名.setTime()・・・1970年からの経過ミリ秒数を設定する

日付オブジェクト名.getYear()・・・1900年からの経過年数を取得する

日付オブジェクト名.setYear()・・・1900年からの経過年数を設定する

日付オブジェクト名.toLocaleString()・・・指定日時を文字列に変換する

日付オブジェクト名.toUTCString()・・・協定世界時に基づいて指定日時を文字列に変換する

日付オブジェクト名.toGMTString()・・・グリニッジ標準時に基づいて指定日時を文字列に変換する

ブラウザ関連

navigator.appName・・・ブラウザの公式名称を返す

navigator.appCodeName・・・ブラウザのコード名を返す

navigator.appVersion・・・ブラウザのバージョンを返す

navigator.language・・・ブラウザの使用言語を返す

navigator.platform・・・ブラウザのプラットフォームを返す

navigator.userAgent・・・ブラウザのユーザーエージェントを返す

navigator.onLine・・・ブラウザの接続状態がオンラインかどうかを返す

navigator.cookieEnabled・・・ブラウザでCookieが有効かどうかを返す

navigator.javaEnabled()・・・ブラウザでJavaが有効かどうかを返す

関数

関数名.caller・・・指定した関数の呼び出し元を返す

関数名.length・・・指定した関数に必要な引数の数を返す

arguments・・・関数に渡された引数を参照する

関数名.call()・・・this参照値と個々の引数を渡して関数を呼び出す

関数名.apply()・・・this参照値と配列の引数を渡して関数を呼び出す

スクリーン

screen.width・・・スクリーンの幅を返す

screen.height・・・スクリーンの高さを返す

screen.availWidth・・・スクリーンの有効領域の幅を返す

screen.availHeight・・・スクリーンの有効領域の高さを返す

screen.colorDepth・・・スクリーンの色深度をビット数で返す

screen.pixelDepth・・・スクリーンのビット深度を返す

マウスイベント

マウスイベント名.clientX・・・マウスイベントが発生したX座標を取得する

マウスイベント名.clientY・・・マウスイベントが発生したY座標を取得する

マウスイベント名.layerX・・・マウスイベントが発生したレイヤーのX座標を取得する

マウスイベント名.layerY・・・マウスイベントが発生したレイヤーのY座標を取得する

マウスイベント名.pageX・・・マウスイベントが発生したページ上のX座標を取得する

マウスイベント名.pageY・・・マウスイベントが発生したページ上のY座標を取得する

マウスイベント名.screenX・・・マウスイベントが発生したスクリーン上のX座標を取得する

マウスイベント名.screenY・・・マウスイベントが発生したスクリーン上のX座標を取得する

マウスイベント名.width・・・マウスイベントが発生したウィンドウ上のX座標を取得する

マウスイベント名.height・・・マウスイベントが発生した場所のY座標を取得する

マウスイベント名.keycCode・・・入力されたキーコードUnicode番号

マウスイベント名.modifier・・・マウスイベント発生時に押されていた特殊キー

マウスイベント名.target・・・マウスイベントが最初に送られたオブジェクトを返す

マウスイベント名.type・・・マウスイベントの種類を返す

マウスイベント名.whitch・・・マウスボタンの種類を返す

## よく使うものだけを説明

ウィンドウ名.alert()

windowオブジェクトのalert()メソッドは、［OK］ボタンのみの警告ダイアログを表示します。

ユーザーは［OK］ボタンか［×］ボタンをクリックしてダイアログを閉じるまで、次の処理に進んだりブラウザを操作することはできません。

構文

ウィンドウ名.alert("メッセージの内容");

サンプル・・・