**JavaScript**

**JavaScriptとは**

|  |  |
| --- | --- |
| **ECMAScript** | どの実行環境でも共通な動作のみが定義されている |
| **ECMAScript 2015（ES2015）** | 大きくアップデートされた仕様 |
| **"use strict";** | strict mode  文の先頭に書くやつ |

**変数と宣言**

|  |  |
| --- | --- |
| **const** | 値の再代入不可能な変数を宣言 |
| **let** | 値の再代入可能な変数を宣言 |

**値の評価と表示**

|  |  |
| --- | --- |
| **REPL** | Webコンソールにある機能 |
| 入力された値を評価し表示する |
| consoleの変数をクリア | F5でリロード |

**Scriptファイルを保存するときの文字コードと改行コード**

|  |  |
| --- | --- |
| 文字コード | **UTF-8** |
| 改行コード | **LF** |

|  |  |
| --- | --- |
| HTMLでJavascript読み込むタグ | **<script src=""></script>** |

**エラーついて**

|  |  |
| --- | --- |
| SyntaxError: missing ) after argument list[詳細] index.js:1:13 | ・)がない。  ・1行目13列目  ・行は1からカウント  ・列は0からカウント |

**データ型**

**・変数の型**はない。**値の型**はある

**プリミティブ型**

|  |  |
| --- | --- |
| **Boolean** | true、false |
| **Number** | 数値 |
| **String** | 文字列 |
| **undefined** | 値が未定義であることを意味するデータ型 |
| **null** | 値が存在しないことを意味するデータ型 |
| **Symbol** | ES2015から追加された一意で不変な値のデータ型 |

**文字列リテラル**

|  |  |
| --- | --- |
| **'** | **シングルクオート** |
| **"** | **ダブルクオート** |
| **`** | **バッククオート** |
| **改行ができる** |
| **${x}で変数を埋め込める** |

**オブジェクト**

|  |  |
| --- | --- |
| **リテラル** | **{}** |

|  |  |  |
| --- | --- | --- |
| **プロパティへのアクセス** | **ドット記法** | **object.key** |
| **ブラケット** | **object["key"]** |

**配列**

|  |  |
| --- | --- |
| **リテラル** | **[]** |
| **要素へのアクセス** | **array[index]** |

**正規表現**

|  |  |
| --- | --- |
| **リテラル** | **//** |
| **/で正規表現を囲む**  **例)数字**  **/\d/** |

|  |  |
| --- | --- |
| **===** | **厳密等価演算子** |
| **!==** | **厳密不等価演算子** |
| **==** | **等価演算子** |
| **!=** | **不等価演算子** |

・==では**暗黙的な型変換**を行ってから比較する。

console.log(1 == "1"); // => true

**型変換**

|  |  |  |
| --- | --- | --- |
| true、falseに変換 | **Boolean()** | Boolean("string"); // => true  Boolean(0); // => false |

|  |  |
| --- | --- |
| **falsyな値** | false  undefined  null  0  NaN  "" |

|  |  |  |
| --- | --- | --- |
| 文字列に変換 | **String()** | String("str"); // => "str"  String(true); // => "true"  String(null); // => "null" |

※**プリミティブ**型のみ有効

|  |  |  |
| --- | --- | --- |
| 数値に変換 | **Number()** | Number("1"); // => 1 |

|  |  |  |
| --- | --- | --- |
| NaN | Not a Number |  |

|  |  |
| --- | --- |
| NaNかどうか判定 | Number.isNaN(x) |

**関数**

|  |  |
| --- | --- |
| **関数宣言** | **Function**キーワードを使う |
| **静**的 |
| **関数式** | **変数**に**匿名関数**を代入する |
| **動**的 |

|  |  |
| --- | --- |
| **arguments** | 与えられた全ての引数を保持する |
| **arrayライク**なオブジェクト |
| **関数ブロック**でのみ利用可能 |

・returnが省略された場合**undefined**を返す

・javaScriptは引数の個数をチェックしない

・varがない変数は全てグローバル変数となる

**ES2015**

|  |  |
| --- | --- |
| **アロー関数** | **変数**に**匿名関数**を代入する |
| **動**的 |

|  |  |
| --- | --- |
| **可変長引数** | **...**を仮引数の前につける |
| numsは完璧な**配列** |

|  |  |
| --- | --- |
| **引数のデフォルト値** | **仮引数=x**でできる |

|  |  |
| --- | --- |
| **コールバック関数** | 引数として渡される関数 |
| **高階関数** | 引数がコールバック関数な関数 |

例）

setTimeout(function(){

console.log("Hello")

}, 2000);

**ビルトインオブジェクト**

**Array**

|  |  |  |
| --- | --- | --- |
| **length** | Number | 配列の長さ |
| **Array.isArray(object)** | Boolean | objectが配列かどうか判定 |
| **push() \*** | Number | ・配列の末尾に要素を追加  ・戻り値は配列の長さ |
| **pop() \*** | 要素 | ・配列の末尾の要素を取得・削除  ・戻り値は要素 |
| **shift() \*** | Number | ・配列の先頭に要素を追加  ・戻り値は配列の長さ |
| **unshift() \*** | 要素 | ・配列の先頭の要素を取得・削除  ・戻り値は要素 |
| **sort() \*** | Array | ・配列を昇順にソート  ・戻り値は配列 |
| **reverse() \*** | Array | ・配列をリバースする  ・戻り値は配列 |
| **indexOf()** | Number | ・要素のインデックスを取得する |
| **concat()** | Array | ・配列を取り込む  ・戻り値は配列 |
| **join()** | String | ・配列の要素をdelimiterで区切った文字列に変換  ・delimiterを空文字にすると区切られず連結される  ・戻り値は文字列 |
| **of()** | Array | ・引数に渡された値から配列を生成  ・戻り値は配列 |
| **includes()** | Boolean | ・引数に渡された値を含むかどうか判定  ・戻り値はboolean |

|  |  |  |
| --- | --- | --- |
| **forEach()** | undefined | 要素を順に処理 |
| **map()** | Array | 要素を順に加工 |
| **filter()** | Array | 要素をフィルタリング |

・**\***が付いているメソッドは元の配列の値が変わる。

・ついていないメソッドは元の配列の値に影響しない。

**Object**

|  |  |  |
| --- | --- | --- |
| **Object.keys()** | **Array** | ・objectの列挙可能なkeyを取得  ・戻り値はArray |
| **Object.getOwnPropertyNames()** | **Array** | ・objectの全てのkeyを取得  ・戻り値はArray |
| **Object.create()** | **object** | ・既存のobjectから新しいobjectを生成  ・戻り値はobject |