



Cette page a été traduite à partir de l'anglais par la communauté. Vous pouvez également contribuer en rejoignant la communauté francophone sur MDN Web Docs.

## Objets globaux

Cette partie référence tous les objets natifs standards JavaScript, avec leurs propriétés et méthodes.

Le terme « objets globaux » (ou objets natifs standards) ne doit pas ici être confondu avec l'\*objet global\*. Ici, « objets globaux » se réfère aux *objets de portée globale*. L'objet global lui-même peut être accédé en utilisant [this](#) dans la portée globale (uniquement lorsque [le mode strict](#) n'est pas utilisé, sinon, il renvoie [undefined](#)). En réalité, la portée globale *consiste des* propriétés de l'objet global (avec ses propriétés héritées, s'il en a).

**Note :** En [mode strict](#), la portée globale représentée par `this` sera [undefined](#).

Les autres objets de la portée globale sont [créés par les scripts utilisateurs](#) ou fournis par l'application hôte dans laquelle s'exécute JavaScript. Les objets mis à disposition par les navigateurs web sont documentés dans la [référence API](#). Pour plus d'informations sur la distinction entre le [DOM](#) et JavaScript, voir [l'aperçu des technologies JavaScript](#).

## Objets globaux standards (par catégorie)

### Propriétés - valeurs

Les propriétés globales renvoient une valeur simple, elles ne possèdent aucune propriété ou méthode :

- [Infinity](#)
- [NaN](#)
- [undefined](#)
- le littéral [null](#)
- [globalThis](#)

### Propriétés - fonctions

Les fonctions globales, appelées globalement (et non par rapport à un objet), renvoient directement leur résultat à l'objet appelant.

- [eval\(\)](#)
- [uneval\(\)](#) 🗑️
- [isFinite\(\)](#)
- [isNaN\(\)](#)
- [parseFloat\(\)](#)
- [parseInt\(\)](#)
- [decodeURI\(\)](#)
- [decodeURIComponent\(\)](#)
- [encodeURIComponent\(\)](#)
- [encodeURIComponent\(\)](#)
- [escape\(\)](#) 🗑️
- [unescape\(\)](#) 🗑️

## Objets fondamentaux

Ces objets sont les objets fondamentaux de JavaScript. Parmi ces objets, on retrouve les objets génériques, les fonctions et les erreurs.

- [Object](#)
- [Function](#)
- [Boolean](#)
- [Symbol](#)
- [Error](#)
- [EvalError](#)
- [InternalError](#) 🗑️
- [RangeError](#)
- [ReferenceError](#)
- [StopIteration](#)
- [SyntaxError](#)
- [TypeError](#)
- [URIError](#)

## Nombres et dates

Ces objets permettent de manipuler les nombres, dates et calculs mathématiques.

- [Number](#)
- [BigInt](#)
- [Math](#)
- [Date](#)

## Manipulation de textes

Ces objets permettent de manipuler des chaînes de caractères.

- [String](#)
- [RegExp](#)

## Collections indexées

Ces objets sont des collections ordonnées par un index. Cela inclut les tableaux (typés) et les objets semblables aux tableaux.

- [Array](#)
- [Int8Array](#)
- [Uint8Array](#)
- [Uint8ClampedArray](#)
- [Int16Array](#)
- [Uint16Array](#)
- [Int32Array](#)
- [Uint32Array](#)
- [Float32Array](#)
- [Float64Array](#)
- [BigInt64Array](#)
- [BigUint64Array](#)

## Collections avec clefs



Ces objets représentent des collections d'objets avec clefs. Ils contiennent des éléments itérables, dans leur ordre d'insertion.

- [Map](#)
- [Set](#)


- [WeakMap](#)
- [WeakSet](#)

## Données structurées

Ces objets permettent de représenter et de manipuler des tampons de données (*buffers*) et des données utilisant la notation JSON (**J**ava**S**cript **O**bject **N**otation).

- [ArrayBuffer](#)
- [SharedArrayBuffer](#) 
- [Atomics](#) 
- [DataView](#)
- [JSON](#)

## Objets de contrôle d'abstraction

- [Promise](#)
- [Generator](#)
- [GeneratorFunction](#)
- [AsyncFunction](#) 

## Introspection

- [Reflect](#)
- [Proxy](#)

## Internationalisation

Ces objets ont été ajoutés à ECMAScript pour des traitements dépendants de particularités linguistiques. Ils possèdent leur propre spécification.

- [Intl](#)
- [Intl.Collator](#)
- [Intl.DateTimeFormat](#)
- [Intl.ListFormat](#)
- [Intl.NumberFormat](#)
- [Intl.PluralRules](#)
- [Intl.RelativeTimeFormat](#)
- [Intl.Locale](#)

## WebAssembly

- [WebAssembly](#)
- [WebAssembly.Module](#)
- [WebAssembly.Instance](#)
- [WebAssembly.Memory](#)
- [WebAssembly.Table](#)
- [WebAssembly.CompileError](#)
- [WebAssembly.LinkError](#)
- [WebAssembly.RuntimeError](#)

## Autres

- [arguments](#)

**Last modified:** 28 janv. 2022, [by MDN contributors](#)