

Filter

TypedArray.prototype.values()
TypedArray.prototype.with()

Related pages:

Uint8Array

Uint8ClampedArray

Int16Array

Uint16Array

Int32Array

Uint32Array

Float32Array

Float64Array

BigInt64Array

Uint8Array

The `Uint8Array` typed array represents an array of 8-bit unsigned integers. The contents are initialized to `0`. Once established, you can reference elements in the array using the object's methods, or using standard array index syntax (that is, using bracket notation).

`Uint8Array` is a subclass of the hidden `TypedArray` class.

Constructor

`Uint8Array()`

Creates a new `Uint8Array` object.

Static properties

Also inherits static properties from its parent `TypedArray`.

`Uint8Array.BYTES_PER_ELEMENT`

Returns a number value of the element size. `1` in the case of `Uint8Array`.

Static methods

Inherits static methods from its parent `TypedArray`.

Instance properties

Also inherits instance properties from its parent `TypedArray`.

These properties are defined on `Uint8Array.prototype` and shared by all `Uint8Array` instances.

`Uint8Array.prototype.BYTES_PER_ELEMENT`

Returns a number value of the element size. `1` in the case of a `Uint8Array`.

`Uint8Array.prototype.constructor`

The constructor function that created the instance object. For `Uint8Array` instances, the initial value is the `Uint8Array` constructor.

Instance methods

Inherits instance methods from its parent `TypedArray`.

Examples

Different ways to create a Uint8Array

```
JS

// From a length
const uint8 = new Uint8Array(2);
uint8[0] = 42;
console.log(uint8[0]); // 42
console.log(uint8.length); // 2
console.log(uint8.BYTES_PER_ELEMENT); // 1

// From an array
const x = new Uint8Array([21, 31]);
console.log(x[1]); // 31

// From another TypedArray
const y = new Uint8Array(x);
console.log(y[0]); // 21

// From an ArrayBuffer
const buffer = new ArrayBuffer(8);
const z = new Uint8Array(buffer, 1, 4);
console.log(z.byteOffset); // 1















































































// From an iterable
const iterable = (function* () {
  yield* [1, 2, 3];
})();
const uint8FromIterable = new Uint8Array(iterable);
console.log(uint8FromIterable);
// Uint8Array [1, 2, 3]
```

Specifications

Specification
ECMAScript Language Specification # table-49

Browser compatibility

Report problems with this compatibility data on GitHub

	🖥️					📱					☰		
	 Chrome	 Edge	 Firefox	 Opera	 Safari	 Chrome Android	 Firefox for Android	 Opera Android	 Safari on iOS	 Samsung Internet	 WebView Android	 Deno	 Node.js
<code>Uint8Array</code>													
	7	12	4	11.6	5.1	18	4	12	4.2	1.0	4	1.0	0.10.0
<code>Uint8Array()</code> constructor													
	7	12	4	11.6	5.1	18	4	12	4.2	1.0	4	1.0	0.10.0
Constructor without parameters													
	7	12	55	11.6	5.1	18	55	12	5	1.0	37	1.0	0.10.0
<code>new</code> <code>Uint8Array(iterable)</code>													
	39	14	52	26	10	39	52	26	10	4.0	39	1.0	4.0.0
<code>Uint8Array()</code> without <code>new</code> throws													
	7	14	44	15	5.1	18	44	14	5	1.0	37	1.0	0.12.0

Tip: you can click/tap on a cell for more information.

✓ Full support

See also

- Polyfill of `Uint8Array` in `core-js`
- JavaScript typed arrays
- `TypedArray`
- `ArrayBuffer`
- `DataView`

Found a content problem with this page?

- Edit the page on GitHub.
- Report the content issue.
- View the source on GitHub.

Want to get more involved? Learn how to contribute.

This page was last modified on Aug 14, 2023 by MDN contributors.