



JavaScript Numeric Separator

Summary: in this tutorial, you'll learn how to use the JavaScript numeric separator to make the numeric literals more readable.

Introduction to the JavaScript numeric separator

The numeric separator allows you to create a visual separation between groups of digits by using underscores (`_`) as separators.

For example, the following number is challenging to read especially when it contains long digit repetitions:

```
const budget = 1000000000;
```

Is this a billion or a hundred million?

The numeric separator fixes this readability issue as follows:

```
const budget = 1_000_000_000;
```

As you can see, the number is now very easy to interpret.

JavaScript allows you to use numeric separators for both integer and floating-point [numbers](#). For example:

```
let amount = 120_201_123.05; // 120201123.05
let expense = 123_450; // 123450
let fee = 12345_00; // 1234500
```

It's important to note that all numbers in JavaScript are floating-point numbers.

Also, you can use the numeric separators for factional and exponent parts. For example:

```
let amount = 0.000_001; // 1 millionth
```

It's important to notice that you can use the numeric separator for `bigint` literal, binary literal, octal literal, and hex literal. For example:

```
// BigInt
const max = 9_223_372_036_854_775_807n;

// binary
let nibbles = 0b1011_0101_0101;

// octal
let val = 0o1234_5670;

// hex
let message = 0xD0_E0_F0;
```

Quiz

Summary

- Use underscores (`_`) as the numeric separators to create a visual separation between groups of digits.