

Numbers in Javascript

Relevant links

- Flanagan's book, section 3.1.
- MDN's reference for Numbers¹
- MDN's reference for the Math object²
- IEEE 754 standard³ (optional)
- The group working on the standard⁴ (optional)
- Wikipedia page on IEEE 754 standard⁵ (optional)

Numbers in Javascript

- Unlike most languages, in Javascript there is only one type of numbers, namely 64-bit floating point numbers defined in the IEEE 754 standard.
- Certain operations (e.g. bitwise) are performed treating the numbers as 32-bit integers instead.
- Number literals are much like in other languages. You can use 0x for hexadecimal.
- Avoid using octal literals (those starting with a 0).
- The Math object contains a number of useful functions. Refer to the MDN documentation for these functions for details. As practice, learn how to do the following: (avoid functions tagged as “experimental”):
 - Raising to a power (so compute for example $3^{2.2}$. Should get about 11.2)
 - Maximum/Minimum (compute for example the maximum between 2, -3 and 1e2)
 - Random numbers
 - * Ask the system to generate a random number between 0 and 1
 - * Obtain a random integer between 0 and 4 as follows:
 - Start with a random number between 0 and 1
 - Multiply it by 5
 - Use the floor function to round it down
 - Learn how to round numbers up, and also what the normal round would do
 - Logarithm and exponentials if you have time
 - Trigonometric functions if you have time
- The Math object also contains a number of mathematical constants (pi, e).
- The Number object contains constants for the largest integer, as well as Infinity and NaN (not a number).

¹https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Number

²https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Math

³<http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=4610933>

⁴<http://grouper.ieee.org/groups/754/>

⁵http://en.wikipedia.org/wiki/IEEE_floating_point

- Arithmetic operations never raise errors, instead producing either Infinity, -Infinity or NaN.
- NaN in particular is a very unusual value. It represents a numeric value that is indeterminate.
 - So it *is* a number, even though it stands for “not a number”.
 - You can see that by typing `typeof NaN`.
 - NaN is not equal to any value, including itself.
 - You can see that by trying `NaN == NaN`.
 - `Number.isNaN` can be used to check if a *number* is actually NaN.
- Some behavior to watch out for (try to explain it): `Number.MAX_VALUE == Number.MAX_VALUE + 1`
- Make sure to read about rounding errors, that are unavoidable whenever you do floating point arithmetic.