Lecture Template

Built-In Objects

Topics

Number

- **ParseInt**
- **IsNaN**

Math

- **Floor**
- **Ceil**
- **Abs**
- **Round**
- **Random**

Number

Number

When used as a function, Number(value) converts a string or other value to the Number type. If the value can't be converted, it returns NaN.

Number is a primitive wrapper object used to represent and manipulate numbers like 37 or -9.25.

```
Number('123')  // returns the number 123
Number('123') === 123  // true

Number("unicorn")  // NaN
Number(undefined)  // NaN
```

parseInt

The **Number.parseInt()** method parses a string argument and returns an integer of the specified radix or base.

```
parseInt("10");          // returns 10
parseInt("10.33");       // returns 10
parseInt("10 20 30");    // returns 10
parseInt("10 years");    // returns 10
parseInt("years 10");    // returns NaN
```

isNaN

The Number.isNaN() method determines whether the passed value is NaN and its type is Number. It is a more robust version of the original, global isNaN().

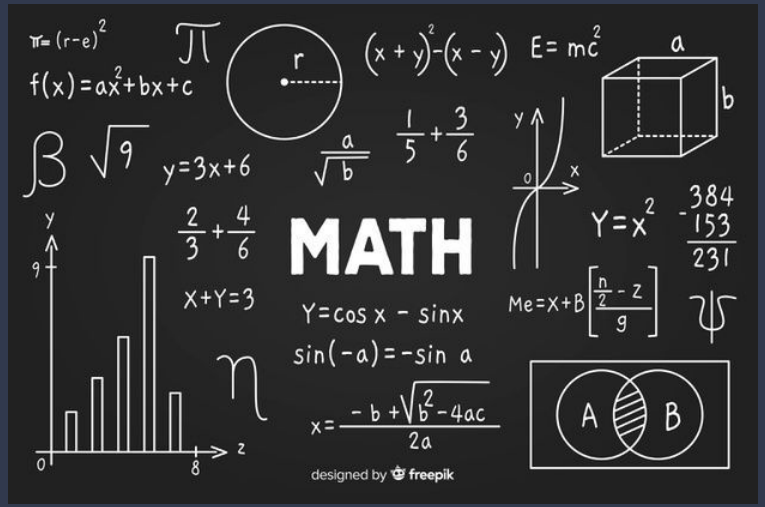
```
isNaN(123) //false
isNaN(-1.23) //false
isNaN(5-2) //false
isNaN(0) //false
isNaN('123') //false
```

```
isNaN('Hello') //true
isNaN('2005/12/12') //true
isNaN('') //false
isNaN(true) //false
isNaN(undefined) //true
isNaN('NaN') //true
```

Video: parseInt

Video: NaN

Math



Math is one of JavaScript's global or standard built-in objects, and can be used anywhere you can use JavaScript.

It contains useful constants like π and Euler's constant and functions such as floor(), round(), and ceil().

Math.Max()

Math.max() is a function that returns the largest value from a list of numeric values passed as parameters. If a non-numeric value is passed as a parameter, Math.max() will return **NaN**.

An array of numeric values can be passed as a single parameter to Math.max() using either spread (...) or apply. Either of these methods can, however, fail when the amount of array values gets too high.

```
Math.max(value1, value2, value3, ...);
```

Math.Max() cont..

Parameters

Numbers, or limited array of numbers.

Return Value

The greatest of given numeric values, or NaN if any given value is non-numeric.

```
Math.max(4, 13, 27, 0, -5); // returns 27
```

```
Math.max(4, 13, 27, 'eight', -5); // returns NaN
```

Math.Min()

The Math.min() function returns the smallest of zero or more numbers.

You can pass it any number of arguments.

```
Math.min(7, 2, 9, -6);
// returns -6
```

Math PI

Math.PI is a static property of the Math object and is defined as the ratio of a circle's circumference to its diameter.

Pi is approximately 3.14149, and is often represented by the Greek letter π.

```
Math.PI \\ 3.141592653589793
```

Math Pow

Math.pow() returns the value of a number to the power of another number.

Syntax

Math.pow(base, exponent), where base is the base number and exponent is the number by which to raise the base.

```
Math.pow(5, 2); // 25
Math.pow(7, 4); // 2401
Math.pow(9, 0.5); // 3
Math.pow(-8, 2); // 64
Math.pow(-4, 3); // -64
```

Math Sqrt

The function Math.sqrt() returns the square root of a number.

If a negative number is entered, NaN is returned.

Syntax

Math.sqrt(x), where x is a number.

```
Math.sqrt(25); // 5
Math.sqrt(169); // 13
Math.sqrt(3); // 1.732050807568
Math.sqrt(1); // 1
Math.sqrt(-5); // NaN
```

Math Ceiling

The Math.ceil() is a method of the Math standard object that rounds a given number upwards to the next integer.

Take note that for negative numbers this means that the number will get rounded “towards 0” instead of the number of greater absolute value

```
Math.ceil(0.1)  //  1
Math.ceil(1.3)  //  2
Math.ceil(-0.9) // -0
Math.ceil(-1.5) // -1
```

Math Floor

Math.floor() is a method of the Math standard object that rounds a given number downwards to the next integer.

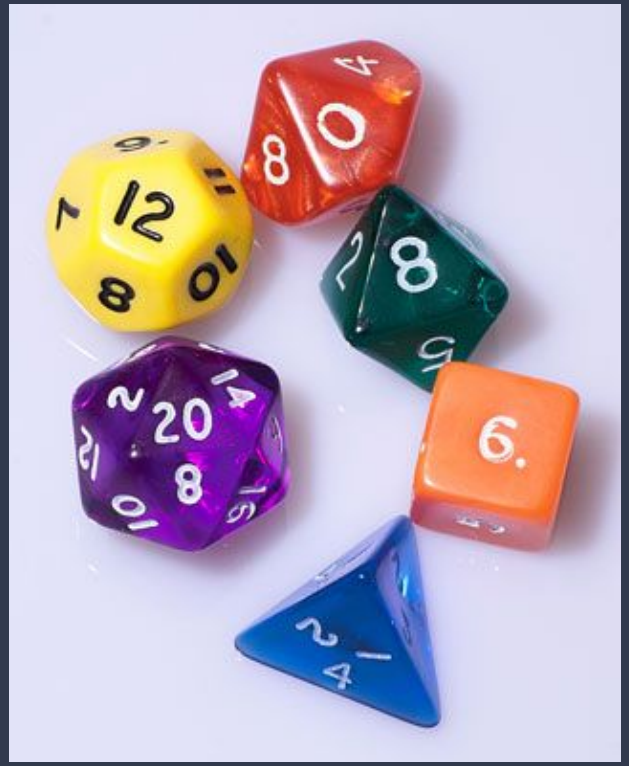
Math.floor() returns the largest integer less than or equal to the given number.

```
Math.floor(0.9)  //  0
Math.floor(1.3)  //  1
Math.floor(0.5)  //  0
Math.floor(-0.9) // -1
Math.floor(-1.3) // -2
```

Video - Math

Math.Random()

Generate Random Numbers



Often while developing projects, you will find yourself looking for ways to generate random numbers.

The most common use cases for generating random numbers are games of chance like rolling dice, shuffling playing cards, and spinning roulette wheels.

Math.random()

The **Math** object in JavaScript is a built-in object that has properties and methods for performing mathematical calculations.

A common use of the Math object is to create a random number using the random() method.

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

```
const randomValue = Math.random();
```

Math.random() cont..

But the Math.random() method doesn't actually return a whole number. Instead, it returns a floating-point value between 0 (inclusive) and 1 (exclusive). Also, note that the value returned from Math.random() is pseudo-random in nature.

Random numbers generated by Math.random() might seem random, but those numbers will repeat and eventually display a non-random pattern over a period of time.

This is because algorithmic random number generation can never be truly random in nature. This is why we call them pseudo-random number generators (PRNGs).

Random Number Generator Function

Here is a Math.random() method to create a function that will return a random integer between two values (inclusive).

```
const getRandomNumber = (min, max) => {
  return Math.floor(Math.random() * (max - min + 1)) + min;
};
```

Logic Breakdown

The **Math.random()** method will return a floating-point number between 0 and 1 (exclusive).

So the intervals would be as follows:

```
[0 ..................................... 1)

[min ..................................... max)
```

To factor the second interval, subtract min from both ends. So that would give you an interval between 0 and max-min.

```
[0 ..................................... 1)

[0 ..................................... max - min)
```

So now, to get a random value you would do the following:

```
const x = Math.random() * (max - min)
```

Here x is the random value.