

Front-end

JavaScript

JS. Numbers

- Работа с числами
- Math

JS. Numbers

Метод `toString()`

Метод `toString()` возвращает число в виде строки.

JS. Numbers

МетодtoFixed()

МетодtoFixed() возвращает строку с числом, записанным с указанным количеством десятичных знаков

```
let a = 8;  
let b = 3;  
  
const result = a / b;  
  
console.log(result); // => 2.6666666666666665 - number  
console.log(result.toFixed(2)) // => '2.67' - string
```

JS. Numbers

Преобразование переменных в числа

Есть 3 метода JavaScript, которые можно использовать для преобразования переменных в числа:

Метод Number()

Метод parseInt() - анализирует его аргумент и возвращает целое число

Метод parseFloat() - анализирует его аргумент и возвращает число с плавающей запятой

Если число не может быть преобразовано, возвращается NaN

JS. Numbers

Метод parseInt()

Метод `parseInt()` анализирует строку и возвращает целое число.
Разрешены пробелы. Возвращается только первое число

Функция `parseInt` удобна тем, что если в строке после числа есть текст, то она его исключает и возвращает только число.

Если же в строке текст расположен перед числом, то функция `parseInt` возвращает `NaN`.

Функция `parseInt` возвращает ЦЕЛОЕ число

JS. Numbers

Метод parseFloat()

Метод parseFloat() анализирует строку и возвращает число. Разрешены пробелы. Возвращается только первое число

Функция parseFloat аналогична функции parseInt. Но есть одно отличие - parseFloat возвращает дробное число.

Функция parseFloat - возвращает дробное число

JS. Numbers

Второй аргумент parseInt

Аргумент radix - основание системы счисления

Описание, примеры

Функции parseInt, toString преобразуют первый аргумент в число по указанному основанию, а если это невозможно - возвращает NaN. Например, radix=10 даст десятичное число, 16 - шестнадцатеричное и т.п. Для radix > 10 цифры после девяти представлены буквами латинского алфавита.

Если в процессе преобразования parseInt обнаруживает цифру, которая не является цифрой в системе счисления с основанием radix, например G в 16-ричной системе или A в десятичной, то процесс преобразования тут же завершается и возвращается значение, полученное из строки на данный момент.

parseInt округляет дробные числа, т.к. останавливается на десятичной точке.

Если radix не указан или равен 0, то JavaScript предполагает следующее:

Если входная строка начинается с "0x", то radix = 16

Если входная строка начинается с "0", то radix = 8. Этот пункт зависит от реализации и в некоторых браузерах (Google Chrome) отсутствует.

В любом другом случае radix=10

Если преобразовать в число не удается, parseInt возвращает NaN

Чтобы представить число в виде строки нужной системы счисления, используйте intValue.toString(основание).

JS. Numbers

Метод isInteger()

```
console.log(Number.isInteger(2)) // => true
```

Определяет, является ли значение целым числом.

Этот метод возвращает `true`, если значение имеет тип `Number` и является целым числом. В противном случае метод возвращает `false`.

```
console.log(Number.isInteger('2')) // => false
```

JS. Numbers

Метод Number.isFinite()

Определяет, является ли значение конечным числом.

Возвращает `false`, если аргумент является `NaN`, положительной или отрицательной бесконечностью (`Infinity` или `-Infinity`). Иначе возвращает `true`.

`Number.isFinite()` отличается от глобальной функции `isFinite()`. Глобальная функция `isFinite()` сначала преобразует тестируемое значение в число, а затем проверяет его.

`Number.isFinite()` не преобразует значения в число и не возвращает `true` для любого значения, которое не относится к типу `Number`.

JS. Math

Math — встроенный в JS объект, который содержит свойства и методы с математическим уклоном.

JS. Math

Основные свойства объекта Math

- **Math.E** – константа е (число Эйлера)
- **Math.LN2** – натуральный логарифм 2
- **Math.LN10** – натуральный логарифм 10
- **Math.LOG2E** – натуральный логарифм числа е (Эйлера) по основанию 2
- **Math.LOG10E** - натуральный логарифм числа е (Эйлера) по основанию 10
- **Math.PI** – константа π
- **Math.SQRT1_2** – корень квадратный из $\frac{1}{2}$
- **Math.SQRT2** – корень квадратный из 2

```
const PI = Math.PI  
console.log(PI)
```

JS. Math

Основные методы объекта Math

Тригонометрические функции:

- `Math.sin(параметр)`
- `Math.cos(параметр)`
- `Math.tan(параметр)`
- `Math.asin(параметр)`
- `Math.acos(параметр)`
- `Math.atan(параметр)`

JS. Math

Основные методы объекта Math

Преобразование дробных чисел в целые:

- **Math.ceil(параметр)** – округление в большую сторону
- **Math.floor(параметр)** – округление в меньшую сторону
- **Math.round(параметр)** – математическое округление, т.е. когда дробная часть числа больше или равна 0.5, то оно округляется в большую сторону, а иначе в меньшую сторону
- **Math.trunc(параметр)** – отбрасывает дробную часть числа

JS. Math

Основные методы объекта Math

Math.abs(параметр) - модуль числа

Math.exp(параметр) - возвращает натуральный логарифм по основанию е и аргументу

Math.log(параметр) - вычисление натурального логарифма числа, указанного в качестве параметра

Math.pow(x,y) - возводит первый аргумент в степень (второй аргумент)

Math.sqrt(параметр) - вычисление корня квадратного из числа

Math.sign(параметр) - определение знака числа

JS. Math

Основные методы объекта Math

Math.random() - генерация случайного числа с плавающей точкой от 0 (включая 0) до 1 (не включая 1). Всегда возвращает число меньше 1

Math.max(параметр_1, параметр_2, ...) - возвращения максимального значения из чисел, указанных в качестве параметров

Math.min(параметр_1, параметр_2, ...) - возвращения минимального значения из чисел, указанных в качестве параметров

JS. Math

Случайные (рандомные) целые числа

Math.random() используется с Math.floor() и может использоваться для возврата случайных целых чисел.

Math.floor(Math.random() * 10); - случайное целое число от 0 до 9

Math.floor(Math.random() * (max - min)) + min; - min (включено) до max (исключено)

Math.floor(Math.random() * (max - min + 1)) + min; - от min до max (оба включены)

JS. Math

Math

```
// password base 36
console.log(Math.random().toString(36).slice(2, 10)) // 0-9 a-z
```

```
const randomColor = Math.floor(Math.random() * 16777215).toString(16);

test.style.backgroundColor = '#' + randomColor;

console.log('#' + randomColor)
```