

Объект **Math** в JavaScript

1. **Math.sqrt(x)** - Вычисление квадратного корня:

```
console.log(Math.sqrt(36)); // 6
console.log(36 ** (1 / 2)); // 6
```

- **Math.sqrt(x)** используется для вычисления квадратного корня числа **x**. Второй вариант представляет собой эквивалентное выражение с использованием оператора "**".

2. **Math.cbrt(x)** - Вычисление кубического корня:

```
console.log(Math.cbrt(27)); // 3
```

- **Math.cbrt(x)** вычисляет кубический корень числа **x**.

3. **Math.pow(x, y)** - Возведение в степень:

```
console.log(Math.pow(16, 1 / 4)); // 2
console.log(16 ** (1 / 4)); // 2
```

- **Math.pow(x, y)** позволяет возвести число **x** в степень **y**. Также мы можем использовать оператор "**" для выполнения той же операции.

4. **Math.sign(x)** - Получение знака числа:

```
console.log(Math.sign(-100)); // -1
console.log(Math.sign(35)); // 1
```

- **Math.sign(x)** возвращает знак числа **x**. Он возвращает **-1** для отрицательных чисел, **1** для положительных чисел и **0** для нуля.

5. **Math.abs(x)** - Получение абсолютного значения числа:

```
console.log(Math.abs(-100)); // 100
console.log(Math.abs(100)); // 100
```

- `Math.abs(x)` возвращает абсолютное (положительное) значение числа `x`.

6. `Math.exp(x)` - Вычисление экспоненты:

```
console.log(Math.exp(3)); // 20.085536923187668
```

- `Math.exp(x)` вычисляет экспоненту (e^x), где `e` - это математическая константа, приближенно равная 2.71828.

7. `Math.max(...args)` и `Math.min(...args)` - Нахождение максимального и минимального значения:

```
console.log(Math.max(1, -2, 10, 0, 19)); // 19
console.log(Math.min(1, -2, '10', true, '19')); // 19
console.log(Math.max(1, -2, '10sdfs', true, '19')); // NaN
console.log(Math.min(1, -2, 'abc', true, '19')); // NaN
```

- `Math.max(...args)` возвращает наибольшее значение среди переданных аргументов. Если все аргументы могут быть корректно преобразованы в числа (например, строки, которые представляют числа, или булевы значения), то возвращается максимальное число. Однако, если хотя бы один из аргументов не может быть преобразован в число (например, невалидная строка или объект), результатом будет NaN.
- `Math.min(...args)` делает аналогичное, но для минимального значения.

8. Использование оператора расширения (spread operator) для массивов:

```
const arr = [1, -2, 10, 0, 19];
console.log(Math.max(arr)); // NaN
console.log(Math.max(...arr)); // 19
```

- Вы можете использовать оператор расширения (`...`) для преобразования элементов массива в аргументы функции `Math.max()` или `Math.min()`.

9. `Math.random()` - Генерация случайного числа:

```
console.log(Math.random());
```

- `Math.random()` возвращает случайное число в диапазоне от 0 (включительно) до 1 (исключительно).