Math Functions - Math.random():

Math.random() is a built-in function in JavaScript that returns a random number between 0 (inclusive) and 1 (exclusive). The number generated by Math.random() is a pseudo random float (decimal) number.

Here's an example of using Math.random() to generate a random number:

```JavaScript
let randomNumber = Math.random();
console.log(randomNumber); // Output: a number between 0 and 1, e.g. 0.23452345
```

You can use this function to generate a random number within a specific range by multiplying the result of Math.random() by the desired range and then using the Math.floor() function to round down to the nearest whole number.

For example, to generate a random number between 1 and 10:

```JavaScript
let randomNumber = Math.floor(Math.random() * 10) + 1;
console.log(randomNumber); // Output: a number between 1 and 10, e.g. 5
```

To generate a random number between a specific range, you can use the following formula:

```JavaScript
let randomNumber = Math.floor(Math.random() * (max - min + 1)) + min;
```

Where max is the maximum number of the range and min is the minimum number of the range.

Note: Math.random() uses a deterministic algorithm, so it is not truly random. It is suitable for most use cases. For cryptographic purposes you should use a cryptographically secure random number generator.

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution is prohibited.

Math Functions - Min and Max Functions in JS:

Math.min() and Math.max() are built-in functions in JavaScript that are used to find the minimum and maximum values in a set of numbers, respectively.

The Math.min() function takes any number of arguments and returns the smallest of them. Here's an example:

```javascript
let minimum = Math.min(5, 3, 8, 1, 9);
console.log(minimum); // Output: 1
```

The Math.max() function also takes any number of arguments and returns the largest of them. Here's an example:

```javascript
let maximum = Math.max(5, 3, 8, 1, 9);
console.log(maximum); // Output: 9
```

You can also use the Math.min() and Math.max() functions to find the minimum and maximum values in an array of numbers:

```javascript
let myArray = [5, 3, 8, 1, 9];
let minimum = Math.min(...myArray);
let maximum = Math.max(...myArray);
console.log(minimum); // Output: 1
console.log(maximum); // Output: 9
```

Here "..." is the spread operator, it allows an iterable such as an array to be expanded in places where zero or more arguments (for function calls) or elements (for array literals) are expected.

In short, Math.min() function returns the smallest number of the given numbers and Math.max() function returns the largest number of the given numbers.

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution is prohibited.

Math functions - Ceil, Floor and Power methods in JS:

Math.ceil(), Math.floor(), and Math.pow() are built-in functions in JavaScript that are used to perform mathematical operations on numbers.

The Math.ceil() function is used to round a number up to the nearest integer. Here's an example:

```javascript
let number = 5.2;
let rounded = Math.ceil(number);
console.log(rounded); // Output: 6
```

The Math.floor() function is used to round a number down to the nearest integer. Here's an example:

```javascript
let number = 5.8;
let rounded = Math.floor(number);
console.log(rounded); // Output: 5
```

The Math.pow() function is used to calculate a number raised to the power of another number. It takes two arguments: the number to be raised to a power and the power to which the number is raised. Here's an example:

```javascript
let number = 2;
let power = 3;
let result = Math.pow(number, power);
console.log(result); // Output: 8
```

In short, Math.ceil() function rounds a number up to the nearest integer, Math.floor() function rounds a number down to the nearest integer, and Math.pow() function calculates a number raised to the power of another number.

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution is prohibited.

Note: These methods are part of the Math object, so you need to use the Math keyword before calling them.

Number functions - ParseInt and ParseFloat methods:

parseInt() and parseFloat() are built-in functions in JavaScript that are used to convert strings to numbers.

The parseInt() function is used to convert a string to an integer. It takes a string as an argument and returns the parsed integer. If the string cannot be parsed as an integer, it returns NaN (Not a Number). Here's an example:

```javascript
let myString = "123";
let myNumber = parseInt(myString);
console.log(myNumber); // Output: 123
```

The parseFloat() function is used to convert a string to a floating-point number. It takes a string as an argument and returns the parsed number. If the string cannot be parsed as a number, it returns NaN (Not a Number). Here's an example:

```javascript
let myString = "3.14";
let myNumber = parseFloat(myString);
console.log(myNumber); // Output: 3.14
```

In short, parseInt() function is used to convert a string to an integer, and parseFloat() function is used to convert a string to a floating-point number.

Note: parseInt() and parseFloat() functions are global functions and not part of any object, so you can call them directly without referencing any object.

Also, you can use the Number() function to convert a variable to number type, but it only works with variables, not strings, and it also returns NaN when it can't convert the variable to number.

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution is prohibited.