Number

Feb 17, 2024 22:28

Supported Clients

SmartClient WebClient NGClient MobileClient

Property Summary

Number	MAX_VALUE	The largest representable number.
Number	MIN_VALUE	The smallest representable number.
Number	NEGATIVE_INFINITY	Special value representing negative infinity; returned on overflow.
Object	NaN	Special "not a number" value.
Number	POSITIVE_INFINITY	Special value representing infinity; returned on overflow.

Methods Summary

String	toExponential()	Returns a string representing the number in exponential notation.
String	toExponential(fractionDigits)	Returns a string representing the number in exponential notation.
String	toFixed()	Returns a string representing the number in fixed-point notation.
String	toFixed(digits)	Returns a string representing the number in fixed-point notation.
String	toLocaleString()	Converts the number into a string which is suitable for presentation in the given locale.
String	toPrecision()	Returns a string representing the number to a specified precision in fixed-point or exponential notation.
String	toPrecision(precision)	Returns a string representing the number to a specified precision in fixed-point or exponential notation.
String	toString()	Returns a string representing the specified Number object.
String	toString(radix)	Returns a string representing the specified Number object.

Property Details

MAX_VALUE

`The largest representable number.`

Returns

Number

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
application.output("Largest number: " + Number.MAX_VALUE);
```

MIN_VALUE

`The smallest representable number.`

Returns

Number

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
application.output("Smallest number: " + Number.MIN_VALUE);
```

NEGATIVE_INFINITY

`Special value representing negative infinity; returned on overflow.`

Returns

Number

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
application.output("Negative infinity: " + Number.NEGATIVE_INFINITY);
```

NaN

Special "not a number" value.

Returns

Object

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
application.output("NaN: " + Number.NaN);
```

POSITIVE_INFINITY

Special value representing infinity; returned on overflow.

Returns

Number

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
application.output("Positive infinity: " + Number.POSITIVE_INFINITY);
```

Methods Details

toExponential()

Returns a string representing the number in exponential notation.

Returns

String A string representing the number in exponential notation.

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var n = 123.45678;
application.output(n.toExponential(3));
```

toExponential(fractionDigits)

Returns a string representing the number in exponential notation.

Parameters

Number fractionDigits An integer specifying the number of digits after the decimal point. Defaults to as many digits as necessary to specify the number.

Returns

String A string representing the number in exponential notation.

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var n = 123.45678;
application.output(n.toExponential(3));
```

toFixed()

Returns a string representing the number in fixed-point notation.

Returns

String A string representing the number in fixed-point notation.

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var n = 123.45678;
application.output(n.toFixed(3));
```

toFixed(digits)

Returns a string representing the number in fixed-point notation.

Parameters

Number digits The number of digits to appear after the decimal point. Defaults to 0.

Returns

String A string representing the number in fixed-point notation.

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var n = 123.45678;
application.output(n.toFixed(3));
```

toLocaleString()

Converts the number into a string which is suitable for presentation in the given locale.

Returns

String A string representing the number in the current locale.

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var n = 1000000;
application.output(n.toLocaleString());
```

toPrecision()

Returns a string representing the number to a specified precision in fixed-point or exponential notation.

Returns

String A string representing the number to a specified precision in fixed-point or exponential notation.

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var n = 123.45678;
application.output(n.toPrecision(5));
```

toPrecision(precision)

Returns a string representing the number to a specified precision in fixed-point or exponential notation.

Parameters

Number precision An integer specifying the number of significant digits.

Returns

String A string representing the number to a specified precision in fixed-point or exponential notation.

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var n = 123.45678;
application.output(n.toPrecision(5));
```

toString()

Returns a string representing the specified Number object.

Returns

String A string representing the specified Number object.

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var n = 7;
application.output(n.toString()); //displays "7"
application.output(n.toString(2)); //displays "111"
```

toString(radix)

Returns a string representing the specified Number object.

Parameters

Number radix An integer between 2 and 36 specifying the base to use for representing numeric values

Returns

String A string representing the specified Number object.

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var n = 7;
application.output(n.toString()); //displays "7"
application.output(n.toString(2)); //displays "111"
```