xtype.js

Elegant, highly efficient data validation for JavaScript

GitHub

Basic Types

null

compact | `null`

The null value.

type id	`xtype.NULL`
category	simple type
derived	no

undefined

compact | **undef**

The undefined value.

type id	`xtype.UNDEFINED`
category	simple type
derived	no

nan

compact | **nan**

The NaN value.

type id	`xtype.NAN`
category	simple type
derived	no

symbol

compact | **symb**

A symbol value.

type id	`xtype.SYMBOL`
category	simple type
derived	no

function

compact | **func**

A function object.

type id	`xtype.FUNCTION`
category	simple type
derived	no

date

compact | **date**

A date object.

type id	`xtype.DATE`
category	simple type
derived	no

error

compact | **err**

An object of any of the error types.

basic
object
string
array
number
boolean
group
more

type id	`xtype.ERROR`
category	simple type
derived	no

regexp

compact | **regex**

A regular expression object.

type id	`xtype.REGEXP`
category	simple type
derived	no

Object Types

object

compact | **obj**

All other objects that are not one of the other specific object types.

type id	`xtype.OBJECT`
category	simple type
derived	yes
derivation	`Match Any:` `empty_object` `single_prop_object` `multi_prop_object`

empty_object

compact | **obj0**

An object with no own properties.

type id	`xtype.EMPTY_OBJECT`
category	extended type
derived	no

single_prop_object

compact | **obj1**

An object with exactly one own property.

type id	`xtype.SINGLE_PROP_OBJECT`
category	extended type
derived	no

multi_prop_object

compact | **obj2+**

An object with more than one own property.

type id	`xtype.MULTI_PROP_OBJECT`
category	extended type
derived	no

non_empty_object

compact `-obj0`

An object with one or more own properties.

type id	`xtype.NON_EMPTY_OBJECT`
category	extended type
derived	yes
derivation	`Match Any:` `single_prop_object` `multi_prop_object`

String Types

string

compact `str`

A string primitive or object.

type id	`xtype.STRING`
category	simple type
derived	yes
derivation	`Match Any:` `empty_string` `whitespace` `single_char_string` `multi_char_string`

whitespace

compact `str_`

A string with one or more of only whitespace characters.

type id	`xtype.WHITESPACE`
category	extended type
derived	no

single_char_string

compact `str1`

A string with exactly one non-whitespace character and zero or more whitespace characters.

type id	`xtype.SINGLE_CHAR_STRING`
category	extended type
derived	no

multi_char_string

compact `str2+`

A string with more than one non-whitespace character and zero or more whitespace characters.

type id	`xtype.MULTI_CHAR_STRING`
category	extended type
derived	no

empty_string

compact `str0`

A string with zero characters.

type id	`xtype.EMPTY_STRING`
category	extended type
derived	no

blank_string

compact `str0_`

A string with zero or more of only whitespace characters.

type id	`xtype.BLANK_STRING`
category	extended type
derived	yes
derivation	`Match Any:` `empty_string` `whitespace`

non_empty_string

compact `-str0`

A string with at least one whitespace or other character.

type id	`xtype.NON_EMPTY_STRING`
category	extended type
derived	yes
derivation	`Match Any:` `whitespace` `single_char_string` `multi_char_string`

non_blank_string

compact `-str0_`

A string with at least one non-whitespace character.

type id	`xtype.NON_BLANK_STRING`
category	extended type
derived	yes
derivation	`Match Any:` `single_char_string` `multi_char_string`

Array Types

array

compact | `arr`

An array object.

type id	`xtype.ARRAY`
category	simple type
derived	yes
derivation	`Match Any:` `empty_array` `single_elem_array` `multi_elem_array`

empty_array

compact | `arr0`

An array with no elements.

type id	`xtype.EMPTY_ARRAY`
category	extended type
derived	no

single_elem_array

compact | `arr1`

An array with exactly one element.

type id	`xtype.SINGLE_ELEM_ARRAY`
category	extended type
derived	no

multi_elem_array

compact | `arr2+`

An array with more than one element.

type id	`xtype.MULTI_ELEM_ARRAY`
category	extended type
derived	no

non_empty_array

compact | `-arr0`

An array with one or more elements.

type id	`xtype.NON_EMPTY_ARRAY`
category	extended type
derived	yes
derivation	`Match Any:` `single_elem_array` `multi_elem_array`

Number Types

number

compact | `num`

A number primitive or object.

type id	`xtype.NUMBER`
category	simple type
derived	yes
derivation	`Match Any:` `zero` `positive_integer` `positive_float` `positive_infinity` `negative_integer` `negative_float` `negative_infinity`

integer

compact | `int`

A number primitive or object with a whole number value.

type id	`xtype.INTEGER`
category	extended type
derived	yes
derivation	`Match Any:` `zero` `positive_integer` `negative_integer`

float

compact | `float`

A number primitive or object with a fractional component.

type id	`xtype.FLOAT`
category	extended type
derived	yes
derivation	`Match Any:` `positive_float` `negative_float`

positive_number

compact | `num+`

A number primitive or object greater than zero.

type id	`xtype.POSITIVE_NUMBER`
category	extended type
derived	yes
derivation	`Match Any:` `positive_integer` `positive_float` `positive_infinity`

negative_number

compact `num-`

A number primitive or object less than zero.

type id	`xtype.NEGATIVE_NUMBER`
category	extended type
derived	yes
derivation	`Match Any:` `negative_integer` `negative_float` `negative_infinity`

infinite_number

compact `inf`

A number primitive or object with a positive or negative infinite value.

type id	`xtype.INFINITE_NUMBER`
category	extended type
derived	yes
derivation	`Match Any:` `positive_infinity` `negative_infinity`

zero

compact `num0`

A number primitive or object equal to zero.

type id	`xtype.ZERO`
category	extended type
derived	no

non_zero_number

compact `-num0`

A number primitive or object with a value other than zero.

type id	`xtype.NON_ZERO_NUMBER`
category	extended type
derived	yes
derivation	`Match Any:` `positive_integer` `positive_float` `positive_infinity` `negative_integer` `negative_float` `negative_infinity`

positive_integer

compact `int+`

basic
object
string
array
number
boolean
group
more

A finite whole number primitive or object greater than zero.

type id	`xtype.POSITIVE_INTEGER`
category	extended type
derived	no

negative_integer

compact | **int-**

A finite whole number primitive or object less than zero.

type id	`xtype.NEGATIVE_INTEGER`
category	extended type
derived	no

positive_float

compact | **float+**

A finite number primitive or object greater than zero with a fractional component.

type id	`xtype.POSITIVE_FLOAT`
category	extended type
derived	no

negative_float

compact | **float-**

A finite number primitive or object less than zero with a fractional component.

type id	`xtype.NEGATIVE_FLOAT`
category	extended type
derived	no

positive_infinity

compact | **inf+**

A number primitive or object with the value of the JavaScript positive Infinity.

type id	`xtype.POSITIVE_INFINITY`
category	extended type
derived	no

negative_infinity

compact | **inf-**

A number primitive or object with the value of the JavaScript negative Infinity.

type id	`xtype.NEGATIVE_INFINITY`
category	extended type
derived	no

basic
object
string
array
number
boolean
group
more

non_positive_number

compact | `-num+`

A number primitive or object less than or equal to zero.

type id	`xtype.NON_POSITIVE_NUMBER`
category	extended type
derived	yes
derivation	`Match Any:` `zero` `negative_integer` `negative_float` `negative_infinity`

non_negative_number

compact | `-num-`

A number primitive or object greater than or equal to zero.

type id	`xtype.NON_NEGATIVE_NUMBER`
category	extended type
derived	yes
derivation	`Match Any:` `zero` `positive_integer` `positive_float` `positive_infinity`

non_infinite_number

compact | `-inf`

A number primitive or object with a non-infinite value.

type id	`xtype.NON_INFINITE_NUMBER`
category	extended type
derived	yes
derivation	`Match Any:` `zero` `positive_integer` `positive_float` `negative_integer` `negative_float`

Boolean Types

boolean

compact | `bool`

A boolean primitive or object.

type id	`xtype.BOOLEAN`
category	simple type
derived	yes
derivation	`Match Any:` `true` `false`

basic
object
string
array
number
boolean
group
more

true

compact `true`

The boolean primitive or object that is true.

type id	`xtype.TRUE`
category	extended type
derived	no

false

compact `false`

The boolean primitive or object that is false.

type id	`xtype.FALSE`
category	extended type
derived	no

Group Types

primitive

compact `prim`

A primitive type other than null or undefined.

type id	`xtype.PRIMITIVE`
category	extended type
derived	yes
derivation	`Match Any:` `true` `false` `empty_string` `whitespace` `single_char_string` `multi_char_string` `zero` `positive_integer` `positive_float` `positive_infinity` `negative_integer` `negative_float` `negative_infinity` `symbol`

nothing

compact `nil`

The null or undefined value.

type id	`xtype.NOTHING`
category	extended type
derived	yes
derivation	`Match Any:` `null` `undefined`

any

compact `any`

Any type.

type id	`xtype.ANY`
category	extended type
derived	yes
derivation	`Match Any:` `null` `undefined` `nan` `true` `false` `empty_string` `whitespace` `single_char_string` `multi_char_string` `zero` `positive_integer` `positive_float` `positive_infinity` `negative_integer` `negative_float` `negative_infinity` `empty_array` `single_elem_array` `multi_elem_array` `empty_object` `single_prop_object` `multi_prop_object` `symbol` `date` `error` `regexp` `function`

basic
object
string
array
number
boolean
group
more

none

compact | **none**

Not of any type.

type id	`xtype.NONE`
category	extended type
derived	no

More Types

Instance Types

Any constructor function is an Instance type. Each constructor function is the Instance type for objects of that constructor.

type id	*not applicable*
category	instance type
derived	no

Self-Validated Types

Self-validated types are user-defined types that provide an implementation of the arbitrary custom validation used in validating that type. Any number of self-validated types can be created and used like, and/or together with, all other types.

type id	*not applicable*
category	self-validated type
derived	no

Derived Types

Derived types are types created from a combination of multiple types, and may be created to validate either on a match of all the component types, or of any of the component types.

type id	*based on composition - only applicable for derivations using only built-in types*
category	derived type
derived	yes
derivation	*component types*

basic
object
string
array
number
boolean
group
more

github | star | watch | xtypejs | licensed under the MIT license

top