JavaScript Coercions Grid

Exception!

Value Coercion WTF

Fixed WTF

This tool is designed to help visualize JS's value coercion corner cases, specifically the ones that are WTFs, and to see what effect fixing some or all of them can have on the landscape of values.

For lots more information on this topic, please check out Fixing Coercion, Not The Symptoms and *You Don't Know JS: Types & Grammar*.

▶ **settings** ← You should try this!

	String(x)	x + ''	x.toString()	Number(x), +x	x * 1	x + 0
0	'0'	'0'	'0'	0	0	0
.0	'0'	'0'	'0'	0	0	0
-0	'0'	'0'	'0'	-0	-0	0
NaN	'NaN'	'NaN'	'NaN'	NaN	NaN	NaN
''	''	''	''	0	0	'0'
' '	' '	' '	' '	0	0	' 0'
'\n\n'	'\n\n'	'\n\n'	'\n\n'	0	0	'\n\n0'
null	'null'	'null'		0	0	0
undefined	'undefined'	'undefined'		NaN	NaN	NaN
true	'true'	'true'	'true'	1	1	1
false	'false'	'false'	'false'	0	0	0
/ /	'/ /'	'/ /'	'/ /'	NaN	NaN	'/ /0'
Infinity	'Infinity'	'Infinity'	'Infinity'	Infinity	Infinity	Infinity
-Infinity	'-Infinity'	'-Infinity'	'-Infinity'	-Infinity	-Infinity	-Infinity
'Infinity'	'Infinity'	'Infinity'	'Infinity'	Infinity	Infinity	'Infinity0'
'-Infinity'	'-Infinity'	'-Infinity'	'-Infinity'	-Infinity	-Infinity	'-Infinity0'
function(){}	'function(){}'	'function(){}'	'function(){}'	NaN	NaN	'function(){}0'
Symbol('')	'Symbol()'		'Symbol()'			
[]	''	''	''	0	0	'0'
[0]	'0'	'0'	'0'	0	0	'00'
[.0]	'0'	'0'	'0'	0	0	'00'
[-0]	'0'	'0'	'0'	0	0	'00'
[NaN]	'NaN'	'NaN'	'NaN'	NaN	NaN	'NaN0'
['']	''	''	''	0	0	'0'
[' ']	' '	' '	' '	0	0	' 0'
['\n\n']	'\n\n'	'\n\n'	'\n\n'	0	0	'\n\n0'
[null]	''	''	''	0	0	'0'
[undefined]	''	''	''	0	0	'0'
[false]	'false'	'false'	'false'	NaN	NaN	'false0'
[true]	'true'	'true'	'true'	NaN	NaN	'true0'
[/ /]	'/ /'	'/ /'	'/ /'	NaN	NaN	'/ /0'
[,]	''	''	''	0	0	'0'
[[]]	''	''	''	0	0	'0'
[Infinity]	'Infinity'	'Infinity'	'Infinity'	Infinity	Infinity	'Infinity0'
[-Infinity]	'-Infinity'	'-Infinity'	'-Infinity'	-Infinity	-Infinity	'-Infinity0'
['Infinity']	'Infinity'	'Infinity'	'Infinity'	Infinity	Infinity	'Infinity0'
['-Infinity']	'-Infinity'	'-Infinity'	'-Infinity'	-Infinity	-Infinity	'-Infinity0'
[function(){}]	'function(){}'	'function(){}'	'function(){}'	NaN	NaN	'function(){}0'
[Symbol('')]						
{}	'[object Object]'	'[object Object]'	'[object Object]'	NaN	NaN	'[object Object]0'
Object.create(null)						
{'':null}	'[object Object]'	'[object Object]'	'[object Object]'	NaN	NaN	'[object Object]0'
{' ':null}	'[object Object]'	'[object Object]'	'[object Object]'	NaN	NaN	'[object Object]0'
{'\n\n':null}	'[object Object]'	'[object Object]'	'[object Object]'	NaN	NaN	'[object Object]0'
{'':undefined}	'[object Object]'	'[object Object]'	'[object Object]'	NaN	NaN	'[object Object]0'
{'':function(){}}	'[object Object]'	'[object Object]'	'[object Object]'	NaN	NaN	'[object Object]0'
{Symbol(''):null}	'[object Object]'	'[object Object]'	'[object Object]'	NaN	NaN	'[object Object]0'

Extended Coercions

Note: These are all deliberately tricky objects and as such the "proper" coercive behavior is a lot more subject to debate. I'll leave these unmarked (for now).

	String(x)	x + ''	x.toString()	Number(x), +x	x * 1	x +
{ toString: 0 }	'0'	'0'	0	0	0	0
{ toString: .0 }	'0'	'0'	0	0	0	0
{ toString: -0 }	'0'	'0'	-0	-0	-0	0
{ toString: NaN }	'NaN'	'NaN'	NaN	NaN	NaN	NaN
{ toString: '' }	''	''	''	0	0	'0'
{ toString: ' ' }	' '	' '	' '	0	0	' 0'
{ toString: '\n\n' }	'\n\n'	'\n\n'	'\n\n'	0	0	'\n\n0'
{ toString: null }	'null'	'null'	null	0	0	0
{ toString: undefined }	'undefined'	'undefined'	undefined	NaN	NaN	NaN
{ toString: false }	'false'	'false'	false	0	0	0
{ toString: true }	'true'	'true'	true	1	1	1
{ toString: [] }			[]			
{ toString: {} }			{}			
{ toString: / / }			{}			
{ toString: Infinity }	'Infinity'	'Infinity'	Infinity	Infinity	Infinity	Infinity
{ toString: -Infinity }	'-Infinity'	'-Infinity'	-Infinity	-Infinity	-Infinity	-Infinity
{ toString: 'Infinity' }	'Infinity'	'Infinity'	'Infinity'	Infinity	Infinity	'Infinity0'
{ toString: '-Infinity' }	'-Infinity'	'-Infinity'	'-Infinity'	-Infinity	-Infinity	'-Infinity0
{ toString: function(){} }			function(){}			
{ toString: Symbol('') }			Symbol()			
{ toString: 0, valueOf: {} }	'0'	'0'	0	0	0	0
{ toString: .0, valueOf: {} }	'0'	'0'	0	0	0	0
{ toString: -0, valueOf: {} }	'0'	'0'	-0	-0	-0	0
{ toString: NaN, valueOf: {} }	'NaN'	'NaN'	NaN	NaN	NaN	NaN
{ toString: '', valueOf: {} }	''	''	''	0	0	'0'
{ toString: ' ', valueOf: {} }	' '	' '	' '	0	0	' 0'
{ toString: '\n\n', valueOf: {}…	'\n\n'	'\n\n'	'\n\n'	0	0	'\n\n0'
{ toString: null, valueOf: {} }	'null'	'null'	null	0	0	0
{ toString: undefined, valueOf:…	'undefined'	'undefined'	undefined	NaN	NaN	NaN
{ toString: false, valueOf: {} }	'false'	'false'	false	0	0	0
{ toString: true, valueOf: {} }	'true'	'true'	true	1	1	1
{ toString: [], valueOf: {} }			[]			
{ toString: {}, valueOf: {} }			{}			
{ toString: / /, valueOf: {} }			{}			
{ toString: Infinity, valueOf: …	'Infinity'	'Infinity'	Infinity	Infinity	Infinity	Infinity
{ toString: -Infinity, valueOf:…	'-Infinity'	'-Infinity'	-Infinity	-Infinity	-Infinity	-Infinity
{ toString: 'Infinity', valueOf…	'Infinity'	'Infinity'	'Infinity'	Infinity	Infinity	'Infinity0'
{ toString: '-Infinity', valueO…	'-Infinity'	'-Infinity'	'-Infinity'	-Infinity	-Infinity	'-Infinity0
{ toString: function(){}, value…			function(){}			
{ toString: Symbol(''), valueOf…			Symbol()			
{ toString: NaN, valueOf: Infin…	'NaN'	'Infinity'	NaN	Infinity	Infinity	Infinity
{ valueOf: 0 }	'[object Object]'	'0'	'[object Object]'	0	0	0
{ valueOf: .0 }	'[object Object]'	'0'	'[object Object]'	0	0	0
{ valueOf: -0 }	'[object Object]'	'0'	'[object Object]'	-0	-0	0
{ valueOf: NaN }	'[object Object]'	'NaN'	'[object Object]'	NaN	NaN	NaN
{ valueOf: '' }	'[object Object]'	''	'[object Object]'	0	0	'0'
{ valueOf: ' ' }	'[object Object]'	' '	'[object Object]'	0	0	' 0'
{ valueOf: '\n\n' }	'[object Object]'	'\n\n'	'[object Object]'	0	0	'\n\n0'
{ valueOf: null }	'[object Object]'	'null'	'[object Object]'	0	0	0
{ valueOf: undefined }	'[object Object]'	'undefined'	'[object Object]'	NaN	NaN	NaN
{ valueOf: false }	'[object Object]'	'false'	'[object Object]'	0	0	0
{ valueOf: true }	'[object Object]'	'true'	'[object Object]'	1	1	1
{ valueOf: [] }	'[object Object]'	'[object Object]'	'[object Object]'	NaN	NaN	'[object Ob
{ valueOf: {} }	'[object Object]'	'[object Object]'	'[object Object]'	NaN	NaN	'[object Ob

{ valueOf: / / }	'[object Object]'	'[object Object]'	'[object Object]'	NaN	NaN	'[object Ob
{ valueOf: Infinity }	'[object Object]'	'Infinity'	'[object Object]'	Infinity	Infinity	Infinity
{ valueOf: -Infinity }	'[object Object]'	'-Infinity'	'[object Object]'	-Infinity	-Infinity	-Infinity
{ valueOf: 'Infinity' }	'[object Object]'	'Infinity'	'[object Object]'	Infinity	Infinity	'Infinity0'
{ valueOf: '-Infinity' }	'[object Object]'	'-Infinity'	'[object Object]'	-Infinity	-Infinity	'-Infinity0
{ valueOf: function(){} }	'[object Object]'	'[object Object]'	'[object Object]'	NaN	NaN	'[object Ob
{ valueOf: Symbol('') }	'[object Object]'		'[object Object]'			
{ valueOf: 0, toString: '' }	''	'0'	''	0	0	0
{ valueOf: .0, toString: '' }	''	'0'	''	0	0	0
{ valueOf: -0, toString: '' }	''	'0'	''	-0	-0	0
{ valueOf: NaN, toString: '' }	''	'NaN'	''	NaN	NaN	NaN
{ valueOf: '', toString: '' }	''	''	''	0	0	'0'
{ valueOf: ' ', toString: ' ' }	' '	' '	' '	0	0	' 0'
{ valueOf: '\n\n', toString: '\…	'\n\n'	'\n\n'	'\n\n'	0	0	'\n\n0'
{ valueOf: null, toString: '' }	''	'null'	''	0	0	0
{ valueOf: undefined, toString:…	''	'undefined'	''	NaN	NaN	NaN
{ valueOf: false, toString: '' }	''	'false'	''	0	0	0
{ valueOf: true, toString: '' }	''	'true'	''	1	1	1
{ valueOf: [], toString: '' }	''	''	''	0	0	'0'
{ valueOf: {}, toString: '' }	''	''	''	0	0	'0'
{ valueOf: / /, toString: '' }	''	''	''	0	0	'0'
{ valueOf: Infinity, toString: …	''	'Infinity'	''	Infinity	Infinity	Infinity
{ valueOf: -Infinity, toString:…	''	'-Infinity'	''	-Infinity	-Infinity	-Infinity
{ valueOf: 'Infinity', toString…	'Infinity'	'Infinity'	'Infinity'	Infinity	Infinity	'Infinity0'
{ valueOf: '-Infinity', toStrin…	'-Infinity'	'-Infinity'	'-Infinity'	-Infinity	-Infinity	'-Infinity0
{ valueOf: function(){}, toStri…			function(){}			
{ valueOf: Symbol(''), toString…			Symbol()			