



Class viewer

Sources	Report format	Report include	Filter
Tree	Full	All inherited	
2018	CLASSES	SHEET	SCRIPT
Search	<input type="text"/> <input type="button" value="Start"/> <input checked="" type="checkbox"/> Classes <input checked="" type="checkbox"/> Sources <input checked="" type="checkbox"/> In depth		

test	CJC	Cdialog	Chistory	Cloom	Cselect
2018	CasyncQueue	Cdrag	ChistoryEvent	Cmenu	Csource
AsyncQ	CceTable	CfileReader	Churdles	Cmenu_Row	CsourceIndex
CasyncQueue.js	Ccell	CfileStore	Cindex	Cmenus	Cstack
ClassViewer	CclassDef	CfileStore_ReadCatalogue	CjobOwner	CobjSig	Cstacks
CclassViewer.js	CclassDef_elements	CfileStore_ReadCatalogue	CjobOwners	Coptions	Cstore
Cclasses.js	CclassElement	_Item	Cjoinder	CoptionsElement	Ctask
CcvCfg.js	CclassFileParser	CfileStore_WPending	CjoinderAs	Corganisation	Ctasks
Csample.js	CclassIndex	CfileStore_WPending_Item	CjoinderCounter	Coutline	Cteam
Csources.js	CclassViewer	CfileWriter	CjoinderDelay	Cpage	Cteam_Member
FileStore	CcolDef	Cflags	CjoinderLogic	CphpFileWriter	Cteams
CfileReader.js	CcolDefs	CformElement	CkeyWatcher	Cprogress	CtickBox
CfileWriter.js	CcommandTree	Cgrid	CkeyWatcher_Watch	Cproject	Ctree
Hurdles	CcommonColDef	Cgroup	Clatch	Cprojects	Cuser
Ccategory.js	CcvCfg	Cgroups	Clist	Crotator	Cusers
Cflags.js	CcvCfg_package	Chelper1	CliveList	Crow	Cwire
ChurdleStructs.js	CcvCfg_package_index	Chelper2	ClocalStore	Csample	
Churdles.js					
Cowner.js					
Ctask.js					
ImpExp					
CJC.js					
dist					
developer					
CJC.js					
api_files					
CJC.js					
CasyncQueue.js					

/test/2018/ImpExp/dist/developer/api_files/CJC.js

CJC

552	f Block	Class ClassName	Alias for .doNotStringify()
27	f DeepCopy	OriginalObj	Copy a functional object deeply ie. every bit is a copy not a copy of a reference
378	f DumpConstructorNames		Debugging aid to list all registered constructors Returns a string.
190	f FromJ	Jstring	Convert a JSON string into a complete 'classy' object (if that's what it represents.) Practically a drop-in replacement for JSON.parse() Note: If Jstring isn't a string this returns "CJC.FromJ": EXPECTED STRING. Given <type>" If the JSON.parse fails then thus returns "CJC.FromJ": PARSE FAILED. Info <info>" So as 99% of the time you're expecting an object to be returned, testing for a string or Object.isObject() or instanceof will trap issues by being out-of-band and you can handle them there. This shouldn't throw any exceptions.
365	f IsConstructorRegistered	ClassName	Is the constructor registered?
358	f RegisterConstructors	ArrayOfClasses	Tell CJC about classes it needs to know how to reconstruct For example: CJC.RegisterConstructors([CmyClass,CmyOtherClass]);
484	f RemoveComponentUplinks	Parent Label ArraysAlso	Remove references to Parent which may exist in contained objects. The Label string is how the contained objects refer to the parent. To search arrays for elements which point to the parent container add the optional ArraysAlso argument as true. Example: <pre>class Cfoo{ get saveString(){ CJC.RemoveComponentUplinks(this,'myContainer',true); return CJC.ToJ(this); } }</pre>
499	f ReplaceComponentUplinks	Parent Label ArraysAlso	This re-links components to the Parent container. Typically call at end of ccontainer constructor. Example: <pre>class Cfoo{ constructor(){ stuff gets constructed CJC.ReplaceComponentUplinks(this,'myContainer',true); } }</pre>
50	f ToJ	FromThing	Export something, typically a 'classy' object to JSON Returns a JSON string. This is a drop-in replacement for JSON.stringify() FromThing is whatever you want to convert to JSON. Note: Deals with recursion, Dates, RegExp.
384	about		What's the version? Returns a date string.
547	f doNotStringify	Class ClassName	Prevent certain classes being stringified. Some objects are far too complex, unnecessary and screw-up reconstruction so drop these and replace with an information message. A typical case is where a DOM element

			is stored by an active component. This is only of transient use. A good idea for instances of major libraries such as jQuery.
175	f fromJSON	<i>Jstring</i>	Alias for CJC.FromJ()
40	f toJSON	<i>FromThing</i>	Alias for CJC.ToJ()
70	f _AddClassNames	<i>ListRef</i> <i>FromThing</i>	This where the hard work of preparing for stringifying goes on. It works down the object values, sub-objects and sub-sub-objects recursively ListRef points to our private list of what we've already exported. If we find we're repeating ourselves then we'll substitute a place-holder with a reference (in a property .CJC_PLACEHOLDER) which points to the listed object. Importantly, we embed the name of any object's constructor in the object itself (as .class_name_) so we'll know how to reconstruct it when performing .FromJ() . We also deal with some of the issues that JSON doesn't handle. In particular Dates, RegExps, null and NaN.
319	f _AddToDeferred	<i>ListRef</i> <i>AnObj</i>	Helper for _ApplyDeferred()
163	f _AddToExportedList	<i>ListRef</i> <i>AnObj</i>	Helper for _AddClassNames() Append object (with .CJC-IDENTITY property) to already exported list
272	f _ApplyDeferred	<i>ListRef</i> <i>Obj</i>	At this point we've got an object (or sub-object) which might have one or more place-holders in it. The procedure is to make a note of likely objects with their .CJC_IDENTITY properties so we can find them when we come across a place-holder and then substitute. This recurses down the object tree in the same way as when we were exporting so we're fairly certain of reaching place-holders AFTER what they represent.
306	f _FetchDeferred	<i>ListRef</i> <i>Id</i>	Helper for _ApplyDeferred() Deferred objects are full objects with a CJC_IDENTITY property Return the object to replace the place-holder, or if not found, its an error that should never happen, an error message string.
371	f _GetConstructor	<i>ClassName</i>	Fetch a constructor by name
153	f _InExportedList	<i>ListRef</i> <i>AnObj</i>	Helper for _AddClassNames() Return false or object's CJC-IDENTITY depending on whether we've already got the object in the list.
393	f _JsonErrToString	<i>Err</i> <i>Jstring</i>	JSON has thrown an exception which we've caught. Convert it into a string ready to be returned as an out-of-band response by FromJ
329	f _NewObjList		Create a new list for objects and return a random id which will be used to reference it. We go to this trouble in case there are multiple threads which have to be kept separate.
222	f _Reconstructor	<i>Key</i> <i>Val</i>	This is called by JSON. parse after each element is read. It is a bottom-up process with sub-sub-elements being processed before sub-elements etc. We do three things: * Deal with the special tokens we need for null and NaN * Deal with properly reconstituting Dates and RegExps * Deal with creating a proper class from a raw object
339	f _RemoveObjList	<i>ListRef</i>	Remove an object list. Note : This also removes the CJC_IDENTITY properties from the objects so as to tidy them up.
510	f _ReplaceUpLink	<i>Parent</i> <i>Label</i> <i>With</i> <i>ArraysAlso</i>	Seeks out immediate properties with a Label property and sets those properties to With . With will be null or the Parent container. If ArraysAlso is truthy then do the same for elements inside arrays also.