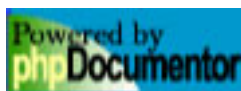


# dbdMVC



# Contents

<a href="#">Package dbdCommon Procedural Elements</a>	2
<a href="#">FlashLoaderSWFO.php</a>	2
<a href="#">GDColor.php</a>	3
<a href="#">InputCSS.php</a>	4
<a href="#">QTLoder.php</a>	5
<a href="#">wmArrays.php</a>	6
<a href="#">wmFileSystem.php</a>	7
<a href="#">wmSort.php</a>	8
<a href="#">wmString.php</a>	9
<a href="#">Package dbdCommon Classes</a>	10
<a href="#">Class FlashLoaderSWFO</a>	10
<a href="#">Constructor construct</a>	10
<a href="#">Method addMovie</a>	11
<a href="#">Method getCommonJS</a>	11
<a href="#">Method getCommonJSPath</a>	11
<a href="#">Method getMovie</a>	11
<a href="#">Method getMovies</a>	12
<a href="#">Method hasMovies</a>	12
<a href="#">Method loadMovie</a>	12
<a href="#">Method loadMovies</a>	13
<a href="#">Method setAtts</a>	13
<a href="#">Class GDColor</a>	13
<a href="#">Constructor construct</a>	14
<a href="#">Method getAlpha</a>	14
<a href="#">Method getBlue</a>	14
<a href="#">Method getColor</a>	14
<a href="#">Method getGreen</a>	15
<a href="#">Method getHex</a>	15
<a href="#">Method getRed</a>	15
<a href="#">Method hex2arr</a>	15
<a href="#">Method int2arr</a>	16
<a href="#">Method destruct</a>	16
<a href="#">Class InputCSS</a>	16
<a href="#">Method button</a>	16
<a href="#">Method checkbox</a>	17
<a href="#">Method file</a>	17
<a href="#">Method radio</a>	18
<a href="#">Method reset</a>	18
<a href="#">Class QTLoder</a>	19
<a href="#">Constructor construct</a>	19
<a href="#">Method addMovie</a>	20
<a href="#">Method getCommonJS</a>	20
<a href="#">Method getCommonJSPath</a>	20

<a href="#">Method <code>getMovie</code></a>	20
<a href="#">Method <code>getMovies</code></a>	21
<a href="#">Method <code>hasMovies</code></a>	21
<a href="#">Method <code>loadMovie</code></a>	21
<a href="#">Method <code>loadMovies</code></a>	22
<a href="#">Method <code>setAtts</code></a>	22
<a href="#">Class <code>wmArrays</code></a>	22
<a href="#">Method <code>arrayMergeRecursiveSumFields</code></a>	23
<a href="#">Method <code>arrayRemoveValue</code></a>	23
<a href="#">Method <code>arraySearchRecursive</code></a>	23
<a href="#">Method <code>arrayValuesRecursive</code></a>	24
<a href="#">Method <code>explodeNewline</code></a>	24
<a href="#">Method <code>implodeObjArray</code></a>	25
<a href="#">Method <code>pregExplode</code></a>	25
<a href="#">Class <code>wmFileSystem</code></a>	25
<a href="#">Method <code>bytesToHuman</code></a>	26
<a href="#">Method <code>dirSize</code></a>	26
<a href="#">Method <code>contentType</code></a>	26
<a href="#">Method <code>scanDir</code></a>	27
<a href="#">Class <code>wmSort</code></a>	27
<a href="#">Method <code>multiASort</code></a>	28
<a href="#">Method <code>multiSort</code></a>	28
<a href="#">Class <code>wmString</code></a>	29
<a href="#">Class Constant <code>TIME_LEN_FORMAT_LONG</code></a>	29
<a href="#">Class Constant <code>TIME_LEN_FORMAT_SHORT</code></a>	29
<a href="#">Class Constant <code>TIME_LEN_LABEL_AGO</code></a>	29
<a href="#">Class Constant <code>TIME_LEN_LABEL_DAY</code></a>	29
<a href="#">Class Constant <code>TIME_LEN_LABEL_HRS</code></a>	29
<a href="#">Class Constant <code>TIME_LEN_LABEL_LAST</code></a>	29
<a href="#">Class Constant <code>TIME_LEN_LABEL_MIN</code></a>	29
<a href="#">Class Constant <code>TIME_LEN_LABEL_MON</code></a>	29
<a href="#">Class Constant <code>TIME_LEN_LABEL_SEC</code></a>	29
<a href="#">Class Constant <code>TIME_LEN_LABEL_TODAY</code></a>	29
<a href="#">Class Constant <code>TIME_LEN_LABEL_WEEK</code></a>	30
<a href="#">Class Constant <code>TIME_LEN_LABEL_YEAR</code></a>	30
<a href="#">Class Constant <code>TIME_LEN_LABEL_YESTERDAY</code></a>	30
<a href="#">Class Constant <code>TIME_LEN_TIME_DAY</code></a>	30
<a href="#">Class Constant <code>TIME_LEN_TIME_HRS</code></a>	30
<a href="#">Class Constant <code>TIME_LEN_TIME_MIN</code></a>	30
<a href="#">Class Constant <code>TIME_LEN_TIME_MON</code></a>	30
<a href="#">Class Constant <code>TIME_LEN_TIME_SEC</code></a>	30
<a href="#">Class Constant <code>TIME_LEN_TIME_WEEK</code></a>	30
<a href="#">Class Constant <code>TIME_LEN_TIME_YEAR</code></a>	30
<a href="#">Class Constant <code>TIME_LEN_TYPE_DAY</code></a>	30
<a href="#">Class Constant <code>TIME_LEN_TYPE_HRS</code></a>	30
<a href="#">Class Constant <code>TIME_LEN_TYPE_MIN</code></a>	30
<a href="#">Class Constant <code>TIME_LEN_TYPE_MON</code></a>	30
<a href="#">Class Constant <code>TIME_LEN_TYPE_SEC</code></a>	31
<a href="#">Class Constant <code>TIME_LEN_TYPE_WEEK</code></a>	31

<a href="#">Class Constant TIME_LEN_TYPE_YEAR</a>	31
<a href="#">Class Constant VALID_DATE</a>	31
<a href="#">Class Constant VALID_EMAIL</a>	31
<a href="#">Method dateValid</a>	31
<a href="#">Method dayOfWeek</a>	31
<a href="#">Method emailEncode</a>	32
<a href="#">Method emailValid</a>	32
<a href="#">Method hanukkahStart</a>	32
<a href="#">Method jsSpecialChars</a>	33
<a href="#">Method lcFirst</a>	33
<a href="#">Method makePlural</a>	34
<a href="#">Method makeSingular</a>	34
<a href="#">Method numberSuffix</a>	34
<a href="#">Method randomWord</a>	35
<a href="#">Method superCap2UnderLower</a>	35
<a href="#">Method superCapAddSpaces</a>	35
<a href="#">Method timeFormat</a>	36
<a href="#">Method timeLengthFormat</a>	36
<a href="#">Method timePastFormat</a>	36
<a href="#">Method truncate</a>	37
<a href="#">Method underLower2SuperCap</a>	37
<a href="#">Method wkt2kml</a>	38
<b><a href="#">Package FirePHP Procedural Elements</a></b>	40
<a href="#">FirePHP.class.php</a>	40
<b><a href="#">Package FirePHP Classes</a></b>	42
<a href="#">Class FirePHP</a>	42
<a href="#">Class Constant DUMP</a>	42
<a href="#">Class Constant ERROR</a>	42
<a href="#">Class Constant EXCEPTION</a>	43
<a href="#">Class Constant GROUP_END</a>	43
<a href="#">Class Constant GROUP_START</a>	43
<a href="#">Class Constant INFO</a>	43
<a href="#">Class Constant LOG</a>	43
<a href="#">Class Constant TABLE</a>	43
<a href="#">Class Constant TRACE</a>	43
<a href="#">Class Constant VERSION</a>	44
<a href="#">Class Constant WARN</a>	44
<a href="#">Var \$convertAssertionErrorsToExceptions</a>	44
<a href="#">Var \$enabled</a>	44
<a href="#">Var \$inExceptionHandler</a>	44
<a href="#">Var \$instance</a>	45
<a href="#">Var \$messageIndex</a>	45
<a href="#">Var \$objectFilters</a>	45
<a href="#">Var \$objectStack</a>	46
<a href="#">Var \$options</a>	46
<a href="#">Var \$throwAssertionExceptions</a>	46
<a href="#">Var \$throwErrorExceptions</a>	46
<a href="#">Constructor __construct</a>	47

<a href="#">Method assertionHandler</a>	47
<a href="#">Method detectClientExtension</a>	47
<a href="#">Method dump</a>	47
<a href="#">Method encodeObject</a>	48
<a href="#">Method encodeTable</a>	48
<a href="#">Method error</a>	48
<a href="#">Method errorHandler</a>	49
<a href="#">Method exceptionHandler</a>	49
<a href="#">Method fb</a>	50
<a href="#">Method getEnabled</a>	50
<a href="#">Method getInstance</a>	50
<a href="#">Method getOptions</a>	51
<a href="#">Method getUserAgent</a>	51
<a href="#">Method group</a>	51
<a href="#">Method groupEnd</a>	51
<a href="#">Method info</a>	52
<a href="#">Method init</a>	52
<a href="#">Method is_utf8</a>	52
<a href="#">Method jsonEncode</a>	53
<a href="#">Method log</a>	53
<a href="#">Method newException</a>	53
<a href="#">Method registerAssertionHandler</a>	54
<a href="#">Method registerErrorHandler</a>	54
<a href="#">Method registerExceptionHandler</a>	55
<a href="#">Method setEnabled</a>	55
<a href="#">Method setHeader</a>	55
<a href="#">Method setObjectFilter</a>	55
<a href="#">Method setOptions</a>	56
<a href="#">Method setProcessorUrl</a>	56
<a href="#">Method setRendererUrl</a>	57
<a href="#">Method table</a>	57
<a href="#">Method trace</a>	57
<a href="#">Method warn</a>	58
<a href="#">Method _escapeTrace</a>	58
<a href="#">Method _escapeTraceFile</a>	59
<a href="#">Method _standardizePath</a>	59
<a href="#">Method _sleep</a>	59
<a href="#">Package JSMin Classes</a>	60
<a href="#">Class JSMin</a>	60
<a href="#">Class Constant ORD_LF</a>	61
<a href="#">Class Constant ORD_SPACE</a>	61
<a href="#">Var \$a</a>	61
<a href="#">Var \$b</a>	61
<a href="#">Var \$input</a>	61
<a href="#">Var \$inputIndex</a>	62
<a href="#">Var \$inputLength</a>	62
<a href="#">Var \$lookAhead</a>	62
<a href="#">Var \$output</a>	62
<a href="#">Constructor _construct</a>	62

<a href="#">Method action</a>	63
<a href="#">Method get</a>	63
<a href="#">Method isAlphaNum</a>	63
<a href="#">Method min</a>	63
<a href="#">Method minify</a>	64
<a href="#">Method next</a>	64
<a href="#">Method peek</a>	64
<a href="#">Class JSMinException</a>	64
<b><a href="#">Package dbdMVC Procedural Elements</a></b>	66
<a href="#">dbdMVC.php</a>	66
<a href="#">Function dbdLog</a>	66
<a href="#">Define DBD_CACHE_DIR</a>	67
<a href="#">Define DBD_DEBUG_ALL</a>	67
<a href="#">Define DBD_DEBUG_CSS</a>	67
<a href="#">Define DBD_DEBUG_DB</a>	67
<a href="#">Define DBD_DEBUG_HTML</a>	67
<a href="#">Define DBD_DEBUG_JS</a>	67
<a href="#">Define DBD_DS</a>	67
<a href="#">Define DBD_FIREPHP_DIR</a>	67
<a href="#">Define DBD_MVC_CLI</a>	67
<a href="#">Define DBD_PLUG_DIR</a>	67
<a href="#">Define DBD_SMARTY_DIR</a>	67
<a href="#">Define DBD_SMARTY_PLUG_DIR</a>	67
<a href="#">Define DBD_XAJAX_DIR</a>	67
<a href="#">fb.php</a>	69
<a href="#">Function fb</a>	69
<a href="#">dbconn.inc</a>	70
<a href="#">Define DBCONN_DEFAULT_DB</a>	70
<a href="#">Define DBCONN_INCLUDED</a>	70
<a href="#">Define DB_DRIVER</a>	70
<a href="#">Define DB_HOST</a>	70
<a href="#">Define DB_PASS</a>	70
<a href="#">Define DB_PORT</a>	70
<a href="#">Define DB_USER</a>	70
<a href="#">Index.php</a>	71
<a href="#">index.php</a>	72
<a href="#">Define DBD_APP_DIR</a>	72
<a href="#">Define DBD_DOC_ROOT</a>	72
<a href="#">Define DBD_MVC_DIR</a>	72
<a href="#">dbdController.php</a>	73
<a href="#">dbdCSS.php</a>	74
<a href="#">dbdDB.php</a>	75
<a href="#">dbdDispatcher.php</a>	76
<a href="#">dbdEmptyController.php</a>	77
<a href="#">dbdError.php</a>	78
<a href="#">dbdException.php</a>	79
<a href="#">dbdHoldableException.php</a>	80
<a href="#">dbdInfo.php</a>	81
<a href="#">dbdJS.php</a>	82

<a href="#">dbdLoader.php</a>	83
Function <a href="#">autoload</a>	83
<a href="#">dbdModel.php</a>	84
<a href="#">dbdRequest.php</a>	85
<a href="#">dbdRouter.php</a>	86
<a href="#">dbdSmarty.php</a>	87
<a href="#">dbdURI.php</a>	88
<a href="#">dbdXajax.php</a>	89
<a href="#">dbdXajaxJS.php</a>	90
<a href="#">wmGD.php</a>	91
Function <a href="#">colorizeImage</a>	91
Function <a href="#">cropImage</a>	91
Function <a href="#">cropImageReal</a>	91
Function <a href="#">cropResizeImage</a>	91
Function <a href="#">imageCreateFrom</a>	91
Function <a href="#">imagefilledellipsealpha</a>	92
Function <a href="#">imageLineThick</a>	92
Function <a href="#">imageType</a>	92
Function <a href="#">imageTypesArray</a>	92
Function <a href="#">outputAndDestroy</a>	92
Function <a href="#">reflectImage</a>	92
Function <a href="#">resizeImage</a>	93
Function <a href="#">stretchCopyButton</a>	93
<b><a href="#">Package dbdMVC Classes</a></b>	94
Class <a href="#">dbdController</a>	94
Var <a href="#">\$model</a>	94
Var <a href="#">\$router</a>	95
Var <a href="#">\$view</a>	95
Var <a href="#">\$xajax</a>	95
Var <a href="#">\$xajaxResponse</a>	95
Constructor <a href="#">construct</a>	96
Method <a href="#">assignAllParams</a>	96
Method <a href="#">autoExec</a>	96
Method <a href="#">autoRender</a>	97
Method <a href="#">dnit</a>	97
Method <a href="#">doDefault</a>	97
Method <a href="#">doXajax</a>	98
Method <a href="#">forward</a>	98
Method <a href="#">getAction</a>	98
Method <a href="#">getAppDir</a>	98
Method <a href="#">getController</a>	99
Method <a href="#">getParam</a>	99
Method <a href="#">getParams</a>	99
Method <a href="#">getTemplate</a>	99
Method <a href="#">getUploadInfo</a>	100
Method <a href="#">getURL</a>	100
Method <a href="#">init</a>	100
Method <a href="#">noRender</a>	101
Method <a href="#">registerXajaxActions</a>	101

<a href="#">Method setParam</a>	101
<a href="#">Method setTemplate</a>	102
<a href="#">Method unsetParam</a>	102
<a href="#">Method destruct</a>	102
<a href="#">Class dbdCSS</a>	103
<a href="#">Class Constant BLOCK_CLOSE_REGEX</a>	103
<a href="#">Class Constant BLOCK_OPEN_REGEX</a>	103
<a href="#">Class Constant BUTT_DEF_TAG_REGEX</a>	104
<a href="#">Class Constant BUTT_GEN_TAG_REGEX</a>	104
<a href="#">Class Constant BUTT_GEN_TPL_TAG_REGEX</a>	104
<a href="#">Class Constant BUTT_PROP_DEF_DISABLED_REGEX</a>	104
<a href="#">Class Constant BUTT_PROP_DEF_HOVER_REGEX</a>	104
<a href="#">Class Constant BUTT_PROP_DEF_REGEX</a>	104
<a href="#">Class Constant BUTT_TAG_REGEX</a>	104
<a href="#">Class Constant CACHE_INFO_FILES</a>	105
<a href="#">Class Constant CACHE_INFO_SPRITE</a>	105
<a href="#">Class Constant CSS_EXT_REGEX</a>	105
<a href="#">Class Constant DEBUG_REGEX</a>	105
<a href="#">Class Constant DIR_DELIM_REGEX</a>	105
<a href="#">Class Constant IMPORT_REGEX</a>	105
<a href="#">Class Constant MATH_CALL_REGEX</a>	105
<a href="#">Class Constant MATH_SAFE_REGEX</a>	106
<a href="#">Class Constant VAR_CALL_REGEX</a>	106
<a href="#">Class Constant VAR_DEF_REGEX</a>	106
<a href="#">Class Constant VAR_TAG_REGEX</a>	106
<a href="#">Method doCombine</a>	106
<a href="#">Method genURL</a>	106
<a href="#">Method init</a>	107
<a href="#">Class dbdDB</a>	107
<a href="#">Class Constant COMP_BETWEEN</a>	107
<a href="#">Class Constant COMP_EQ</a>	107
<a href="#">Class Constant COMP_GT</a>	107
<a href="#">Class Constant COMP_GTEQ</a>	107
<a href="#">Class Constant COMP_IN</a>	108
<a href="#">Class Constant COMP_LIKE</a>	108
<a href="#">Class Constant COMP_LT</a>	108
<a href="#">Class Constant COMP_LTEQ</a>	108
<a href="#">Class Constant COMP_NBETWEEN</a>	108
<a href="#">Class Constant COMP_NEQ</a>	108
<a href="#">Class Constant COMP_NIN</a>	108
<a href="#">Class Constant COMP_NLIKE</a>	108
<a href="#">Class Constant COMP_NULLEQ</a>	108
<a href="#">Class Constant COMP_TYPE</a>	108
<a href="#">Class Constant DATE_FORMAT</a>	108
<a href="#">Class Constant DBCONN_PATH</a>	108
<a href="#">Class Constant TIME_FORMAT</a>	109
<a href="#">Method date</a>	109
<a href="#">Method getInstance</a>	109
<a href="#">Method nextAutoID</a>	110



<a href="#">Method prepExec</a>	110
<a href="#">Method time</a>	111
<a href="#">Method call</a>	111
<a href="#">Class dbdDispatcher</a>	111
<a href="#">Class Constant ACTION_PREFIX</a>	112
<a href="#">Class Constant BUILTIN_PREFIX</a>	112
<a href="#">Class Constant DEFAULT_ACTION</a>	112
<a href="#">Class Constant DEFAULT_CONTROLLER</a>	112
<a href="#">Class Constant ERROR_CONTROLLER</a>	112
<a href="#">Class Constant FALLBACK_CONTROLLER</a>	112
<a href="#">Class Constant MAGIC_ACTION</a>	113
<a href="#">Class Constant XAJAX_ACTION_PREFIX</a>	113
<a href="#">Constructor construct</a>	113
<a href="#">Method dispatch</a>	113
<a href="#">Class dbdEmptyController</a>	114
<a href="#">Class dbdError</a>	114
<a href="#">Class Constant ERR_400</a>	114
<a href="#">Class Constant ERR_401</a>	114
<a href="#">Class Constant ERR_403</a>	114
<a href="#">Class Constant ERR_404</a>	115
<a href="#">Class Constant ERR_408</a>	115
<a href="#">Class Constant ERR_500</a>	115
<a href="#">Method doDefault</a>	115
<a href="#">Method doError</a>	115
<a href="#">Class dbdException</a>	116
<a href="#">Constructor construct</a>	116
<a href="#">Class dbdHoldableException</a>	116
<a href="#">Var \$held</a>	117
<a href="#">Var \$hold</a>	117
<a href="#">Method getHeld</a>	117
<a href="#">Method hold</a>	117
<a href="#">Method intercept</a>	118
<a href="#">Method release</a>	118
<a href="#">Class dbdInfo</a>	118
<a href="#">Method doDefault</a>	119
<a href="#">Class dbdJS</a>	119
<a href="#">Class Constant DIR_DELIM_REGEX</a>	119
<a href="#">Method addFile</a>	120
<a href="#">Method doCombine</a>	120
<a href="#">Method doDefault</a>	120
<a href="#">Method doGet</a>	120
<a href="#">Method genURL</a>	121
<a href="#">Method init</a>	121
<a href="#">Method output</a>	121
<a href="#">Class dbdLoader</a>	121
<a href="#">Class Constant DEFAULT_CONTROLLER_DIR</a>	122
<a href="#">Class Constant DEFAULT_MODEL_DIR</a>	122
<a href="#">Method addControllerDir</a>	122
<a href="#">Method addPath</a>	122

<a href="#">Method <u>getControllerDirs</u></a>	123
<a href="#">Method <u>getInstance</u></a>	123
<a href="#">Method <u>getModelDir</u></a>	123
<a href="#">Method <u>getPaths</u></a>	123
<a href="#">Method <u>load</u></a>	124
<a href="#">Method <u>loadClass</u></a>	124
<a href="#">Method <u>search</u></a>	124
<a href="#">Method <u>setControllerDir</u></a>	125
<a href="#">Method <u>setModelDir</u></a>	125
<a href="#">Class <u>dbdModel</u></a>	126
<a href="#">Class Constant <u>CONST_TABLE_KEY</u></a>	126
<a href="#">Class Constant <u>CONST_TABLE_NAME</u></a>	126
<a href="#">Var <u>\$data</u></a>	126
<a href="#">Var <u>\$db</u></a>	127
<a href="#">Var <u>\$id</u></a>	127
<a href="#">Constructor <u>__construct</u></a>	127
<a href="#">Method <u>delete</u></a>	127
<a href="#">Method <u>ensureDB</u></a>	128
<a href="#">Method <u>getAll</u></a>	128
<a href="#">Method <u>getCount</u></a>	128
<a href="#">Method <u>getData</u></a>	129
<a href="#">Method <u>getDB</u></a>	129
<a href="#">Method <u>getID</u></a>	129
<a href="#">Method <u>init</u></a>	129
<a href="#">Method <u>initFields</u></a>	130
<a href="#">Method <u>save</u></a>	130
<a href="#">Method <u>setDB</u></a>	130
<a href="#">Method <u>__call</u></a>	131
<a href="#">Method <u>__get</u></a>	131
<a href="#">Method <u>__isset</u></a>	131
<a href="#">Method <u>__set</u></a>	132
<a href="#">Method <u>__unset</u></a>	132
<a href="#">Class <u>dbdMVC</u></a>	132
<a href="#">Class Constant <u>VERSION</u></a>	133
<a href="#">Method <u>debugMode</u></a>	133
<a href="#">Method <u>e</u></a>	133
<a href="#">Method <u>exposePHPInfo</u></a>	134
<a href="#">Method <u>getAppDir</u></a>	134
<a href="#">Method <u>getAppName</u></a>	134
<a href="#">Method <u>getErrorHandler</u></a>	134
<a href="#">Method <u>getErrorLog</u></a>	135
<a href="#">Method <u>getExecutionTime</u></a>	135
<a href="#">Method <u>getFallbackController</u></a>	135
<a href="#">Method <u>getInstance</u></a>	135
<a href="#">Method <u>getRequest</u></a>	136
<a href="#">Method <u>getRouter</u></a>	136
<a href="#">Method <u>logExecutionTime</u></a>	136
<a href="#">Method <u>PHPisExposed</u></a>	136
<a href="#">Method <u>run</u></a>	137

<a href="#">Method setAppDir</a>	137
<a href="#">Method setAppName</a>	137
<a href="#">Method setDebugMode</a>	138
<a href="#">Method setErrorController</a>	138
<a href="#">Method setErrorLog</a>	139
<a href="#">Method setFallbackController</a>	139
<a href="#">Class dbdRequest</a>	139
<a href="#">Constructor construct</a>	140
<a href="#">Method get</a>	140
<a href="#">Method getCookie</a>	141
<a href="#">Method getEnv</a>	141
<a href="#">Method getHeader</a>	141
<a href="#">Method getPost</a>	142
<a href="#">Method getQuery</a>	142
<a href="#">Method getRequestURI</a>	143
<a href="#">Method getServer</a>	143
<a href="#">Method has</a>	143
<a href="#">Method isXmlHttpRequest</a>	144
<a href="#">Method set</a>	144
<a href="#">Method setRequestURI</a>	144
<a href="#">Method isset</a>	145
<a href="#">Class dbdRouter</a>	145
<a href="#">Constructor construct</a>	145
<a href="#">Method addParam</a>	146
<a href="#">Method getAction</a>	146
<a href="#">Method getController</a>	146
<a href="#">Method getParam</a>	146
<a href="#">Method getParams</a>	147
<a href="#">Method getURL</a>	147
<a href="#">Method setAction</a>	147
<a href="#">Method setController</a>	148
<a href="#">Method setParam</a>	148
<a href="#">Method unsetParam</a>	148
<a href="#">Class dbdSmarty</a>	149
<a href="#">Class Constant CACHE_DIR</a>	149
<a href="#">Class Constant COMPILED_TEMPLATE_DIR</a>	149
<a href="#">Class Constant CONFIG_DIR</a>	150
<a href="#">Class Constant TEMPLATE_DIR</a>	150
<a href="#">Class Constant VIEW_DIR</a>	150
<a href="#">Constructor construct</a>	150
<a href="#">Method addCSS</a>	150
<a href="#">Method addCSSVar</a>	151
<a href="#">Method addFlashMovie</a>	151
<a href="#">Method addJS</a>	151
<a href="#">Method addJSVar</a>	152
<a href="#">Method addQTMovie</a>	152
<a href="#">Method clearCSS</a>	152
<a href="#">Method clearJS</a>	153
<a href="#">Method dbdBeautify</a>	153

<a href="#">Method dbdIncludeFiles</a>	153
<a href="#">Method dbdMinify</a>	153
<a href="#">Method dbdTag</a>	154
<a href="#">Method display</a>	154
<a href="#">Method fetch</a>	155
<a href="#">Method getCSSVar</a>	155
<a href="#">Method getJSVar</a>	155
<a href="#">Method initFlash</a>	156
<a href="#">Method initQuickTime</a>	156
<a href="#">Method setCsshHost</a>	156
<a href="#">Method setJSHost</a>	157
<a href="#">Method wasRendered</a>	157
<a href="#">Class dbdURI</a>	157
<a href="#">Method create</a>	158
<a href="#">Method getAction</a>	158
<a href="#">Method getController</a>	158
<a href="#">Method getInstance</a>	159
<a href="#">Method getParams</a>	159
<a href="#">Method replace</a>	159
<a href="#">Method set</a>	159
<a href="#">Method setParam</a>	160
<a href="#">Method setPositionList</a>	160
<a href="#">Class dbdXajax</a>	161
<a href="#">Constructor construct</a>	161
<a href="#">Method xmlToArray</a>	161
<a href="#">Class dbdXajaxJS</a>	162
<a href="#">Method doDefault</a>	162
<a href="#">Class FB</a>	163
<a href="#">Method dump</a>	163
<a href="#">Method error</a>	163
<a href="#">Method getEnabled</a>	164
<a href="#">Method getOptions</a>	164
<a href="#">Method group</a>	164
<a href="#">Method groupEnd</a>	164
<a href="#">Method info</a>	165
<a href="#">Method log</a>	165
<a href="#">Method send</a>	166
<a href="#">Method setEnabled</a>	166
<a href="#">Method setObjectFilter</a>	166
<a href="#">Method setOptions</a>	167
<a href="#">Method table</a>	167
<a href="#">Method trace</a>	168
<a href="#">Method warn</a>	168
<a href="#">Class Index</a>	168
<a href="#">Method doDefault</a>	169
<a href="#">Class wavEdit</a>	169
<a href="#">Var \$afiles</a>	169
<a href="#">Var \$arrData</a>	169
<a href="#">Var \$bdColor</a>	169

<a href="#">Var \$bgColor</a>	169
<a href="#">Var \$Db</a>	170
<a href="#">Var \$gridColor</a>	170
<a href="#">Var \$grpColor</a>	170
<a href="#">Var \$header</a>	170
<a href="#">Var \$imgBorder</a>	170
<a href="#">Var \$imgHeight</a>	170
<a href="#">Var \$imgWidth</a>	170
<a href="#">Var \$visual_graph_mode</a>	170
<a href="#">Var \$waves</a>	170
<a href="#">Constructor wavEdit</a>	170
<a href="#">Method cropFiles</a>	170
<a href="#">Method doClip</a>	170
<a href="#">Method doCrop</a>	170
<a href="#">Method doDrawImg</a>	171
<a href="#">Method doEcho</a>	171
<a href="#">Method doFade</a>	171
<a href="#">Method doJoin</a>	171
<a href="#">Method doSetVol8Bit</a>	171
<a href="#">Method doSetVol16Bit</a>	171
<a href="#">Method doViewHeader</a>	171
<a href="#">Method drawImgWave</a>	172
<a href="#">Method getBytes</a>	172
<a href="#">Method getDataArr</a>	172
<a href="#">Method getFileName</a>	172
<a href="#">Method getFiles</a>	172
<a href="#">Method getHeader</a>	172
<a href="#">Method getStructure</a>	172
<a href="#">Method joinFiles</a>	172
<a href="#">Method setEcho</a>	173
<a href="#">Method setFade</a>	173
<a href="#">Method setVolume</a>	173
<a href="#">Method unpackData</a>	173
<a href="#">Method unpackHeader</a>	173
<a href="#">Method viewHeader</a>	173
<a href="#">Method viewImage</a>	173
<a href="#">Method writeWavFile</a>	174
<a href="#">Appendices</a>	175
<a href="#">Appendix A - Class Trees</a>	176
<a href="#">dbdCommon</a>	176
<a href="#">dbdMVC</a>	177
<a href="#">JSMin</a>	179
<a href="#">FirePHP</a>	179
<a href="#">Appendix D - Todo List</a>	181



# Package dbdCommon Procedural Elements

## FlashLoaderSWFO.php

FlashLoaderSWFO.php :: Flash movie loading/embedding class

- **Package** dbdCommon
- **Author** Will Mason < [will@dontblinkdesign.com](mailto:will@dontblinkdesign.com)>
- **Version** 1.3
- **Copyright** Copyright (c) 2006-2007 by Don't Blink Design

# GDColor.php

## GDColor.php :: The GD Color Class

GDColor version 1.1 Copyright (c) 2006-2007 by Don't Blink Design  
<http://www.dontblinkdesign.com>

GDColor is released under the terms of the LGPL license  
<http://www.gnu.org/copyleft/lesser.html>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

- **Package** dbdCommon
- **Author** Will Mason < [will@dontblinkdesign.com](mailto:will@dontblinkdesign.com) >
- **Version** 1.1
- **Copyright** Copyright (c) 2007 by Don't Blink Design
- **License** <http://www.gnu.org/copyleft/lesser.html>



# InputCSS.php

## InputCSS.php :: The Fancy XHTML/CSS Form Input Class

- **Package** dbdCommon
- **Author** Will Mason < [will@dontblinkdesign.com](mailto:will@dontblinkdesign.com)>
- **Version** 1.3
- **Copyright** Copyright (c) 2009 by Don't Blink Design

# QTLoader.php

QTLoader.php :: QuickTime movie loading/embedding class

- **Package** dbdCommon
- **Author** Will Mason < [will@dontblinkdesign.com](mailto:will@dontblinkdesign.com)>
- **Version** 1.1
- **Copyright** Copyright (c) 2006-2007 by Don't Blink Design

# wmArrays.php

## wmArrays.php :: Will Mason's array processing class

- **Package** dbdCommon
- **Author** Will Mason < [will@dontblinkdesign.com](mailto:will@dontblinkdesign.com)>
- **Version** 1.0
- **Copyright** Copyright (c) 2006-2007 by Don't Blink Design

# wmFileSystem.php

wmFileSystem.php :: Will Mason's file system class

- **Package** dbdCommon
- **Author** Will Mason < [will@dontblinkdesign.com](mailto:will@dontblinkdesign.com)>
- **Version** 1.2
- **Copyright** Copyright (c) 2006-2007 by Don't Blink Design

require\_once ["wmSort.php"](#)*[line 10]*

# wmSort.php

## wmSort.php :: Will Mason's array sorting class

- **Package** dbdCommon
- **Author** Will Mason < [will@dontblinkdesign.com](mailto:will@dontblinkdesign.com)>
- **Version** 1.1
- **Copyright** Copyright (c) 2006-2007 by Don't Blink Design

# wmString.php

## wmString.php :: Will Mason's string manipulation class

- **Package** dbdCommon
- **Author** Will Mason < [will@dontblinkdesign.com](mailto:will@dontblinkdesign.com)>
- **Version** 1.3
- **Copyright** Copyright (c) 2006-2007 by Don't Blink Design

# Package dbdCommon Classes

## Class FlashLoaderSWFO

[line 28]

The **FlashLoaderSWFO** class is used to effortlessly embedd Flash movies in a (X)HTML Document.

**REQUIRED ITEM:** SWFObject.js

*Simple Usage:*

```
1  $FL = new FlashLoaderSWFO("version=6,0,29,0" , "quality=high" );
2  $FL-> loadMovie("src=swfs/movie.swf" , "element_id=mainFlash" ,
"width=800" , "height=600" );
```

*Advanced Usage:*

```
1  $FL = new FlashLoaderSWFO("version=6,0,29,0" , "quality=high" );
2  $FL-> addMovie("src=swfs/movie.swf" , "element_id=mainFlash" ,
"width=800" , "height=600" );
3  $FL-> loadMovies();
```

- **Package** dbdCommon

Constructor *void* function FlashLoaderSWFO::\_\_construct(\$attr,...) [line 116]

**Function Parameters:**

- *string* **\$attr**,... "attribute=value",... optional arguments

The purpose of the constructor is to initialize the main attribute list and then fill it with the arguments (if any).

- **Access** public

*void function FlashLoaderSWFO::addMovie(\$attr,...) [line 235]*

**Function Parameters:**

- *string* **\$attr**,... "attribute=value",... optional arguments

**Adds movie arguments to list of movies to be embedded later by loadMovies().**

- **See** function loadMovies
- **Access** public

*void function FlashLoaderSWFO::getCommonJS() [line 217]*

**Prints include to JavaScript library.**

- **Access** public

*void function FlashLoaderSWFO::getCommonJSPath() [line 225]*

**Prints include to JavaScript library.**

- **Access** public

*void function FlashLoaderSWFO::getMovie(\$attr,...) [line 270]*

**Function Parameters:**



- *mixed* **\$attr**,... "attribute=value",... optional arguments or array \$atts list

**Gets (embeds) flash movie in (X)HTML document.**

- **Access** public

*void* function FlashLoaderSWFO::getMovies() [*line 251*]

**Gets movies in list created by addMovie()**

- **See** function loadMovie
- **Access** public

*boolean* function FlashLoaderSWFO::hasMovies() [*line 243*]

**Check to see if there are any movies to load.**

- **Access** public

*void* function FlashLoaderSWFO::loadMovie(\$attr,...) [*line 297*]

**Function Parameters:**

- *mixed* **\$attr**,... "attribute=value",... optional arguments or array \$atts list

**Loads (embeds) flash movie in (X)HTML document.**

- **Access** public

*void function FlashLoaderSWFO::loadMovies() [line 262]*

**Loads movies in list created by addMovie()**

- **See** function loadMovie
- **Access** public

*void function FlashLoaderSWFO::setAtts(\$attr,...) [line 126]*

**Function Parameters:**

- *string* **\$attr**,... "attribute=value",... optional arguments

**Fill the attribute list with the arguments (if any).**

- **Access** public

## Class GDColor

*[line 37]*

**The GDColor Class is used to easily handle colors while using the GD Library.**

- **Package** dbdCommon

Constructor *void* function GDColor::\_\_construct(\$i, \$hex) [line 64]

**Function Parameters:**

- *resource* **\$i** of image from imagecreate...
- *string* **\$hex** color value

**The constructor's job is to convert HEX color to its integer parts and then allocate the color.**

- **Access public**

*int* function GDColor::getAlpha() [line 132]

**Get the integer color value for the Alpha channel.**

- **Access public**

*int* function GDColor::getBlue() [line 123]

**Get the integer color value for the Blue channel.**

- **Access public**

*int* function GDColor::getColor() [line 87]

**Get the integer color value for use in GD functions.**

- **Access public**

*int* function GDColor::getGreen() [*line 114*]

**Get the integer color value for the Green channel.**

- **Access** public

*string* function GDColor::getHex() [*line 96*]

**Get the HEX color value of the allocated color.**

- **Access** public

*int* function GDColor::getRed() [*line 105*]

**Get the integer color value for the Red channel.**

- **Access** public

*array* function GDColor::hex2arr(\$hex) [*line 161*]

**Function Parameters:**

- *string* **\$hex** color

**Static function for breaking a HEX color into its various channels.**

- **Static**
- **Access** public

array function GDColor::int2arr(\$int) [*line 144*]

**Function Parameters:**

- *int* **\$int** color

**Static function for breaking an integer color into its various channels.**

- **Static**
- **Access** public

void function GDColor::\_\_destruct() [*line 77*]

**Clean up the allocated colors.**

- **Access** public

## Class InputCSS

[*line 15*]

**The InputCSS Class is to generate the markup and JQuery calls to make a fancy CSS form input.**

- **Package** dbdCommon

string function InputCSS::button(\$id, \$name, \$value, [\$disabled = false], [\$extra = ""]) [*line 137*]

**Function Parameters:**

- *string* **\$id**
- *string* **\$name**
- *string* **\$value**
- *bool* **\$disabled**
- *string* **\$extra**

**Generate a submit button.**

- **Static**
- **Access** public

*string* function InputCSS::checkbox(\$id, \$name, \$value, [\$disabled = false], [\$checked = false], [\$extra = ""]) [*line 180*]

**Function Parameters:**

- *string* **\$id**
- *string* **\$name**
- *string* **\$value**
- *bool* **\$disabled**
- *bool* **\$checked**
- *string* **\$extra**

**Generate a checkbox input.**

- **Static**
- **Access** public

*string* function InputCSS::file(\$id, \$name, [\$disabled = false], [\$extra = ""], \$value) [*line 165*]

**Function Parameters:**

- *string* **\$id**
- *string* **\$name**
- *string* **\$value**

- *bool* **\$disabled**
- *string* **\$extra**

**Generate a file/browse input.**

- **Static**
- **Access** public

*string* function InputCSS::radio(\$id, \$name, \$value, [\$disabled = false], [\$checked = false], [\$extra = ""])  
[line 195]

**Function Parameters:**

- *string* **\$id**
- *string* **\$name**
- *string* **\$value**
- *bool* **\$disabled**
- *bool* **\$checked**
- *string* **\$extra**

**Generate a radio input.**

- **Static**
- **Access** public

*string* function InputCSS::reset(\$id, \$name, \$value, [\$disabled = false], [\$extra = ""]) [line 151]

**Function Parameters:**

- *string* **\$id**
- *string* **\$name**
- *string* **\$value**
- *bool* **\$disabled**
- *string* **\$extra**

**Generate a reset button.**

- **Static**
- **Access** public

## Class QTLoder

[line 29]

The QTLoder class is used to effortlessly embedd QuickTime movies in a (X)HTML Document.

**REQUIRED ITEM:** qtobject.js

*Simple Usage:*

```
1 $QT = new QTLoder();
2 $QT-> loadMovie("src=movies/movie.mov" , "element_id=mainQuickTime" ,
"width=800" , "height=600" );
```

*Advanced Usage:*

```
1 $QT = new QTLoder();
2 $QT-> addMovie("src=movies/movie1.mov" , "element_id=movie1" ,
"width=800" , "height=600" );
3 $QT-> addMovie("src=movies/movie2.mov" , "element_id=movie2" ,
"width=800" , "height=600" );
4 $QT-> loadMovies();
```

- **Package** dbdCommon

Constructor void function QTLoder::\_\_construct(\$attr,...) [line 100]

**Function Parameters:**

- *string* **\$attr**,... "attribute=value",... optional arguments

**The purpose of the constructor is to initialize the main attribute list and then fill it with**



the arguments (if any).

- **Access** public

*void function QTLoder::addMovie(\$attr,...) [line 219]*

**Function Parameters:**

- *string* **\$attr**,... "attribute=value",... optional arguments

**Adds movie arguments to list of movies to be embedded later by loadMovies().**

- **See** function loadMovies
- **Access** public

*void function QTLoder::getCommonJS() [line 201]*

**Prints include to JavaScript library.**

- **Access** public

*void function QTLoder::getCommonJSPath() [line 209]*

**Prints include to JavaScript library.**

- **Access** public

*void function QTLoder::getMovie(\$attr,...) [line 254]*

**Function Parameters:**

- *mixed* **\$attr**,... "attribute=value",... optional arguments or array \$atts list

**Gets (embeds) quicktime movie in (X)HTML document.**

- **Access** public

*void* function QTLoder::getMovies() [*line 235*]

**Gets movies in list created by addMovie()**

- **See** function loadMovie
- **Access** public

*boolean* function QTLoder::hasMovies() [*line 227*]

**Check to see if there are any movies to load.**

- **Access** public

*void* function QTLoder::loadMovie(\$attr,...) [*line 281*]

**Function Parameters:**

- *mixed* **\$attr**,... "attribute=value",... optional arguments or array \$atts list

**Loads (embeds) quicktime movie in (X)HTML document.**

- **Access** public

*void function QTLoder::loadMovies() [line 246]*

**Loads movies in list created by addMovie()**

- **See** function loadMovie
- **Access** public

*void function QTLoder::setAtts(\$attr,...) [line 110]*

**Function Parameters:**

- *string* **\$attr**,... "attribute=value",... optional arguments

**Fill the attribute list with the arguments (if any).**

- **Access** public

## Class wmArrays

*[line 15]*

**The wmArrays class is used to preform complex routines on arrays.**

- **Package** dbdCommon
- **Static**

array function `wmArrays::arrayMergeRecursiveSumFields($arr1, $arr2, $fields)` [line 79]

**Function Parameters:**

- array **\$arr1** first array
- array **\$arr2** second array
- array **\$fields** list of string keys of values to sum

**Merges multidimensional arrays, and sums the integer values of fields in given list.**

*Usage:*

```
1  $arr1 = array(  
2      "page"           =>  "index"           ,  
3      "hits"           =>  10  
4  );  
5  $arr2 = array(  
6      "page"           =>  "index"           ,  
7      "hits"           =>  20  
8  );  
9  $marr = wmArrays::arrayMergeRecursiveSumFields($arr1, $arr2, array("hits"    ));  
10 //result  
11 $marr = array(  
12     "page"           =>  "index"           ,  
13     "hits"           =>  30  
14 );
```

- **Static**
- **Access** public

void function `wmArrays::arrayRemoveValue($arr, $value)` [line 111]

**Function Parameters:**

- **\$arr**
- **\$value**

- **Static**
- **Access** public

array function `wmArrays::arraySearchRecursive($arr, $val, [$key = false], [$max_depth = -1])` [line 30]

### **Function Parameters:**

- *array* **\$arr** array to search through
- *mixed* **\$val** value to search for in array
- *mixed* **\$key** optional key to match value with
- *int* **\$max\_depth** optional maximum depth of recursion

**Searches through a multidimensional array for a given value and if found, returns an array of keys to its location.**

*Usage:*

```
1 $index = wmArrays::arraySearchRecursive($arr, "will@dontblinkdesign.com" );
```

- **Static**
- **Access** public

*void function* [wmArrays::arrayValuesRecursive](#)(\$arr) [*line 96*]

### **Function Parameters:**

- **\$arr**

- **Static**
- **Access** public

*void function* [wmArrays::explodeNewline](#)(\$string) [*line 129*]

### **Function Parameters:**

- **\$string**

- **Static**
- **Access** public

*void function* `wmArrays::implodeObjArray($glue, $arr, $method)` *[line 134]*

**Function Parameters:**

- **\$glue**
- **\$arr**
- **\$method**

- **Static**
- **Access** public

*void function* `wmArrays::pregExplode($pattern, $string, [$limit = -1])` *[line 124]*

**Function Parameters:**

- **\$pattern**
- **\$string**
- **\$limit**

- **Static**
- **Access** public

## Class `wmFileSystem`

*[line 17]*

The `wmFileSystem` class is collection of specialized file system utility methods.

- **Package** dbdCommon
- **Static**

*string* function `wmFileSystem::bytesToHuman($size)` [line 101]

**Function Parameters:**

- *int* **\$size** byte count

**Makes byte counts human readable (i.e. 1024 -> 1KB).**

*Usage:*

```
1 $size = wmFileSystem::bytesToHuman\(\$size\);
```

- **Static**
- **Access** public

*integer* function `wmFileSystem::dirSize($dir, [$exclude = array()])` [line 79]

**Function Parameters:**

- *string* **\$dir**
- **\$exclude**

**Get the file size of a directory and its subcontents in bytes**

- **Static**
- **Access** public

*string* function `wmFileSystem::mimeType($file)` [line 117]

**Function Parameters:**

- *string* **\$file**

## Get the mime type from mime\_magic

- **Static**
- **Access** public

array function wmFileSystem::scanDir(\$dir, [\$max\_depth = -1]) [*line 30*]

### **Function Parameters:**

- *string* **\$dir** target directory
- *int* **\$max\_depth** maximum depth for recursion

**Reads directory contents recursively and returns them as an associative array.**

### *Usage:*

```
1 $files = wmFileSystem::scanDir("./somedir/" );
```

- **Static**
- **Access** public

## Class wmSort

[*line 15*]

**The wmSort class is used for advanced two-dimentional array sorting.**

- **Package** dbdCommon
- **Static**



void function wmSort::multiASort(&\$arr, \$sortby, \$arr) [line 43]

**Function Parameters:**

- array **\$arr** reference to associative array to be sorted
- string **\$sortby** key(s) to sort by and direction
- **&\$arr**

**Sorts a two-dimentional array by one to many key "indexes" and maintains its associative indexes.**

*Usage:*

```
1  wmSort::multiASort(\$arr, "name:desc,date:asc" );
```

- **Static**
- **Access** public

void function wmSort::multiSort(&\$arr, \$sortby, \$arr) [line 27]

**Function Parameters:**

- array **\$arr** reference to array to be sorted
- string **\$sortby** key(s) to sort by and direction
- **&\$arr**

**Sorts a two-dimentional array by one to many key "indexes".**

*Usage:*

```
1  wmSort::multiSort(\$arr, "name:desc,date:asc" );
```

- **Static**
- **Access** public

# Class wmString

*[line 15]*

The **wmString** class is used to manipulate strings in various ways.

- **Package** dbdCommon
- **Static**

**wmString::TIME\_LEN\_FORMAT\_LONG**

= 1 *[line 43]*

**wmString::TIME\_LEN\_FORMAT\_SHORT**

= 0 *[line 42]*

**wmString::TIME\_LEN\_LABEL\_AGO**

= "Ago" *[line 39]*

**wmString::TIME\_LEN\_LABEL\_DAY**

= "Day" *[line 34]*

**wmString::TIME\_LEN\_LABEL\_HRS**

= "Hour" *[line 33]*

**wmString::TIME\_LEN\_LABEL\_LAST**

= "Last" *[line 38]*

**wmString::TIME\_LEN\_LABEL\_MIN**

= "Minute" *[line 32]*

**wmString::TIME\_LEN\_LABEL\_MON**

= "Month" *[line 36]*

**wmString::TIME\_LEN\_LABEL\_SEC**

= "Second" *[line 31]*

**wmString::TIME\_LEN\_LABEL\_TODAY**

= "Today" *[line 41]*

**wmString::TIME\_LEN\_LABEL\_WEEK**

= "Week" *[line 35]*

**wmString::TIME\_LEN\_LABEL\_YEAR**

= "Year" *[line 37]*

**wmString::TIME\_LEN\_LABEL\_YESTERDAY**

= "Yesterday" *[line 40]*

**wmString::TIME\_LEN\_TIME\_DAY**

= 86400 *[line 27]*

**wmString::TIME\_LEN\_TIME\_HRS**

= 3600 *[line 26]*

**wmString::TIME\_LEN\_TIME\_MIN**

= 60 *[line 25]*

**wmString::TIME\_LEN\_TIME\_MON**

= 2592000 *[line 29]*

**wmString::TIME\_LEN\_TIME\_SEC**

= 1 *[line 24]*

**wmString::TIME\_LEN\_TIME\_WEEK**

= 604800 *[line 28]*

**wmString::TIME\_LEN\_TIME\_YEAR**

= 31536000 *[line 30]*

**wmString::TIME\_LEN\_TYPE\_DAY**

= 3 *[line 20]*

**wmString::TIME\_LEN\_TYPE\_HRS**

= 2 *[line 19]*

**wmString::TIME\_LEN\_TYPE\_MIN**

= 1 *[line 18]*

**wmString::TIME\_LEN\_TYPE\_MON**

= 5 *[line 22]*

**wmString::TIME\_LEN\_TYPE\_SEC**

$$= 0 \text{ [line 17]}$$

wmString::TIME\_LEN\_TYPE\_WEEK

$= 4$  [line 21]

wmString::TIME\_LEN\_TYPE\_YEAR

$$= 6 \text{ [line 23]}$$

**wmString::VALID\_DATE**

```
= '\A(?:\d{2}([02468][048])|(13579[26]))[\-\\s]?(((0?[13578])|(1[02]))[\-\\s]?((0?[1-9])|([1-2][0-9])|(3[01])))(((0?[469])|(11))[\-\\s]?((0?[1-9])|([1-2][0-9])|(30)))|(0?2[\-\\s]?((0?[1-9])|([1-2][0-9]))))\d{2}([02468][1235679])|(13579[01345789]))[\-\\s]?(((0?[13578])|(1[02]))[\-\\s]?((0?[1-9])|([1-2][0-9])|(3[01])))(((0?[469])|(11))[\-\\s]?((0?[1-9])|([1-2][0-9])|(30)))|(0?2[\-\\s]?((0?[1-9])|([10-9])|(2[0-8]))))\s(((0?[0-9])|([10-9])|(2[0-3]))\:(-[0-5][0-9])\s\:(\:(-[0-5][0-9]))?))?$'\z' [line 45]
```

**wmString::VALID\_EMAIL**

```
= '[a-z0-9!#$%&'*+V=?^_`{}~.-]+(?:\.[a-z0-9!#$%&'*+V=?^_`{}~.-]+)*@(?:[a-z0-9](?:[a-z0-9-]*[a-z0-9])?\.)+[a-z0-9](?:[a-z0-9-]*[a-z0-9])?/i' [line 44]
```

```
boolean function wmString::dateValid($date) [line 369]
```

### Function Parameters:

- *string* \$date

## Test if a date is valid.

- **Static**
- **Access** public

```
string function wmString::dayOfWeek([$day_num = 0], [$format_string = "l"]) [line 268]
```

**Function Parameters:**

- *integer* **\$day\_num**
- *string* **\$format\_string**

**Give the day of week given day number 0-6.**

- **Static**
- **Access** public

*string* function wmString::emailEncode(\$str) [line 60]

**Function Parameters:**

- *string* **\$str** email address and/or mailto

**Takes a string (mailto:email@domain.com) and converts it to unicode to protect it from spiders.**

*Usage:*

```
1  $email = wmString::emailEncode("mailto:email@domain.com");
```

- **Static**
- **Access** public

*boolean* function wmString::emailValid(\$email) [line 359]

**Function Parameters:**

- *string* **\$email**

**Test if an email address is valid.**

- **Static**
- **Access** public

*integer* function wmString::hanukkahStart([\$year = null]) [line 284]

**Function Parameters:**

- *integer* **\$year**

**Get the unix timestamp of Hanukkah for this year.**

- **Static**
- **Access** public

*string* function wmString::jsSpecialChars(\$str) [*line 153*]

**Function Parameters:**

- *string* **\$str**

**Escapes special characters for use in javascript function calls.**

- **Static**
- **Access** public

*string* function wmString::lcFirst(\$str) [*line 141*]

**Function Parameters:**

- *string* **\$str**

**Turns the first characher of a string to lower case**

- **Static**
- **Access** public

*string* function `wmString::makePlural($str)` [line 90]

**Function Parameters:**

- *string* **\$str**

**Takes a string and converts to the plural version.**

- **Static**
- **Access** public

*string* function `wmString::makeSingular($str)` [line 74]

**Function Parameters:**

- *string* **\$str**

**Takes a string and converts to the singular version.**

- **Static**
- **Access** public

*string* function `wmString::numberSuffix($num)` [line 296]

**Function Parameters:**

- *mixed* **\$num**

**Add suffixes to numbers.**

- **Static**

- **Access** public

*string* function wmString::randomWord([\$length = 5]) [*line 317*]

**Function Parameters:**

- *integer* **\$length**

**Generate a random readable word.**

- **Static**
- **Access** public

*string* function wmString::superCap2UnderLower(\$str) [*line 118*]

**Function Parameters:**

- *string* **\$str**

**Takes a super capped string (SuperCapString) and adds underscores and makes lowercase**

- **Static**
- **Access** public

*string* function wmString::superCapAddSpaces(\$str) [*line 106*]

**Function Parameters:**

- *string* **\$str**

**Takes a super capped string (SuperCapString) and adds spaces**



- **Static**
- **Access** public

*string* function `wmString::timeFormat($seconds)` [line 247]

**Function Parameters:**

- *mixed* **\$seconds**

**Format the amount of time in seconds to hh::mm:ss**

- **Static**
- **Access** public

*string* function `wmString::timeLengthFormat($time, [$type_in = self::TIME_LEN_TYPE_SEC], [$type_out = self::TIME_LEN_TYPE_SEC], [$format = self::TIME_LEN_FORMAT_LONG])` [line 168]

**Function Parameters:**

- *integer* **\$time**
- *integer* **\$type\_in**
- *integer* **\$type\_out**
- *integer* **\$format**

**Format a length of time.**

- **Static**
- **Access** public

*string* function `wmString::timePastFormat($date, [$use_today = false])` [line 211]

### ***Function Parameters:***

- *mixed* **\$date**
- **\$use\_today**

**Format the amount of time past.**

- **Static**
- **Access** public

*string* function wmString::truncate(\$str, [\$len = 80], [\$etc = "..."], [\$break\_words = false], [\$middle = false]) [*line 338*]

### ***Function Parameters:***

- *string* **\$str**
- *integer* **\$len**
- *string* **\$etc**
- *boolean* **\$break\_words**
- *boolean* **\$middle**

**Truncate a string...**

- **Static**
- **Access** public

*string* function wmString::underLower2SuperCap(\$str) [*line 129*]

### ***Function Parameters:***

- *string* **\$str**

**Takes an underscored lower case string (under\_lower\_string) converts it to SuperCaps**

- **Static**
- **Access** public

*string* function wmString::wkt2kml(\$wkt, [\$name = false], [\$desc = false], [\$style\_id = false]) [*line 382*]

**Function Parameters:**

- *string* **\$wkt**
- *string* **\$name**
- *string* **\$desc**
- *string* **\$style\_id**

**Convert WKT to KML.**

- **Static**
- **Access** public



# Package FirePHP Procedural Elements

## FirePHP.class.php

\*\*\* BEGIN LICENSE BLOCK \*\*\*\*\*

This file is part of FirePHP (<http://www.firephp.org/>).

Software License Agreement (New BSD License)

Copyright (c) 2006-2009, Christoph Dorn All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

\* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

\* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

\* Neither the name of Christoph Dorn nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

\*\*\*\*\* END LICENSE BLOCK \*\*\*\*\*

- **Package** FirePHP
- **Author** Christoph Dorn < [christoph@christophdorn.com](mailto:christoph@christophdorn.com)>
- **Copyright** Copyright (C) 2007-2009 Christoph Dorn
- **License** <http://www.opensource.org/licenses/bsd-license.php>

# Package FirePHP Classes

## Class FirePHP

*[line 58]*

**Sends the given data to the FirePHP Firefox Extension.**

The data can be displayed in the Firebug Console or in the "Server" request tab.

For more information see: <http://www.firephp.org/>

- **Package** FirePHP
- **Author** Christoph Dorn < [christoph@christophdorn.com](mailto:christoph@christophdorn.com) >
- **Copyright** Copyright (C) 2007-2009 Christoph Dorn
- **License** <http://www.opensource.org/licenses/bsd-license.php>

### FirePHP::DUMP

= 'DUMP' *[line 108]*

**Dumps a variable to firebug's server panel**

### FirePHP::ERROR

= 'ERROR' *[line 101]*

### Firebug ERROR level

Logs a message to firebug console, displays an error icon before the message and colors the line yellow. Also increments the firebug error count.

#### **FirePHP::EXCEPTION**

= 'EXCEPTION' [*line 124*]

**Displays an exception in firebug console**  
Increments the firebug error count.

#### **FirePHP::GROUP\_END**

= 'GROUP\_END' [*line 145*]

**Ends a group in firebug console**

#### **FirePHP::GROUP\_START**

= 'GROUP\_START' [*line 138*]

**Starts a group in firebug console**

#### **FirePHP::INFO**

= 'INFO' [*line 83*]

**Firebug INFO level**  
Logs a message to firebug console and displays an info icon before the message.

#### **FirePHP::LOG**

= 'LOG' [*line 74*]

**Firebug LOG level**  
Logs a message to firebug console.

#### **FirePHP::TABLE**

= 'TABLE' [*line 131*]

**Displays an table in firebug console**

#### **FirePHP::TRACE**

= 'TRACE' [*line 115*]

**Displays a stack trace in firebug console**



## FirePHP::VERSION

= '0.3' *[line 65]*

## FirePHP version

## FirePHP::WARN

= 'WARN' *[line 92]*

## Firebug WARN level

Logs a message to firebug console, displays an warning icon before the message and colors the line turquoise.

## FirePHP::\$convertAssertionErrorsToExceptions

*boolean* = true *[line 173]*

## Flag whether to convert PHP assertion errors to Exceptions

- **Access** protected

## FirePHP::\$enabled

*boolean* = true *[line 218]*

## Flag to enable/disable logging

- **Access** protected

## FirePHP::\$inExceptionHandler

*boolean* = false *[line 159]*

## Flag whether we are logging from within the exception handler

- **Access** protected

#### **FirePHP::\$instance**

*FirePHP = null [line 152]*

#### **Singleton instance of FirePHP**

- **Static**
- **Access** protected

#### **FirePHP::\$messageIndex**

*int = 1 [line 187]*

#### **Wildfire protocol message index**

- **Access** protected

#### **FirePHP::\$objectFilters**

*array = array() [line 204]*

#### **Filters used to exclude object members when encoding**

- **Access** protected

#### **FirePHP::\$objectStack**

*object = array() [line 211]*

## A stack of objects used to detect recursion during object encoding

- **Access** protected

### FirePHP::\$options

```
array = array('maxObjectDepth' => 10,  
              'maxArrayDepth' => 20,  
              'useNativeJsonEncode' => true,  
              'includeLineNumbers' => true) [line 194]
```

## Options for the library

- **Access** protected

### FirePHP::\$throwAssertionExceptions

*boolean = false [line 180]*

## Flag whether to throw PHP assertion errors that have been converted to Exceptions

- **Access** protected

### FirePHP::\$throwErrorExceptions

*boolean = true [line 166]*

## Flag whether to throw PHP errors that have been converted to Exceptions

- **Access** protected

Constructor *void* function FirePHP::\_\_construct() [*line 223*]

### The object constructor

*void* function FirePHP::assertionHandler(\$file, \$line, \$code) [*line 421*]

#### **Function Parameters:**

- *string* **\$file** File source of assertion
- *int* **\$line** Line source of assertion
- *mixed* **\$code** Assertion code

### FirePHP's assertion handler

Logs all assertions to your firebug console and then stops the script.

- **Access** public

*boolean* function FirePHP::detectClientExtension() [*line 596*]

### Check if FirePHP is installed on client

- **Access** public

*true* function FirePHP::dump(\$Key, \$Variable) [*line 562*]

#### **Function Parameters:**

- *string* **\$Key**
- *mixed* **\$Variable**

### Dumps key and variable to firebug server panel

- See [FirePHP::DUMP](#)
- **Throws** Exception
- **Access** public

*array* function FirePHP::encodeObject(\$Object, [\$ObjectDepth = 1], [\$ArrayDepth = 1], \$Depth) [*line 1011*]

**Function Parameters:**

- *Object* **\$Object** The object to be encoded
- *int* **\$Depth** The current traversal depth
- **\$ObjectDepth**
- **\$ArrayDepth**

**Encodes an object including members with protected and private visibility**

- **Access** protected

*array* function FirePHP::encodeTable(\$Table) [*line 982*]

**Function Parameters:**

- *array* **\$Table** The table to be encoded

**Encodes a table by encoding each row and column with encodeObject()**

- **Access** protected

*true* function FirePHP::error(\$Object, [\$Label = null]) [*line 549*]

**Function Parameters:**

- *mixes* **\$Object**
- *string* **\$Label**

## Log object with label to firebug console

- See [FirePHP::ERROR](#)
- **Throws** Exception
- **Access** public

*void function* FirePHP::errorHandler(\$errno, \$errstr, \$errfile, \$errline, \$errcontext) [*line 344*]

### **Function Parameters:**

- *int* **\$errno**
- *string* **\$errstr**
- *string* **\$errfile**
- *int* **\$errline**
- *array* **\$errcontext**

## FirePHP's error handler

Throws exception for each php error that will occur.

- **Access** public

*void function* FirePHP::exceptionHandler(\$Exception) [*line 382*]

### **Function Parameters:**

- *Exception* **\$Exception**

## FirePHP's exception handler

Logs all exceptions to your firebug console and then stops the script.

- **Throws** Exception

*true* function FirePHP::fb(\$Object) [line 613]

**Function Parameters:**

- *mixed* **\$Object** The variable to be logged

## Log variable to Firebug

- **See** <http://www.firephp.org/Wiki/Reference/Fb>
- **Throws** Exception
- **Access** public

*boolean* function FirePHP::getEnabled() [line 272]

## Check if logging is enabled

- **Access** public

*FirePHP* function FirePHP::getInstance([\$AutoCreate = false]) [line 241]

**Function Parameters:**

- *boolean* **\$AutoCreate**

## Gets singleton instance of FirePHP

- **Static**
- **Access** public

*array* function FirePHP::getOptions() [*line 310*]

### **Get options from the library**

- **Access** public

*string|false* function FirePHP::getUserAgent() [*line 938*]

### **Get user agent**

- **Access** protected

*true* function FirePHP::group(\$Name, [\$Options = null]) [*line 473*]

### **Function Parameters:**

- *string* **\$Name**
- *array* **\$Options** OPTIONAL Instructions on how to log the group

### **Start a group for following messages.**

Options:   Collapsed: [true|false]   Color:   [#RRGGBB|ColorName]

- **Throws** Exception
- **Access** public

*true* function FirePHP::groupEnd() [*line 497*]

### **Ends a group you have started before**



- **Throws** Exception
- **Access** public

*true* function FirePHP::info(\$Object, [\$Label = null]) [*line 523*]

**Function Parameters:**

- *mixes* **\$Object**
- *string* **\$Label**

**Log object with label to firebug console**

- See [FirePHP::INFO](#)
- **Throws** Exception
- **Access** public

*FirePHP* function FirePHP::init() [*line 253*]

**Creates FirePHP object and stores it for singleton access**

- **Static**
- **Access** public

*boolean* function FirePHP::is\_utf8(\$str) [*line 1150*]

**Function Parameters:**

- *mixed* **\$str** String to be tested

**Returns true if \$string is valid UTF-8 and false otherwise.**

- **Static**
- **Access** protected

*string* function FirePHP::jsonEncode(\$Object, [\$skipObjectEncode = false]) [*line 961*]

**Function Parameters:**

- *object* **\$Object** The object to be encoded
- **\$skipObjectEncode**

### Encode an object into a JSON string

Uses PHP's jeson\_encode() if available

- **Access** public

*true* function FirePHP::log(\$Object, [\$Label = null]) [*line 510*]

**Function Parameters:**

- *mixes* **\$Object**
- *string* **\$Label**

### Log object with label to firebug console

- See [FirePHP::LOG](#)
- **Throws** Exception
- **Access** public

*Exception* function FirePHP::newException(\$Message) [*line 949*]

**Function Parameters:**

- *string* **\$Message**

## Returns a new exception

- **Access** protected

*mixed* function FirePHP::registerAssertionHandler([\$convertAssertionErrorsToExceptions = true], [\$throwAssertionExceptions = false]) [*line 400*]

### **Function Parameters:**

- *boolean* **\$convertAssertionErrorsToExceptions**
- *boolean* **\$throwAssertionExceptions**

## Register FirePHP driver as your assert callback

- **Access** public

*mixed* function FirePHP::registerErrorHandler([\$throwErrorExceptions = true]) [*line 321*]

### **Function Parameters:**

- **\$throwErrorExceptions**

## Register FirePHP as your error handler

Will throw exceptions for each php error.

- **Access** public

*mixed* function FirePHP::registerExceptionHandler() [*line 369*]

### **Register FirePHP as your exception handler**

- **Access** public

*void* function FirePHP::setEnabled(\$Enabled) [*line 263*]

#### **Function Parameters:**

- *boolean* **\$Enabled** TRUE to enable, FALSE to disable

### **Enable and disable logging to Firebug**

- **Access** public

*void* function FirePHP::setHeader(\$Name, \$Value) [*line 929*]

#### **Function Parameters:**

- *string* **\$Name**
- *string\_type* **\$Value**

### **Send header**

- **Access** protected

*void* function FirePHP::setObjectFilter(\$Class, \$Filter) [*line 285*]

### ***Function Parameters:***

- *string* **\$Class** The class name of the object
- *array* **\$Filter** An array of members to exclude

### **Specify a filter to be used when encoding an object**

Filters are used to exclude object members.

- **Access** public

*void* function FirePHP::setOptions(\$Options) [*line 301*]

### ***Function Parameters:***

- *array* **\$Options** The options to be set

### **Set some options for the library**

Options:

- **maxObjectDepth**: The maximum depth to traverse objects (default: 10)
- **maxArrayDepth**: The maximum depth to traverse arrays (default: 20)
- **useNativeJsonEncode**: If true will use `json_encode()` (default: true)
- **includeLineNumbers**: If true will include line numbers and filenames (default: true)

- **Access** public

*void* function FirePHP::setProcessorUrl(\$URL) [*line 446*]

### ***Function Parameters:***

- *string* **\$URL**

## Set custom processor url for FirePHP

- **Access** public

*void* function FirePHP::setRendererUrl(\$URL) [*line 456*]

### **Function Parameters:**

- *string* \$URL

## Set custom renderer url for FirePHP

- **Access** public

*true* function FirePHP::table(\$Label, \$Table) [*line 587*]

### **Function Parameters:**

- *string* \$Label
- *string* \$Table

## Log a table in the firebug console

- **See** [FirePHP::TABLE](#)
- **Throws** Exception
- **Access** public

*true* function FirePHP::trace(\$Label) [*line 574*]

### **Function Parameters:**

- *string* **\$Label**

## Log a trace in the firebug console

- See [FirePHP::TRACE](#)
- **Throws** Exception
- **Access** public

*true* function FirePHP::warn(\$Object, [\$Label = null]) [*line 536*]

### **Function Parameters:**

- *mixes* **\$Object**
- *string* **\$Label**

## Log object with label to firebug console

- See [FirePHP::WARN](#)
- **Throws** Exception
- **Access** public

*array* function FirePHP::\_escapeTrace(\$Trace) [*line 892*]

### **Function Parameters:**

- *array* **\$Trace**

## Escape trace path for windows systems

- **Access** protected

*string* function FirePHP::\_escapeTraceFile(\$File) [*line 911*]

**Function Parameters:**

- *string* **\$File**

**Escape file information of trace for windows systems**

- **Access** protected

*string* function FirePHP::\_standardizePath(\$Path) [*line 882*]

**Function Parameters:**

- *string* **\$Path**

**Standardizes path for windows systems.**

- **Access** protected

*array* function FirePHP::\_\_sleep() [*line 231*]

**When the object gets serialized only include specific object members.**

- **Access** public



# Package JSMIn Classes

## Class JSMIn

*[line 48]*

### **jsmin.php - PHP implementation of Douglas Crockford's JSMIn.**

This is pretty much a direct port of jsmin.c to PHP with just a few PHP-specific performance tweaks. Also, whereas jsmin.c reads from stdin and outputs to stdout, this library accepts a string as input and returns another string as output.

PHP 5 or higher is required.

Permission is hereby granted to use this version of the library under the same terms as jsmin.c, which has the following license:

-- Copyright (c) 2002 Douglas Crockford ([www.crockford.com](http://www.crockford.com))

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE. --

- **Package** JSMIn
- **Author** Ryan Grove < [ryan@wonko.com](mailto:ryan@wonko.com) >
- **Version** 1.1.0 (2007-06-01)
- **Copyright** 2002 Douglas Crockford <douglas@crockford.com> (jsmin.c)
- **Copyright** 2007 Ryan Grove <ryan@wonko.com> (PHP port)
- **Link** <http://code.google.com/p/jsmin-php/>
- **License** [MIT License](#)

#### JSMIn::ORD\_LF

= 10 [*line 49*]

#### JSMIn::ORD\_SPACE

= 32 [*line 50*]

#### JSMIn::\$a

*mixed* = " [*line 52*]

- **Access** protected

#### JSMIn::\$b

*mixed* = " [*line 53*]

- **Access** protected

#### JSMIn::\$input

*mixed* = " [*line 54*]

- **Access** protected

### **JSMIn::\$inputIndex**

*mixed* = 0 [line 55]

- **Access** protected

### **JSMIn::\$inputLength**

*mixed* = 0 [line 56]

- **Access** protected

### **JSMIn::\$lookAhead**

*mixed* = null [line 57]

- **Access** protected

### **JSMIn::\$output**

*mixed* = array() [line 58]

- **Access** protected

Constructor *void* function JSMIn::\_\_construct(\$input) [line 69]

#### **Function Parameters:**

- **\$input**

- **Access** public

*void function JSMin::action(\$d) [line 76]*

***Function Parameters:***

- **\$d**

- **Access** protected

*void function JSMin::get() [line 138]*

- **Access** protected

*void function JSMin::isAlphaNum(\$c) [line 163]*

***Function Parameters:***

- **\$c**

- **Access** protected

*void function JSMin::min() [line 167]*

- **Access** protected

*void function JSMin::minify(\$js) [line 62]*

**Function Parameters:**

- **\$js**

- **Static**
- **Access** public

*void function JSMin::next() [line 249]*

- **Access** protected

*void function JSMin::peek() [line 288]*

- **Access** protected

## Class JSMinException

*[line 295]*

- **Package** JSMin



# Package dbdMVC Procedural Elements

## dbdMVC.php

### dbdMVC.php :: dbdMVC Include File & Front Controller Class

dbdMVC version 1.8.2 Copyright (c) 2006-2009 by Don't Blink Design <http://dbdmvc.com>

dbdMVC is a light weight model-view-controller framework.

dbdMVC is released under the terms of the LGPL license  
<http://www.gnu.org/copyleft/lesser.html>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com) >
- **Version** 1.8.2
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design
- **TODO** Create readme and update dbdMVC description
- **License** <http://www.gnu.org/copyleft/lesser.html>

*void function dbdLog([\$msg = ""], [\$type = null], [\$dest = null]) [line 90]*

#### ***Function Parameters:***

- **\$msg**
- **\$type**

- **\$dest**

DBD\_CACHE\_DIR = DBD\_APP\_DIR."cache".DBD\_DS *[line 68]*  
**dbdMVC cache directory**

DBD\_DEBUG\_ALL = 255 *[line 76]*  
**Turn on all debugging**

DBD\_DEBUG\_CSS = 8 *[line 83]*  
 DBD\_DEBUG\_DB = 1 *[line 80]*  
**Turn on all debugging**

DBD\_DEBUG\_HTML = 2 *[line 81]*  
 DBD\_DEBUG\_JS = 4 *[line 82]*  
 DBD\_DS = DIRECTORY\_SEPARATOR *[line 39]*  
**Shorthand for DIRECTORY\_SEPARATOR**

DBD\_FIREPHP\_DIR = DBD\_MVC\_DIR."FirePHPCore-0.3.1".DBD\_DS."lib".DBD\_DS *[line 60]*  
**FirePHP core directory**

DBD\_MVC\_CLI = php\_sapi\_name()=="cli" *[line 72]*  
**Are we running in CLI mode?**

DBD\_PLUG\_DIR = DBD\_MVC\_DIR."plugins".DBD\_DS *[line 64]*  
**dbdMVC plug-in directory**

DBD\_SMARTY\_DIR = DBD\_MVC\_DIR."Smarty-2.6.18".DBD\_DS."libs".DBD\_DS *[line 48]*  
**Smarty package directory**

DBD\_SMARTY\_PLUG\_DIR = DBD\_SMARTY\_DIR.DBD\_DS."plugins\_dbd".DBD\_DS *[line 52]*  
**Smarty addOn plug-in directory (used to avoid disrupting standard plug-in dir)**

DBD\_XAJAX\_DIR = DBD\_MVC\_DIR."xajax\_0.2.5".DBD\_DS *[line 56]*  
**xAjax package directory**

require\_once ["lib/dbdLoader.php"](#) *[line 111]*

**After the dbdLoader is included, all further dbdMVC includes use dbdLoader::load() or  
 \_\_autoload()**



require\_once [FirePHP.class.php](#)*[line 89]*

**error\_log** function that handles complex types too

# fb.php

- **Package** dbdMVC

*true* function fb([\$Object = ]) [*line 57*]

## **Function Parameters:**

- *mixed* **\$Object**

## **Sends the given data to the FirePHP Firefox Extension.**

The data can be displayed in the Firebug Console or in the "Server" request tab.

- **See** <http://www.firephp.org/Wiki/Reference/Fb>
- **Throws** Exception

require\_once **dirname(\_\_FILE\_\_).'../FirePHP.class.php'** [*line 45*]

## dbconn.inc :: The Database string constants

- **Package** dbdMVC
- **Author** Will Mason < [will@dontblinkdesign.com](mailto:will@dontblinkdesign.com)>
- **Version** 1.2
- **Copyright** Copyright (c) 2009 by Don't Blink Design

DBCONN\_DEFAULT\_DB = " *[line 18]*

**Constant default database name if none is specified at connection time**

DBCONN\_INCLUDED = true *[line 13]*

**Constant boolean to flag file inclusion.**

DB\_DRIVER = 'mysql' *[line 23]*

**Database connection strings**

DB\_HOST = 'localhost' *[line 24]*

DB\_PASS = 'i\$th3\$hit' *[line 27]*

DB\_PORT = '3306' *[line 25]*

DB\_USER = 'dbdmvc' *[line 26]*

# Index.php

## Index.php :: Index Controller Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.1
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# index.php

## index.php :: dbdMVC Bootstrap Sample File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.4
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

```
DBD_APP_DIR = realpath(DBD_DOC_ROOT."../dbdApp")."/" [line 11]
DBD_DOC_ROOT = dirname(__FILE__) [line 10]
DBD_MVC_DIR = "/var/www/dbdMVC/" [line 12]
require_once DBD\_MVC\_DIR."dbdMVC.php [line 15]
```

# dbdController.php

## dbdController.php :: dbdController Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.9
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdCSS.php

## dbdCSS.php :: dbdCSS Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.12
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdDB.php

dbdDB.php :: dbdDB Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.3
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design



# dbdDispatcher.php

## dbdDispatcher.php :: dbdDispatcher Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.8
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdEmptyController.php

## dbdEmptyController.php :: dbdEmptyController Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.1
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdError.php

## dbdError.php :: dbdError Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.2
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdException.php

## dbdException.php :: dbdException Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.3
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdHoldableException.php

## dbdHoldableException.php :: dbdHoldableException Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.1
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdInfo.php

## dbdInfo.php :: dbdInfo Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.1
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdJS.php

## dbdJS.php :: dbdJS Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.7
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdLoader.php

## dbdLoader.php :: dbdLoader Class File & \_\_autoload() Function

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.6
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

require\_once ["dbdException.php"](#)*[line 14]*

### dbdException class

void function \_\_autoload(\$class) *[line 237]*

#### **Function Parameters:**

- *string* **\$class**

**Autoload classes using dbdLoader::loadClass();**



# dbdModel.php

dbdModel.php :: dbdModel Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.11
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdRequest.php

## dbdRequest.php :: dbdRequest Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.2
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdRouter.php

## dbdRouter.php :: dbdRouter Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.3
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdSmarty.php

## dbdSmarty.php :: dbdSmarty Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.14
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdURI.php

## dbdURI.php :: dbdURI Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.8
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdXajax.php

dbdXajax.php :: dbdXajax Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.1
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# dbdXajaxJS.php

## dbdXajaxJS.php :: dbdXajaxJS Class File

- **Package** dbdMVC
- **Author** Don't Blink Design < [info@dontblinkdesign.com](mailto:info@dontblinkdesign.com)>
- **Version** 1.2
- **Copyright** Copyright (c) 2006-2009 by Don't Blink Design

# wmGD.php

- **Package** dbdMVC

*void function* colorizeImage(&\$im, \$color) [*line 100*]

**Function Parameters:**

- **&\$im**
- [\*GDColor\*](#) **\$color**

*void function* cropImage(&\$src, \$w, \$h) [*line 168*]

**Function Parameters:**

- **&\$src**
- **\$w**
- **\$h**

*void function* cropImageReal(&\$src, \$x, \$y, \$w, \$h) [*line 239*]

**Function Parameters:**

- **&\$src**
- **\$x**
- **\$y**
- **\$w**
- **\$h**

*void function* cropResizeImage(&\$src, \$w, \$h, [\$align = 1]) [*line 192*]

**Function Parameters:**

- **&\$src**
- **\$w**
- **\$h**
- **\$align**

*void function* imageCreateFrom(\$filename) [*line 4*]

**Function Parameters:**



- **\$filename**

*void function imagefilledellipsealpha(&\$im, \$cx, \$cy, \$w, \$h, \$color) [line 281]*

**Function Parameters:**

- **&\$im**
- **\$cx**
- **\$cy**
- **\$w**
- **\$h**
- **\$color**

*void function imageLineThick(&\$im, \$x1, \$y1, \$x2, \$y2, \$color, [\$thick = 1]) [line 260]*

**Function Parameters:**

- **&\$im**
- **\$x1**
- **\$y1**
- **\$x2**
- **\$y2**
- **\$color**
- **\$thick**

*void function imageType(\$filename) [line 76]*

**Function Parameters:**

- **\$filename**

*void function imageTypesArray() [line 82]*

*void function outputAndDestroy(&\$im, [\$type = "png"], [\$cache\_file = null], [\$no\_output = false]) [line 20]*

**Function Parameters:**

- **&\$im**
- **\$type**
- **\$cache\_file**
- **\$no\_output**

*void function reflectImage(&\$src, [\$ref\_height = 0.50], [\$fade\_start = 80], [\$fade\_end = 0], [\$tint = "7f7f7f"], [\$bg = "ffffff"], [\$type = "png"]) [line 355]*

**Function Parameters:**

- **&\$src**
- **\$ref\_height**
- **\$fade\_start**
- **\$fade\_end**
- **\$tint**
- **\$bg**
- **\$type**

require\_once ["GDColor.php"](#) *[line 2]*

*void function* `resizeImage(&$src, $w, $h, [$force_upscale = false])` *[line 142]*

***Function Parameters:***

- **&\$src**
- **\$w**
- **\$h**
- **\$force\_upscale**

*void function* `stretchCopyButton($dst_im, $src_im, $dst_wd, $src_wd, $src_ht, [$y = 0])` *[line 248]*

***Function Parameters:***

- **\$dst\_im**
- **\$src\_im**
- **\$dst\_wd**
- **\$src\_wd**
- **\$src\_ht**
- **\$y**

# Package dbdMVC Classes

## Class dbdController

[line 22]

**Abstract Parent Controller Class** All controller classes must extend this class or a child of this class.

- **Package** dbdMVC
- **TODO** Use Response object for output.
- **Abstract Element**
- **Usedby** [dbdInfo](#)
- **Usedby** [dbdJS](#)
- **Usedby** [dbdDispatcher](#)
- **Usedby** [Index](#)
- **Usedby** [dbdError](#)
- **Usedby** [dbdCSS](#)
- **Uses** [dbdLoader](#)
- **Uses** [dbdSmarty](#)
- **Uses** [dbdXajax](#)
- **Uses** [dbdController::\\$xajaxResponse](#)
- **Usedby** [dbdEmptyController](#)

**dbdController::\$model**

*object = null [line 59]*

**Instance of some model.**

**Note:** May not be used.

- **Access** protected

#### **dbdController::\$router**

*dbdRouter = null [line 53]*

#### **Instance of dbdRouter**

- **Access** protected

#### **dbdController::\$view**

*dbdSmarty = null [line 64]*

#### **Instance of dbdSmarty**

- **Access** protected

#### **dbdController::\$xajax**

*dbdXajax = null [line 69]*

#### **Instance of dbdXajax**

- **Access** protected

#### **dbdController::\$xajaxResponse**

*dbdXajaxResponse = null [line 75]*

#### **Instance of xajaxResponse object**

- **TODO** make dbdXajaxResponse object
- **Access** protected
- **Usedby** [dbdController](#)

Constructor *void* function dbdController::\_\_construct(\$router, \$app\_dir) [line 87]

**Function Parameters:**

- [dbdRouter](#) **\$router**
- *string* **\$app\_dir**

**Prepare controller for action.**

Set the router and app\_dir, create the view, and then call init(). **Note:** The constructor is declared final to prevent corruption of object. Any further construction needed by child classes can be accomplished by overriding dbdController::init().

- **Final**
- **Access** public

*void* function dbdController::assignAllParams() [line 239]

**Assign all request parameters to the view as an associative array.**

**Note:** Slashes are stripped.

- **Access** protected

*void* function dbdController::autoExec() [line 144]

**Auto execute method to be executed after the action.**

Can be overridden for additional automatic functionality (maybe assign some variable to the view). **Note:** If overridden, parent::autoExec(); must be called to maintain current functionality!

- **Access** public

*void* function dbdController::autoRender() [*line 126*]

#### **Auto render the view.**

Can be disabled by using `$this->noRender()`; If a template has not been set, it will attempt to convert the controller name into a template filename.

- **Final**
- **Throws** dbdException
- **Access** public

*void* function dbdController::dnit() [*line 171*]

#### **Default dnit method.**

This method is to be overridden for additional object destruction and de-initialization (maybe cleaning up the `$model` property).

- **Access** protected

*void* function dbdController::doDefault() [*line 177*]

#### **Default action method.**

Can be preformed without the presense of a child controller.

- **Access** public

*void* function dbdController::doXajax() [*line 184*]

**Prepare controller for xajax request.**

Ajax actions are called by dbdXajax.

- **Final**
- **Access** public

*void* function dbdController::forward([\$url = "/"]) [*line 248*]

**Function Parameters:**

- *string* \$url

**Perform an HTTP Location change.**

Disables auto render.

- **Access** protected

*string* function dbdController::getAction() [*line 273*]

**Get action name.**

- **Access** protected

*string* function dbdController::getAppDir() [*line 257*]

**Get application directory.**

- **Access** protected

*string* function dbdController::getController() [*line 265*]

**Get controller name.**

- **Access** protected

*mixed* function dbdController::getParam(\$name) [*line 299*]

**Function Parameters:**

- *string* **\$name**

**Get request parameter.**

- **Access** protected

*array* function dbdController::getParams() [*line 308*]

**Get all parameters.**

- **See** [dbdRouter::getParams\(\)](#)
- **Access** protected

*string* function dbdController::getTemplate() [*line 231*]

**Get template filename.**



- **Access** protected

*array* function dbdController::getUploadInfo(\$id) [*line 330*]

**Function Parameters:**

- *string* \$id

**Get the progress information on a current upload via APC.**

- **Link** <http://us.php.net/manual/en/apc.configuration.php#ini.apc.rfc1867>
- **Access** protected
- **Uses** php-pecl-apc

*string* function dbdController::getURL([\$get\_params = false], [\$host = false]) [*line 319*]

**Function Parameters:**

- *boolean* \$get\_params
- *boolean* \$host

**Get current request url.**

If optional flag is passed, parameters are included.

- **Access** protected

*void* function dbdController::init() [*line 162*]

**Default init method.**

This method is to be overridden for additional object construction and initialization (maybe setting the \$model property).

- **Access** protected

*void function dbdController::noRender() [line 151]*

**Disable auto render on destruction.**

- **Access** public

*void function dbdController::registerXajaxActions([\$controller = null]) [line 204]*

**Function Parameters:**

- *string* **\$controller**

### **Register controller actions with dbdXajax.**

If controller name is not passed it defaults to the current one. JavaScript is assigned to view as \$xajax. **Note:** Action methods must use the dbdDispatcher::XAJAX\_ACTION\_PREFIX.

- **TODO** Auto insert JS into header of view output using some filter.
- **Final**
- **Access** protected

*void function dbdController::setParam(\$name, \$value) [line 282]*

**Function Parameters:**

- *string* **\$name**
- *mixed* **\$value** value

## Set request parameter.

- **Access** protected

*void* function dbdController::setTemplate(\$tpl) [*line 223*]

### **Function Parameters:**

- *string* **\$tpl**

## Set template filename.

- **Access** protected

*void* function dbdController::unsetParam(\$name) [*line 290*]

### **Function Parameters:**

- *string* **\$name**

## Unset request parameter.

- **Access** protected

*void* function dbdController::\_\_destruct(\$router, \$app\_dir) [*line 114*]

### **Function Parameters:**

- [\*dbdRouter\*](#) **\$router**
- *string* **\$app\_dir**

### Conclude the controller action.

Call `dnit()`. **Note:** The destructor is declared final to prevent corruption of object. Any further destruction needed by child classes can be accomplished by overriding `dbdController::dnit()`.

- **Final**
- **Access** public

## Class dbdCSS

*[line 63]*

### Controller class for compressing and serving css files.

Can be used to combine files using `@import` statments and the `doCombine` action. Also extends `CSS` by adding internal/external variable support and simple math. **Set Variable Example:**

- **Package** dbdMVC
- **Uses** [dbdException](#)
- **Uses** [dbdController](#)

### dbdCSS::BLOCK\_CLOSE\_REGEX

= `'/{\}}'` *[line 85]*

### Css block close pattern.

### dbdCSS::BLOCK\_OPEN\_REGEX

= `'/{\{'` *[line 81]*

## Css block open pattern.

dbdCSS::BUTT\_DEF\_TAG\_REGEX

= '\b([\#.][a-z][a-z0-9#.-]+ )?(a|input)([\#.])([a-z][a-z0-9#.-]\*)[,]?/i' *[line 122]*

## Css button definition pattern.

dbdCSS::BUTT\_GEN\_TAG\_REGEX

= '/[@]generated/i' *[line 114]*

## Css generated buttons at-tag pattern.

dbdCSS::BUTT\_GEN\_TPL\_TAG\_REGEX

= '/#[a-z][a-z0-9]\*)/i' *[line 118]*

## Css generated buttons template tag pattern.

dbdCSS::BUTT\_PROP\_DEF\_DISABLED\_REGEX

= '/[-]disabled\$/i' *[line 134]*

## Css button property disabled pattern.

dbdCSS::BUTT\_PROP\_DEF\_HOVER\_REGEX

= '/[-]hover\$/i' *[line 130]*

## Css button property hover pattern.

dbdCSS::BUTT\_PROP\_DEF\_REGEX

= '/([a-z][a-z-]\*)[ \t\n]\*[:][ \t\n]\*([^;]+);/i' *[line 126]*

## Css button property definition pattern.

dbdCSS::BUTT\_TAG\_REGEX

= '/^[@]buttons/i' *[line 110]*

## Css buttons at-tag pattern.

dbdCSS::CACHE\_INFO\_FILES

= '/^[ ]?\\\*[ ]?[@]files (.+)\$/' *[line 142]*

**Css caching info file list pattern.**

**dbdCSS::CACHE\_INFO\_SPRITE**

= '/^[ ]?\\\*[ ]?[@]sprite (.+)\$/' *[line 138]*

**Css caching info sprite pattern.**

**dbdCSS::CSS\_EXT\_REGEX**

= '/^+\\. [c][s][s]\$/' *[line 77]*

**Css file extension pattern.**

**dbdCSS::DEBUG\_REGEX**

= '/^\\^\*\\. \*[@]debug\\. \*\\^/' *[line 73]*

**Debug comment to turn off minification.**

**dbdCSS::DIR\_DELIM\_REGEX**

= '/[,\\]' *[line 146]*

**Directory delimiter for passing a string of files**

**dbdCSS::IMPORT\_REGEX**

= '/^[@]import[ ]?["']([a-z0-9-\_\\.]+[\\.[c][s][s)]["'];/i' *[line 69]*

**Css import regular expression.**

- **TODO** allow for url() syntax

**dbdCSS::MATH\_CALL\_REGEX**

= '/math[(.+)[]]/ie' *[line 102]*

**Css math call pattern. (for simple equations)**

**dbdCSS::MATH\_SAFE\_REGEX**

= '!^[ 0-9()%/+\*~]+\$!' *[line 106]*

**Css math safe characters pattern. (for simple equations)**

**dbdCSS::VAR\_CALL\_REGEX**

= '/var[()([a-z][a-z0-9-\_]\*)]/ie' *[line 98]*

**Css variable call pattern. (for dereferencing)**

**dbdCSS::VAR\_DEF\_REGEX**

= '/([a-z][a-z0-9-\_]\*)[ \t\n]\*[:][ \t\n]\*([^\;]+);/i' *[line 94]*

**Css variable definition pattern.**

**dbdCSS::VAR\_TAG\_REGEX**

= '/^[@]variables/i' *[line 90]*

**Css variables at-tag pattern.**

Uses implimentation specs from <http://disruptive-innovations.com/zoo/cssvariables/>

*void* function dbdCSS::doCombine() *[line 1168]*

**Serve multiple css files as one**

- **Access** public

*string* function dbdCSS::genURL([\$files = array()], [\$vars = array()]) *[line 1200]*

**Function Parameters:**

- *array* **\$files**
- *array* **\$vars**

**Generate a combine url from an array of files**

- **Static**
- **Access** public

*void* function dbdCSS::init() [*line 1141*]

**Set debug**

- **Access** public

## Class dbdDB

[*line 15*]

**dbdMVC Database Abstraction Layer**

- **Package** dbdMVC

**dbdDB::COMP\_BETWEEN**

= 12 [*line 45*]

**dbdDB::COMP\_EQ**

= 1 [*line 34*]

**dbdDB::COMP\_GT**

= 5 [*line 38*]

**dbdDB::COMP\_GTEQ**

= 4 [*line 37*]



**dbdDB::COMP\_IN**

= 8 *[line 41]*

**dbdDB::COMP\_LIKE**

= 10 *[line 43]*

**dbdDB::COMP\_LT**

= 7 *[line 40]*

**dbdDB::COMP\_LTEQ**

= 6 *[line 39]*

**dbdDB::COMP\_NBETWEEN**

= 13 *[line 46]*

**dbdDB::COMP\_NEQ**

= 2 *[line 35]*

**dbdDB::COMP\_NIN**

= 9 *[line 42]*

**dbdDB::COMP\_NLIKE**

= 11 *[line 44]*

**dbdDB::COMP\_NULLEQ**

= 3 *[line 36]*

**dbdDB::COMP\_TYPE**

= "dbdDB::COMP\_TYPE" *[line 33]*

**dbdDB::DATE\_FORMAT**

= "Y-m-d H:i:s" *[line 26]*

## **SQL compliant date formatting string**

- **Var** constant

**dbdDB::DBCONN\_PATH**

= "constant/dbconn.inc" *[line 21]*

## path to database connection string file

- **Var** constant

**dbdDB::TIME\_FORMAT**

= "H:i:s" [*line 31*]

## SQL compliant time formatting string

- **Var** constant

*string* function dbdDB::date([\$timestamp = null]) [*line 146*]

### **Function Parameters:**

- *integer* **\$timestamp**

## Format date for SQL insert

- **Static**
- **Access** public

*dbdDB* function dbdDB::getInstance([\$dsn = ""], [\$user = ""], [\$pass = ""],  
[\$driver\_options = array()]) [*line 98*]

### **Function Parameters:**

- *string* **\$dsn**
- *string* **\$user**
- *string* **\$pass**
- *array* **\$driver\_options**

## Include db\_conn string file.

Key method of Singleton Pattern!!

- **Static**
- **Access** public

*int* function dbdDB::nextAutoID(\$table) [*line 178*]

### **Function Parameters:**

- **\$table**

**Selects the next auto\_increment id**

- **Access** public

*PDOStatement* function dbdDB::prepExec(\$statement, [\$input\_parameters = array()], [\$driver\_options = array()]) [*line 167*]

### **Function Parameters:**

- *string* **\$statement**
- *array* **\$input\_parameters**
- *array* **\$driver\_options**

**Prepare & excute a query statement**

- **Throws** dbdException
- **Access** public

*string* function dbdDB::time([\$timestamp = null]) [*line 155*]

**Function Parameters:**

- *integer* **\$timestamp**

## Format time for SQL insert

- **Static**
- **Access** public

*mixed* function dbdDB::\_\_call(\$name, \$args) [*line 192*]

**Function Parameters:**

- *string* **\$name**
- *array* **\$args**

## Magic function to call PDO functions and rethrow any exceptions as dbdException

- **Throws** dbdException
- **Access** public

# Class dbdDispatcher

[*line 19*]

## Dispatch controller actions.

- **Package** dbdMVC
- **Usedby** [dbdMVC](#)
- **Usedby** [dbdXajax](#)
- **Uses** [dbdRouter](#)
- **Uses** [dbdLoader](#)
- **Uses** [dbdEmptyController](#)
- **Uses** [dbdController](#)

### **dbdDispatcher::ACTION\_PREFIX**

= "do" *[line 48]*

#### **Action method name prefix**

### **dbdDispatcher::BUILTIN\_PREFIX**

= "dbd" *[line 24]*

#### **Builtin controller class prefix**

### **dbdDispatcher::DEFAULT\_ACTION**

= "default" *[line 40]*

#### **Default action name**

### **dbdDispatcher::DEFAULT\_CONTROLLER**

= "index" *[line 36]*

#### **Default controller name**

### **dbdDispatcher::ERROR\_CONTROLLER**

= "dbdError" *[line 32]*

#### **Error controller class if is set**

### **dbdDispatcher::FALLBACK\_CONTROLLER**

= "dbdEmptyController" *[line 28]*

## Fallback controller class if none exists

**dbdDispatcher::MAGIC\_ACTION**

= "action" *[line 44]*

## Magic action name

**dbdDispatcher::XAJAX\_ACTION\_PREFIX**

= "xDo" *[line 52]*

## Xajax action method name prefix

Constructor *void* function dbdDispatcher::\_\_construct([\$router = null], [\$app\_dir = null]) *[line 83]*

### **Function Parameters:**

- [\*dbdRouter\*](#) **\$router**
- *string* **\$app\_dir**

## Set router, app\_dir, and controller\_dir.

- **Access** public

*void* function dbdDispatcher::dispatch() *[line 95]*

## Dispatch actions

- **Access** public

# Class dbdEmptyController

[line 16]

Empty controller class for rendering templates without a controller!

- **Package** dbdMVC
- **Usedby** [dbdDispatcher](#)
- **Uses** [dbdController](#)

# Class dbdError

[line 16]

Error handling controller class.

- **Package** dbdMVC
- **Uses** [dbdController](#)

**dbdError::ERR\_400**

= "Bad Request" [line 21]

**HTTP Status 400 Description**

**dbdError::ERR\_401**

= "Unauthorized" [line 25]

**HTTP Status 401 Description**

**dbdError::ERR\_403**

= "Forbidden" [line 29]

**HTTP Status 403 Description**

**dbdError::ERR\_404**

= "Not Found" *[line 33]*

### **HTTP Status 404 Description**

**dbdError::ERR\_408**

= "Request Timeout" *[line 37]*

### **HTTP Status 408 Description**

**dbdError::ERR\_500**

= "Internal Server Error" *[line 41]*

### **HTTP Status 500 Description**

*void function dbdError::doDefault() [line 45]*

**Attept to render error.tpl or just echo error.**

- **Access** public

*void function dbdError::doError(\$that) [line 54]*

#### **Function Parameters:**

- [\*dbdController\*](#) **\$that**

**Crazy static action method expects instnace of dbdController for dbdError extension.**

- **Static**
- **Access** public



## Class dbdException

[line 15]

**dbdMVC Exception class.**

- **Package** dbdMVC
- **Usedby** [dbdLoader](#)
- **Usedby** [dbdXajaxJS](#)
- **Usedby** [dbdJS](#)
- **Usedby** [dbdCSS](#)

Constructor *void* function dbdException::\_\_construct([\$message = null], [\$code = 500]) [line 22]

**Function Parameters:**

- *string* **\$message**
- *integer* **\$code**

**Pass paremeters on to parent exception with prefixed message.**

- **Access** public

## Class dbdHoldableException

[line 15]

**dbdMVC Exception class.**

- **Package** dbdMVC

#### dbdHoldableException::\$held

*mixed* = array() [line 26]

#### Array of held exceptions

- **Static Variable Used** array 0: dbdHoldableException
- **Static**
- **Access** protected

#### dbdHoldableException::\$hold

*mixed* = false [line 21]

#### Flag holding status

- **Static Variable Used** boolean 0:
- **Static**
- **Access** protected

array function dbdHoldableException::getHeld() [line 73]

#### Get array of held exceptions

- **Static**
- **Access** public

void function dbdHoldableException::hold() [line 31]

#### Enable holding

- **Static**
- **Access** public

*void function dbdHoldableException::intercept(\$e) [line 46]*

**Function Parameters:**

- [dbdHoldableException](#) \$e

**Exception handler and method to call when ready to throw**

- **Throws** dbdHoldableException
- **Static**
- **Access** public

*void function dbdHoldableException::release() [line 58]*

**Disable holding and throw the first held excetion, if any**

- **Throws** dbdHoldableException
- **Static**
- **Access** public

## Class dbdInfo

*[line 18]*

**Controller for displaying phpInfo.**

Use /dbdInfo/.

- **Package** dbdMVC
- **Uses** [dbdMVC](#)
- **Uses** [dbdController](#)

*void* function dbdInfo::doDefault() [*line 23*]

**Render phpinfo() if PHPisExposed or forward home.**

- **Access** public

## Class dbdJS

[*line 18*]

**Controller class for compressing and serving js files.**

Can be used to combine files using the doCombine action. Files are combined, minified, and cached once for later use.

- **Package** dbdMVC
- **Usedby** [dbdXajaxJS](#)
- **Uses** [dbdException](#)
- **Uses** [dbdController](#)

**dbdJS::DIR\_DELIM\_REGEX**

= "[,\\]" [*line 23*]

**Directory delimiter for passing a string of files**

*void function dbdJS::addFile(\$file, \$dir) [line 163]*

**Function Parameters:**

- *string \$file*
- *string \$dir*

**Check a file for existence and then add it to the array for later processing.**

- **Access** protected

*void function dbdJS::doCombine() [line 218]*

**Serve multiple js files as one**

- **Access** public

*void function dbdJS::doDefault() [line 203]*

**Alias of doGet()**

- **Access** public

*void function dbdJS::doGet() [line 210]*

**Serve js files...**

- **Access** public

*string* function dbdJS::genURL([\$files = array()], [\$vars = array()]) [*line 238*]

**Function Parameters:**

- *array* **\$files**
- *array* **\$vars**

**Generate a combine url from an array of files**

- **Static**
- **Access** public

*void* function dbdJS::init() [*line 191*]

**Set no render, cache\_dir, debug, and load JSMin**

- **Access** protected

*void* function dbdJS::output() [*line 175*]

**Set headers, minify, and echo buffer.**

- **Access** protected

## Class dbdLoader

[*line 20*]

### dbdMVC File and Class Loader

- **Package** dbdMVC
- **Usedby** [dbdMVC](#)
- **Usedby** [dbdDispatcher](#)
- **Usedby** [dbdController](#)
- **Uses** [dbdException](#)

**dbdLoader::DEFAULT\_CONTROLLER\_DIR**

= "controllers" [*line 25*]

### Controller directory

**dbdLoader::DEFAULT\_MODEL\_DIR**

= "models" [*line 29*]

### Model directory

*void function* dbdLoader::addControllerDir(\$dir) [*line 135*]

#### **Function Parameters:**

- *string* **\$dir**

### Add a controller directory

- **Static**
- **Access** public

*void function* dbdLoader::addPath(\$path) [*line 111*]

#### **Function Parameters:**

- *string* **\$path**

## Add path to dbdLoader and php.ini

- **Static**
- **Access** public

*string* function dbdLoader::getControllerDirs() [*line 151*]

### **Get controller directory**

- **Static**
- **Access** public

*object instance* function dbdLoader::getInstance() [*line 101*]

### **Get the active instance or create one.**

- **Static**
- **Access** public

*string* function dbdLoader::getModelDir() [*line 159*]

### **Get model directory**

- **Static**
- **Access** public

*array* function dbdLoader::getPaths() [*line 119*]

### **Get all paths registered with dbdLoader**



- **Static**
- **Access** public

*boolean* function dbdLoader::load(\$file, [\$dir = null]) [*line 193*]

**Function Parameters:**

- *string* **\$file**
- *string* **\$dir**

**Load a php file using require\_once if not already loaded.**

- **Throws** dbdException
- **Static**
- **Access** public

*void* function dbdLoader::loadClass(\$class) [*line 211*]

**Function Parameters:**

- *string* **\$class**

**Load a class file. Calls dbdLoader::load().**

Called by \_\_autoload()

- **Static**
- **Access** public

*mixed* function dbdLoader::search(\$file, [\$dir = null]) [*line 170*]

**Function Parameters:**

- *string* **\$file**
- *string* **\$dir**

### Search for a file in the list of path(s).

Returns path to file if found and false upon failure.

- **Static**
- **Access** public

*void* function dbdLoader::setControllerDir(\$dir) [*line 127*]

#### **Function Parameters:**

- *string* **\$dir**

### Set controller directory

- **Static**
- **Access** public

*void* function dbdLoader::setModelDir(\$dir) [*line 143*]

#### **Function Parameters:**

- *string* **\$dir**

### Set model directory

- **Static**
- **Access** public

# Class dbdModel

[line 35]

## dbdMVC Database Table Model

### Simple Model Example

```
1  class Widget extends dbdModel
2  {
3      const TABLE_NAME = "widgets"          ;
4      const TABLE_KEY = "widget_id"         ;
5
6      public function __construct($id = 0)
7      {
8          parent::__construct(__CLASS__, $id);
9      }
10
11     public static function getAll()
12     {
13         return parent::getAll(__CLASS__);
14     }
15 }
```

- **Package** dbdMVC
- **Abstract Element**

#### dbdModel::CONST\_TABLE\_KEY

= 'TABLE\_KEY' [line 38]

#### dbdModel::CONST\_TABLE\_NAME

= 'TABLE\_NAME' [line 37]

#### dbdModel::\$data

array = array() [line 68]

### Array of fields from row

- **Access** protected

**dbdModel::\$db**

*dbdDB = null [line 48]*

### Static instance of dbdDB

- **Static**
- **Access** protected

**dbdModel::\$id**

*integer = 0 [line 63]*

### Row id

- **Access** protected

Constructor *void* function dbdModel::\_\_construct(\$class, [\$id = 0]) [line 75]

#### **Function Parameters:**

- *string* **\$class**
- *integer* **\$id**

**The constructor's job is to set the table info, get the db, and call the initializer.**

- **Access** public

*void* function dbdModel::delete() [line 147]

## Delete this row

- **Access** public

*void* function dbdModel::ensureDB() [*line 296*]

### Make sure we have an instance of the db

- **Static**
- **Access** public

*array* function dbdModel::getAll(\$class, [\$table\_keys = array()], [\$order = null], [\$limit = null], [\$ids\_only = false]) [*line 175*]

#### **Function Parameters:**

- *string* **\$order**
- *string* **\$limit**
- *boolean* **\$ids\_only**
- **\$class**
- **\$table\_keys**

## Get all rows from this table

- **Static**
- **Access** public

*array* function dbdModel::getCount(\$class, [\$table\_keys = array()]) [*line 160*]

#### **Function Parameters:**

- *string* **\$class**
- *array* **\$table\_keys**

## Get a count of all rows from this table

- **Static**
- **Access** public

*array* function dbdModel::getData() [*line 330*]

## Get the fields for this row

- **Access** public

*dbdDB* function dbdModel::getDB() [*line 305*]

## Get the database object

- **Static**
- **Access** public

*integer* function dbdModel::getID() [*line 322*]

## Get the row id

- **Access** public

*void* function dbdModel::init() [*line 89*]

## Select all the fields for this row

- **Access** protected

*void function dbdModel::initFields() [line 97]*

**Select all the fields names for this table**

- **Access** protected

*void function dbdModel::save([\$fields = array()]) [line 110]*

**Function Parameters:**

- *array* **\$fields**

**Save all the fields for this row**

- **Access** public

*dbdDB function dbdModel::setDB(\$db) [line 314]*

**Function Parameters:**

- [\*dbdDB\*](#) **\$db**

**Set the database object**

- **Static**
- **Access** public

*mixed* function dbdModel::\_\_call(\$name, \$args) [*line 377*]

**Function Parameters:**

- *string* **\$name**
- *array* **\$args**

**Magic function allows calling of extra magic functions for testing field values.**

Includes: has, is, get, and set.

- **Throws** dbdException
- **Access** public

*mixed* function dbdModel::\_\_get(\$name) [*line 348*]

**Function Parameters:**

- *string* **\$name**

**Magic function for getting field values**

- **Access** public

*boolean* function dbdModel::\_\_isset(\$name) [*line 357*]

**Function Parameters:**

- *string* **\$name**

**Magic function to check if a field is set**



- **Access** public

*void function dbdModel::\_\_set(\$name, \$value) [line 339]*

***Function Parameters:***

- *string* **\$name**
- *mixed* **\$value**

## **Magic function for setting field values**

- **Access** public

*void function dbdModel::\_\_unset(\$name) [line 365]*

***Function Parameters:***

- *string* **\$name**

## **Magic function to unset a field**

- **Access** public

# **Class dbdMVC**

*[line 129]*

## **dbdMVC Front Controller Class**

- **Package** dbdMVC
- **Usedby** [dbdInfo](#)
- **Usedby** [dbdSmarty](#)
- **Uses** [dbdRouter](#)
- **Uses** [dbdRequest](#)
- **Uses** [dbdLoader](#)
- **Uses** [dbdDispatcher](#)

## dbdMVC::VERSION

= "1.8.2" [*line 135*]

### Current version number

**Note:** May not match file @version number

*boolean* function dbdMVC::debugMode(\$mode) [*line 465*]

#### **Function Parameters:**

- **\$mode**

**Check if debug mode is set.**

- **Static**
- **Access** public

*void* function dbdMVC::e(\$e) [*line 345*]

#### **Function Parameters:**

- *Exception* **\$e**

**Default ecption handler**

- **Static**
- **Access** public

*void* function dbdMVC::exposePHPInfo() [*line 475*]

**Allow phpinfo() display via dbdInfo.**

**WARNING:** Allowing system information to be view by the public can be a major security risk. This should be used with caution for development only!

- See [dbdInfo](#)
- **Static**
- **Access** public

*string* function dbdMVC::getAppDir() [*line 399*]

**Get application directory.**

- **Static**
- **Access** public

*string* function dbdMVC::getAppName() [*line 391*]

**Get application name.**

- **Static**
- **Access** public

*string* function dbdMVC::getErrorController() [*line 415*]

**Get error controller name.**

- **Static**
- **Access** public

*float* function dbdMVC::getErrorLog() [*line 423*]

**Return the error\_log file name.**

- **Static**
- **Access** public

*float* function dbdMVC::getExecutionTime() [*line 447*]

**Calculate execution time of the application and return.**

- **Static**
- **Access** public

*string* function dbdMVC::getFallbackController() [*line 407*]

**Get fallback controller name.**

- **Static**
- **Access** public

*object instance* function dbdMVC::getInstance() [*line 335*]

**Get the active instance of dbdMVC or create one.**

- **Static**
- **Access** public

*object dbdRequest* function dbdMVC::getRequest() [*line 431*]

**Get the instance of dbdRequest.**

- **Static**
- **Access** public

*object dbdRouter* function dbdMVC::getRouter() [*line 439*]

**Get the instance of dbdRouter.**

- **Static**
- **Access** public

*void* function dbdMVC::logExecutionTime() [*line 455*]

**Log execution time of the application to the error\_log.**

- **Throws** dbdException
- **Static**
- **Access** public

*boolean* function dbdMVC::PHPisExposed() [*line 483*]

**Check if phpinfo display is allowed.**

- **Static**
- **Access** public

*void function dbdMVC::run(\$app\_dir, [\$app\_name = null], [\$fallback\_controller = null], [\$error\_controller = null], [\$error\_log = null], [\$debug\_mode = null]) [line 369]*

**Function Parameters:**

- *string* **\$app\_dir**
- *string* **\$app\_name**
- **\$fallback\_controller**
- **\$error\_controller**
- **\$error\_log**
- **\$debug\_mode**

**Run dbdMVC on the given dbdApp dir.**

This is the main method of dbdMVC.

- **Static**
- **Access** public

*void function dbdMVC::setAppDir(\$app\_dir) [line 317]*

**Function Parameters:**

- *string* **\$app\_dir**

**Set application directory property.**

**Note:** Accepts paths with or without trailing slash.

- **Static**
- **Access** public

*void function dbdMVC::setAppName(\$app\_name) [line 284]*

**Function Parameters:**

- *string* **\$app\_name**

**Set application name property.**

- **Static**
- **Access** public

*void function dbdMVC::setDebugMode(\$mode) [line 327]*

**Function Parameters:**

- *integer* **\$mode**

**Set debug mode.**

- **Static**
- **Access** public

*void function dbdMVC::setErrorHandler(\$error) [line 300]*

**Function Parameters:**

- *string* **\$error**

**Set error\_controller property.**

- **Static**
- **Access** public

*void function dbdMVC::setErrorLog(\$path, \$error) [line 308]*

**Function Parameters:**

- *string* **\$error**
- **\$path**

**Set error\_log property.**

- **Static**
- **Access** public

*void function dbdMVC::setFallbackController(\$fallback) [line 292]*

**Function Parameters:**

- *string* **\$fallback**

**Set fallback\_controller property.**

- **Static**
- **Access** public

## Class dbdRequest

[line 16]

### HTTP Request Class.

Based on Zend Framework Request Class



- **Package** dbdMVC
- **Usedby** [dbdMVC](#)
- **Usedby** [dbdRouter](#)

Constructor *void* function dbdRequest::\_\_construct([\$uri = null]) [*line 34*]

**Function Parameters:**

- *string* **\$uri**

**Prepare the request object for use by setting the request uri.**

If a string uri is not provided, it is pulled from \$\_SERVER['REQUEST\_URI'].

- **Access** public

*mixed* function dbdRequest::get(\$key) [*line 80*]

**Function Parameters:**

- *string* **\$key**

**Access values contained in the superglobals as public members**

Order of precedence: 1. GET, 2. POST, 3. COOKIE, 4. SERVER, 5. ENV

- **See** <http://msdn.microsoft.com/en-us/library/system.web.httprequest.item.aspx>
- **Access** public

*mixed* function dbdRequest::getCookie([\$key = null], [\$default = null]) [*line 176*]

**Function Parameters:**

- *string* **\$key**
- *mixed* **\$default** Default value to use if key not found

**Retrieve a member of the \$\_COOKIE superglobal.**

If no \$key is passed, returns the entire \$\_COOKIE array.

- **TODO** How to retrieve from nested arrays
- **Access** public

*mixed* function dbdRequest::getEnv([\$key = null], [\$default = null]) [*line 202*]

**Function Parameters:**

- *string* **\$key**
- *mixed* **\$default** Default value to use if key not found

**Retrieve a member of the \$\_ENV superglobal.**

If no \$key is passed, returns the entire \$\_ENV array.

- **Access** public

*string|false* function dbdRequest::getHeader(\$header) [*line 216*]

**Function Parameters:**

- *string* **\$header** HTTP header name

**Return the value of the given HTTP header. Pass the header name as the plain, HTTP-specified header name. Ex.: Ask for 'Accept' to get the Accept header, 'Accept-**

## Encoding' to get the Accept-Encoding header.

- **Throws** dbdException
- **Access** public

*mixed* function dbdRequest::getPost([\$key = null], [\$default = null]) [*line 162*]

### **Function Parameters:**

- *string* **\$key**
- *mixed* **\$default** Default value to use if key not found

## Retrieve a member of the \$\_POST superglobal.

If no \$key is passed, returns the entire \$\_POST array.

- **TODO** How to retrieve from nested arrays
- **Access** public

*mixed* function dbdRequest::getQuery([\$key = null], [\$default = null]) [*line 148*]

### **Function Parameters:**

- *string* **\$key**
- *mixed* **\$default** Default value to use if key not found

## Retrieve a member of the \$\_GET superglobal.

If no \$key is passed, returns the entire \$\_GET array.

- **TODO** How to retrieve from nested arrays
- **Access** public

*void* function dbdRequest::getRequestURI(\$uri) [*line 60*]

**Function Parameters:**

- *string* **\$uri**

## Get request uri

- **Access** public

*mixed* function dbdRequest::getServer([\$key = null], [\$default = null]) [*line 189*]

**Function Parameters:**

- *string* **\$key**
- *mixed* **\$default** Default value to use if key not found

## Retrieve a member of the \$\_SERVER superglobal.

If no \$key is passed, returns the entire \$\_COOKIE array.

- **Access** public

*boolean* function dbdRequest::has(\$key) [*line 135*]

**Function Parameters:**

- *string* **\$key**

## Alias to \_\_isset()

- **Access** public

*boolean* function dbdRequest::isXmlHttpRequest() [*line 241*]

**Is the request a Javascript XMLHttpRequest? Should work with Prototype/Script.aculo.us, possibly others.**

- **Access** public

*void* function dbdRequest::set(\$key, \$val) [*line 69*]

**Function Parameters:**

- *string* **\$key**
- *mixed* **\$val**

## Set parameters at runtime

- **Access** public

*void* function dbdRequest::setRequestURI(\$uri) [*line 52*]

**Function Parameters:**

- *string* **\$uri**

## Set request uri

- **Access** public

*boolean function dbdRequest::\_\_isset(\$key) [line 109]*

**Function Parameters:**

- *string \$key*

**Check to see if a property is set**

- **Access** public

## Class dbdRouter

*[line 17]*

**Request routing class**

- **Package** dbdMVC
- **Usedby** [dbdXajax](#)
- **Usedby** [dbdMVC](#)
- **Usedby** [dbdDispatcher](#)
- **Uses** [dbdURI](#)
- **Uses** [dbdRequest](#)

*Constructor void function dbdRouter::\_\_construct([\$request = null]) [line 53]*

**Function Parameters:**

- [dbdRequest](#) **\$request**

**Set request object, parse request uri, and build base url**

- **Access** public

*mixed* function dbdRouter::addParam(\$name, \$value) [*line 130*]

**Function Parameters:**

- *string* **\$name**
- *mixed* **\$value**

## Add value to array parameter

- **Access** public

*string* function dbdRouter::getAction() [*line 153*]

## Get action name

- **Access** public

*string* function dbdRouter::getController() [*line 145*]

## Get controller name

- **Access** public

*mixed* function dbdRouter::getParam(\$name) [*line 162*]

**Function Parameters:**

- *string* **\$name**

## Get named parameter

- **Access** public

*array* function dbdRouter::getParams() [*line 173*]

## Get all paramters

- **Access** public

*string* function dbdRouter::getURL([*\$get\_params* = false]) [*line 186*]

### **Function Parameters:**

- *boolean* **\$get\_params**

## Get current url.

If optional paramter is passed, all current paramters are included in the url.

- **Access** public

*string* function dbdRouter::setAction(*\$action*) [*line 98*]

### **Function Parameters:**

- *string* **\$action**



## Set action name

- **Access** public

*string* function dbdRouter::setController(\$controller) [*line 87*]

### **Function Parameters:**

- *string* **\$controller**

## Set controller name

- **Access** public

*mixed* function dbdRouter::setParam(\$name, \$value) [*line 110*]

### **Function Parameters:**

- *string* **\$name**
- *mixed* **\$value**

## Set named parameter

- **Access** public

*void* function dbdRouter::unsetParam(\$name) [*line 119*]

### **Function Parameters:**

- *string* **\$name**

## Unset named parameter

- **Access** public

# Class dbdSmarty

*[line 22]*

## dbdMVC Smarty subclass

- **Package** dbdMVC
- **TODO** move custom plugins to separate directory and then register them
- **Usedby** [dbdController](#)
- **Uses** Smarty
- **Uses** [dbdMVC](#)

## dbdSmarty::CACHE\_DIR

= ".cache" *[line 43]*

**Path to directory where smarty cache will live relative to view dir.**

## dbdSmarty::COMPILED\_TEMPLATE\_DIR

= ".templates\_c" *[line 35]*

**Path to directory where smarty compiled templates will live relative to view dir.**

## dbdSmarty::CONFIG\_DIR

= "configs" *[line 39]*

**Path to directory where smarty configs will live relative to view dir.**

**dbdSmarty::TEMPLATE\_DIR**

= "templates" *[line 31]*

**Path to directory where smarty templates will live relative to view dir.**

**dbdSmarty::VIEW\_DIR**

= "views" *[line 27]*

**Path to directory where smarty templates will live relative to application dir.**

Constructor *void* function dbdSmarty::\_\_construct(\$app\_dir) *[line 122]*

***Function Parameters:***

- **\$app\_dir**

**Smarty contructor that sets implimentation specific variables**

- **Access** public

*void* function dbdSmarty::addCSS(\$files) *[line 207]*

***Function Parameters:***

- *mixed* **\$files**

**Add CSS files to the list**

- **Access** public

*void function dbdSmarty::addCSSVar(\$name, \$value) [line 218]*

***Function Parameters:***

- *string* **\$name**
- *mixed* **\$value**

## **Add CSS variable to the list**

- **Access** public

*void function dbdSmarty::addFlashMovie(0) [line 306]*

***Function Parameters:***

- *mixed* **0**

## **Add a flash movie to FlashLoader**

- **Access** public

*void function dbdSmarty::addJS(\$files) [line 250]*

***Function Parameters:***

- *mixed* **\$files**

## **Add JS files to the list**

- **Access** public

*void function dbdSmarty::addJSVar(\$name, \$value) [line 270]*

***Function Parameters:***

- *string* **\$name**
- *mixed* **\$value**

## **Get a JS variable from the list**

- **Access** public

*void function dbdSmarty::addQTMovie(0) [line 337]*

***Function Parameters:***

- *mixed* **0**

## **Add a flash movie to FlashLoader**

- **Access** public

*void function dbdSmarty::clearCSS() [line 234]*

## **Clear CSS file list**

- **Access** public

*void function dbdSmarty::clearJS() [line 277]*

### **Clear JS file list**

- **Access public**

*string function dbdSmarty::dbdBeautify(\$tpl, \$smarty) [line 424]*

#### **Function Parameters:**

- *string* **\$tpl**
- *object* **\$smarty**

**Beautify buffer.**

- **Access public**

*string function dbdSmarty::dbdIncludeFiles(\$tpl, \$smarty) [line 357]*

#### **Function Parameters:**

- *string* **\$tpl**
- *object* **\$smarty**

### **Include CSS & JS Files Automatically**

- **Access public**

*string function dbdSmarty::dbdMinify(\$tpl, \$smarty) [line 436]*

#### **Function Parameters:**

- *string* **\$tpl**
- *object* **\$smarty**

**Minify buffer.**

- **Access** public

*string* function dbdSmarty::dbdTag(\$tpl, \$smarty) [line 448]

**Function Parameters:**

- *string* **\$tpl**
- *object* **\$smarty**

**Smarty output filter to add dbdMVC credits to source**

- **Access** public

*void* function dbdSmarty::display(\$resource\_name, [\$cache\_id = null], [\$compile\_id = null], [\$no\_includes = false])  
[line 168]

**Function Parameters:**

- *string* **\$resource\_name**
- *string* **\$cache\_id**
- *string* **\$compile\_id**
- **\$no\_includes**

**Executes & displays the template results.**

Also sets rendered flag.

- **Access** public

*void* function dbdSmarty::fetch(\$resource\_name, [\$cache\_id = null], [\$compile\_id = null], [\$display = false], [\$no\_includes = false], [\$no\_minify = false]) [*line 182*]

**Function Parameters:**

- *string* **\$resource\_name**
- *string* **\$cache\_id**
- *string* **\$compile\_id**
- *boolean* **\$display**
- *boolean* **\$no\_includes**
- *boolean* **\$no\_minify**

**Executes & returns the template result.**

- **Access** public

*mixed* function dbdSmarty::getCSSVar(\$name) [*line 227*]

**Function Parameters:**

- *string* **\$name**

**Get a CSS variable from the list**

- **Access** public

*mixed* function dbdSmarty::getJSVar(\$name) [*line 261*]

**Function Parameters:**

- *string* **\$name**



## Add JS variable to the list

- **Access** public

*void function dbdSmarty::initFlash(0) [line 285]*

### ***Function Parameters:***

- *mixed 0*

## Initialize FlashLoaderSWFO

- **Access** public

*void function dbdSmarty::initQuickTime(0) [line 316]*

### ***Function Parameters:***

- *mixed 0*

## Initialize QTLoader

- **Access** public

*void function dbdSmarty::setCssHost(\$host) [line 199]*

### ***Function Parameters:***

- *string \$host*

## Set CSS host for inclusion

- **Access** public

*void* function dbdSmarty::setJSHost(\$host) [*line 242*]

### **Function Parameters:**

- *string* **\$host**

## Set JS host for inclusion

- **Access** public

*boolean* function dbdSmarty::wasRendered() [*line 347*]

### **Has a template been rendered?**

- **Access** public

## Class dbdURI

[*line 15*]

### **dbdMVC URI Parsing and creating class**

- **Package** dbdMVC
- **Usedby** [dbdRouter](#)

*string* function dbdURI::create([\$controller = null], [\$action = null], [\$args = array()]) [*line 158*]

**Function Parameters:**

- *string* **\$controller**
- *string* **\$action**
- *associative* **\$args** array \$args

**Create a new url based on the given components**

- **Static**
- **Access** public

*string* function dbdURI::getAction() [*line 139*]

**Get action name**

- **Static**
- **Access** public

*string* function dbdURI::getController() [*line 131*]

**Get controller name**

- **Static**
- **Access** public

*object instance* function dbdURI::getInstance() [*line 55*]

**Get the active instance or create one.**

- **Static**
- **Access** public

*array* function dbdURI::getParams() [*line 147*]

**Get all parameter**

- **Static**
- **Access** public

*string* function dbdURI::replace([\$baseUri = ""], [\$controller = ""], [\$action = ""],  
[\$args = array()]) [*line 207*]

**Function Parameters:**

- *string* **\$baseUri**
- *string* **\$controller**
- *string* **\$action**
- *associative array* **\$args** array \$args

**Replace the given url with the new components given**

- **Static**
- **Access** public

*void* function dbdURI::set(\$uri) [*line 65*]

**Function Parameters:**

- *string* **\$uri**

## Set request uri

- **Static**
- **Access** public

*void* function dbdURI::setParam(\$name, \$value) [*line 91*]

### **Function Parameters:**

- *string* **\$name**
- *mixed* **\$value**

## Set request parameter, recursively

- **Static**
- **Access** public

*void* function dbdURI::setPositionList(\$controller, \$action, [\$params = array()]) [*line 120*]

### **Function Parameters:**

- *string* **\$controller**
- *string* **\$action**
- *array* **\$params**

## Set a parameter position list for a specific controller/action.

This allows for key-less parameterized urls

- **Static**
- **Access** public

## Class dbdXajax

[line 22]

**dbdMVC Xajax subclass**

- **Package** dbdMVC
- **Usedby** [dbdController](#)
- **Uses** xajax
- **Uses** [dbdRouter](#)
- **Uses** [dbdDispatcher](#)

Constructor *void* function dbdXajax::\_\_construct(\$router, [\$sRequestURI = ""], [\$sWrapperPrefix = dbdDispatcher::XAJAX\_ACTION\_PREFIX], [\$sEncoding = XAJAX\_DEFAULT\_CHAR\_ENCODING], [\$bDebug = false]) [line 38]

### **Function Parameters:**

- [dbdRouter](#) **\$router**
- *string* **\$sRequestURI**
- *string* **\$sWrapperPrefix**
- *string* **\$sEncoding**
- *boolean* **\$bDebug**

**Set router and construct xajax object**

- **Access** public

*string* function dbdXajax::\_xmlToArray(\$rootTag, \$sXml) [line 50]

### **Function Parameters:**

- *string* **\$rootTag**
- *string* **\$sXml**

Wrapper for `xajax::_xmlToArray()` to set paramaters with `dbdRouter`

- **Access** public

## Class dbdXajaxJS

*[line 17]*

Controller class for serving xajax javascript file.

- **Package** dbdMVC
- **Uses** [dbdJS](#)
- **Uses** [dbdException](#)

*void* function `dbdXajaxJS::doDefault()` *[line 23]*

**Serve xajax javascript file**

- **Throws** `dbdException`
- **Access** public

# Class FB

[line 66]

- **Package** dbdMVC

*true* function FB::dump(\$Key, \$Variable) [line 231]

## **Function Parameters:**

- *string* **\$Key**
- *mixed* **\$Variable**

**Dumps key and variable to firebug server panel**

- See [FirePHP::DUMP](#)
- **Static**
- **Throws** Exception
- **Access** public

*true* function FB::error(\$Object, [\$Label = null]) [line 218]

## **Function Parameters:**

- *mixes* **\$Object**
- *string* **\$Label**

**Log object with label to firebug console**

- See [FirePHP::ERROR](#)
- **Static**
- **Throws** Exception
- **Access** public



*boolean* function FB::getEnabled() [*line 86*]

### **Check if logging is enabled**

- **See** FirePHP->getEnabled()
- **Static**
- **Access** public

*array* function FB::getOptions() [*line 124*]

### **Get options for the library**

- **See** FirePHP->getOptions()
- **Static**
- **Access** public

*true* function FB::group(\$Name, [\$Options = null]) [*line 155*]

#### **Function Parameters:**

- *string* **\$Name**
- *array* **\$Options** OPTIONAL Instructions on how to log the group

### **Start a group for following messages**

Options: Collapsed: [true|false] Color: [#RRGGBB|ColorName]

- **Static**
- **Access** public

*true* function FB::groupEnd() [*line 166*]

### **Ends a group you have started before**

- **Throws** Exception
- **Static**
- **Access** public

*true* function FB::info(\$Object, [\$Label = null]) [*line 192*]

**Function Parameters:**

- *mixes* **\$Object**
- *string* **\$Label**

## Log object with label to firebug console

- See [FirePHP::INFO](#)
- **Static**
- **Throws** Exception
- **Access** public

*true* function FB::log(\$Object, [\$Label = null]) [*line 179*]

**Function Parameters:**

- *mixes* **\$Object**
- *string* **\$Label**

## Log object with label to firebug console

- See [FirePHP::LOG](#)
- **Static**
- **Throws** Exception
- **Access** public

true function FB::send(\$Object) [line 137]

**Function Parameters:**

- *mixed* **\$Object**

## Log object to firebug

- **See** <http://www.firephp.org/Wiki/Reference/Fb>
- **Static**
- **Throws** Exception
- **Access** public

void function FB::setEnabled(\$Enabled) [line 75]

**Function Parameters:**

- *boolean* **\$Enabled** TRUE to enable, FALSE to disable

## Enable and disable logging to Firebug

- **See** FirePHP->setEnabled()
- **Static**
- **Access** public

void function FB::setObjectFilter(\$Class, \$Filter) [line 101]

**Function Parameters:**

- *string* **\$Class** The class name of the object
- *array* **\$Filter** An array of members to exclude

## Specify a filter to be used when encoding an object

Filters are used to exclude object members.

- **See** FirePHP->setObjectFilter()
- **Static**
- **Access** public

*void* function FB::setOptions(\$Options) [*line 113*]

**Function Parameters:**

- *array* **\$Options** The options to be set

## Set some options for the library

- **See** FirePHP->setOptions()
- **Static**
- **Access** public

*true* function FB::table(\$Label, \$Table) [*line 256*]

**Function Parameters:**

- *string* **\$Label**
- *string* **\$Table**

## Log a table in the firebug console

- **See** [FirePHP::TABLE](#)
- **Static**
- **Throws** Exception
- **Access** public

*true* function FB::trace(\$Label) [*line 243*]

**Function Parameters:**

- *string* \$Label

## Log a trace in the firebug console

- See [FirePHP::TRACE](#)
- **Static**
- **Throws** Exception
- **Access** public

*true* function FB::warn(\$Object, [\$Label = null]) [*line 205*]

**Function Parameters:**

- *mixes* \$Object
- *string* \$Label

## Log object with label to firebug console

- See [FirePHP::WARN](#)
- **Static**
- **Throws** Exception
- **Access** public

# Class Index

[*line 16*]

## dbdMVC Index Sample Controller Class

- **Package** dbdMVC
- **Uses** [dbdController](#)

*void function Index::doDefault() [line 21]*

**Set a welcome message for the view**

- **Access** public

## Class wavEdit *[line 27]*

- **Package** dbdMVC

**wavEdit::\$files**

*mixed = array() [line 28]*

**wavEdit::\$arrData**

*mixed = array() [line 29]*

**wavEdit::\$bdColor**

*mixed = [line 38]*

**wavEdit::\$bgColor**

*mixed = [line 36]*

**wavEdit::\$Db**

*mixed = [line 42]*

#### **wavEdit::\$gridColor**

*mixed = [line 39]*

#### **wavEdit::\$grpColor**

*mixed = [line 37]*

#### **wavEdit::\$header**

*mixed = "" [line 32]*

#### **wavEdit::\$imgBorder**

*mixed = [line 40]*

#### **wavEdit::\$imgHeight**

*mixed = [line 35]*

#### **wavEdit::\$imgWidth**

*mixed = [line 34]*

#### **wavEdit::\$visual\_graph\_mode**

*mixed = [line 41]*

#### **wavEdit::\$wavs**

*mixed = array() [line 31]*

Constructor *void* function wavEdit::wavEdit() *[line 44]*

*void* function wavEdit::cropFiles(\$arr) *[line 205]*

##### **Function Parameters:**

- **\$arr**

*void* function wavEdit::doClip(\$val, \$min, \$max) *[line 432]*

##### **Function Parameters:**

- **\$val**
- **\$min**
- **\$max**

*void* function wavEdit::doCrop(\$pos, \$time) *[line 218]*

##### **Function Parameters:**

- \$pos
- \$time

*void function wavEdit::doDrawImg(\$output) [line 498]*

***Function Parameters:***

- \$output

*void function wavEdit::doEcho(\$bytes, \$echo, \$inte) [line 352]*

***Function Parameters:***

- \$bytes
- \$echo
- \$inte

*void function wavEdit::doFade(\$pos, \$bytes) [line 274]*

***Function Parameters:***

- \$pos
- \$bytes

*void function wavEdit::doJoin() [line 460]*

*void function wavEdit::doSetVol8Bit(\$val) [line 408]*

***Function Parameters:***

- \$val

*void function wavEdit::doSetVol16Bit(\$val) [line 419]*

***Function Parameters:***

- \$val

*void function wavEdit::doViewHeader(\$head) [line 650]*

***Function Parameters:***

- \$head



*void function wavEdit::drawImgWave([\$val = ""]) [line 483]*

***Function Parameters:***

- **\$val**

*void function wavEdit::getBytes(\$val) [line 245]*

***Function Parameters:***

- **\$val**

*void function wavEdit::getDataArr(\$arr) [line 186]*

***Function Parameters:***

- **\$arr**

*void function wavEdit::getFileName(\$path) [line 85]*

***Function Parameters:***

- **\$path**

*void function wavEdit::getFiles(\$arr) [line 68]*

***Function Parameters:***

- **\$arr**

*void function wavEdit::getHeader() [line 136]*

*void function wavEdit::getStructure(\$binstr) [line 105]*

***Function Parameters:***

- **\$binstr**

*void function wavEdit::joinFiles([\$fname = ""]) [line 443]*

***Function Parameters:***

- **\$fname**

*void function wavEdit::setEcho(\$arr) [line 329]*

***Function Parameters:***

- **\$arr**

*void function wavEdit::setFade(\$arr) [line 259]*

***Function Parameters:***

- **\$arr**

*void function wavEdit::setVolume(\$val, [\$fname = ""]) [line 390]*

***Function Parameters:***

- **\$val**
- **\$fname**

*void function wavEdit::unpackData(\$strbin) [line 176]*

***Function Parameters:***

- **\$strbin**

*void function wavEdit::unpackHeader(\$str, [\$opt = ""]) [line 165]*

***Function Parameters:***

- **\$str**
- **\$opt**

*void function wavEdit::viewHeader([\$fname = ""]) [line 632]*

***Function Parameters:***

- **\$fname**

*void function wavEdit::viewImage([\$name = ""], [\$opt = ""]) [line 589]*

***Function Parameters:***

- **\$name**
- **\$opt**

`void function wavEdit::writeWavFile([$val = ""]) [line 616]`

***Function Parameters:***

- **\$val**

# Appendices

# Appendix A - Class Trees

## Package dbdCommon

### FlashLoaderSWFO

- [FlashLoaderSWFO](#)

### GDColor

- [GDColor](#)

### InputCSS

- [InputCSS](#)

### QTLoader

- [QTLoader](#)

### wmArrays

- [wmArrays](#)

### wmFileSystem

- [wmFileSystem](#)

## wmSort

- [wmSort](#)

## wmString

- [wmString](#)

## Package dbdMVC

## dbdController

- [dbdController](#)
  - [dbdCSS](#)
  - [dbdEmptyController](#)
  - [dbdError](#)
  - [dbdInfo](#)
  - [dbdJS](#)
    - [dbdXajaxJS](#)
- [Index](#)

## dbdDB

- [dbdDB](#)

## dbdDispatcher

- [dbdDispatcher](#)

## dbdException

- [dbdException](#)
  - [dbdHoldableException](#)

## dbdLoader

- [dbdLoader](#)

## dbdModel

- [dbdModel](#)

## dbdMVC

- [dbdMVC](#)

## dbdRequest

- [dbdRequest](#)

## dbdRouter

- [dbdRouter](#)

## dbdSmarty

- Smarty
  - [dbdSmarty](#)

## dbdURI

- [dbdURI](#)

## dbdXajax

- xajax
  - [dbdXajax](#)

## FB

- [FB](#)

## wavEdit

- [wavEdit](#)

## Package JSMin

## JSMin

- [JSMin](#)

## JSMinException

- Exception
  - [JSMinException](#)

## Package FirePHP



# FirePHP

- [FirePHP](#)

# Appendix D - Todo List

## In Package dbdMVC

In [dbdController::\\$ajaxResponse](#)

- make dbdXajaxResponse object

In [dbdController](#)

- Use Response object for output.

In [dbdMVC.php](#)

- Create readme and update dbdMVC description

In [dbdSmarty](#)

- move custom plugins to separate directory and then register them

In [dbdRequest::getCookie\(\)](#)

- How to retrieve from nested arrays

In [dbdRequest::getPost\(\)](#)

- How to retrieve from nested arrays

In [dbdRequest::getQuery\(\)](#)

- How to retrieve from nested arrays

In [dbdCSS::IMPORT\\_REGEX](#)

- allow for url() syntax

In [dbdController::registerXajaxActions\(\)](#)

- Auto insert JS into header of view output using some filter.

# Index

## C

<a href="#">constructor dbdModel::__construct()</a>	127
<i>The constructor's job is to set the table info, get the db, and call the initializer.</i>	
<a href="#">constructor dbdException::__construct()</a>	116
<i>Pass paremeters on to parent exception with prefixed message.</i>	
<a href="#">constructor dbdDispatcher::__construct()</a>	113
<i>Set router, app_dir, and controller_dir.</i>	
<a href="#">constructor dbdRequest::__construct()</a>	140
<i>Prepare the request object for use by setting the request uri.</i>	
<a href="#">constructor dbdRouter::__construct()</a>	145
<i>Set request object, parse request uri, and build base url</i>	
<a href="#">constructor wavEdit::wavEdit()</a>	170
<a href="#">constructor dbdXajax::__construct()</a>	161
<i>Set router and construct xajax object</i>	
<a href="#">constructor dbdSmarty::__construct()</a>	150
<i>Smarty constructor that sets implimentation specific variables</i>	
<a href="#">constructor dbdController::__construct()</a>	96
<i>Prepare controller for action.</i>	
<a href="#">cropResizelImage()</a>	91
<a href="#">constructor FirePHP::__construct()</a>	47
<i>The object constructor</i>	
<a href="#">constructor QTLoader::__construct()</a>	19
<i>The purpose of the constructor is to initialize the main attribute list and then fill it with the arguments (if any).</i>	
<a href="#">constructor GDColor::__construct()</a>	14
<i>The constructor's job is to convert HEX color to its integer parts and then allocate the color.</i>	
<a href="#">constructor JSMin::__construct()</a>	62
<a href="#">colorizelImage()</a>	91
<a href="#">cropImageReal()</a>	91
<a href="#">cropImage()</a>	91
<a href="#">constructor FlashLoaderSWFO::__construct()</a>	10
<i>The purpose of the constructor is to initialize the main attribute list and then fill it with the arguments (if any).</i>	

## D

<a href="#">dbdModel::__call()</a>	131
<i>Magic function allows calling of extra magic functions for testing field values.</i>	
<a href="#">dbdModel::__setDB()</a>	130
<i>Set the database object</i>	
<a href="#">dbdModel::__save()</a>	130
<i>Save all the fields for this row</i>	

<a href="#">dbdModel::__get()</a>	131
<i>Magic function for getting field values</i>	
<a href="#">dbdModel::__isset()</a>	131
<i>Magic function to check if a field is set</i>	
<a href="#">dbdModel::__unset()</a>	132
<i>Magic function to unset a field</i>	
<a href="#">dbdModel::__set()</a>	132
<i>Magic function for setting field values</i>	
<a href="#">dbdModel::initFields()</a>	130
<i>Select all the fields names for this table</i>	
<a href="#">dbdModel::init()</a>	129
<i>Select all the fields for this row</i>	
<a href="#">dbdModel::getAll()</a>	128
<i>Get all rows from this table</i>	
<a href="#">dbdModel::ensureDB()</a>	128
<i>Make sure we have an instance of the db</i>	
<a href="#">dbdModel::delete()</a>	127
<i>Delete this row</i>	
<a href="#">dbdModel::getCount()</a>	128
<i>Get a count of all rows from this table</i>	
<a href="#">dbdModel::getData()</a>	129
<i>Get the fields for this row</i>	
<a href="#">dbdModel::getID()</a>	129
<i>Get the row id</i>	
<a href="#">dbdModel::getDB()</a>	129
<i>Get the database object</i>	
<a href="#">dbdMVC</a>	132
<i>dbdMVC Front Controller Class</i>	
<a href="#">dbdMVC::VERSION</a>	133
<i>Current version number</i>	
<a href="#">dbdMVC::getRouter()</a>	136
<i>Get the instance of dbdRouter.</i>	
<a href="#">dbdMVC::getRequest()</a>	136
<i>Get the instance of dbdRequest.</i>	
<a href="#">dbdMVC::getInstance()</a>	135
<i>Get the active instance of dbdMVC or create one.</i>	
<a href="#">dbdMVC::logExecutionTime()</a>	136
<i>Log execution time of the application to the error_log.</i>	
<a href="#">dbdMVC::PHPisExposed()</a>	136
<i>Check if phpinfo display is allowed.</i>	
<a href="#">dbdMVC::setAppDir()</a>	137
<i>Set application directory property.</i>	
<a href="#">dbdMVC::run()</a>	137
<i>Run dbdMVC on the given dbdApp dir.</i>	
<a href="#">dbdMVC::getFallbackController()</a>	135
<i>Get fallback controller name.</i>	
<a href="#">dbdMVC::getExecutionTime()</a>	135
<i>Calculate execution time of the application and return.</i>	
<a href="#">dbdMVC::exposePHPInfo()</a>	134
<i>Allow phpinfo() display via dbdInfo.</i>	
<a href="#">dbdMVC::e()</a>	133
<i>Default exception handler</i>	
<a href="#">dbdMVC::debugMode()</a>	133

Check if debug mode is set.	
<a href="#">dbdMVC::getAppDir()</a>	134
Get application directory.	
<a href="#">dbdMVC::getAppName()</a>	134
Get application name.	
<a href="#">dbdMVC::getErrorLog()</a>	135
Return the error_log file name.	
<a href="#">dbdMVC::getErrorController()</a>	134
Get error controller name.	
<a href="#">dbdModel::\$id</a>	127
Row id	
<a href="#">dbdModel::\$db</a>	127
Static instance of dbdDB	
<a href="#">dbdJS::doDefault()</a>	120
Alias of doGet()	
<a href="#">dbdJS::doCombine()</a>	120
Serve multiple js files as one	
<a href="#">dbdJS::addFile()</a>	120
Check a file for existence and then add it to the array for later processing.	
<a href="#">dbdJS::doGet()</a>	120
Serve js files...	
<a href="#">dbdJS::genURL()</a>	121
Generate a combine url from an array of files	
<a href="#">dbdJS::output()</a>	121
Set headers, minify, and echo buffer.	
<a href="#">dbdJS::init()</a>	121
Set no render, cache_dir, debug, and load JSMin	
<a href="#">dbdJS::DIR_DELIM_REGEX</a>	119
Directory delimiter for passing a string of files	
<a href="#">dbdJS</a>	119
Controller class for compressing and serving js files.	
<a href="#">dbdHoldableException::hold()</a>	117
Enable holding	
<a href="#">dbdHoldableException::getHeld()</a>	117
Get array of held exceptions	
<a href="#">dbdHoldableException::\$hold</a>	117
Flag holding status	
<a href="#">dbdHoldableException::intercept()</a>	118
Exception handler and method to call when ready to throw	
<a href="#">dbdHoldableException::release()</a>	118
Disable holding and throw the first held excetion, if any	
<a href="#">dbdInfo::doDefault()</a>	119
Render phpinfo() if PHPisExposed or forward home.	
<a href="#">dbdInfo</a>	118
Controller for displaying phpInfo.	
<a href="#">dbdLoader</a>	121
dbdMVC File and Class Loader	
<a href="#">dbdLoader::DEFAULT_CONTROLLER_DIR</a>	122
Controller directory	
<a href="#">dbdLoader::setModelDir()</a>	125
Set model directory	
<a href="#">dbdLoader::setControllerDir()</a>	125

Set controller directory	124
<a href="#">dbdLoader::search()</a>	124
Search for a file in the list of path(s).	
<a href="#">dbdModel</a>	126
dbdMVC Database Table Model	
<a href="#">dbdModel::CONST_TABLE_KEY</a>	126
<a href="#">dbdModel::\$data</a>	126
Array of fields from row	
<a href="#">dbdModel::CONST_TABLE_NAME</a>	126
<a href="#">dbdLoader::loadClass()</a>	124
Load a class file. Calls dbdLoader::load().	
<a href="#">dbdLoader::load()</a>	124
Load a php file using require_once if not already loaded.	
<a href="#">dbdLoader::addPath()</a>	122
Add path to dbdLoader and php.ini	
<a href="#">dbdLoader::addControllerDir()</a>	122
Add a controller directory	
<a href="#">dbdLoader::DEFAULT_MODEL_DIR</a>	122
Model directory	
<a href="#">dbdLoader::getControllerDirs()</a>	123
Get controller directory	
<a href="#">dbdLoader::getInstance()</a>	123
Get the active instance or create one.	
<a href="#">dbdLoader::getPaths()</a>	123
Get all paths registered with dbdLoader	
<a href="#">dbdLoader::getModelDir()</a>	123
Get model directory	
<a href="#">dbdMVC::setAppName()</a>	137
Set application name property.	
<a href="#">dbdMVC::setDebugMode()</a>	138
Set debug mode.	
<a href="#">dbdSmarty::display()</a>	154
Executes & displays the template results.	
<a href="#">dbdSmarty::dbdTag()</a>	154
Smarty output filter to add dbdMVC credits to source	
<a href="#">dbdSmarty::dbdMinify()</a>	153
Minify buffer.	
<a href="#">dbdSmarty::fetch()</a>	155
Executes & returns the template result.	
<a href="#">dbdSmarty::getCSSVar()</a>	155
Get a CSS variable from the list	
<a href="#">dbdSmarty::initFlash()</a>	156
Initialize FlashLoaderSWFO	
<a href="#">dbdSmarty::getJSVar()</a>	155
Add JS variable to the list	
<a href="#">dbdSmarty::dbdIncludeFiles()</a>	153
Include CSS & JS Files Automatically	
<a href="#">dbdSmarty::dbdBeautify()</a>	153
Beautify buffer.	
<a href="#">dbdSmarty::addJS()</a>	151
Add JS files to the list	
<a href="#">dbdSmarty::addFlashMovie()</a>	151
Add a flash movie to FlashLoader	

<a href="#">dbdSmarty::addCSSVar()</a>	151
<i>Add CSS variable to the list</i>	
<a href="#">dbdSmarty::addJSVar()</a>	152
<i>Get a JS variable from the list</i>	
<a href="#">dbdSmarty::addQTMovie()</a>	152
<i>Add a flash movie to FlashLoader</i>	
<a href="#">dbdSmarty::clearJS()</a>	153
<i>Clear JS file list</i>	
<a href="#">dbdSmarty::clearCSS()</a>	152
<i>Clear CSS file list</i>	
<a href="#">dbdSmarty::initQuickTime()</a>	156
<i>Initialize QTLoader</i>	
<a href="#">dbdSmarty::setCssHost()</a>	156
<i>Set CSS host for inclusion</i>	
<a href="#">dbdURI::setPositionList()</a>	160
<i>Set a parameter position list for a specific controller/action.</i>	
<a href="#">dbdURI::setParam()</a>	160
<i>Set request parameter, recursively</i>	
<a href="#">dbdURI::set()</a>	159
<i>Set request uri</i>	
<a href="#">dbdXajax</a>	161
<i>dbdMVC Xajax subclass</i>	
<a href="#">dbdXajax::xmlToArray()</a>	161
<i>Wrapper for xajax::_xmlToArray() to set paramaters with dbdRouter</i>	
<a href="#">dbdXajaxJS::doDefault()</a>	162
<i>Serve xajax javascript file</i>	
<a href="#">dbdXajaxJS</a>	162
<i>Controller class for serving xajax javascript file.</i>	
<a href="#">dbdURI::replace()</a>	159
<i>Replace the given url with the new components given</i>	
<a href="#">dbdURI::getParams()</a>	159
<i>Get all parameter</i>	
<a href="#">dbdURI</a>	157
<i>dbdMVC URI Parsing and creating class</i>	
<a href="#">dbdSmarty::wasRendered()</a>	157
<i>Has a trmplate been rendered?</i>	
<a href="#">dbdSmarty::setJSHost()</a>	157
<i>Set JS host for inclusion</i>	
<a href="#">dbdURI::create()</a>	158
<i>Create a new url based on the given components</i>	
<a href="#">dbdURI::getAction()</a>	158
<i>Get action name</i>	
<a href="#">dbdURI::getInstance()</a>	159
<i>Get the active instance or create one.</i>	
<a href="#">dbdURI::getController()</a>	158
<i>Get controller name</i>	
<a href="#">dbdSmarty::addCSS()</a>	150
<i>Add CSS files to the list</i>	
<a href="#">dbdSmarty::VIEW_DIR</a>	150
<i>Path to directory where smarty templates will live relative to application dir.</i>	
<a href="#">dbdRequest::getServer()</a>	143
<i>Retrieve a member of the \$_SERVER superglobal.</i>	
<a href="#">dbdRequest::getRequestURI()</a>	143



Get request uri	142
<a href="#">dbdRequest::getQuery()</a>	142
Retrieve a member of the \$_GET superglobal.	
<a href="#">dbdRequest::has()</a>	143
Alias to __isset()	
<a href="#">dbdRequest::isXmlHttpRequest()</a>	144
Is the request a Javascript XMLHttpRequest?	
Should work with Prototype/Script.aculo.us, possibly others.	
<a href="#">dbdRequest::setRequestURI()</a>	144
Set request uri	
<a href="#">dbdRequest::set()</a>	144
Set parameters at runtime	
<a href="#">dbdRequest::getPost()</a>	142
Retrieve a member of the \$_POST superglobal.	
<a href="#">dbdRequest::getHeader()</a>	141
Return the value of the given HTTP header. Pass the header name as the plain, HTTP-specified header name. Ex.: Ask for 'Accept' to get the Accept header, 'Accept-Encoding' to get the Accept-Encoding header.	
<a href="#">dbdMVC::setFallbackController()</a>	139
Set fallback_controller property.	
<a href="#">dbdMVC::setErrorLog()</a>	139
Set error_log property.	
<a href="#">dbdMVC::setErrorController()</a>	138
Set error_controller property.	
<a href="#">dbdRequest</a>	139
HTTP Request Class.	
<a href="#">dbdRequest::get()</a>	140
Access values contained in the superglobals as public members	
<a href="#">dbdRequest::getEnv()</a>	141
Retrieve a member of the \$_ENV superglobal.	
<a href="#">dbdRequest::getCookie()</a>	141
Retrieve a member of the \$_COOKIE superglobal.	
<a href="#">dbdRequest:: __isset()</a>	145
Check to see if a property is set	
<a href="#">dbdRouter</a>	145
Request routing class	
<a href="#">dbdSmarty</a>	149
dbdMVC Smarty subclass	
<a href="#">dbdRouter::unsetParam()</a>	148
Unset named parameter	
<a href="#">dbdRouter::setParam()</a>	148
Set named parameter	
<a href="#">dbdSmarty::CACHE_DIR</a>	149
Path to directory where smarty cache will live relative to view dir.	
<a href="#">dbdSmarty::COMPILED_TEMPLATE_DIR</a>	149
Path to directory where smarty compiled templates will live relative to view dir.	
<a href="#">dbdSmarty::TEMPLATE_DIR</a>	150
Path to directory where smarty templates will live relative to view dir.	
<a href="#">dbdSmarty::CONFIG_DIR</a>	150
Path to directory where smarty configs will live relative to view dir.	
<a href="#">dbdRouter::setController()</a>	148
Set controller name	
<a href="#">dbdRouter::setAction()</a>	147

Set action name	
<a href="#">dbdRouter::getAction()</a>	146
Get action name	
<a href="#">dbdRouter::addParam()</a>	146
Add value to array parameter	
<a href="#">dbdRouter::getController()</a>	146
Get controller name	
<a href="#">dbdRouter::getParam()</a>	146
Get named parameter	
<a href="#">dbdRouter::getURL()</a>	147
Get current url.	
<a href="#">dbdRouter::getParams()</a>	147
Get all paramters	
<a href="#">dbdHoldableException::\$held</a>	117
Array of held exceptions	
<a href="#">dbdHoldableException</a>	116
dbdMVC Exception class.	
<a href="#">dbdController::\$view</a>	95
Instance of dbdSmarty	
<a href="#">dbdController::\$router</a>	95
Instance of dbdRouter	
<a href="#">dbdController::\$model</a>	94
Instance of some model.	
<a href="#">dbdController::\$xajax</a>	95
Instance of dbdXajax	
<a href="#">dbdController::\$xajaxResponse</a>	95
Instance of xajaxResponse object	
<a href="#">dbdController::autoExec()</a>	96
Auto execute method to be executed after the action.	
<a href="#">dbdController::assignAllParams()</a>	96
Assign all request parameters to the view as an associative array.	
<a href="#">dbdController</a>	94
Abstract Parent Controller Class	
All controller classes must extend this class or a child of this class.	
<a href="#">dbdXajaxJS.php</a>	90
dbdXajaxJS.php :: dbdXajaxJS Class File	
<a href="#">dbdRequest.php</a>	85
dbdRequest.php :: dbdRequest Class File	
<a href="#">dbdModel.php</a>	84
dbdModel.php :: dbdModel Class File	
<a href="#">dbdLoader.php</a>	83
dbdLoader.php :: dbdLoader Class File & __autoload() Function	
<a href="#">dbdRouter.php</a>	86
dbdRouter.php :: dbdRouter Class File	
<a href="#">dbdSmarty.php</a>	87
dbdSmarty.php :: dbdSmarty Class File	
<a href="#">dbdXajax.php</a>	89
dbdXajax.php :: dbdXajax Class File	
<a href="#">dbdURI.php</a>	88
dbdURI.php :: dbdURI Class File	
<a href="#">dbdController::autoRender()</a>	97
Auto render the view.	
<a href="#">dbdController::dnit()</a>	97

<i>Default dnit method.</i>	
<a href="#">dbdController::init()</a>	100
<i>Default init method.</i>	
<a href="#">dbdController::getURL()</a>	100
<i>Get current request url.</i>	
<a href="#">dbdController::getUploadInfo()</a>	100
<i>Get the progress information on a current upload via APC.</i>	
<a href="#">dbdController::noRender()</a>	101
<i>Disable auto render on destruction.</i>	
<a href="#">dbdController::registerXajaxActions()</a>	101
<i>Register controller actions with dbdXajax.</i>	
<a href="#">dbdController::setTemplate()</a>	102
<i>Set template filename.</i>	
<a href="#">dbdController::setParam()</a>	101
<i>Set request parameter.</i>	
<a href="#">dbdController::getTemplate()</a>	99
<i>Get template filename.</i>	
<a href="#">dbdController::getParams()</a>	99
<i>Get all parameters.</i>	
<a href="#">dbdController::forward()</a>	98
<i>Preform an HTTP Location change.</i>	
<a href="#">dbdController::doXajax()</a>	98
<i>Prepare controller for xajax request.</i>	
<a href="#">dbdController::doDefault()</a>	97
<i>Default action method.</i>	
<a href="#">dbdController::getAction()</a>	98
<i>Get action name.</i>	
<a href="#">dbdController::getAppDir()</a>	98
<i>Get application directory.</i>	
<a href="#">dbdController::getParam()</a>	99
<i>Get request parameter.</i>	
<a href="#">dbdController::getController()</a>	99
<i>Get controller name.</i>	
<a href="#">dbdJS.php</a>	82
<i>dbdJS.php :: dbdJS Class File</i>	
<a href="#">dbdInfo.php</a>	81
<i>dbdInfo.php :: dbdInfo Class File</i>	
<a href="#">DBD_SMARTY_DIR</a>	67
<i>Smarty package directory</i>	
<a href="#">DBD_PLUG_DIR</a>	67
<i>dbdMVC plug-in directory</i>	
<a href="#">DBD_MVC_CLI</a>	67
<i>Are we running in CLI mode?</i>	
<a href="#">DBD_SMARTY_PLUG_DIR</a>	67
<i>Smarty addOn plug-in directory directory (used to avoid disrupting standard plug-in dir)</i>	
<a href="#">DBD_XAJAX_DIR</a>	67
<i>xAjax package directory</i>	
<a href="#">DBCONN_DEFAULT_DB</a>	70
<i>Constant default database name if none is specified at connection time</i>	
<a href="#">dbconn.inc</a>	70
<i>dbconn.inc :: The Database string constants</i>	
<a href="#">DBD_FIREPHP_DIR</a>	67
<i>FirePHP core directory</i>	

<a href="#"><u>DBD_DS</u></a>	67
<i>Shorthand for DIRECTORY_SEPARATOR</i>	
<a href="#"><u>DBD_DEBUG_ALL</u></a>	67
<i>Turn on all debugging</i>	
<a href="#"><u>DBD_CACHE_DIR</u></a>	67
<i>dbdMVC cache directory</i>	
<a href="#"><u>dbdLog()</u></a>	66
<a href="#"><u>DBD_DEBUG_CSS</u></a>	67
<a href="#"><u>DBD_DEBUG_DB</u></a>	67
<i>Turn on all debugging</i>	
<a href="#"><u>DBD_DEBUG_JS</u></a>	67
<a href="#"><u>DBD_DEBUG_HTML</u></a>	67
<a href="#"><u>DBCONN_INCLUDED</u></a>	70
<i>Constant boolean to flag file inclusion.</i>	
<a href="#"><u>DB_DRIVER</u></a>	70
<i>Database connection strings</i>	
<a href="#"><u>dbdDispatcher.php</u></a>	76
<i>dbdDispatcher.php :: dbdDispatcher Class File</i>	
<a href="#"><u>dbdDB.php</u></a>	75
<i>dbdDB.php :: dbdDB Class File</i>	
<a href="#"><u>dbdCSS.php</u></a>	74
<i>dbdCSS.php :: dbdCSS Class File</i>	
<a href="#"><u>dbdEmptyController.php</u></a>	77
<i>dbdEmptyController.php :: dbdEmptyController Class File</i>	
<a href="#"><u>dbdError.php</u></a>	78
<i>dbdError.php :: dbdError Class File</i>	
<a href="#"><u>dbdHoldableException.php</u></a>	80
<i>dbdHoldableException.php :: dbdHoldableException Class File</i>	
<a href="#"><u>dbdException.php</u></a>	79
<i>dbdException.php :: dbdException Class File</i>	
<a href="#"><u>dbdController.php</u></a>	73
<i>dbdController.php :: dbdController Class File</i>	
<a href="#"><u>DBD_MVC_DIR</u></a>	72
<a href="#"><u>DB_PASS</u></a>	70
<a href="#"><u>DB_HOST</u></a>	70
<a href="#"><u>DB_PORT</u></a>	70
<a href="#"><u>DB_USER</u></a>	70
<a href="#"><u>DBD_DOC_ROOT</u></a>	72
<a href="#"><u>DBD_APP_DIR</u></a>	72
<a href="#"><u>dbdController::unsetParam()</u></a>	102
<i>Unset request parameter.</i>	
<a href="#"><u>dbdController::__destruct()</u></a>	102
<i>Conclude the controller action.</i>	
<a href="#"><u>dbdDB::time()</u></a>	111
<i>Format time for SQL insert</i>	
<a href="#"><u>dbdDB::prepExec()</u></a>	110
<i>Prepare &amp; excute a query statement</i>	
<a href="#"><u>dbdDB::nextAutoID()</u></a>	110
<i>Selects the next auto_increment id</i>	
<a href="#"><u>dbdDB::__call()</u></a>	111
<i>Magic function to call PDO functions and rethrow any</i>	
<a href="#"><u>dbdDispatcher</u></a>	111
<i>Dispatch controller actions.</i>	

<a href="#"><u>dbdDispatcher::BUILTIN_PREFIX</u></a>	112
<i>Builtin controller class prefix</i>	
<a href="#"><u>dbdDispatcher::ACTION_PREFIX</u></a>	112
<i>Action method name prefix</i>	
<a href="#"><u>dbdDB::getInstance()</u></a>	109
<i>Include db_conn string file.</i>	
<a href="#"><u>dbdDB::date()</u></a>	109
<i>Format date for SQL insert</i>	
<a href="#"><u>dbdDB::COMP_NULLEQ</u></a>	108
<a href="#"><u>dbdDB::COMP_NLIKE</u></a>	108
<a href="#"><u>dbdDB::COMP_NIN</u></a>	108
<a href="#"><u>dbdDB::COMP_TYPE</u></a>	108
<a href="#"><u>dbdDB::DATE_FORMAT</u></a>	108
<i>SQL compliant date formating string</i>	
<a href="#"><u>dbdDB::TIME_FORMAT</u></a>	109
<i>SQL compliant time formating string</i>	
<a href="#"><u>dbdDB::DBCONN_PATH</u></a>	108
<i>path to database connection string file</i>	
<a href="#"><u>dbdDispatcher::DEFAULT_ACTION</u></a>	112
<i>Default action name</i>	
<a href="#"><u>dbdDispatcher::DEFAULT_CONTROLLER</u></a>	112
<i>Default controller name</i>	
<a href="#"><u>dbdError::ERR_408</u></a>	115
<i>HTTP Status 408 Description</i>	
<a href="#"><u>dbdError::ERR_404</u></a>	115
<i>HTTP Status 404 Description</i>	
<a href="#"><u>dbdError::ERR_403</u></a>	114
<i>HTTP Status 403 Description</i>	
<a href="#"><u>dbdError::ERR_500</u></a>	115
<i>HTTP Status 500 Description</i>	
<a href="#"><u>dbdError::doDefault()</u></a>	115
<i>Attept to render error.tpl or just echo error.</i>	
<a href="#"><u>dbdException</u></a>	116
<i>dbdMVC Exception class.</i>	
<a href="#"><u>dbdError::doError()</u></a>	115
<i>Crazy static action method expects intstance of dbdController for dbdError extension.</i>	
<a href="#"><u>dbdError::ERR_401</u></a>	114
<i>HTTP Status 401 Description</i>	
<a href="#"><u>dbdError::ERR_400</u></a>	114
<i>HTTP Status 400 Description</i>	
<a href="#"><u>dbdDispatcher::MAGIC_ACTION</u></a>	113
<i>Magic action name</i>	
<a href="#"><u>dbdDispatcher::FALLBACK_CONTROLLER</u></a>	112
<i>Fallback controller class if none exists</i>	
<a href="#"><u>dbdDispatcher::ERROR_CONTROLLER</u></a>	112
<i>Error controller class if is set</i>	
<a href="#"><u>dbdDispatcher::XAJAX_ACTION_PREFIX</u></a>	113
<i>Xajax action method name prefix</i>	
<a href="#"><u>dbdDispatcher::dispatch()</u></a>	113
<i>Dispatch actions</i>	
<a href="#"><u>dbdError</u></a>	114
<i>Error handling controller class.</i>	
<a href="#"><u>dbdEmptyController</u></a>	114

<i>Empty controller class for rendering templates without a controller!</i>	108
<a href="#">dbdDB::COMP_NEQ</a>	108
<a href="#">dbdDB::COMP_NBETWEEN</a>	108
<a href="#">dbdCSS::CACHE_INFO_SPRITE</a>	105
<i>Css caching info sprite pattern.</i>	
<a href="#">dbdCSS::CACHE_INFO_FILES</a>	105
<i>Css caching info file list pattern.</i>	
<a href="#">dbdCSS::BUTT_TAG_REGEX</a>	104
<i>Css buttons at-tag pattern.</i>	
<a href="#">dbdCSS::CSS_EXT_REGEX</a>	105
<i>Css file extension pattern.</i>	
<a href="#">dbdCSS::DEBUG_REGEX</a>	105
<i>Debug comment to turn off minification.</i>	
<a href="#">dbdCSS::IMPORT_REGEX</a>	105
<i>Css import regular expression.</i>	
<a href="#">dbdCSS::DIR_DELIM_REGEX</a>	105
<i>Directory delimiter for passing a string of files</i>	
<a href="#">dbdCSS::BUTT_PROP_DEF_REGEX</a>	104
<i>Css button property definition pattern.</i>	
<a href="#">dbdCSS::BUTT_PROP_DEF_HOVER_REGEX</a>	104
<i>Css button property hover pattern.</i>	
<a href="#">dbdCSS::BLOCK_OPEN_REGEX</a>	103
<i>Css block open pattern.</i>	
<a href="#">dbdCSS::BLOCK_CLOSE_REGEX</a>	103
<i>Css block close pattern.</i>	
<a href="#">dbdCSS</a>	103
<i>Controller class for compressing and serving css files.</i>	
<a href="#">dbdCSS::BUTT_DEF_TAG_REGEX</a>	104
<i>Css button definition pattern.</i>	
<a href="#">dbdCSS::BUTT_GEN_TAG_REGEX</a>	104
<i>Css generated buttons at-tag pattern.</i>	
<a href="#">dbdCSS::BUTT_PROP_DEF_DISABLED_REGEX</a>	104
<i>Css button property disabled pattern.</i>	
<a href="#">dbdCSS::BUTT_GEN_TPL_TAG_REGEX</a>	104
<i>Css generated buttons template tag pattern.</i>	
<a href="#">dbdCSS::MATH_CALL_REGEX</a>	105
<i>Css math call pattern. (for simple equations)</i>	
<a href="#">dbdCSS::MATH_SAFE_REGEX</a>	106
<i>Css math safe characters pattern. (for simple equations)</i>	
<a href="#">dbdDB::COMP_GTEQ</a>	107
<a href="#">dbdDB::COMP_GT</a>	107
<a href="#">dbdDB::COMP_EQ</a>	107
<a href="#">dbdDB::COMP_IN</a>	108
<a href="#">dbdDB::COMP_LIKE</a>	108
<a href="#">dbdDB::COMP_LTEQ</a>	108
<a href="#">dbdDB::COMP_LT</a>	108
<a href="#">dbdDB::COMP_BETWEEN</a>	107
<a href="#">dbdDB</a>	107
<i>dbdMVC Database Abstraction Layer</i>	
<a href="#">dbdCSS::VAR_DEF_REGEX</a>	106
<i>Css variable definition pattern.</i>	
<a href="#">dbdCSS::VAR_CALL_REGEX</a>	106
<i>Css variable call pattern. (for dereferencing)</i>	

<a href="#">dbdCSS::VAR_TAG_REGEX</a>	106
<i>Css variables at-tag pattern.</i>	
<a href="#">dbdCSS::doCombine()</a>	106
<i>Serve multiple css files as one</i>	
<a href="#">dbdCSS::init()</a>	107
<i>Set debug</i>	
<a href="#">dbdCSS::genURL()</a>	106
<i>Generate a combine url from an array of files</i>	
<a href="#">dbdMVC.php</a>	66
<i>dbdMVC.php :: dbdMVC Include File &amp; Front Controller Class</i>	

## F

<a href="#">FirePHP::setObjectFilter()</a>	55
<i>Specify a filter to be used when encoding an object</i>	
<a href="#">FirePHP::setHeader()</a>	55
<i>Send header</i>	
<a href="#">FirePHP::setEnabled()</a>	55
<i>Enable and disable logging to Firebug</i>	
<a href="#">FirePHP::registerExceptionHandler()</a>	55
<i>Register FirePHP as your exception handler</i>	
<a href="#">FirePHP::setOptions()</a>	56
<i>Set some options for the library</i>	
<a href="#">FirePHP::setProcessorUrl()</a>	56
<i>Set custom processor url for FirePHP</i>	
<a href="#">FirePHP::warn()</a>	58
<i>Log object with label to firebug console</i>	
<a href="#">FirePHP::trace()</a>	57
<i>Log a trace in the firebug console</i>	
<a href="#">FirePHP::table()</a>	57
<i>Log a table in the firebug console</i>	
<a href="#">FirePHP::setRendererUrl()</a>	57
<i>Set custom renderer url for FirePHP</i>	
<a href="#">FirePHP::registerErrorHandler()</a>	54
<i>Register FirePHP as your error handler</i>	
<a href="#">FirePHP::registerAssertionHandler()</a>	54
<i>Register FirePHP driver as your assert callback</i>	
<a href="#">FirePHP::info()</a>	52
<i>Log object with label to firebug console</i>	
<a href="#">FirePHP::groupEnd()</a>	51
<i>Ends a group you have started before</i>	
<a href="#">FirePHP::group()</a>	51
<i>Start a group for following messages.</i>	
<a href="#">FirePHP::getUserAgent()</a>	51
<i>Get user agent</i>	
<a href="#">FirePHP::init()</a>	52
<i>Creates FirePHP object and stores it for singleton access</i>	
<a href="#">FirePHP::is_utf8()</a>	52
<i>Returns true if \$string is valid UTF-8 and false otherwise.</i>	
<a href="#">FirePHP::newException()</a>	53
<i>Returns a new exception</i>	
<a href="#">FirePHP::log()</a>	53



<a href="#"><i>Log object with label to firebug console</i></a>	
<a href="#">FirePHP::jsonEncode()</a>	53
<i>Encode an object into a JSON string</i>	
<a href="#">FirePHP::escapeTrace()</a>	58
<i>Escape trace path for windows systems</i>	
<a href="#">FirePHP::escapeTraceFile()</a>	59
<i>Escape file information of trace for windows systems</i>	
<a href="#">FB::setEnabled()</a>	166
<i>Enable and disable logging to Firebug</i>	
<a href="#">FB::send()</a>	166
<i>Log object to firebug</i>	
<a href="#">FB::log()</a>	165
<i>Log object with label to firebug console</i>	
<a href="#">FB::info()</a>	165
<i>Log object with label to firebug console</i>	
<a href="#">FB::setObjectFilter()</a>	166
<i>Specify a filter to be used when encoding an object</i>	
<a href="#">FB::setOptions()</a>	167
<i>Set some options for the library</i>	
<a href="#">FB::warn()</a>	168
<i>Log object with label to firebug console</i>	
<a href="#">FB::trace()</a>	168
<i>Log a trace in the firebug console</i>	
<a href="#">FB::table()</a>	167
<i>Log a table in the firebug console</i>	
<a href="#">FB::groupEnd()</a>	164
<i>Ends a group you have started before</i>	
<a href="#">FB::group()</a>	164
<i>Start a group for following messages</i>	
<a href="#">fb()</a>	69
<i>Sends the given data to the FirePHP Firefox Extension.</i>	
<a href="#">fb.php</a>	69
<a href="#">FirePHP::sleep()</a>	59
<i>When the object gets serialized only include specific object members.</i>	
<a href="#">FirePHP::standardizePath()</a>	59
<i>Standardizes path for windows systems.</i>	
<a href="#">FB</a>	163
<a href="#">FB::dump()</a>	163
<i>Dumps key and variable to firebug server panel</i>	
<a href="#">FB::getOptions()</a>	164
<i>Get options for the library</i>	
<a href="#">FB::getEnabled()</a>	164
<i>Check if logging is enabled</i>	
<a href="#">FB::error()</a>	163
<i>Log object with label to firebug console</i>	
<a href="#">FirePHP::getOptions()</a>	51
<i>Get options from the library</i>	
<a href="#">FirePHP::getInstance()</a>	50
<i>Gets singleton instance of FirePHP</i>	
<a href="#">FirePHP::EXCEPTION</a>	43
<i>Displays an exception in firebug console</i>	
<a href="#">FirePHP::ERROR</a>	42
<i>Firebug ERROR level</i>	



<a href="#">FirePHP::DUMP</a>	42
<i>Dumps a variable to firebug's server panel</i>	
<a href="#">FirePHP</a>	42
<i>Sends the given data to the FirePHP Firefox Extension.</i>	
<a href="#">FirePHP::GROUP_END</a>	43
<i>Ends a group in firebug console</i>	
<a href="#">FirePHP::GROUP_START</a>	43
<i>Starts a group in firebug console</i>	
<a href="#">FirePHP::TRACE</a>	43
<i>Displays a stack trace in firebug console</i>	
<a href="#">FirePHP::TABLE</a>	43
<i>Displays an table in firebug console</i>	
<a href="#">FirePHP::LOG</a>	43
<i>Firebug LOG level</i>	
<a href="#">FirePHP::INFO</a>	43
<i>Firebug INFO level</i>	
<a href="#">FirePHP.class.php</a>	40
<i>*** BEGIN LICENSE BLOCK *****</i>	
<a href="#">FlashLoaderSWFO::setAtts()</a>	13
<i>Fill the attribute list with the arguments (if any).</i>	
<a href="#">FlashLoaderSWFO::getCommonJSPath()</a>	11
<i>Prints include to JavaScript library.</i>	
<a href="#">FlashLoaderSWFO::getCommonJS()</a>	11
<i>Prints include to JavaScript library.</i>	
<a href="#">FlashLoaderSWFO::addMovie()</a>	11
<i>Adds movie arguments to list of movies to be embedded later by loadMovies().</i>	
<a href="#">FlashLoaderSWFO</a>	10
<i>The FlashLoaderSWFO class is used to effortlessly embedd Flash movies in a (X)HTML Document.</i>	
<a href="#">FlashLoaderSWFO::getMovie()</a>	11
<i>Gets (embbeds) flash movie in (X)HTML document.</i>	
<a href="#">FlashLoaderSWFO::getMovies()</a>	12
<i>Gets movies in list created by addMovie()</i>	
<a href="#">FlashLoaderSWFO::loadMovies()</a>	13
<i>Loads movies in list created by addMovie()</i>	
<a href="#">FlashLoaderSWFO::loadMovie()</a>	12
<i>Loads (embbeds) flash movie in (X)HTML document.</i>	
<a href="#">FlashLoaderSWFO::hasMovies()</a>	12
<i>Check to see if there are any movies to load.</i>	
<a href="#">FirePHP::VERSION</a>	44
<i>FirePHP version</i>	
<a href="#">FirePHP::WARN</a>	44
<i>Firebug WARN level</i>	
<a href="#">FirePHP::encodeTable()</a>	48
<i>Encodes a table by encoding each row and column with encodeObject()</i>	
<a href="#">FirePHP::encodeObject()</a>	48
<i>Encodes an object including members with protected and private visibility</i>	
<a href="#">FirePHP::dump()</a>	47
<i>Dumps key and variable to firebug server panel</i>	
<a href="#">FirePHP::detectClientExtension()</a>	47
<i>Check if FirePHP is installed on client</i>	
<a href="#">FirePHP::error()</a>	48

<i>Log object with label to firebug console</i>	49
<a href="#">FirePHP::errorHandler()</a>	49
<i>FirePHP's error handler</i>	
<a href="#">FirePHP::getEnabled()</a>	50
<i>Check if logging is enabled</i>	
<a href="#">FirePHP::fb()</a>	50
<i>Log variable to Firebug</i>	
<a href="#">FirePHP::exceptionHandler()</a>	49
<i>FirePHP's exception handler</i>	
<a href="#">FirePHP::assertionHandler()</a>	47
<i>FirePHP's assertion handler</i>	
<a href="#">FirePHP::\$throwErrorExceptions</a>	46
<i>Flag whether to throw PHP errors that have been converted to Exceptions</i>	
<a href="#">FirePHP::\$instance</a>	45
<i>Singleton instance of FirePHP</i>	
<a href="#">FirePHP::\$inExceptionHandler</a>	44
<i>Flag whether we are logging from within the exception handler</i>	
<a href="#">FirePHP::\$enabled</a>	44
<i>Flag to enable/disable logging</i>	
<a href="#">FirePHP::\$convertAssertionErrorsToExceptions</a>	44
<i>Flag whether to convert PHP assertion errors to Exceptions</i>	
<a href="#">FirePHP::\$messageIndex</a>	45
<i>Wildfire protocol message index</i>	
<a href="#">FirePHP::\$objectFilters</a>	45
<i>Filters used to exclude object members when encoding</i>	
<a href="#">FirePHP::\$throwAssertionExceptions</a>	46
<i>Flag whether to throw PHP assertion errors that have been converted to Exceptions</i>	
<a href="#">FirePHP::\$options</a>	46
<i>Options for the library</i>	
<a href="#">FirePHP::\$objectStack</a>	46
<i>A stack of objects used to detect recursion during object encoding</i>	
<a href="#">FlashLoaderSWFObject.php</a>	2
<i>FlashLoaderSWFObject.php :: Flash movie loading/embedding class</i>	

## G

<a href="#">GdColor::getRed()</a>	15
<i>Get the integer color value for the Red channel.</i>	
<a href="#">GdColor::hex2arr()</a>	15
<i>Static function for breaking a HEX color into its various channels.</i>	
<a href="#">GdColor::int2arr()</a>	16
<i>Static function for breaking an integer color into its various channels.</i>	
<a href="#">GdColor::__destruct()</a>	16
<i>Clean up the allocated colors.</i>	
<a href="#">GdColor::getHex()</a>	15
<i>Get the HEX color value of the allocated color.</i>	
<a href="#">GdColor::getGreen()</a>	15
<i>Get the integer color value for the Green channel.</i>	
<a href="#">GdColor</a>	13
<i>The GdColor Class is used to easily</i>	

<i>handle colors while using the GD Library.</i>	
<a href="#">GDColor::getAlpha()</a>	14
<i>Get the integer color value for the Alpha channel.</i>	
<a href="#">GDColor::getBlue()</a>	14
<i>Get the integer color value for the Blue channel.</i>	
<a href="#">GDColor::getColor()</a>	14
<i>Get the integer color value for use in GD functions.</i>	
<a href="#">GDColor.php</a>	3
<i>GDColor.php :: The GD Color Class</i>	

## I

<a href="#">imageLineThick()</a>	92
<a href="#">imagefilledellipsealpha()</a>	92
<a href="#">imageCreateFrom()</a>	91
<a href="#">imageType()</a>	92
<a href="#">imageTypesArray()</a>	92
<a href="#">Index::doDefault()</a>	169
<i>Set a welcome message for the view</i>	
<a href="#">Index</a>	168
<i>dbdMVC Index Sample Controller Class</i>	
<a href="#">index.php</a>	72
<i>index.php :: dbdMVC Bootstrap Sample File</i>	
<a href="#">Index.php</a>	71
<i>Index.php :: Index Controller Class File</i>	
<a href="#">InputCSS::button()</a>	16
<i>Generate a submit button.</i>	
<a href="#">InputCSS</a>	16
<i>The InputCSS Class is to generate the markup and JQuery calls to make a fancy CSS form input.</i>	
<a href="#">InputCSS::checkbox()</a>	17
<i>Generate a checkbox input.</i>	
<a href="#">InputCSS::file()</a>	17
<i>Generate a file/browse input.</i>	
<a href="#">InputCSS::reset()</a>	18
<i>Generate a reset button.</i>	
<a href="#">InputCSS::radio()</a>	18
<i>Generate a radio input.</i>	
<a href="#">InputCSS.php</a>	4
<i>InputCSS.php :: The Fancy XHTML/CSS Form Input Class</i>	

## J

<a href="#">JSMIn::isAlphaNum()</a>	63
<a href="#">JSMIn::get()</a>	63
<a href="#">JSMIn::action()</a>	63
<a href="#">JSMIn::min()</a>	63
<a href="#">JSMIn::minify()</a>	64
<a href="#">JSMInException</a>	64
<a href="#">JSMIn::peek()</a>	64
<a href="#">JSMIn::next()</a>	64

<a href="#">JSMIn::\$output</a>	62
<a href="#">JSMIn::\$lookAhead</a>	62
<a href="#">JSMIn::\$a</a>	61
<a href="#">JSMIn::ORD_SPACE</a>	61
<a href="#">JSMIn::ORD_LF</a>	61
<a href="#">JSMIn::\$b</a>	61
<a href="#">JSMIn::\$input</a>	61
<a href="#">JSMIn::\$inputLength</a>	62
<a href="#">JSMIn::\$inputIndex</a>	62
<a href="#">JSMIn</a>	60
<i>jsmIn.php - PHP implementation of Douglas Crockford's JSMIn.</i>	

## O

<a href="#">outputAndDestroy()</a>	92
------------------------------------	----

## Q

<a href="#">QTLoader::hasMovies()</a>	21
<i>Check to see if there are any movies to load.</i>	
<a href="#">QTLoader::loadMovie()</a>	21
<i>Loads (embeds) quicktime movie in (X)HTML document.</i>	
<a href="#">QTLoader::loadMovies()</a>	22
<i>Loads movies in list created by addMovie()</i>	
<a href="#">QTLoader::setAtts()</a>	22
<i>Fill the attribute list with the arguments (if any).</i>	
<a href="#">QTLoader::getMovies()</a>	21
<i>Gets movies in list created by addMovie()</i>	
<a href="#">QTLoader::getMovie()</a>	20
<i>Gets (embeds) quicktime movie in (X)HTML document.</i>	
<a href="#">QTLoader</a>	19
<i>The QTLoader class is used to effortlessly embedd QuickTime movies in a (X)HTML Document.</i>	
<a href="#">QTLoader::addMovie()</a>	20
<i>Adds movie arguments to list of movies to be embedded later by loadMovies().</i>	
<a href="#">QTLoader::getCommonJS()</a>	20
<i>Prints include to JavaScript library.</i>	
<a href="#">QTLoader::getCommonJSPath()</a>	20
<i>Prints include to JavaScript library.</i>	
<a href="#">QTLoader.php</a>	5
<i>QTLoader.php :: QuickTime movie loading/embedding class</i>	

## R

<a href="#">resizeImage()</a>	93
<a href="#">reflectImage()</a>	92

## S

## W

<a href="#">wavEdit::\$bdColor</a>	169
<a href="#">wavEdit::\$bgColor</a>	169
<a href="#">wavEdit::\$arrData</a>	169
<a href="#">wavEdit::\$files</a>	169
<a href="#">wavEdit</a>	169
<a href="#">wavEdit::\$Db</a>	170
<a href="#">wavEdit::\$gridColor</a>	170
<a href="#">wavEdit::\$imgHeight</a>	170
<a href="#">wavEdit::\$imgWidth</a>	170
<a href="#">wavEdit::\$imgBorder</a>	170
<a href="#">wavEdit::\$header</a>	170
<a href="#">wavEdit::\$grpColor</a>	170
<a href="#">wmGD.php</a>	91
<a href="#">wmString::wkt2kml()</a>	38
<i>Convert WKT to KML.</i>	
<a href="#">wmString::numberSuffix()</a>	34
<i>Add suffixes to numbers.</i>	
<a href="#">wmString::randomWord()</a>	35
<i>Generate a random readable word.</i>	
<a href="#">wmString::makeSingular()</a>	34
<i>Takes a string and converts to the singular version.</i>	
<a href="#">wmString::makePlural()</a>	34
<i>Takes a string and converts to the plural version.</i>	
<a href="#">wmString::lcFirst()</a>	33
<i>Turns the first characher of a string to lower case</i>	
<a href="#">wmString::superCap2UnderLower()</a>	35
<i>Takes a super capped string (SuperCapString) and adds underscores and makes lowercase</i>	
<a href="#">wmString::superCapAddSpaces()</a>	35
<i>Takes a super capped string (SuperCapString) and adds spaces</i>	
<a href="#">wmString::truncate()</a>	37
<i>Truncate a string...</i>	
<a href="#">wmString::underLower2SuperCap()</a>	37
<i>Takes an underscored lower case string (under_lower_string) converts it to SuperCaps</i>	
<a href="#">wmString::timePastFormat()</a>	36
<i>Format the about of time past.</i>	
<a href="#">wmString::timeLengthFormat()</a>	36
<i>Format a length of time.</i>	
<a href="#">wmString::timeFormat()</a>	36
<i>Format the amount of time in seconds to hh::mm:ss</i>	
<a href="#">wavEdit::\$visual_graph_mode</a>	170
<a href="#">wavEdit::\$waves</a>	170
<a href="#">wavEdit::joinFiles()</a>	172
<a href="#">wavEdit::setEcho()</a>	173
<a href="#">wavEdit::getStructure()</a>	172
<a href="#">wavEdit::getHeader()</a>	172
<a href="#">wavEdit::getFiles()</a>	172
<a href="#">wavEdit::setFade()</a>	173
<a href="#">wavEdit::setVolume()</a>	173

<a href="#">wavEdit::viewImage()</a>	173
<a href="#">wavEdit::writeWavFile()</a>	174
<a href="#">wavEdit::viewHeader()</a>	173
<a href="#">wavEdit::unpackHeader()</a>	173
<a href="#">wavEdit::unpackData()</a>	173
<a href="#">wavEdit::getFileName()</a>	172
<a href="#">wavEdit::getDataArr()</a>	172
<a href="#">wavEdit::doDrawImg()</a>	171
<a href="#">wavEdit::doEcho()</a>	171
<a href="#">wavEdit::doCrop()</a>	170
<a href="#">wavEdit::doClip()</a>	170
<a href="#">wavEdit::cropFiles()</a>	170
<a href="#">wavEdit::doFade()</a>	171
<a href="#">wavEdit::doJoin()</a>	171
<a href="#">wavEdit::drawImgWave()</a>	172
<a href="#">wavEdit::getBytes()</a>	172
<a href="#">wavEdit::doViewHeader()</a>	171
<a href="#">wavEdit::doSetVol16Bit()</a>	171
<a href="#">wavEdit::doSetVol8Bit()</a>	171
<a href="#">wmString::jsSpecialChars()</a>	33
<i>Escapes special characters for use in javascript function calls.</i>	
<a href="#">wmString::hanukkahStart()</a>	32
<i>Get the unix timestamp of Hanukkah for this year.</i>	
<a href="#">wmSort::multiASort()</a>	28
<i>Sorts a two-dimensional array by one to many key "indexes" and maintains its associative indexes.</i>	
<a href="#">wmSort::multiSort()</a>	28
<i>Sorts a two-dimensional array by one to many key "indexes".</i>	
<a href="#">wmSort</a>	27
<i>The wmSort class is used for advanced two-dimensional array sorting.</i>	
<a href="#">wmFileSystem::scanDir()</a>	27
<i>Reads directory contents recursively and returns them as an associative array.</i>	
<a href="#">wmFileSystem::mimeType()</a>	26
<i>Get the mime type from mime_magic</i>	
<a href="#">wmString</a>	29
<i>The wmString class is used to manipulate strings in various ways.</i>	
<a href="#">wmString::TIME_LEN_FORMAT_LONG</a>	29
<a href="#">wmString::TIME_LEN_LABEL_HRS</a>	29
<a href="#">wmString::TIME_LEN_LABEL_LAST</a>	29
<a href="#">wmString::TIME_LEN_LABEL_DAY</a>	29
<a href="#">wmString::TIME_LEN_LABEL_AGO</a>	29
<a href="#">wmString::TIME_LEN_FORMAT_SHORT</a>	29
<a href="#">wmFileSystem::dirSize()</a>	26
<i>Get the file size of a directory and its subcontents in bytes</i>	
<a href="#">wmFileSystem::bytesToHuman()</a>	26
<i>Makes byte counts human readable (i.e. 1024 -&gt; 1KB).</i>	
<a href="#">wmArrays</a>	22
<i>The wmArrays class is used to perform complex routines on arrays.</i>	
<a href="#">wmArrays::arrayMergeRecursiveSumFields()</a>	23
<i>Merges multidimensional arrays, and sums the integer values of fields in given list.</i>	
<a href="#">wmString.php</a>	9

<a href="#">wmString.php</a> :: Will Mason's string manipulation class	8
<a href="#">wmSort.php</a> :: Will Mason's array sorting class	7
<a href="#">wmFileSystem.php</a> :: Will Mason's file system class	23
<a href="#">wmArrays::arrayRemoveValue()</a>	23
<a href="#">wmArrays::arraySearchRecursive()</a>	25
<i>Searches through a multidimensional array for a given value and if found, returns an array of keys to its location.</i>	
<a href="#">wmArrays::pregExplode()</a>	25
<a href="#">wmFileSystem</a>	25
<i>The wmFileSystem class is collection of specialized file system utility methods.</i>	
<a href="#">wmArrays::implodeObjArray()</a>	24
<a href="#">wmArrays::explodeNewline()</a>	24
<a href="#">wmArrays::arrayValuesRecursive()</a>	29
<a href="#">wmString::TIME_LEN_LABEL_MIN</a>	29
<a href="#">wmString::TIME_LEN_LABEL_MON</a>	31
<a href="#">wmString::TIME_LEN_TYPE_SEC</a>	31
<a href="#">wmString::TIME_LEN_TYPE_WEEK</a>	30
<a href="#">wmString::TIME_LEN_TYPE_MON</a>	30
<a href="#">wmString::TIME_LEN_TYPE_MIN</a>	30
<a href="#">wmString::TIME_LEN_TYPE_HRS</a>	31
<a href="#">wmString::TIME_LEN_TYPE_YEAR</a>	31
<a href="#">wmString::VALID_DATE</a>	32
<a href="#">wmString::emailEncode()</a>	32
<i>Takes a string (mailto:email@domain.com) and converts it to unicode to protect it from spiders.</i>	
<a href="#">wmString::emailValid()</a>	31
<i>Test if an email address is valid.</i>	
<a href="#">wmString::dayOfWeek()</a>	31
<i>Give the day of week given day number 0-6.</i>	
<a href="#">wmString::dateValid()</a>	31
<i>Test if a date is valid.</i>	
<a href="#">wmString::VALID_EMAIL</a>	30
<a href="#">wmString::TIME_LEN_TYPE_DAY</a>	30
<a href="#">wmString::TIME_LEN_TIME_YEAR</a>	30
<a href="#">wmString::TIME_LEN_LABEL_YEAR</a>	30
<a href="#">wmString::TIME_LEN_LABEL_YESTERDAY</a>	30
<a href="#">wmString::TIME_LEN_LABEL_WEEK</a>	29
<a href="#">wmString::TIME_LEN_LABEL_TODAY</a>	29
<a href="#">wmString::TIME_LEN_LABEL_SEC</a>	30
<a href="#">wmString::TIME_LEN_TIME_DAY</a>	30
<a href="#">wmString::TIME_LEN_TIME_HRS</a>	30
<a href="#">wmString::TIME_LEN_TIME_WEEK</a>	30
<a href="#">wmString::TIME_LEN_TIME_SEC</a>	30
<a href="#">wmString::TIME_LEN_TIME_MON</a>	30
<a href="#">wmString::TIME_LEN_TIME_MIN</a>	30
<a href="#">wmArrays.php</a> :: Will Mason's array processing class	6

<a href="#">__autoload()</a>	83
<i>Autoload classes using dbdLoader::loadClass();</i>	