

# Cheat Sheet

LNXMCP CHEAT SHEET (v3.5.0)							
Site Url	https://github.com/linhunix/lnxmcp	Doc Url	http://lnxmcp.uk/	Phar Url	https://github.com/linhunix/lnxmcp/blob/3.5.0/dist/lnxmcp.phar		
Cfg			Sys Command	mcpBaseModelClass			
file  (app.php) resource	cfg/mcp.settings.json	lnxmcp() →	controller(\$proc,\$sproc,\$scope,\$mod,\$subc,\$ven)	\$this →	getRes(\$resname)	getArgCtl(\$name)	
	/app.php		api(\$srv, \$sproc, \$scope, \$mod, \$subc, \$ven)		getCfg(\$cfigname)	getArgOut(\$name)	
	\$lnxmcp_phar		service(\$srv,\$sproc,\$scope,\$mod,\$subc,\$ven)		getDriver(\$drvlabel)	setArgOut(\$name, \$value)	
	app.def		page(\$page,\$scope,\$mod,\$ven,\$pathtpl,\$hasreturn)		getCommon(\$name)	setReturn(\$return)	
	\$app_path		block(\$page,\$scope,\$mod,\$ven,\$pathtpl,\$hasreturn)		callCmd(array \$scopeCtl, array \$scopeIn)		
	app.path		render(\$page,\$scope,\$mod,\$ven,\$pathtpl,\$hasreturn)		callTag(\$action,\$scopeIn, \$buffer)		
	app.lang		driver(\$name,\$sproc,\$scope,\$mod,\$subc,\$ven,\$path)		debug(\$message)	info(\$message)	
	app.menu.InitCommon		shell(\$scriproc,\$scope,\$mod,\$subc,\$ven)		warning(\$message)	error(\$message)	
app.debug and app.level		remote(\$proc,\$scopeIn,\$mod,\$subcall,\$ven)	pdoDriver				
Sys Cfg Settings			Sys Debug	ScopeIn			
lnxmcp() →	getCfg(\$name)	lnxmcp() →	debug(\$message)	Type	"E" as Driver envs	"Q" as Query	
	setCfg(\$name,\$value)		info(\$message)		"V" value	"T" type	
	getResource(\$name)		warning(\$message)		"e" execute	"q" simpleQuery	
	getCommon(\$name)		error(\$message)		"f" firstRow	"c" simpleCount	
	setCommon(\$name,\$value)		rem(\$msg [\$msg2])		"er" executeWithRollback	's' return sql	
			dump( \$msg )		Inxmcp() → runCommand() / InxmcpCmd		
					cmd list	extTemplate	extFile
						page	showPage
		showFullCommonBlock					
		blockShell	blockRemote				
		block	blockCommon				
		showBlock	showCommonBlock				
		render	renderCommon				
		service	serviceCommon				
Sys Tools			Database				
lnxmcp() → (legacy move)	header(\$string,\$end,\$retcode,	lnxmcp() →	resource	("Driver.[db label])			
	move(\$file,\$filedef,\$ext,\$path,\$end)		queryJsonR(\$name,\$scope,\$module)				
	Rem(\$var,[\$var])		query(\$db,\$sprocload,\$scope,\$mod,\$subc,\$ven)				
	supportmail(\$message)		queryR(\$db,\$sprocload,\$scope,\$mod,\$subc,\$ven)				
	mail(\$page,\$scope,module,\$vendor)		queryCommonR(\$db,\$sprocload,\$scope,\$mod,\$subc)				
	escapeClear(\$string)		queryArrayR(\$scopeIn)				
	ConvertToAscii(\$string)						
	lnxmcpExit((\$message))						
function			exTemplate tags				
			[scope-<vars>]	[scope-dump]			
			[common-<vars>]	[server-<vars>]			
			[lnxmcp-<tags>]	[<label>-<tags>]			
			name	module			
			vendor	type			
			disable-rem	Block-type			
			config	json			
Sys Admin							
resource	mcp.web.api	<lnxmcp>					
	mcp.web.admin		</lnxmcp>				
function	lnxmcpadm(\$cmd)	(block-type)					
	lnxmcp-adm \$cmd						
shell web	/lnxmcpadm		common	scope			
	/lnxmcpapi		translate	javascript			
web function	home , form ,mail		print_r				
shell function	http, checksyntax,						
Example Controller:				Example Service:			
<?php /** ** <DESCRIPTION> **/ namespace App\<Module>\Controller; use LinHUniX\Mcp\Model\mcpBaseModelClass; class <Name>Controller extends mcpBaseModelClass { /** * Ideally this method should be used to first execution */ protected function moduleInit(){ \$this->spacename= __NAMESPACE__; \$this->classname= __CLASS__; } /** * Ideally this method should be used to insert * the model code and the other are to be used only as normal */ protected function moduleCore() { \$mydataarray=\$this->argIn; /// is empty wait to be implemented \$this->argOut=\$arrayout; } /** * Ideally this method should be used to insert * the model code and the other are to be used only as normal */ protected function moduleSingleTon() { /// is empty wait to be implemented } }				<?php /** ** <DESCRIPTION> **/ namespace App\<Module>\Service; use LinHUniX\Mcp\Model\mcpServiceModelClass; Class <Name> extends mcpServiceModelClass { /** * Ideally this method should be used to first execution */ Protected function moduleInit(){ \$this->spacename= __NAMESPACE__; \$this->classname= __CLASS__; } /** * standard 1 shot user */ protected function moduleSingleTon() {} /** * function <T>_<e>(){ * [T]=xxx * [E]=yyy */ public function xxx_yyy(){ } }			
Json Query:							
{ "E": "db label", "T": "query type", "Q": "query sql [field] [R:raw field] [S:stripslashed field] " }							
Change Path module							
lnxmcp() → setCfg('app.mod.path.<vendro>.<module>','fullpaht');							