

localhost Ltd



ALPHA CMS

::: A.P.I Reference ::: (Version 10.0)

localhost Ltd - ALPHA CMS is an A.P.I – free, MVC based Content Management System. ALPHA CMS architecture gives the ability to easily create advanced web pages, add-ons or even a new A.P.I and a new CMS. ALPHA CMS is based on ALPHA Framework which is developed on PHP and MySQL. ALPHA CMS also uses Javascript and AJAX.

A.P.I – free does not mean that ALPHA CMS does not have an A.P.I! Instead, ALPHA CMS offers a very useful, concrete and easy A.P.I, based and extended on the ALPHA Framework, which also includes PHP, Javascript and AJAX extensions.

ALPHA CMS is open in such a way that lets programmers to extend it or even to develop their A.P.I freely on top of it while ALPHA CMS continues to function properly, unlike other CMS such as Joomla! and Drupal.

ALPHA CMS automatically loads the ALPHA Framework and uses the extended static class “ALPHA_CMS” for all the PHP files that load from the “main.php” file. For any custom usage or usage that is separate from the “main.php” file you just need to include both the “alpha.php” and the “alpha_cms.php” files.

localhost Ltd – ALPHA CMS A.P.I (PHP)

Function: *Setup_Wizard*
Arguments: *0*
Returns: *false* on error or the *setup wizard* on success
Description: *This function is useful only at first run of ALPHA CMS on a new system. It sets up the ALPHA CMS to the server with the help of a wizard.*

Function: *Get_Default_Language*
Arguments: *0*
Returns: *false* on error or the *default language code* on success
Description: *This function returns the default language code as this was selected in the administration panel of ALPHA CMS. The predefined language code is always 'en'.*

Function: *Execute_SQL_Command*
Arguments: *sql_com* [any valid SQL command as a string]
log [null, 1 or 'on' (default: null)]
Returns: *false* on error or an *array with results* on success
Description: *This function executes any valid SQL commands and returns a multi-dimensional array that has at least two dimensions. The first dimension is always an index of the SQL rows that were read and the other dimensions may contain a result or other arrays that contain data or other arrays and so on. This function also logs any errors to the ALPHA CMS Logs table if an error occurs. The logging facility is deactivated by default.*

Function: *Execute_SQL_Script_File*
Arguments: *sql_file* [any valid SQL script]
log [null, 1 or 'on' (default: null)]
Returns: *false* on error or an *array with results* on success
Description: *This function executes any valid SQL script and returns a multi-dimensional array that has at least two dimensions. The first dimension is always an index of the SQL rows that were read and the other dimensions may contain a result or other arrays that contain data or other arrays and so on. This function also logs any errors to the ALPHA CMS Logs table if an error occurs. The logging facility is deactivated by default.*

Function: *Create_Menu*
Arguments: *caller* [* or any valid string]
parent_menu_id [any valid id as integer]
lang_code [*, en, gr, ...]
order_field [null or any valid field (default: null)]
sort_type [null, ASC or DESC (default: null)]
Returns: *false* on error or an *array* on success
Description: *This function creates a menu or part of a menu, designed in the alpha_menu table, as an associative array map.*

Function: *Get_Menu*
Arguments: *0*
Returns: *false* on error or a *menu array* on success
Description: *This function returns a previously created menu as an array.*

Function: *Load_Language*
Arguments: *lang_code* [*, en, gr, ...]
Returns: *false* on error or *language(s)* on success
Description: *This function loads all languages as an array with '*' or the specified language based on language code.*

Function: *Load_Common*
Arguments: *option* [*id, site_title, site_description, site_keywords, company_name, company_site, footer_info, lang_id, is_protected*]
binded_route [the corresponding binded route]
lang_code [en, gr, ...]
Returns: *false* on error or a *common option* on success
Description: *This function loads a selected common option for the corresponding binded route and language.*

Function: *Load_Menu*
Arguments: *caller_id* [* or any valid string]
lang_code [en, gr, ...]
order_field [null or any valid field (default: null)]
sort_type [null, ASC or DESC (default: null)]
Returns: *false* on error or *menu elements* on success
Description: *This function loads a menu related to a caller,*

*which might be a number or a link in the site,
or all the menus with '*' as an array
for the selected language or for every language with '*'.*

Function:	<i>Load_Content</i>
Arguments:	<i>page</i> [any content page] <i>option</i> [id, page, content, keywords, lang_id, is_protected, is_route] <i>lang_code</i> [en, gr, ...]
Returns:	<i>false</i> on error or <i>page content</i> on success
Description:	<i>This function loads a specified content related to a page for the selected language.</i>
Function:	<i>Load_Users</i>
Arguments:	<i>option</i> [* , id, username, email, password, type]
Returns:	<i>false</i> on error or <i>all fields or selected fields of users</i> on success
Description:	<i>This function loads all or a specified field from users table for all the users. The users are most commonly useful for the administration panel but you can make use of them in the front-end.</i>
Function:	<i>Insert_Language</i>
Arguments:	<i>lang_code</i> [en, gr, ...] <i>language</i> [any valid language in text] <i>sort_order</i> [1, 2, ...] <i>is_default</i> [null, 0 or 1 (default: null)] <i>is_protected</i> [null, 0 or 1 (default: null)]
Returns:	<i>false</i> on error or <i>true</i> on success
Description:	<i>This function inserts a new language.</i>
Function:	<i>Insert_Common</i>
Arguments:	<i>site_title</i> [the site title] <i>site_description</i> [the site description] <i>site_keywords</i> [any comma separated keywords] <i>company_name</i> [the company name] <i>company_site</i> [the company site] <i>footer_info</i> [the site footer info] <i>binded_route</i> [the corresponding binded route]

Function: *lang_id [1, 2, ...]*
Arguments: *is_protected [null, 0 or 1 (default: null)]*
Returns: *false* on error or *true* on success
Description: *This function inserts new common options.*

Function: *Insert_Menu*
Arguments: *caller [any valid string]*
parent_menu_id [0, 1, 2, ...]
menu_name [any valid string]
menu_link [any valid link or null]
lang_id [1, 2, ...]
sort_order [1, 2, ...]
is_protected [null, 0 or 1 (default: null)]
Returns: *false* on error or *true* on success
Description: *This function inserts a new menu item.*

Function: *Insert_Content*
Arguments: *page [any content page]*
content [any HTML, PHP or JS content]
keywords [any comma separated keywords]
lang_id [1, 2, ...]
is_protected [null, 0 or 1 (default: null)]
is_route [null, 0 or 1 (default: null)]
Returns: *false* on error or *true* on success
Description: *This function inserts a new content.*

Function: *Insert_User*
Arguments: *username [any valid string]*
email [any valid email as a string]
password [any valid string]
type [0, 1, 2]
Returns: *false* on error or *true* on success
Description: *This function inserts a new user of a specific type for the administration panel.*
Administrator: 0 (Full access)
Editor: 1 (Only editing)
Auditor: 2 (Only auditing)

Function: *Update_Language*
Arguments: *id* [any valid record id]
lang_code [en, gr, ...]
language [any valid string]
sort_order [1, 2, ...]
is_default [null, 0 or 1 (default: null)]
is_protected [null, 0 or 1 (default: null)]
Returns: *false* on error or *true* on success
Description: *This function updates languages.*

Function: *Update_Common*
Arguments: *id* [any valid record id]
site_title [the site title]
site_description [the site description]
site_keywords [any comma separated keywords]
company_name [the company name]
company_site [the company site]
footer_info [the site footer info]
binded_route [the corresponding binded route]
lang_id [1, 2, ...]
is_protected [null, 0 or 1 (default: null)]
Returns: *false* on error or *true* on success
Description: *This function updates common options.*

Function: *Update_Menu*
Arguments: *id* [any valid record id]
caller [any valid string]
parent_menu_id [0, 1, 2, ...]
menu_name [any valid string]
menu_link [any valid link or null]
lang_id [1, 2, ...]
sort_order [1, 2, ...]
is_protected [null, 0 or 1 (default: null)]
Returns: *false* on error or *true* on success
Description: *This function updates menu items.*

Function: *Update_Content*
Arguments: *id* [any valid record id]
page [any content page as a string]
content [any HTML, PHP or JS content]

keywords [any comma separated keywords]
lang_id [1, 2, ...]
is_protected [null, 0 or 1 (default: null)]
is_route [null, 0 or 1 (default: null)]

Returns: *false* on error or *true* on success

Description: *This function updates contents.*

Function: *Update_User*

Arguments: *id [any valid record id]*
username [any valid string]
email [any valid email as a string]
password [any valid string]
type [0, 1, 2]

Returns: *false* on error or *true* on success

Description: *This function updates a specific user.*

Function: *Update_Extension*

Arguments: *ext_name [any valid string]*
ext_type [any valid extension type as a string]
ext_keys [any valid array]

Returns: *false* on error or *true* on success

Description: *This function updates a specific extension.*

Function: *Delete_Language*

Arguments: *id [any valid record id]*

Returns: *false* on error or *true* on success

Description: *This function deletes languages.*

Function: *Delete_Common*

Arguments: *id [any valid record id]*

Returns: *false* on error or *true* on success

Description: *This function deletes common options.*

Function: *Delete_Menu*

Arguments: *id [any valid record id]*

Returns: *false* on error or *true* on success

Description: *This function deletes menu items.*
Function: *Delete_Content*
Arguments: *id [any valid record id]*
Returns: *false on error or true on success*
Description: *This function deletes contents.*

Function: *Delete_User*
Arguments: *id [any valid record id]*
Returns: *false on error or true on success*
Description: *This function deletes a specific user.
Notice that the 'admin' can't be deleted!*

Function: *Delete_Extension*
Arguments: *ext_name [any valid string]*
ext_type [any valid extension type as a string]
Returns: *false on error or true on success*
Description: *This function deletes a specific extension.*

Function: *Log_Error*
Arguments: *log_type [php , mysql]*
Returns: *false on error or true on success*
Description: *This function logs PHP and MySQL errors in the Logs table of ALPHA CMS database. Notice that in case of a MySQL error the 'Line' field is being set to -1.*

Function: *Show_Log*
Arguments: *id [* or any valid record id]*
option [, id, entity, error_code, message, file, line, date_time]*
order_field [null or any valid field (default: null)]
sort_type [null, ASC or DESC (default: null)]
Returns: *false on error or logs as an array on success*
Description: *This function shows any stored logs from ALPHA CMS database.
Notice that if '*' is used for id and option in conjunction with 'null' for order and sort, then the result will be a typical selection with an ascending sort order.*

Function: *Delete_Log*
Arguments: *id* [*** or any valid record id]
from [null or any valid date (default: null)]
to [null or any valid date (default: null)]
Returns: *false* on error or *true* on success
Description: *This function deletes all or any specified stored logs from ALPHA CMS database. You may also set a range between dates in order to delete specific records.*

Function: *Go_To_Route*
Arguments: *route* [any valid MVC route]
mvc_args [null, any valid arguments as a string or an array of arguments (default: null)]
Returns: *false* on error or *true* on success
Description: *This function redirects you to an existing physical MVC route passing any arguments.*

Function: *Get_All_Routes*
Arguments: *lang_code* [***, *en*, *gr*, ...]
Returns: *false* on error or *true* on success
Description: *This function returns all the physical MVC routes when you put '*' as an argument or all the physical MVC routes per language if you provide a language code.*

localhost Ltd – ALPHA CMS A.P.I (Javascript - Standard Extensions)

Extension: *languages_manager*
Function: *Init_Languages*
Arguments: *0*
Returns: *false* on error or *true* on success
Description: *This function is useful only at the main HTML body of ALPHA CMS. It initializes all the predefined languages and starts the language management system.*

Extension: *languages_manager*
Function: *Change_Language*
Arguments: *lang_code [en, gr, ...]*
Returns: *false* on error or *true* on success
Description: *This function changes the current language.*

Extension: *languages_manager*
Function: *Setup_Language*
Arguments: *index [0, 1, 2, ...]*
lang_code [en, gr, ...]
Returns: *false* on error or *true* on success
Description: *This function sets up selected language codes.*

Extension: *languages_manager*
Function: *Set_Default_Lang*
Arguments: *lang_code [en, gr, ...]*
Returns: *false* on error or *true* on success
Description: *This function sets the default language.*

Extension: *languages_manager*
Function: *Check_Language*
Arguments: *lang_code [en, gr, ...]*
state [true, false]
Returns: *false* on error or *true* on success
Description: *This function checks or unchecks a language.*
Extension: *languages_manager*
Function: *Setup_Lang_Pix*
Arguments: *index [0, 1, 2, ...]*

id [any HTML element id]

Returns: *false* on error or *true* on success

Description: *This function sets up a language flag picture for the given HTML element.*

Extension: *languages_manager*

Function: *Setup_Lang_Label*

Arguments: *index [0, 1, 2, ...]*
label_name [any valid string]

Returns: *false* on error or *true* on success

Description: *This function sets up a language label.*

Extension: *languages_manager*

Function: *Set_Lang_Text_Focus*

Arguments: *element_id [any HTML element id]*
lang_code [en, gr, ...]
focus_color [any valid CSS color or HEX color]

Returns: *false* on error or *true* on success

Description: *This function focuses the selected language label by coloring it with the specified color.*

Extension: *languages_manager*

Function: *Load_Lang_Pix*

Arguments: *id [any HTML element id]*
lang_code [en, gr, ...]

Returns: *false* on error or *true* on success

Description: *This function loads a language flag picture for the given HTML element.*

Extension: *languages_manager*

Function: *Lang_Pix_Over*

Arguments: *id [any HTML element id]*
lang_code [en, gr, ...]

Returns: *false* on error or *true* on success

Description: *This function changes the mouse over flag picture for the given language.*

Extension: *languages_manager*
Function: *Lang_Pix_Out*
Arguments: *id* [any HTML element id]
lang_code [en, gr, ...]
Returns: *false* on error or *true* on success
Description: *This function changes the mouse out flag picture for the given language.*

Extension: *key_manager*
Function: *Scan_Keys*
Arguments: *key* [any keyboard key]
Returns: *false* on error or *true* on success
Description: *This function reads any keyboard stroke and stores its key code.*

Extension: *key_manager*
Function: *Get_Keys*
Arguments: *0*
Returns: *false* on error or a *key code* on success
Description: *This function returns a key code.*

localhost Ltd – ALPHA CMS A.P.I (AJAX - Standard Extensions)

Extension: *bull*
Function: *AJAX_Init*
Arguments: *0*
Returns: *alert* on error or an *AJAX object* on success
Description: *This function initializes the AJAX subsystem and creates an AJAX object. It should only be used once before any AJAX request. The ALPHA Framework initializes 8 AJAX objects automatically. The first 5 objects are used by the AJAX_Transfer_Data function and the rest 3 by AJAX_Transfer_Response function. Normally you would not need any more AJAX objects but if you do then you have to use AJAX_Init.*

Extension: *bull*
Function: *AJAX_Transfer_Data*
Arguments: *url [any valid URL]*
data [any valid key pairs (format: name1=value1&name2=value2...)]
element_id [any HTML element id]
ajax_mode [1, 2]
ajax_session [1, 2, 3, 4, 5]
Returns: *false* on error or *true* on success
Description: *This function transfers data with AJAX inside an HTTP request and returns the result in the specified HTML element. The requests are based on POST. Asynchronous mode: 1 Synchronous mode: 2 AJAX session number: 1 to 5*

Extension: *bull*
Function: *AJAX_Transfer_Response*
Arguments: *url [any valid URL]*
data [any valid key pairs (format: name1=value1&name2=value2...)]
ajax_session [1, 2, 3]
Returns: *false* on error or *data* on success
Description: *This function transfers responses with AJAX inside an HTTP request and returns the result. The responses are always fetched in synchronous mode. The requests are based on POST. AJAX session number: 1 to 3*

Extension: *menu_manager*
Function: *AJAX_Bind_Menu*
Arguments: *element_id* [any HTML element id]
caller [any valid string]
lang_code [en, gr, ...]
ajax_mode [1, 2]
ajax_session [1, 2, 3, 4, 5]
Returns: *false* on error or *true* on success
Description: *This function binds a menu on a HTML element specified by element_id and relates it to a caller specified by caller. The requests are based on POST. Asynchronous mode: 1 Synchronous mode: 2 AJAX session number: 1 to 5*

Extension: *menu_manager*
Function: *AJAX_Load_Menu_Content*
Arguments: *element_id* [any HTML element id]
menu_link [corresponding menu link]
lang_code [en, gr, ...]
ajax_mode [1, 2]
ajax_session [1, 2, 3, 4, 5]
Returns: *false* on error or *true* on success
Description: *This function loads content on a HTML element specified by element_id and related to the menu specified by menu_link. The requests are based on POST. Asynchronous mode: 1 Synchronous mode: 2 AJAX session number: 1 to 5*

Extension: *content_manager*
Function: *AJAX_Load_Content*
Arguments: *element_id* [any HTML element id]
content_id [any valid element id]
lang_code [en, gr, ...]
ajax_mode [1, 2]
ajax_session [1, 2, 3, 4, 5]
Returns: *false* on error or *true* on success
Description: *This function loads content on a HTML element specified by element_id that is related to a content specified by the content_id. The requests are based on POST.*

Asynchronous mode: 1
Synchronous mode: 2
AJAX session number: 1 to 5

Extension: *content_fetcher*
Function: *AJAX_Fetch_Content*
Arguments: *element_id* [any HTML element id]
content_url [any valid URL]
ajax_mode [1, 2]
ajax_session [1, 2, 3, 4, 5]
Returns: *false* on error or *true* on success
Description: *This function fetches data from a URL specified by content_url to a HTML element specified by element_id.*
The requests are based on POST.
Asynchronous mode: 1
Synchronous mode: 2
AJAX session number: 1 to 5