

```
<?php
//Default Configuration
$CONFIG =
{"lang":"en","error_reporting":false,"show_hidden":false,"hide_Cols":false,"theme":"light"};

/**
 * H3K ~ Tiny File Manager V2.6
 * @author CCP Programmers
 * @github https://github.com/prasathmani/tinyfilemanager
 * @link https://tinyfilemanager.github.io
 */

//TFM version
define('VERSION', '2.6');

//Application Title
define('APP_TITLE', 'Tiny File Manager');

// --- EDIT BELOW CONFIGURATION CAREFULLY ---

// Auth with login/password
// set true/false to enable/disable it
// Is independent from IP white- and blacklisting
$use_auth = true;

// Login user name and password
// Users: array('Username' => 'Password', 'Username2' => 'Password2', ...)
// Generate secure password hash - https://tinyfilemanager.github.io/docs/pwd.html
$auth_users = array(
    'admin' => '$2y$10$/K.hjNr84lLNDt8fTXjol.DBp6PpeyoJ.mGwrrLuCZfAwfSAGqhOW',
    //admin@123
    'user' => '$2y$10$Fg6Dz8oH9fPoZ2jJan5tZuv6Z4Kp7avtQ9bDfrdRntXtPeiMAZyGO' //12345
);

// Readonly users
// e.g. array('users', 'guest', ...)
$readonly_users = array(
    'user'
```

```
);
```

```
// Global readonly, including when auth is not being used
```

```
$global_readonly = false;
```

```
// user specific directories
```

```
// array('Username' => 'Directory path', 'Username2' => 'Directory path', ...)
```

```
$directories_users = array();
```

```
// Enable highlight.js (https://highlightjs.org/) on view's page
```

```
$use_highlightjs = true;
```

```
// highlight.js style
```

```
// for dark theme use 'ir-black'
```

```
$highlightjs_style = 'vs';
```

```
// Enable ace.js (https://ace.c9.io/) on view's page
```

```
$edit_files = true;
```

```
// Default timezone for date() and time()
```

```
// Doc - http://php.net/manual/en/timezones.php
```

```
$default_timezone = 'Etc/UTC'; // UTC
```

```
// Root path for file manager
```

```
// use absolute path of directory i.e: '/var/www/folder' or $_SERVER['DOCUMENT_ROOT'].'/folder'
```

```
//make sure update $root_url in next section
```

```
$root_path = $_SERVER['DOCUMENT_ROOT'];
```

```
// Root url for links in file manager.Relative to $http_host. Variants: "", 'path/to/subfolder'
```

```
// Will not working if $root_path will be outside of server document root
```

```
$root_url = "";
```

```
// Server hostname. Can set manually if wrong
```

```
// $_SERVER['HTTP_HOST'].'/folder'
```

```
$http_host = $_SERVER['HTTP_HOST'];
```

```
// input encoding for iconv
```

```
$iconv_input_encoding = 'UTF-8';
```

```
// date() format for file modification date
// Doc - https://www.php.net/manual/en/function.date.php
$datetime_format = 'm/d/Y g:i A';

// Path display mode when viewing file information
// 'full' => show full path
// 'relative' => show path relative to root_path
// 'host' => show path on the host
$path_display_mode = 'full';

// Allowed file extensions for create and rename files
// e.g. 'txt,html,css,js'
$allowed_file_extensions = "";

// Allowed file extensions for upload files
// e.g. 'gif,png,jpg,html,txt'
$allowed_upload_extensions = "";

// Favicon path. This can be either a full url to an .PNG image, or a path based on the document
root.
// full path, e.g http://example.com/favicon.png
// local path, e.g images/icons/favicon.png
$favicon_path = "";

// Files and folders to excluded from listing
// e.g. array('myfile.html', 'personal-folder', '*.php', '/path/to/folder', ...)
$exclude_items = array();

// Online office Docs Viewer
// Available rules are 'google', 'microsoft' or false
// Google => View documents using Google Docs Viewer
// Microsoft => View documents using Microsoft Web Apps Viewer
// false => disable online doc viewer
$online_viewer = 'google';

// Sticky Nav bar
// true => enable sticky header
```

```
// false => disable sticky header
$sticky_navbar = true;

// Maximum file upload size
// Increase the following values in php.ini to work properly
// memory_limit, upload_max_filesize, post_max_size
$max_upload_size_bytes = 5000000000; // size 5,000,000,000 bytes (~5GB)

// chunk size used for upload
// eg. decrease to 1MB if nginx reports problem 413 entity too large
$upload_chunk_size_bytes = 2000000; // chunk size 2,000,000 bytes (~2MB)

// Possible rules are 'OFF', 'AND' or 'OR'
// OFF => Don't check connection IP, defaults to OFF
// AND => Connection must be on the whitelist, and not on the blacklist
// OR => Connection must be on the whitelist, or not on the blacklist
$ip_ruleset = 'OFF';

// Should users be notified of their block?
$ip_silent = true;

// IP-addresses, both ipv4 and ipv6
$ip_whitelist = array(
    '127.0.0.1', // local ipv4
    '::1'        // local ipv6
);

// IP-addresses, both ipv4 and ipv6
$ip_blacklist = array(
    '0.0.0.0', // non-routable meta ipv4
    '::'       // non-routable meta ipv6
);

// if User has the external config file, try to use it to override the default config above [config.php]
// sample config - https://tinyfilemanager.github.io/config-sample.txt
$config_file = __DIR__ . '/config.php';
if (is_readable($config_file)) {
    @include($config_file);
}
```

```
}
```

```
// External CDN resources that can be used in the HTML (replace for GDPR compliance)
```

```
$external = array(
    'css-bootstrap' => '<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhY6hW+ALE
wIH" crossorigin="anonymous">',
    'css-dropzone' => '<link
href="https://cdnjs.cloudflare.com/ajax/libs/dropzone/5.9.3/min/dropzone.min.css" rel="stylesheet">',
    'css-font-awesome' => '<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"
crossorigin="anonymous">',
    'css-highlightjs' => '<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/highlight.js/11.9.0/styles/' . $highlightjs_style .
'.min.css">',
    'js-ace' => '<script src="https://cdnjs.cloudflare.com/ajax/libs/ace/1.32.2/ace.js"></script>',
    'js-bootstrap' => '<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"
integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcldslK1eN7N6jleHz"
crossorigin="anonymous"></script>',
    'js-dropzone' => '<script
src="https://cdnjs.cloudflare.com/ajax/libs/dropzone/5.9.3/min/dropzone.min.js"></script>',
    'js-jquery' => '<script src="https://code.jquery.com/jquery-3.6.1.min.js"
integrity="sha256-o88AwQnZB+VDvE9tvIXrMQaPiffSUTR+nldQm1LuPXQ="
crossorigin="anonymous"></script>',
    'js-jquery-datatables' => '<script src="https://cdn.datatables.net/1.13.1/js/jquery.dataTables.min.js"
crossorigin="anonymous" defer></script>',
    'js-highlightjs' => '<script
src="https://cdnjs.cloudflare.com/ajax/libs/highlight.js/11.9.0/highlight.min.js"></script>',
    'pre-jsdelivr' => '<link rel="preconnect" href="https://cdn.jsdelivr.net" crossorigin/><link
rel="dns-prefetch" href="https://cdn.jsdelivr.net"/>',
    'pre-cloudflare' => '<link rel="preconnect" href="https://cdnjs.cloudflare.com" crossorigin/><link
rel="dns-prefetch" href="https://cdnjs.cloudflare.com"/>'
);
```

```
// --- EDIT BELOW CAREFULLY OR DO NOT EDIT AT ALL ---
```

```
// max upload file size
define('MAX_UPLOAD_SIZE', $max_upload_size_bytes);

// upload chunk size
define('UPLOAD_CHUNK_SIZE', $upload_chunk_size_bytes);

// private key and session name to store to the session
if (!defined('FM_SESSION_ID')) {
    define('FM_SESSION_ID', 'filemanager');
}

// Configuration
$cfg = new FM_Config();

// Default language
$lang = isset($cfg->data['lang']) ? $cfg->data['lang'] : 'en';

// Show or hide files and folders that starts with a dot
$show_hidden_files = isset($cfg->data['show_hidden']) ? $cfg->data['show_hidden'] : true;

// PHP error reporting - false = Turns off Errors, true = Turns on Errors
$report_errors = isset($cfg->data['error_reporting']) ? $cfg->data['error_reporting'] : true;

// Hide Permissions and Owner cols in file-listing
$hide_Cols = isset($cfg->data['hide_Cols']) ? $cfg->data['hide_Cols'] : true;

// Theme
$theme = isset($cfg->data['theme']) ? $cfg->data['theme'] : 'light';

define('FM_THEME', $theme);

//available languages
$lang_list = array(
    'en' => 'English'
);

if ($report_errors == true) {
    @ini_set('error_reporting', E_ALL);
```

```

    @ini_set('display_errors', 1);
} else {
    @ini_set('error_reporting', E_ALL);
    @ini_set('display_errors', 0);
}

// if fm included
if (defined('FM_EMBED')) {
    $use_auth = false;
    $sticky_navbar = false;
} else {
    @set_time_limit(600);

    date_default_timezone_set($default_timezone);

    ini_set('default_charset', 'UTF-8');
    if (version_compare(PHP_VERSION, '5.6.0', '<') && function_exists('mb_internal_encoding')) {
        mb_internal_encoding('UTF-8');
    }
    if (function_exists('mb_regex_encoding')) {
        mb_regex_encoding('UTF-8');
    }

    session_cache_limiter('nocache'); // Prevent logout issue after page was cached
    session_name(FM_SESSION_ID);
    function session_error_handling_function($code, $msg, $file, $line)
    {
        // Permission denied for default session, try to create a new one
        if ($code == 2) {
            session_abort();
            session_id(session_create_id());
            @session_start();
        }
    }
    set_error_handler('session_error_handling_function');
    session_start();
    restore_error_handler();
}

```

```

//Generating CSRF Token
if (empty($_SESSION['token'])) {
    if (function_exists('random_bytes')) {
        $_SESSION['token'] = bin2hex(random_bytes(32));
    } else {
        $_SESSION['token'] = bin2hex(openssl_random_pseudo_bytes(32));
    }
}

if (empty($auth_users)) {
    $use_auth = false;
}

$is_https = isset($_SERVER['HTTPS']) && (strtolower($_SERVER['HTTPS']) == 'on' ||
$_SERVER['HTTPS'] == 1)
            || isset($_SERVER['HTTP_X_FORWARDED_PROTO']) &&
$_SERVER['HTTP_X_FORWARDED_PROTO'] == 'https';

// update $root_url based on user specific directories
if (isset($_SESSION[FM_SESSION_ID]['logged']) &&
!empty($directories_users[$_SESSION[FM_SESSION_ID]['logged']])) {
    $wd = fm_clean_path(dirname($_SERVER['PHP_SELF']));
    $root_url = $root_url . $wd . DIRECTORY_SEPARATOR .
$directories_users[$_SESSION[FM_SESSION_ID]['logged']];
}
// clean $root_url
$root_url = fm_clean_path($root_url);

// abs path for site
defined('FM_ROOT_URL') || define('FM_ROOT_URL', ($is_https ? 'https' : 'http') . '://' . $http_host .
(!empty($root_url) ? '/' . $root_url : ''));
defined('FM_SELF_URL') || define('FM_SELF_URL', ($is_https ? 'https' : 'http') . '://' . $http_host .
$_SERVER['PHP_SELF']);

// logout
if (isset($_GET['logout'])) {
    unset($_SESSION[FM_SESSION_ID]['logged']);
}

```

```
unset($_SESSION['token']);  
fm_redirect(FM_SELF_URL);  
}
```

```
// Validate connection IP
```

```
if ($ip_ruleset != 'OFF') {  
    function getClientIP()  
    {  
        if (array_key_exists('HTTP_CF_CONNECTING_IP', $_SERVER)) {  
            return $_SERVER['HTTP_CF_CONNECTING_IP'];  
        } else if (array_key_exists('HTTP_X_FORWARDED_FOR', $_SERVER)) {  
            return $_SERVER['HTTP_X_FORWARDED_FOR'];  
        } else if (array_key_exists('REMOTE_ADDR', $_SERVER)) {  
            return $_SERVER['REMOTE_ADDR'];  
        } else if (array_key_exists('HTTP_CLIENT_IP', $_SERVER)) {  
            return $_SERVER['HTTP_CLIENT_IP'];  
        }  
        return "";  
    }  
}
```

```
$clientIp = getClientIP();  
$proceed = false;  
$whitelisted = in_array($clientIp, $ip_whitelist);  
$blacklisted = in_array($clientIp, $ip_blacklist);
```

```
if ($ip_ruleset == 'AND') {  
    if ($whitelisted == true && $blacklisted == false) {  
        $proceed = true;  
    }  
} else  
if ($ip_ruleset == 'OR') {  
    if ($whitelisted == true || $blacklisted == false) {  
        $proceed = true;  
    }  
}
```

```
if ($proceed == false) {  
    trigger_error('User connection denied from: ' . $clientIp, E_USER_WARNING);  
}
```

```

if ($ip_silent == false) {
    fm_set_msg(Ing('Access denied. IP restriction applicable'), 'error');
    fm_show_header_login();
    fm_show_message();
}
exit();
}
}

// Checking if the user is logged in or not. If not, it will show the login form.
if ($use_auth) {
    if (isset($_SESSION[FM_SESSION_ID]['logged'],
    $auth_users[$_SESSION[FM_SESSION_ID]['logged']])) {
        // Logged
    } elseif (isset($_POST['fm_usr'], $_POST['fm_pwd'], $_POST['token'])) {
        // Logging In
        sleep(1);
        if (function_exists('password_verify')) {
            if (isset($auth_users[$_POST['fm_usr']]) && isset($_POST['fm_pwd']) &&
password_verify($_POST['fm_pwd'], $auth_users[$_POST['fm_usr']]) &&
verifyToken($_POST['token'])) {
                $_SESSION[FM_SESSION_ID]['logged'] = $_POST['fm_usr'];
                fm_set_msg(Ing('You are logged in'));
                fm_redirect(FM_SELF_URL);
            } else {
                unset($_SESSION[FM_SESSION_ID]['logged']);
                fm_set_msg(Ing('Login failed. Invalid username or password'), 'error');
                fm_redirect(FM_SELF_URL);
            }
        } else {
            fm_set_msg(Ing('password_hash not supported, Upgrade PHP version'), 'error');
        }
    } else {
        // Form
        unset($_SESSION[FM_SESSION_ID]['logged']);
        fm_show_header_login();
    }
}
?>

```

```
<section class="h-100">
  <div class="container h-100">
    <div class="row justify-content-md-center align-content-center h-100vh">
      <div class="card-wrapper">
        <div class="card fat" data-bs-theme="<?php echo FM_THEME; ?>">
          <div class="card-body">
            <form class="form-signin" action="" method="post" autocomplete="off">
              <div class="mb-3">
                <div class="brand">
                  <svg version="1.0" xmlns="http://www.w3.org/2000/svg" M1008
width="100%" height="80px" viewBox="0 0 238.000000 140.000000" aria-label="H3K Tiny File
Manager">
                    <g transform="translate(0.000000,140.000000)
scale(0.100000,-0.100000)" fill="#000000" stroke="none">
                        <path d="M160 700 l0 -600 110 0 110 0 0 260 0 260 70 0 70 0 0
-260 0 -260 110 0 110 0 0 600 0 600 -110 0 -110 0 0 -260 0 -260 -70 0 -70 0 0 260 0 260 -110 0
-110 0 0 -600z" />
                        <path fill="#003500" d="M1008 1227 l-108 -72 0 -117 0 -118 110 0
110 0 0 110 0 110 70 0 70 0 0 -180 0 -180 -125 0 c-69 0 -125 -3 -125 -6 0 -3 23 -39 52 -80 l52 -74
73 0 73 0 0 -185 0 -185 -70 0 -70 0 0 115 0 115 -110 0 -110 0 0 -190 0 -190 181 0 181 0 109 73 108
72 1 181 0 181 -69 48 -68 49 68 50 69 49 0 249 0 248 -182 -1 -183 0 -107 -72z" />
                        <path d="M1640 700 l0 -600 110 0 110 0 0 208 0 208 35 34 35 34
35 -34 35 -34 0 -208 0 -208 110 0 110 0 0 212 0 213 -87 87 -88 88 88 88 87 87 0 213 0 212 -110 0
-110 0 0 -208 0 -208 -70 -69 -70 -69 0 277 0 277 -110 0 -110 0 0 -600z" />
                    </g>
                  </svg>
                </div>
                <div class="text-center">
                  <h1 class="card-title"><?php echo APP_TITLE; ?></h1>
                </div>
              </div>
            <hr />
            <div class="mb-3">
              <label for="fm_usr" class="pb-2"><?php echo lng('Username'); ?></label>
              <input type="text" class="form-control" id="fm_usr" name="fm_usr"
required autofocus>
            </div>
```

```

        <div class="mb-3">
            <label for="fm_pwd" class="pb-2"><?php echo lng('Password'); ?></label>
            <input type="password" class="form-control" id="fm_pwd" name="fm_pwd"
required>

        </div>

        <div class="mb-3">
            <?php fm_show_message(); ?>
        </div>

            <input type="hidden" name="token" value="<?php echo
htmlentities($_SESSION['token']); ?>" />
        <div class="mb-3">
            <button type="submit" class="btn btn-success btn-block w-100 mt-4"
role="button">

                <?php echo lng('Login'); ?>
            </button>
        </div>
    </form>
</div>
</div>
<div class="footer text-center">
    &mdash;&mdash; &copy;
            <a href="https://tinyfilemanager.github.io/" target="_blank"
class="text-decoration-none text-muted" data-version="<?php echo VERSION; ?>">CCP
Programmers</a> &mdash;&mdash;
        </div>
    </div>
</div>
</div>
</section>

<?php
    fm_show_footer_login();
    exit;
}
}

// update root path

```

```

if ($use_auth && isset($_SESSION[FM_SESSION_ID]['logged'])) {
    $root_path = isset($directories_users[$_SESSION[FM_SESSION_ID]['logged']]) ?
$directories_users[$_SESSION[FM_SESSION_ID]['logged']] : $root_path;
}

```

```

// clean and check $root_path
$root_path = rtrim($root_path, '\\');
$root_path = str_replace('\\', '/', $root_path);
if (!@is_dir($root_path)) {
    echo "<h1>" . lng('Root path') . " \"{$root_path}\" " . lng('not found!') . " </h1>";
    exit;
}

```

```

defined('FM_SHOW_HIDDEN') || define('FM_SHOW_HIDDEN', $show_hidden_files);
defined('FM_ROOT_PATH') || define('FM_ROOT_PATH', $root_path);
defined('FM_LANG') || define('FM_LANG', $lang);
defined('FM_FILE_EXTENSION') || define('FM_FILE_EXTENSION', $allowed_file_extensions);
defined('FM_UPLOAD_EXTENSION') || define('FM_UPLOAD_EXTENSION',
$allowed_upload_extensions);
defined('FM_EXCLUDE_ITEMS') || define('FM_EXCLUDE_ITEMS',
(version_compare(PHP_VERSION, '7.0.0', '<') ? serialize($exclude_items) : $exclude_items));
defined('FM_DOC_VIEWER') || define('FM_DOC_VIEWER', $online_viewer);
define('FM_READONLY', $global_readonly || ($use_auth && !empty($readonly_users) &&
isset($_SESSION[FM_SESSION_ID]['logged']) &&
in_array($_SESSION[FM_SESSION_ID]['logged'], $readonly_users)));
define('FM_IS_WIN', DIRECTORY_SEPARATOR == '\\');

```

```

// always use ?p=
if (!isset($_GET['p']) && empty($_FILES)) {
    fm_redirect(FM_SELF_URL . '?p=');
}

```

```

// get path
$p = isset($_GET['p']) ? $_GET['p'] : (isset($_POST['p']) ? $_POST['p'] : "");

```

```

// clean path
$p = fm_clean_path($p);

```

```

// for ajax request - save
$input = file_get_contents('php://input');
$_POST = (strpos($input, 'ajax') != FALSE && strpos($input, 'save') != FALSE) ?
json_decode($input, true) : $_POST;

// instead globals vars
define('FM_PATH', $p);
define('FM_USE_AUTH', $use_auth);
define('FM_EDIT_FILE', $edit_files);
defined('FM_ICONV_INPUT_ENC') || define('FM_ICONV_INPUT_ENC', $iconv_input_encoding);
defined('FM_USE_HIGHLIGHTJS') || define('FM_USE_HIGHLIGHTJS', $use_highlightjs);
defined('FM_HIGHLIGHTJS_STYLE') || define('FM_HIGHLIGHTJS_STYLE', $highlightjs_style);
defined('FM_DATETIME_FORMAT') || define('FM_DATETIME_FORMAT', $datetime_format);

unset($p, $use_auth, $iconv_input_encoding, $use_highlightjs, $highlightjs_style);

/***** ACTIONS *****/

// Handle all AJAX Request
if ((isset($_SESSION[FM_SESSION_ID]['logged'],
$auth_users[$_SESSION[FM_SESSION_ID]['logged']] || !FM_USE_AUTH) &&
isset($_POST['ajax'], $_POST['token']) && !FM_READONLY) {
    if (!verifyToken($_POST['token'])) {
        header('HTTP/1.0 401 Unauthorized');
        die("Invalid Token.");
    }

    //search : get list of files from the current folder
    if (isset($_POST['type']) && $_POST['type'] == "search") {
        $dir = $_POST['path'] == "." ? " : $_POST['path'];
        $response = scan(fm_clean_path($dir), $_POST['content']);
        echo json_encode($response);
        exit();
    }

    // save editor file
    if (isset($_POST['type']) && $_POST['type'] == "save") {
        // get current path

```

```

$path = FM_ROOT_PATH;
if (FM_PATH != "") {
    $path .= '/' . FM_PATH;
}
// check path
if (!is_dir($path)) {
    fm_redirect(FM_SELF_URL . '?p=');
}
$file = $_GET['edit'];
$file = fm_clean_path($file);
$file = str_replace('/', ' ', $file);
if ($file == " || !is_file($path . '/' . $file)) {
    fm_set_msg(Ing('File not found'), 'error');
    $FM_PATH = FM_PATH;
    fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}
header('X-XSS-Protection:0');
$file_path = $path . '/' . $file;

$writedata = $_POST['content'];
$fd = fopen($file_path, "w");
$write_results = @fwrite($fd, $writedata);
fclose($fd);
if ($write_results === false) {
    header("HTTP/1.1 500 Internal Server Error");
    die("Could Not Write File! - Check Permissions / Ownership");
}
die(true);
}

// backup files
if (isset($_POST['type']) && $_POST['type'] == "backup" && !empty($_POST['file'])) {
    $fileName = fm_clean_path($_POST['file']);
    $fullPath = FM_ROOT_PATH . '/';
    if (!empty($_POST['path'])) {
        $relativeDirPath = fm_clean_path($_POST['path']);
        $fullPath .= "{$relativeDirPath}/";
    }
}

```

```

$date = date("dMy-His");
$newFileName = "{$fileName}-{$date}.bak";
$fullyQualifiedFileName = $fullPath . $fileName;
try {
    if (!file_exists($fullyQualifiedFileName)) {
        throw new Exception("File {$fileName} not found");
    }
    if (copy($fullyQualifiedFileName, $fullPath . $newFileName)) {
        echo "Backup {$newFileName} created";
    } else {
        throw new Exception("Could not copy file {$fileName}");
    }
} catch (Exception $e) {
    echo $e->getMessage();
}
}

// Save Config
if (isset($_POST['type']) && $_POST['type'] == "settings") {
    global $cfg, $lang, $report_errors, $show_hidden_files, $lang_list, $hide_cols, $theme;
    $newLng = $_POST['js-language'];
    fm_get_translations([]);
    if (!array_key_exists($newLng, $lang_list)) {
        $newLng = 'en';
    }

    $erp = isset($_POST['js-error-report']) && $_POST['js-error-report'] == "true" ? true : false;
    $shf = isset($_POST['js-show-hidden']) && $_POST['js-show-hidden'] == "true" ? true : false;
    $hco = isset($_POST['js-hide-cols']) && $_POST['js-hide-cols'] == "true" ? true : false;
    $te3 = $_POST['js-theme-3'];

    if ($cfg->data['lang'] != $newLng) {
        $cfg->data['lang'] = $newLng;
        $lang = $newLng;
    }
    if ($cfg->data['error_reporting'] != $erp) {
        $cfg->data['error_reporting'] = $erp;
        $report_errors = $erp;
    }
}

```

```

}
if ($cfg->data['show_hidden'] != $shf) {
    $cfg->data['show_hidden'] = $shf;
    $show_hidden_files = $shf;
}
if ($cfg->data['show_hidden'] != $shf) {
    $cfg->data['show_hidden'] = $shf;
    $show_hidden_files = $shf;
}
if ($cfg->data['hide_Cols'] != $hco) {
    $cfg->data['hide_Cols'] = $hco;
    $hide_Cols = $hco;
}
if ($cfg->data['theme'] != $te3) {
    $cfg->data['theme'] = $te3;
    $theme = $te3;
}
$cfg->save();
echo true;
}

```

```

// new password hash
if (isset($_POST['type']) && $_POST['type'] == "pwdhash") {
    $res = isset($_POST['inputPassword2']) && !empty($_POST['inputPassword2']) ?
password_hash($_POST['inputPassword2'], PASSWORD_DEFAULT) : "";
    echo $res;
}

```

```

//upload using url
if (isset($_POST['type']) && $_POST['type'] == "upload" && !empty($_REQUEST["uploadurl"])) {
    $path = FM_ROOT_PATH;
    if (FM_PATH != "") {
        $path .= '/' . FM_PATH;
    }
}

```

```

function event_callback($message)
{
    global $callback;

```

```

        echo json_encode($message);
    }

    function get_file_path()
    {
        global $path, $fileinfo, $temp_file;
        return $path . "/" . basename($fileinfo->name);
    }

    $url = !empty($_REQUEST["uploadurl"]) && preg_match("|^http(s)?://.+?$|",
stripslashes($_REQUEST["uploadurl"])) ? stripslashes($_REQUEST["uploadurl"]) : null;

    //prevent 127.* domain and known ports
    $domain = parse_url($url, PHP_URL_HOST);
    $port = parse_url($url, PHP_URL_PORT);
    $knownPorts = [22, 23, 25, 3306];

    if (preg_match("/^localhost$|^127(?:\.[0-9]+){0,2}\.[0-9]+$|^(?:0*\.)*?:?0*1$/i", $domain) ||
in_array($port, $knownPorts)) {
        $err = array("message" => "URL is not allowed");
        event_callback(array("fail" => $err));
        exit();
    }

    $use_curl = false;
    $temp_file = tempnam(sys_get_temp_dir(), "upload-");
    $fileinfo = new stdClass();
    $fileinfo->name = trim(urldecode(basename($url)), "\x00..\x20");

    $allowed = (FM_UPLOAD_EXTENSION) ? explode(',', FM_UPLOAD_EXTENSION) : false;
    $ext = strtolower(pathinfo($fileinfo->name, PATHINFO_EXTENSION));
    $isFileAllowed = ($allowed) ? in_array($ext, $allowed) : true;

    $err = false;

    if (!$isFileAllowed) {
        $err = array("message" => "File extension is not allowed");
        event_callback(array("fail" => $err));
    }

```

```

    exit();
}

if (!$url) {
    $success = false;
} else if ($use_curl) {
    @$fp = fopen($temp_file, "w");
    @$ch = curl_init($url);
    curl_setopt($ch, CURLOPT_NOPROGRESS, false);
    curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
    curl_setopt($ch, CURLOPT_FILE, $fp);
    @$success = curl_exec($ch);
    $curl_info = curl_getinfo($ch);
    if (!$success) {
        $err = array("message" => curl_error($ch));
    }
    @curl_close($ch);
    fclose($fp);
    $fileinfo->size = $curl_info["size_download"];
    $fileinfo->type = $curl_info["content_type"];
} else {
    $ctx = stream_context_create();
    @$success = copy($url, $temp_file, $ctx);
    if (!$success) {
        $err = error_get_last();
    }
}

if ($success) {
    $success = rename($temp_file, strtok(get_file_path(), '?'));
}

if ($success) {
    event_callback(array("done" => $fileinfo));
} else {
    unlink($temp_file);
    if (!$err) {
        $err = array("message" => "Invalid url parameter");
    }
}

```

```

    }
    event_callback(array("fail" => $err));
}
}
exit();
}

// Delete file / folder
if (isset($_GET['del'], $_POST['token']) && !FM_READONLY) {
    $del = str_replace('/', '', fm_clean_path($_GET['del']));
    if ($del != "" && $del != '..' && $del != '.' && verifyToken($_POST['token'])) {
        $path = FM_ROOT_PATH;
        if (FM_PATH != "") {
            $path .= '/' . FM_PATH;
        }
        $is_dir = is_dir($path . '/' . $del);
        if (fm_rdelete($path . '/' . $del)) {
            $msg = $is_dir ? lng('Folder') . ' <b>%s</b> ' . lng('Deleted') : lng('File') . ' <b>%s</b> ' .
lng('Deleted');
            fm_set_msg(sprintf($msg, fm_enc($del)));
        } else {
            $msg = $is_dir ? lng('Folder') . ' <b>%s</b> ' . lng('not deleted') : lng('File') . ' <b>%s</b> ' .
lng('not deleted');
            fm_set_msg(sprintf($msg, fm_enc($del)), 'error');
        }
    } else {
        fm_set_msg(lng('Invalid file or folder name'), 'error');
    }
    $FM_PATH = FM_PATH;
    fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}

// Create a new file/folder
if (isset($_POST['newfilename'], $_POST['newfile'], $_POST['token']) && !FM_READONLY) {
    $type = urldecode($_POST['newfile']);
    $new = str_replace('/', '', fm_clean_path(strip_tags($_POST['newfilename'])));
    if (fm_isvalid_filename($new) && $new != "" && $new != '..' && $new != '.' &&
verifyToken($_POST['token'])) {

```

```

$path = FM_ROOT_PATH;
if (FM_PATH != "") {
    $path .= '/' . FM_PATH;
}
if ($type == "file") {
    if (!file_exists($path . '/' . $new)) {
        if (fm_is_valid_ext($new)) {
            @fopen($path . '/' . $new, 'w') or die('Cannot open file: ' . $new);
            fm_set_msg(sprintf(Ing('File') . ' <b>%s</b> ' . Ing('Created'), fm_enc($new)));
        } else {
            fm_set_msg(Ing('File extension is not allowed'), 'error');
        }
    } else {
        fm_set_msg(sprintf(Ing('File') . ' <b>%s</b> ' . Ing('already exists'), fm_enc($new)), 'alert');
    }
} else {
    if (fm_mkdir($path . '/' . $new, false) === true) {
        fm_set_msg(sprintf(Ing('Folder') . ' <b>%s</b> ' . Ing('Created'), $new));
    } elseif (fm_mkdir($path . '/' . $new, false) === $path . '/' . $new) {
        fm_set_msg(sprintf(Ing('Folder') . ' <b>%s</b> ' . Ing('already exists'), fm_enc($new)),
'alert');
    } else {
        fm_set_msg(sprintf(Ing('Folder') . ' <b>%s</b> ' . Ing('not created'), fm_enc($new)),
'error');
    }
}
} else {
    fm_set_msg(Ing('Invalid characters in file or folder name'), 'error');
}
$FM_PATH = FM_PATH;
fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}

```

// Copy folder / file

```

if (isset($_GET['copy'], $_GET['finish']) && !FM_READONLY) {
    // from
    $copy = urldecode($_GET['copy']);
    $copy = fm_clean_path($copy);
}

```

```

// empty path
if ($copy == "") {
    fm_set_msg(Ing('Source path not defined'), 'error');
    $FM_PATH = FM_PATH;
    fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}
// abs path from
$from = FM_ROOT_PATH . '/' . $copy;
// abs path to
$dest = FM_ROOT_PATH;
if (FM_PATH != "") {
    $dest .= '/' . FM_PATH;
}
$dest .= '/' . basename($from);
// move?
$move = isset($_GET['move']);
$move = fm_clean_path(urldecode($move));
// copy/move/duplicate
if ($from != $dest) {
    $msg_from = trim(FM_PATH . '/' . basename($from), '/');
    if ($move) { // Move and to != from so just perform move
        $rename = fm_rename($from, $dest);
        if ($rename) {
            fm_set_msg(sprintf(Ing('Moved from') . ' ' . <b>%s</b> ' . Ing('to') . ' ' . <b>%s</b>',
fm_enc($copy), fm_enc($msg_from)));
        } elseif ($rename === null) {
            fm_set_msg(Ing('File or folder with this path already exists'), 'alert');
        } else {
            fm_set_msg(sprintf(Ing('Error while moving from') . ' ' . <b>%s</b> ' . Ing('to') . ' ' . <b>%s</b>',
fm_enc($copy), fm_enc($msg_from)), 'error');
        }
    } else { // Not move and to != from so copy with original name
        if (fm_rcopy($from, $dest)) {
            fm_set_msg(sprintf(Ing('Copied from') . ' ' . <b>%s</b> ' . Ing('to') . ' ' . <b>%s</b>',
fm_enc($copy), fm_enc($msg_from)));
        } else {
            fm_set_msg(sprintf(Ing('Error while copying from') . ' ' . <b>%s</b> ' . Ing('to') . ' ' . <b>%s</b>',
fm_enc($copy), fm_enc($msg_from)), 'error');
        }
    }
}

```

```

    }
}
} else {
    if (!$move) { //Not move and to = from so duplicate
        $msg_from = trim(FM_PATH . '/' . basename($from), '/');
        $fn_parts = pathinfo($from);
        $extension_suffix = "";
        if (!is_dir($from)) {
            $extension_suffix = '.' . $fn_parts['extension'];
        }
        //Create new name for duplicate
        $fn_duplicate = $fn_parts['dirname'] . '/' . $fn_parts['filename'] . '-' . date('YmdHis') .
$extension_suffix;
        $loop_count = 0;
        $max_loop = 1000;
        // Check if a file with the duplicate name already exists, if so, make new name (edge case...)
        while (file_exists($fn_duplicate) & $loop_count < $max_loop) {
            $fn_parts = pathinfo($fn_duplicate);
            $fn_duplicate = $fn_parts['dirname'] . '/' . $fn_parts['filename'] . '-copy' . $extension_suffix;
            $loop_count++;
        }
        if (fm_rcopy($from, $fn_duplicate, False)) {
            fm_set_msg(sprintf('Copied from <b>%s</b> to <b>%s</b>', fm_enc($copy),
fm_enc($fn_duplicate)));
        } else {
            fm_set_msg(sprintf('Error while copying from <b>%s</b> to <b>%s</b>', fm_enc($copy),
fm_enc($fn_duplicate)), 'error');
        }
    } else {
        fm_set_msg(lng('Paths must be not equal'), 'alert');
    }
}
}
$FM_PATH = FM_PATH;
fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}

```

// Mass copy files/ folders

```
if (isset($_POST['file'], $_POST['copy_to'], $_POST['finish'], $_POST['token'])) && !FM_READONLY)
```

```

{

if (!verifyToken($_POST['token'])) {
    fm_set_msg(Ing('Invalid Token.'), 'error');
    die("Invalid Token.");
}

// from
$path = FM_ROOT_PATH;
if (FM_PATH != "") {
    $path .= '/' . FM_PATH;
}
// to
$copy_to_path = FM_ROOT_PATH;
$copy_to = fm_clean_path($_POST['copy_to']);
if ($copy_to != "") {
    $copy_to_path .= '/' . $copy_to;
}
if ($path == $copy_to_path) {
    fm_set_msg(Ing('Paths must be not equal'), 'alert');
    $FM_PATH = FM_PATH;
    fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}
if (!is_dir($copy_to_path)) {
    if (!fm_mkdir($copy_to_path, true)) {
        fm_set_msg('Unable to create destination folder', 'error');
        $FM_PATH = FM_PATH;
        fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
    }
}
// move?
$move = isset($_POST['move']);
// copy/move
$errors = 0;
$files = $_POST['file'];
if (is_array($files) && count($files)) {
    foreach ($files as $f) {
        if ($f != "") {

```

```

    $f = fm_clean_path($f);
    // abs path from
    $from = $path . '/' . $f;
    // abs path to
    $dest = $copy_to_path . '/' . $f;
    // do
    if ($move) {
        $rename = fm_rename($from, $dest);
        if ($rename === false) {
            $errors++;
        }
    } else {
        if (!fm_rcopy($from, $dest)) {
            $errors++;
        }
    }
}

if ($errors == 0) {
    $msg = $move ? 'Selected files and folders moved' : 'Selected files and folders copied';
    fm_set_msg($msg);
} else {
    $msg = $move ? 'Error while moving items' : 'Error while copying items';
    fm_set_msg($msg, 'error');
}

} else {
    fm_set_msg(lng('Nothing selected'), 'alert');
}

$FM_PATH = FM_PATH;
fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}

// Rename
if (isset($_POST['rename_from'], $_POST['rename_to'], $_POST['token']) && !FM_READONLY) {
    if (!verifyToken($_POST['token'])) {
        fm_set_msg("Invalid Token.", 'error');
        die("Invalid Token.");
    }
}

```

```

// old name
$old = urldecode($_POST['rename_from']);
$old = fm_clean_path($old);
$old = str_replace('/', '', $old);
// new name
$new = urldecode($_POST['rename_to']);
$new = fm_clean_path(strip_tags($new));
$new = str_replace('/', '', $new);
// path
$path = FM_ROOT_PATH;
if (FM_PATH != "") {
    $path .= '/' . FM_PATH;
}
// rename
if (fm_isvalid_filename($new) && $old != "" && $new != "") {
    if (fm_rename($path . '/' . $old, $path . '/' . $new)) {
        fm_set_msg(sprintf(lng('Renamed from') . ' <b>%s</b> ' . lng('to') . ' <b>%s</b>',
fm_enc($old), fm_enc($new)));
    } else {
        fm_set_msg(sprintf(lng('Error while renaming from') . ' <b>%s</b> ' . lng('to') . ' <b>%s</b>',
fm_enc($old), fm_enc($new)), 'error');
    }
} else {
    fm_set_msg(lng('Invalid characters in file name'), 'error');
}
$FM_PATH = FM_PATH;
fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}

// Download
if (isset($_GET['dl'], $_POST['token'])) {
    // Verify the token to ensure it's valid
    if (!verifyToken($_POST['token'])) {
        fm_set_msg("Invalid Token.", 'error');
        exit;
    }

    // Clean the download file path

```

```

$dl = urldecode($_GET['dl']);
$dl = fm_clean_path($dl);
$dl = str_replace('/', '', $dl); // Prevent directory traversal attacks


// Define the file path
$path = FM_ROOT_PATH;
if (FM_PATH != "") {
    $path .= '/' . FM_PATH;
}


// Check if the file exists and is valid
if ($dl != "" && is_file($path . '/' . $dl)) {
    // Close the session to prevent session locking
    if (session_status() === PHP_SESSION_ACTIVE) {
        session_write_close();
    }


    // Call the download function
    fm_download_file($path . '/' . $dl, $dl, 1024); // Download with a buffer size of 1024 bytes
    exit;
} else {
    // Handle the case where the file is not found
    fm_set_msg(lng('File not found'), 'error');
    $FM_PATH = FM_PATH;
    fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}
}


// Upload
if (!empty($_FILES) && !FM_READONLY) {
    if (isset($_POST['token'])) {
        if (!verifyToken($_POST['token'])) {
            $response = array('status' => 'error', 'info' => "Invalid Token.");
            echo json_encode($response);
            exit();
        }
    } else {
        $response = array('status' => 'error', 'info' => "Token Missing.");
    }
}

```

```

    echo json_encode($response);
    exit();
}

$chunkIndex = $_POST['dzchunkindex'];
$chunkTotal = $_POST['dztotallchunkcount'];
$fullPathInput = fm_clean_path($_REQUEST['fullpath']);

$f = $_FILES;
$path = FM_ROOT_PATH;
$ds = DIRECTORY_SEPARATOR;
if (FM_PATH != "") {
    $path .= '/' . FM_PATH;
}

$errors = 0;
$uploads = 0;
$allowed = (FM_UPLOAD_EXTENSION) ? explode(',', FM_UPLOAD_EXTENSION) : false;
$response = array(
    'status' => 'error',
    'info'   => 'Oops! Try again'
);

$filename = $f['file']['name'];
$tmp_name = $f['file']['tmp_name'];
$ext = pathinfo($filename, PATHINFO_FILENAME) != "" ? strtolower(pathinfo($filename,
PATHINFO_EXTENSION)) : "";
$isFileAllowed = ($allowed) ? in_array($ext, $allowed) : true;

if (!fm_isvalid_filename($filename) && !fm_isvalid_filename($fullPathInput)) {
    $response = array(
        'status'   => 'error',
        'info'     => "Invalid File name!",
    );
    echo json_encode($response);
    exit();
}

```

```

$targetPath = $path . $ds;
if (is_writable($targetPath)) {
    $fullPath = $path . '/' . $fullPathInput;
    $folder = substr($fullPath, 0, strrpos($fullPath, "/"));

    if (!is_dir($folder)) {
        $old = umask(0);
        mkdir($folder, 0777, true);
        umask($old);
    }

    if (empty($f['file']['error']) && !empty($tmp_name) && $tmp_name != 'none' && $isFileAllowed) {
        if ($chunkTotal) {
            $out = @fopen("{ $fullPath}.part", $chunkIndex == 0 ? "wb" : "ab");
            if ($out) {
                $in = @fopen($tmp_name, "rb");
                if ($in) {
                    if (PHP_VERSION_ID < 80009) {
                        // workaround https://bugs.php.net/bug.php?id=81145
                        do {
                            for (;;) {
                                $buff = fread($in, 4096);
                                if ($buff === false || $buff === "") {
                                    break;
                                }
                            }
                            fwrite($out, $buff);
                        } while (!feof($in));
                    } else {
                        stream_copy_to_stream($in, $out);
                    }
                }
                $response = array(
                    'status' => 'success',
                    'info' => "file upload successful"
                );
            } else {
                $response = array(
                    'status' => 'error',

```

```

        'info' => "failed to open output stream",
        'errorDetails' => error_get_last()
    );
}
@fclose($in);
@fclose($out);
@unlink($tmp_name);

$response = array(
    'status' => 'success',
    'info' => "file upload successful"
);
} else {
    $response = array(
        'status' => 'error',
        'info' => "failed to open output stream"
    );
}

if ($chunkIndex == $chunkTotal - 1) {
    if (file_exists($fullPath)) {
        $ext_1 = $ext ? '.' . $ext : "";
        $fullPathTarget = $path . '/' . basename($fullPathInput, $ext_1) . '_' . date('ymdHis') .
$ext_1;
    } else {
        $fullPathTarget = $fullPath;
    }
    rename("{ $fullPath }.part", $fullPathTarget);
}
} else if (move_uploaded_file($tmp_name, $fullPath)) {
    // Be sure that the file has been uploaded
    if (file_exists($fullPath)) {
        $response = array(
            'status' => 'success',
            'info' => "file upload successful"
        );
    } else {
        $response = array(

```

```

        'status' => 'error',
        'info'  => 'Couldn\'t upload the requested file.'
    );
}
} else {
    $response = array(
        'status'  => 'error',
        'info'    => "Error while uploading files. Uploaded files $uploads",
    );
}
}
} else {
    $response = array(
        'status' => 'error',
        'info'  => 'The specified folder for upload isn\'t writeable.'
    );
}
// Return the response
echo json_encode($response);
exit();
}

// Mass deleting
if (isset($_POST['group'], $_POST['delete'], $_POST['token']) && !FM_READONLY) {

    if (!verifyToken($_POST['token'])) {
        fm_set_msg(Ing("Invalid Token."), 'error');
        die("Invalid Token.");
    }

    $path = FM_ROOT_PATH;
    if (FM_PATH != "") {
        $path .= '/' . FM_PATH;
    }

    $errors = 0;
    $files = $_POST['file'];
    if (is_array($files) && count($files)) {

```

```

foreach ($files as $f) {
    if ($f != "") {
        $new_path = $path . '/' . $f;
        if (!fm_rdelete($new_path)) {
            $errors++;
        }
    }
}
if ($errors == 0) {
    fm_set_msg(Ing('Selected files and folder deleted'));
} else {
    fm_set_msg(Ing('Error while deleting items'), 'error');
}
} else {
    fm_set_msg(Ing('Nothing selected'), 'alert');
}

$FM_PATH = FM_PATH;
fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}

// Pack files zip, tar
if (isset($_POST['group'], $_POST['token']) && (isset($_POST['zip']) || isset($_POST['tar'])) &&
!FM_READONLY) {

    if (!verifyToken($_POST['token'])) {
        fm_set_msg(Ing("Invalid Token."), 'error');
        die("Invalid Token.");
    }

    $path = FM_ROOT_PATH;
    $ext = 'zip';
    if (FM_PATH != "") {
        $path .= '/' . FM_PATH;
    }

    //set pack type
    $ext = isset($_POST['tar']) ? 'tar' : 'zip';

```

```

if (($ext == "zip" && !class_exists('ZipArchive')) || ($ext == "tar" && !class_exists('PharData'))) {
    fm_set_msg(lng('Operations with archives are not available'), 'error');
    $FM_PATH = FM_PATH;
    fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}

$files = $_POST['file'];
$sanitized_files = array();

// clean path
foreach ($files as $file) {
    array_push($sanitized_files, fm_clean_path($file));
}

$files = $sanitized_files;

if (!empty($files)) {
    chdir($path);

    if (count($files) == 1) {
        $one_file = reset($files);
        $one_file = basename($one_file);
        $zipname = $one_file . '_' . date('ymd_His') . '.' . $ext;
    } else {
        $zipname = 'archive_' . date('ymd_His') . '.' . $ext;
    }

    if ($ext == 'zip') {
        $zipper = new FM_Zipper();
        $res = $zipper->create($zipname, $files);
    } elseif ($ext == 'tar') {
        $star = new FM_Zipper_Tar();
        $res = $star->create($zipname, $files);
    }

    if ($res) {
        fm_set_msg(sprintf(lng('Archive') . ' <b>%s</b> ' . lng('Created'), fm_enc($zipname)));
    }
}

```

```

    } else {
        fm_set_msg(lng('Archive not created'), 'error');
    }
} else {
    fm_set_msg(lng('Nothing selected'), 'alert');
}

$FM_PATH = FM_PATH;
fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}

// Unpack zip, tar
if (isset($_POST['unzip'], $_POST['token']) && !FM_READONLY) {

    if (!verifyToken($_POST['token'])) {
        fm_set_msg(lng("Invalid Token."), 'error');
        die("Invalid Token.");
    }

    $unzip = urldecode($_POST['unzip']);
    $unzip = fm_clean_path($unzip);
    $unzip = str_replace('/', '', $unzip);
    $isValid = false;

    $path = FM_ROOT_PATH;
    if (FM_PATH != "") {
        $path .= '/' . FM_PATH;
    }

    if ($unzip != "" && is_file($path . '/' . $unzip)) {
        $zip_path = $path . '/' . $unzip;
        $ext = pathinfo($zip_path, PATHINFO_EXTENSION);
        $isValid = true;
    } else {
        fm_set_msg(lng('File not found'), 'error');
    }

    if (($ext == "zip" && !class_exists('ZipArchive')) || ($ext == "tar" && !class_exists('PharData'))) {

```

```

    fm_set_msg(lng('Operations with archives are not available'), 'error');
    $FM_PATH = FM_PATH;
    fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}

```

```

if ($isValid) {
    //to folder
    $tofolder = "";
    if (isset($_POST['tofolder'])) {
        $tofolder = pathinfo($zip_path, PATHINFO_FILENAME);
        if (fm_mkdir($path . '/' . $tofolder, true)) {
            $path .= '/' . $tofolder;
        }
    }
}

```

```

if ($ext == "zip") {
    $zipper = new FM_Zipper();
    $res = $zipper->unzip($zip_path, $path);
} elseif ($ext == "tar") {
    try {
        $gzipper = new PharData($zip_path);
        if (@$gzipper->extractTo($path, null, true)) {
            $res = true;
        } else {
            $res = false;
        }
    } catch (Exception $e) {
        //TODO:: need to handle the error
        $res = true;
    }
}

```

```

if ($res) {
    fm_set_msg(lng('Archive unpacked'));
} else {
    fm_set_msg(lng('Archive not unpacked'), 'error');
}
} else {

```

```

    fm_set_msg(lng('File not found'), 'error');
}
$FM_PATH = FM_PATH;
fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}

// Change Perms (not for Windows)
if (isset($_POST['chmod'], $_POST['token']) && !FM_READONLY && !FM_IS_WIN) {

    if (!verifyToken($_POST['token'])) {
        fm_set_msg(lng("Invalid Token."), 'error');
        die("Invalid Token.");
    }

    $path = FM_ROOT_PATH;
    if (FM_PATH != "") {
        $path .= '/' . FM_PATH;
    }

    $file = $_POST['chmod'];
    $file = fm_clean_path($file);
    $file = str_replace('/', '', $file);
    if ($file == "" || (!is_file($path . '/' . $file) && !is_dir($path . '/' . $file))) {
        fm_set_msg(lng('File not found'), 'error');
        $FM_PATH = FM_PATH;
        fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
    }

    $mode = 0;
    if (!empty($_POST['ur'])) {
        $mode |= 0400;
    }
    if (!empty($_POST['uw'])) {
        $mode |= 0200;
    }
    if (!empty($_POST['ux'])) {
        $mode |= 0100;
    }
}

```

```

if (!empty($_POST['gr'])) {
    $mode |= 0040;
}
if (!empty($_POST['gw'])) {
    $mode |= 0020;
}
if (!empty($_POST['gx'])) {
    $mode |= 0010;
}
if (!empty($_POST['or'])) {
    $mode |= 0004;
}
if (!empty($_POST['ow'])) {
    $mode |= 0002;
}
if (!empty($_POST['ox'])) {
    $mode |= 0001;
}

if (@chmod($path . '/' . $file, $mode)) {
    fm_set_msg(lng('Permissions changed'));
} else {
    fm_set_msg(lng('Permissions not changed'), 'error');
}

$FM_PATH = FM_PATH;
fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}

```

/***** ACTIONS *****/

```

// get current path
$path = FM_ROOT_PATH;
if (FM_PATH != "") {
    $path .= '/' . FM_PATH;
}

```

```

// check path

```

```

if (!is_dir($path)) {
    fm_redirect(FM_SELF_URL . '?p=');
}

// get parent folder
$parent = fm_get_parent_path(FM_PATH);

$objects = is_readable($path) ? scandir($path) : array();
$folders = array();
$files = array();
$current_path = array_slice(explode("/", $path), -1)[0];
if (is_array($objects) && fm_is_exclude_items($current_path, $path)) {
    foreach ($objects as $file) {
        if ($file == '.' || $file == '..') {
            continue;
        }
        if (!FM_SHOW_HIDDEN && substr($file, 0, 1) === '.') {
            continue;
        }
        $new_path = $path . '/' . $file;
        if (@is_file($new_path) && fm_is_exclude_items($file, $new_path)) {
            $files[] = $file;
        } elseif (@is_dir($new_path) && $file != '.' && $file != '..' && fm_is_exclude_items($file,
$new_path)) {
            $folders[] = $file;
        }
    }
}

if (!empty($files)) {
    natcasesort($files);
}
if (!empty($folders)) {
    natcasesort($folders);
}

// upload form
if (isset($_GET['upload']) && !FM_READONLY) {

```

```

fm_show_header(); // HEADER
fm_show_nav_path(FM_PATH); // current path
//get the allowed file extensions
function getUploadExt()
{
    $extArr = explode(',', FM_UPLOAD_EXTENSION);
    if (FM_UPLOAD_EXTENSION && $extArr) {
        array_walk($extArr, function (&$x) {
            $x = ".$x";
        });
        return implode(',', $extArr);
    }
    return "";
}
?>
<?php print_external('css-dropzone'); ?>
<div class="path">

    <div class="card mb-2 fm-upload-wrapper" data-bs-theme="<?php echo FM_THEME; ?>">
        <div class="card-header">
            <ul class="nav nav-tabs card-header-tabs">
                <li class="nav-item">
                    <a class="nav-link active" href="#fileUploader" data-target="#fileUploader"><i
class="fa fa-arrow-circle-o-up"></i> <?php echo lng('UploadingFiles') ?></a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="#urlUploader" class="js-url-upload"
data-target="#urlUploader"><i class="fa fa-link"></i> <?php echo lng('Upload from URL') ?></a>
                </li>
            </ul>
        </div>
        <div class="card-body">
            <p class="card-text">
                <a href="?p=<?php echo FM_PATH ?>" class="float-right"><i class="fa
fa-chevron-circle-left go-back"></i> <?php echo lng('Back') ?></a>
                <strong><?php echo lng('DestinationFolder') ?></strong>: <?php echo
fm_enc(fm_convert_win(FM_PATH)) ?>
            </p>

```

```

        <form action="<?php echo htmlspecialchars(FM_SELF_URL) . '?p=' . fm_enc(FM_PATH)
?>" class="dropzone card-tabs-container" id="fileUploader" enctype="multipart/form-data">
            <input type="hidden" name="p" value="<?php echo fm_enc(FM_PATH) ?>">
                <input type="hidden" name="fullpath" id="fullpath" value="<?php echo
fm_enc(FM_PATH) ?>">
            <input type="hidden" name="token" value="<?php echo $_SESSION['token']; ?>">
            <div class="fallback">
                <input name="file" type="file" multiple />
            </div>
        </form>

        <div class="upload-url-wrapper card-tabs-container hidden" id="urlUploader">
            <form id="js-form-url-upload" class="row row-cols-lg-auto g-3 align-items-center"
onsubmit="return upload_from_url(this);" method="POST" action="">
                <input type="hidden" name="type" value="upload" aria-label="hidden"
aria-hidden="true">
                <input type="url" placeholder="URL" name="uploadurl" required class="form-control"
style="width: 80%">
                <input type="hidden" name="token" value="<?php echo $_SESSION['token']; ?>">
                <button type="submit" class="btn btn-primary ms-3"><?php echo lng('Upload')
?></button>

                <div class="lds-facebook">
                    <div></div>
                    <div></div>
                    <div></div>
                </div>
            </form>
            <div id="js-url-upload__list" class="col-9 mt-3"></div>
        </div>
    </div>
<?php print_external('js-dropzone'); ?>
<script>
    Dropzone.options.fileUploader = {
        chunking: true,
        chunkSize: <?php echo UPLOAD_CHUNK_SIZE; ?>,

```

```

forceChunking: true,
retryChunks: true,
retryChunksLimit: 3,
parallelUploads: 1,
parallelChunkUploads: false,
timeout: 120000,
maxFileSize: "<?php echo MAX_UPLOAD_SIZE; ?>",
acceptedFiles: "<?php echo getUploadExt() ?>",
init: function() {
    this.on("sending", function(file, xhr, formData) {
        let _path = (file.fullPath) ? file.fullPath : file.name;
        document.getElementById("fullpath").value = _path;
        xhr.ontimeout = (function() {
            toast('Error: Server Timeout');
        });
    }).on("success", function(res) {
        try {
            let _response = JSON.parse(res.xhr.response);

            if (_response.status == "error") {
                toast(_response.info);
            }
        } catch (e) {
            toast("Error: Invalid JSON response");
        }
    }).on("error", function(file, response) {
        toast(response);
    });
}
}
</script>
<?php
    fm_show_footer();
    exit;
}

// copy form POST
if (isset($_POST['copy']) && !FM_READONLY) {

```

```

$copy_files = isset($_POST['file']) ? $_POST['file'] : null;
if (!is_array($copy_files) || empty($copy_files)) {
    fm_set_msg(lng('Nothing selected'), 'alert');
    $FM_PATH = FM_PATH;
    fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}

fm_show_header(); // HEADER
fm_show_nav_path(FM_PATH); // current path
?>
<div class="path">
    <div class="card" data-bs-theme="<?php echo FM_THEME; ?>">
        <div class="card-header">
            <h6><?php echo lng('Copying') ?></h6>
        </div>
        <div class="card-body">
            <form action="" method="post">
                <input type="hidden" name="p" value="<?php echo fm_enc(FM_PATH) ?>">
                <input type="hidden" name="finish" value="1">
                <?php
                foreach ($copy_files as $cf) {
                    echo '<input type="hidden" name="file[]" value="" . fm_enc($cf) . "">' . PHP_EOL;
                }
                ?>
                <p class="break-word"><strong><?php echo lng('Files') ?></strong>: <b><?php echo
implode('</b>', <b>', $copy_files) ?></b></p>
                <p class="break-word"><strong><?php echo lng('SourceFolder') ?></strong>: <?php
echo fm_enc(fm_convert_win(FM_ROOT_PATH . '/' . FM_PATH)) ?><br>
                    <label for="inp_copy_to"><strong><?php echo lng('DestinationFolder')
?></strong>:</label>
                    <?php echo FM_ROOT_PATH ?></input type="text" name="copy_to"
id="inp_copy_to" value="<?php echo fm_enc(FM_PATH) ?>">
                    </p>
                    <p class="custom-checkbox custom-control"><input type="checkbox" name="move"
value="1" id="js-move-files" class="custom-control-input">
                    <label for="js-move-files" class="custom-control-label ms-2"><?php echo lng('Move')
?></label>
                    </p>

```

```

        <p>
            <b><a href="?p=?php echo urlencode(FM_PATH) ?>" class="btn
btn-outline-danger"><i class="fa fa-times-circle"></i> <?php echo lng('Cancel') ?></a></b>&nbsp;
            <input type="hidden" name="token" value="?php echo $_SESSION['token']; ?>">
            <button type="submit" class="btn btn-success"><i class="fa fa-check-circle"></i>
<?php echo lng('Copy') ?></button>
        </p>
    </form>
</div>
</div>
</div>
<?php
    fm_show_footer();
    exit;
}

// copy form
if (isset($_GET['copy']) && !isset($_GET['finish']) && !FM_READONLY) {
    $copy = $_GET['copy'];
    $copy = fm_clean_path($copy);
    if ($copy == " || !file_exists(FM_ROOT_PATH . '/' . $copy)) {
        fm_set_msg(lng('File not found'), 'error');
        $FM_PATH = FM_PATH;
        fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
    }

    fm_show_header(); // HEADER
    fm_show_nav_path(FM_PATH); // current path
?>
    <div class="path">
        <p><b>Copying</b></p>
        <p class="break-word">
            <strong>Source path:</strong> <?php echo fm_enc(fm_convert_win(FM_ROOT_PATH . '/' .
$copy)) ?><br>
            <strong>Destination    folder:</strong>    <?php echo
fm_enc(fm_convert_win(FM_ROOT_PATH . '/' . FM_PATH)) ?>
        </p>
        <p>

```

```

        <b><a href="?p=?php echo urlencode(FM_PATH) ?>&copy=?php echo
urlencode($copy) ?>&finish=1"><i class="fa fa-check-circle"></i> Copy</a></b> &nbsp;
        <b><a href="?p=?php echo urlencode(FM_PATH) ?>&copy=?php echo
urlencode($copy) ?>&finish=1&move=1"><i class="fa fa-check-circle"></i> Move</a></b>
&nbsp;
        <b><a href="?p=?php echo urlencode(FM_PATH) ?>" class="text-danger"><i class="fa
fa-times-circle"></i> Cancel</a></b>
    </p>
    <p><i><?php echo lng('Select folder') ?></i></p>
    <ul class="folders break-word">
        <?php
        if ($parent !== false) {
            ?>
                <li><a href="?p=?php echo urlencode($parent) ?>&copy=?php echo
urlencode($copy) ?>"><i class="fa fa-chevron-circle-left"></i> ..</a></li>
                <?php
                }
                foreach ($folders as $f) {
                    ?>
                        <li>
                            <a href="?p=?php echo urlencode(trim(FM_PATH . '/' . $f, '/')) ?>&copy=?php
echo urlencode($copy) ?>"><i class="fa fa-folder-o"></i> <?php echo fm_convert_win($f) ?></a>
                            </li>
                        <?php
                        }
                    ?>
                </li>
            </ul>
        </div>
    <?php
    fm_show_footer();
    exit;
}

if (isset($_GET['settings']) && !FM_READONLY) {
    fm_show_header(); // HEADER
    fm_show_nav_path(FM_PATH); // current path
    global $cfg, $lang, $lang_list;
    ?>

```

```

<div class="col-md-8 offset-md-2 pt-3">
    <div class="card mb-2" data-bs-theme="<?php echo FM_THEME; ?>">
        <h6 class="card-header d-flex justify-content-between">
            <span><i class="fa fa-cog"></i> <?php echo lng('Settings') ?></span>
            <a href="?p=<?php echo FM_PATH ?>" class="text-danger"><i class="fa
fa-times-circle-o"></i> <?php echo lng('Cancel') ?></a>
        </h6>
        <div class="card-body">
            <form id="js-settings-form" action="" method="post" data-type="ajax" onsubmit="return
save_settings(this)">
                <input type="hidden" name="type" value="settings" aria-label="hidden"
aria-hidden="true">
                <div class="form-group row">
                    <label for="js-language" class="col-sm-3 col-form-label"><?php echo lng('Language')
?></label>
                    <div class="col-sm-5">
                        <select class="form-select" id="js-language" name="js-language">
                            <?php
function getSelected($l)
{
    global $lang;
    return ($lang == $l) ? 'selected' : '';
}
foreach ($lang_list as $k => $v) {
    echo "<option value='$k' " . getSelected($k) . ">$v</option>";
}
?>
                        </select>
                    </div>
                </div>
                <div class="mt-3 mb-3 row ">
                    <label for="js-error-report" class="col-sm-3 col-form-label"><?php echo
lng('ErrorReporting') ?></label>
                    <div class="col-sm-9">
                        <div class="form-check form-switch">
                            <input class="form-check-input" type="checkbox" role="switch"
id="js-error-report" name="js-error-report" value="true" <?php echo $report_errors ? 'checked' : ''; ?>

```

/>

</div>

</div>

</div>

<div class="mb-3 row">

<label for="js-show-hidden" class="col-sm-3 col-form-label"><?php echo lng('ShowHiddenFiles') ?></label>

<div class="col-sm-9">

<div class="form-check form-switch">

<input class="form-check-input" type="checkbox" role="switch" id="js-show-hidden" name="js-show-hidden" value="true" <?php echo \$show_hidden_files ? 'checked' : ''; ?> />

</div>

</div>

</div>

<div class="mb-3 row">

<label for="js-hide-cols" class="col-sm-3 col-form-label"><?php echo lng('HideColumns') ?></label>

<div class="col-sm-9">

<div class="form-check form-switch">

<input class="form-check-input" type="checkbox" role="switch" id="js-hide-cols" name="js-hide-cols" value="true" <?php echo \$hide_Cols ? 'checked' : ''; ?> />

</div>

</div>

</div>

<div class="mb-3 row">

<label for="js-3-1" class="col-sm-3 col-form-label"><?php echo lng('Theme') ?></label>

<div class="col-sm-5">

<select class="form-select w-100 text-capitalize" id="js-3-0" name="js-theme-3">

<option value='light' <?php if (\$theme == "light") {
echo "selected";
} ?>>

<?php echo lng('light') ?>

</option>

```

        <option value='dark' <?php if ($theme == "dark") {
            echo "selected";
        } ?>>
        <?php echo lng('dark') ?>
    </option>
</select>
</div>
</div>

<div class="mb-3 row">
    <div class="col-sm-10">
        <button type="submit" class="btn btn-success"> <i class="fa fa-check-circle"></i>
<?php echo lng('Save'); ?></button>
    </div>
</div>

    <small class="text-body-secondary">* <?php echo lng('Sometimes the save action may
not work on the first try, so please attempt it again') ?>.</small>
</form>
</div>
</div>
</div>
<?php
    fm_show_footer();
    exit;
}

if (isset($_GET['help'])) {
    fm_show_header(); // HEADER
    fm_show_nav_path(FM_PATH); // current path
    global $cfg, $lang;
    ?>

    <div class="col-md-8 offset-md-2 pt-3">
        <div class="card mb-2" data-bs-theme="<?php echo FM_THEME; ?>">
            <h6 class="card-header d-flex justify-content-between">
                <span><i class="fa fa-exclamation-circle"></i> <?php echo lng('Help') ?></span>
                <a href="?p=<?php echo FM_PATH ?>" class="text-danger"><i class="fa

```

```

fa-times-circle-o"></i> <?php echo lng('Cancel') ?></a>
</h6>
<div class="card-body">
    <div class="row">
        <div class="col-xs-12 col-sm-6">
            <p>
                <h3><a href="https://github.com/prasathmani/tinyfilemanager" target="_blank"
class="app-v-title"> Tiny File Manager <?php echo VERSION; ?></a></h3>
            </p>
            <p>Author: PRAŚATH MANI</p>
            <p>Mail Us: <a href="mailto:ccpprogrammers@gmail.com">ccpprogrammers [at]
gmail [dot] com</a> </p>
        </div>
        <div class="col-xs-12 col-sm-6">
            <div class="card">
                <ul class="list-group list-group-flush">
                    <li class="list-group-item"><a
href="https://github.com/prasathmani/tinyfilemanager/wiki" target="_blank"><i class="fa
fa-question-circle"></i> <?php echo lng('Help Documents') ?> </a> </li>
                    <li class="list-group-item"><a
href="https://github.com/prasathmani/tinyfilemanager/issues" target="_blank"><i class="fa
fa-bug"></i> <?php echo lng('Report Issue') ?></a></li>
                    <?php if (!IFM_READONLY) { ?>
                        <li class="list-group-item"><a href="javascript:show_new_pwd();"><i
class="fa fa-lock"></i> <?php echo lng('Generate new password hash') ?></a></li>
                    <?php } ?>
                </ul>
            </div>
        </div>
    </div>
    <div class="row js-new-pwd hidden mt-2">
        <div class="col-12">
            <form class="form-inline" onsubmit="return new_password_hash(this)"
method="POST" action="">
                <input type="hidden" name="type" value="pwdhash" aria-label="hidden"
aria-hidden="true">
                <div class="form-group mb-2">
                    <label for="staticEmail2"><?php echo lng('Generate new password hash')

```

```

?></label>

        </div>
        <div class="form-group mx-sm-3 mb-2">
            <label for="inputPassword2" class="sr-only"><?php echo lng('Password')
?></label>

                <input type="text" class="form-control btn-sm" id="inputPassword2"
name="inputPassword2" placeholder="<?php echo lng('Password') ?>" required>
            </div>

                <button type="submit" class="btn btn-success btn-sm mb-2"><?php echo
lng('Generate') ?></button>
            </form>

            <textarea class="form-control" rows="2" readonly id="js-pwd-result"></textarea>
        </div>
    </div>
</div>
</div>
</div>
</div>
<?php
    fm_show_footer();
    exit;
}

// file viewer
if (isset($_GET['view'])) {
    $file = $_GET['view'];
    $file = fm_clean_path($file, false);
    $file = str_replace('/', '', $file);
    if ($file == "" || !is_file($path . '/' . $file) || !fm_is_exclude_items($file, $path . '/' . $file)) {
        fm_set_msg(lng('File not found'), 'error');
        $FM_PATH = FM_PATH;
        fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
    }

    fm_show_header(); // HEADER
    fm_show_nav_path(FM_PATH); // current path

    $file_url = FM_ROOT_URL . fm_convert_win((FM_PATH != "" ? '/' . FM_PATH : "") . '/' . $file);
    $file_path = $path . '/' . $file;

```

```
$ext = strtolower(pathinfo($file_path, PATHINFO_EXTENSION));
$mime_type = fm_get_mime_type($file_path);
$filesize_raw = fm_get_size($file_path);
$filesize = fm_get_filesize($filesize_raw);

$sis_zip = false;
$sis_gzip = false;
$sis_image = false;
$sis_audio = false;
$sis_video = false;
$sis_text = false;
$sis_onlineViewer = false;

$view_title = 'File';
$filenames = false; // for zip
$content = ""; // for text
$online_viewer = strtolower(FM_DOC_VIEWER);

if ($online_viewer && $online_viewer !== 'false' && in_array($ext, fm_get_onlineViewer_exts())) {
    $sis_onlineViewer = true;
} elseif ($ext == 'zip' || $ext == 'tar') {
    $sis_zip = true;
    $view_title = 'Archive';
    $filenames = fm_get_zif_info($file_path, $ext);
} elseif (in_array($ext, fm_get_image_exts())) {
    $sis_image = true;
    $view_title = 'Image';
} elseif (in_array($ext, fm_get_audio_exts())) {
    $sis_audio = true;
    $view_title = 'Audio';
} elseif (in_array($ext, fm_get_video_exts())) {
    $sis_video = true;
    $view_title = 'Video';
} elseif (in_array($ext, fm_get_text_exts()) || substr($mime_type, 0, 4) == 'text' ||
in_array($mime_type, fm_get_text_mimes())) {
    $sis_text = true;
    $content = file_get_contents($file_path);
```

```

}

?>
<div class="row">
  <div class="col-12">
    <ul class="list-group w-50 my-3" data-bs-theme="<?php echo FM_THEME; ?>">
      <li class="list-group-item active" aria-current="true"><strong><?php echo lng($view_title)
?>:</strong> <?php echo fm_enc(fm_convert_win($file)) ?></li>
      <?php $display_path = fm_get_display_path($file_path); ?>
      <li class="list-group-item"><strong><?php echo $display_path['label']; ?>:</strong> <?php
echo $display_path['path']; ?></li>
      <li class="list-group-item"><strong><?php echo lng('Date Modified') ?>:</strong> <?php
echo date(FM_DATETIME_FORMAT, filemtime($file_path)); ?></li>
      <li class="list-group-item"><strong><?php echo lng('File size') ?>:</strong> <?php echo
($filesize_raw <= 1000) ? "$filesize_raw bytes" : $filesize; ?></li>
      <li class="list-group-item"><strong><?php echo lng('MIME-type') ?>:</strong> <?php echo
$mime_type ?></li>
      <?php
// ZIP info
if (($is_zip || $is_gzip) && $filenames !== false) {
  $total_files = 0;
  $total_comp = 0;
  $total_uncomp = 0;
  foreach ($filenames as $fn) {
    if (!$fn['folder']) {
      $total_files++;
    }
    $total_comp += $fn['compressed_size'];
    $total_uncomp += $fn['filesize'];
  }
  ?>
  <li class="list-group-item"><?php echo lng('Files in archive') ?>: <?php echo $total_files
?></li>
  <li class="list-group-item"><?php echo lng('Total size') ?>: <?php echo
fm_get_filesize($total_uncomp) ?></li>
  <li class="list-group-item"> <?php echo lng('Size in archive') ?>: <?php echo
fm_get_filesize($total_comp) ?></li>
  <li class="list-group-item"><?php echo lng('Compression') ?>: <?php echo

```

```

round(($total_comp / max($total_uncomp, 1)) * 100) ?>%</li>
    <?php
    }
    // Image info
    if ($is_image) {
        $image_size = getimagesize($file_path);
        echo '<li class="list-group-item"><strong>' . lng('Image size') . ' :</strong> ' .
(isset($image_size[0]) ? $image_size[0] : '0') . ' x ' . (isset($image_size[1]) ? $image_size[1] : '0') .
'</li>';
    }
    // Text info
    if ($is_text) {
        $is_utf8 = fm_is_utf8($content);
        if (function_exists('iconv')) {
            if (!$is_utf8) {
                $content = iconv(FM_ICONV_INPUT_ENC, 'UTF-8//IGNORE', $content);
            }
        }
        echo '<li class="list-group-item"><strong>' . lng('Charset') . ' :</strong> ' . ($is_utf8 ?
'utf-8' : '8 bit') . '</li>';
    }
    ?>
</ul>
<div class="btn-group btn-group-sm flex-wrap" role="group">
    <form method="post" class="d-inline mb-0 btn btn-outline-primary" action="?p=<?php
echo urlencode(FM_PATH) ?>&amp;dl=<?php echo urlencode($file) ?>">
        <input type="hidden" name="token" value="<?php echo $_SESSION['token']; ?>">
        <button type="submit" class="btn btn-link btn-sm text-decoration-none fw-bold p-0"><i
class="fa fa-cloud-download"></i> <?php echo lng('Download') ?></button> &nbsp;
    </form>
    <?php if (!FM_READONLY): ?>
        <a class="fw-bold btn btn-outline-primary" title="<?php echo lng('Delete') ?>"
href="?p=<?php echo urlencode(FM_PATH) ?>&amp;del=<?php echo urlencode($file) ?>"
onclick="confirmDailog(event, 1209, '<?php echo lng('Delete') . ' ' . lng('File'); ?>','<?php echo
urlencode($file); ?>', this.href);"> <i class="fa fa-trash"></i> Delete</a>
    <?php endif; ?>
    <a class="fw-bold btn btn-outline-primary" href="<?php echo fm_enc($file_url) ?>"
target="_blank"><i class="fa fa-external-link-square"></i> <?php echo lng('Open') ?></a></b>

```

```

<?php
// ZIP actions
if (!FM_READONLY && ($is_zip || $is_gzip) && $filenames !== false) {
    $zip_name = pathinfo($file_path, PATHINFO_FILENAME);
    ?>
    <form method="post" class="d-inline btn btn-outline-primary mb-0">
        <input type="hidden" name="token" value="<?php echo $_SESSION['token']; ?>">
        <input type="hidden" name="unzip" value="<?php echo urlencode($file); ?>">
        <button type="submit" class="btn btn-link text-decoration-none fw-bold p-0 border-0"
style="font-size: 14px;"><i class="fa fa-check-circle"></i> <?php echo lng('UnZip') ?></button>
    </form>
    <form method="post" class="d-inline btn btn-outline-primary mb-0">
        <input type="hidden" name="token" value="<?php echo $_SESSION['token']; ?>">
        <input type="hidden" name="unzip" value="<?php echo urlencode($file); ?>">
        <input type="hidden" name="tofolder" value="1">
        <button type="submit" class="btn btn-link text-decoration-none fw-bold p-0"
style="font-size: 14px;" title="UnZip to <?php echo fm_enc($zip_name) ?>"><i class="fa
fa-check-circle"></i> <?php echo lng('UnZipToFolder') ?></button>
    </form>
<?php
}
if ($is_text && !FM_READONLY) {
    ?>
        <a class="fw-bold btn btn-outline-primary" href="?p=<?php echo
urlencode(trim(FM_PATH)) ?>&edit=<?php echo urlencode($file) ?>" class="edit-file">
        <i class="fa fa-pencil-square"></i> <?php echo lng('Edit') ?>
    </a>
        <a class="fw-bold btn btn-outline-primary" href="?p=<?php echo
urlencode(trim(FM_PATH)) ?>&edit=<?php echo urlencode($file) ?>&env=ace"
        class="edit-file"><i class="fa fa-pencil-square"></i> <?php echo
lng('AdvancedEditor') ?>
    </a>
    <?php ?>
    <a class="fw-bold btn btn-outline-primary" href="?p=<?php echo urlencode(FM_PATH)
?>"><i class="fa fa-chevron-circle-left go-back"></i> <?php echo lng('Back') ?></a>
</div>
<div class="row mt-3">
    <?php

```

```

if ($is_onlineViewer) {
    if ($online_viewer == 'google') {
        echo '<iframe src="https://docs.google.com/viewer?embedded=true&hl=en&url=' .
fm_enc($file_url) . '" frameborder="no" style="width:100%;min-height:460px"></iframe>';
    } else if ($online_viewer == 'microsoft') {
        echo '<iframe src="https://view.officeapps.live.com/op/embed.aspx?src=' .
fm_enc($file_url) . '" frameborder="no" style="width:100%;min-height:460px"></iframe>';
    }
} elseif ($is_zip) {
    // ZIP content
    if ($filenames !== false) {
        echo '<code class="maxheight">';
        foreach ($filenames as $fn) {
            if ($fn['folder']) {
                echo '<b>' . fm_enc($fn['name']) . '</b><br>';
            } else {
                echo $fn['name'] . ' (' . fm_get_filesize($fn['filesize']) . ')<br>';
            }
        }
        echo '</code>';
    } else {
        echo '<p>' . lng('Error while fetching archive info') . '</p>';
    }
} elseif ($is_image) {
    // Image content
    if (in_array($ext, array('gif', 'jpg', 'jpeg', 'png', 'bmp', 'ico', 'svg', 'webp', 'avif'))) {
        echo '<p><input type="checkbox" id="preview-img-zoomCheck"><label
for="preview-img-zoomCheck"></label></p>';
    }
} elseif ($is_audio) {
    // Audio content
    echo '<p><audio src="' . fm_enc($file_url) . '" controls
preload="metadata"></audio></p>';
} elseif ($is_video) {
    // Video content
    echo '<div class="preview-video"><video src="' . fm_enc($file_url) . '" width="640"
height="360" controls preload="metadata"></video></div>';
}

```

```

    } elseif ($is_text) {
        if (FM_USE_HIGHLIGHTJS) {
            // highlight
            $hljs_classes = array(
                'shtml' => 'xml',
                'htaccess' => 'apache',
                'phtml' => 'php',
                'lock' => 'json',
                'svg' => 'xml',
            );
            $hljs_class = isset($hljs_classes[$ext]) ? 'lang-' . $hljs_classes[$ext] : 'lang-' . $ext;
            if (empty($ext) || in_array(strtolower($file), fm_get_text_names()) ||
preg_match('#\.(min\.(css|js)$#i', $file)) {
                $hljs_class = 'nohighlight';
            }

            $content = '<pre class="with-hljs"><code class="' . $hljs_class . '">' .
fm_enc($content) . '</code></pre>';
        } elseif (in_array($ext, array('php', 'php4', 'php5', 'phtml', 'phps'))) {
            // php highlight
            $content = highlight_string($content, true);
        } else {
            $content = '<pre>' . fm_enc($content) . '</pre>';
        }
        echo $content;
    }
?>
</div>
</div>
</div>
<?php
    fm_show_footer();
    exit;
}

```

// file editor

```

if (isset($_GET['edit']) && !FM_READONLY) {
    $file = $_GET['edit'];
    $file = fm_clean_path($file, false);
}

```

```

$file = str_replace('/', '', $file);
if ($file == "" || !is_file($path . '/' . $file) || !fm_is_exclude_items($file, $path . '/' . $file)) {
    fm_set_msg(lng('File not found'), 'error');
    $FM_PATH = FM_PATH;
    fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
}
$editFile = ' : <i><b>' . $file . '</b></i>';
header('X-XSS-Protection:0');
fm_show_header(); // HEADER
fm_show_nav_path(FM_PATH); // current path

$file_url = FM_ROOT_URL . fm_convert_win((FM_PATH != "" ? '/' . FM_PATH : "") . '/' . $file);
$file_path = $path . '/' . $file;

// normal editor
$isNormalEditor = true;
if (isset($_GET['env'])) {
    if ($_GET['env'] == "ace") {
        $isNormalEditor = false;
    }
}

// Save File
if (isset($_POST['savedata'])) {
    $writedata = $_POST['savedata'];
    $fd = fopen($file_path, "w");
    @fwrite($fd, $writedata);
    fclose($fd);
    fm_set_msg(lng('File Saved Successfully'));
}

$ext = strtolower(pathinfo($file_path, PATHINFO_EXTENSION));
$mime_type = fm_get_mime_type($file_path);
$filesize = filesize($file_path);
$is_text = false;
$content = ""; // for text

if (in_array($ext, fm_get_text_exts()) || substr($mime_type, 0, 4) == 'text' || in_array($mime_type,

```

```

fm_get_text_mimes())) {
    $is_text = true;
    $content = file_get_contents($file_path);
}

?>
<div class="path">
    <div class="row">
        <div class="col-xs-12 col-sm-5 col-lg-6 pt-1">
            <div class="btn-toolbar" role="toolbar">
                <?php if (!$isNormalEditor) { ?>
                    <div class="btn-group js-ace-toolbar">
                        <button data-cmd="none" data-option="fullscreen" class="btn btn-sm
                        btn-outline-secondary" id="js-ace-fullscreen" title="<?php echo lng('Fullscreen') ?>"><i class="fa
                        fa-expand" title="<?php echo lng('Fullscreen') ?>"></i></button>
                        <button data-cmd="find" class="btn btn-sm btn-outline-secondary"
                        id="js-ace-search" title="<?php echo lng('Search') ?>"><i class="fa fa-search" title="<?php echo
                        lng('Search') ?>"></i></button>
                        <button data-cmd="undo" class="btn btn-sm btn-outline-secondary"
                        id="js-ace-undo" title="<?php echo lng('Undo') ?>"><i class="fa fa-undo" title="<?php echo
                        lng('Undo') ?>"></i></button>
                        <button data-cmd="redo" class="btn btn-sm btn-outline-secondary"
                        id="js-ace-redo" title="<?php echo lng('Redo') ?>"><i class="fa fa-repeat" title="<?php echo
                        lng('Redo') ?>"></i></button>
                        <button data-cmd="none" data-option="wrap" class="btn btn-sm
                        btn-outline-secondary" id="js-ace-wordWrap" title="<?php echo lng('Word Wrap') ?>"><i class="fa
                        fa-text-width" title="<?php echo lng('Word Wrap') ?>"></i></button>
                        <select id="js-ace-mode" data-type="mode" title="<?php echo lng('Select
                        Document Type') ?>" class="btn-outline-secondary border-start-0 d-none d-md-block">
                            <option>-- <?php echo lng('Select Mode') ?> --</option>
                        </select>
                        <select id="js-ace-theme" data-type="theme" title="<?php echo lng('Select
                        Theme') ?>" class="btn-outline-secondary border-start-0 d-none d-lg-block">
                            <option>-- <?php echo lng('Select Theme') ?> --</option>
                        </select>
                        <select id="js-ace-fontSize" data-type="fontSize" title="<?php echo lng('Select
                        Font Size') ?>" class="btn-outline-secondary border-start-0 d-none d-lg-block">
                            <option>-- <?php echo lng('Select Font Size') ?> --</option>

```

```

        </select>
    </div>
    <?php } ?>
</div>
</div>
<div class="edit-file-actions col-xs-12 col-sm-7 col-lg-6 text-end pt-1">
    <div class="btn-group">
        <a title=" <?php echo lng('Back') ?>" class="btn btn-sm btn-outline-primary"
href="?p=<?php echo urlencode(trim(FM_PATH)) ?>&view=<?php echo urlencode($file)
?>"><i class="fa fa-reply-all"></i> <?php echo lng('Back') ?></a>

        <a title="<?php echo lng('BackUp') ?>" class="btn btn-sm btn-outline-primary"
href="javascript:void(0);" onclick="backup('<?php echo urlencode(trim(FM_PATH)) ?>', '<?php echo
urlencode($file) ?>')"><i class="fa fa-database"></i> <?php echo lng('BackUp') ?></a>

        <?php if ($is_text) { ?>
            <?php if ($isNormalEditor) { ?>
                <a title="Advanced" class="btn btn-sm btn-outline-primary" href="?p=<?php echo
urlencode(trim(FM_PATH)) ?>&edit=<?php echo urlencode($file) ?>&env=ace"><i
class="fa fa-pencil-square-o"></i> <?php echo lng('AdvancedEditor') ?></a>

                <button type="button" class="btn btn-sm btn-success" name="Save"
data-url="<?php echo fm_enc($file_url) ?>" onclick="edit_save(this, 'nrl')"><i class="fa
fa-floppy-o"></i> Save

                </button>
            <?php } else { ?>
                <a title="Plain Editor" class="btn btn-sm btn-outline-primary" href="?p=<?php echo
urlencode(trim(FM_PATH)) ?>&edit=<?php echo urlencode($file) ?>"><i class="fa
fa-text-height"></i> <?php echo lng('NormalEditor') ?></a>

                <button type="button" class="btn btn-sm btn-success" name="Save"
data-url="<?php echo fm_enc($file_url) ?>" onclick="edit_save(this, 'ace')"><i class="fa
fa-floppy-o"></i> <?php echo lng('Save') ?>

                </button>
            <?php } ?>
        <?php } ?>
    </div>
</div>
</div>
<?php
if ($is_text && $isNormalEditor) {
    echo '<textarea class="mt-2" id="normal-editor" rows="33" cols="120" style="width:

```

```

99.5%;">' . htmlspecialchars($content) . '</textarea>';
        echo '<script>document.addEventListener("keydown", function(e) {if
((window.navigator.platform.match("Mac") ? e.metaKey : e.ctrlKey) && e.keyCode == 83) {
e.preventDefault();edit_save(this,"nrl");}}, false);</script>';
    } elseif ($is_text) {
        echo '<div id="editor" contenteditable="true">' . htmlspecialchars($content) . '</div>';
    } else {
        fm_set_msg(lng('FILE EXTENSION HAS NOT SUPPORTED'), 'error');
    }
?>
</div>
<?php
    fm_show_footer();
    exit;
}

// chmod (not for Windows)
if (isset($_GET['chmod']) && !FM_READONLY && !FM_IS_WIN) {
    $file = $_GET['chmod'];
    $file = fm_clean_path($file);
    $file = str_replace('/', '', $file);
    if ($file == "" || (!is_file($path . '/' . $file) && !is_dir($path . '/' . $file))) {
        fm_set_msg(lng('File not found'), 'error');
        $FM_PATH = FM_PATH;
        fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
    }

    fm_show_header(); // HEADER
    fm_show_nav_path(FM_PATH); // current path

    $file_url = FM_ROOT_URL . (FM_PATH != "" ? '/' . FM_PATH : "") . '/' . $file;
    $file_path = $path . '/' . $file;

    $mode = fileperms($path . '/' . $file);
?>
<div class="path">
    <div class="card mb-2" data-bs-theme="<?php echo FM_THEME; ?>">
        <h6 class="card-header">

```

```

        <?php echo lng('ChangePermissions') ?>
    </h6>
    <div class="card-body">
        <p class="card-text">
            <?php $display_path = fm_get_display_path($file_path); ?>
            <?php echo $display_path['label']; ?>: <?php echo $display_path['path']; ?><br>
        </p>
        <form action="" method="post">
            <input type="hidden" name="p" value="<?php echo fm_enc(FM_PATH) ?>">
            <input type="hidden" name="chmod" value="<?php echo fm_enc($file) ?>">

            <table class="table compact-table" data-bs-theme="<?php echo FM_THEME; ?>">
                <tr>
                    <td></td>
                    <td><b><?php echo lng('Owner') ?></b></td>
                    <td><b><?php echo lng('Group') ?></b></td>
                    <td><b><?php echo lng('Other') ?></b></td>
                </tr>
                <tr>
                    <td style="text-align: right"><b><?php echo lng('Read') ?></b></td>
                    <td><label><input type="checkbox" name="ur" value="1" <?php echo ($mode &
00400) ? ' checked' : " ?>></label></td>
                    <td><label><input type="checkbox" name="gr" value="1" <?php echo ($mode &
00040) ? ' checked' : " ?>></label></td>
                    <td><label><input type="checkbox" name="or" value="1" <?php echo ($mode &
00004) ? ' checked' : " ?>></label></td>
                </tr>
                <tr>
                    <td style="text-align: right"><b><?php echo lng('Write') ?></b></td>
                    <td><label><input type="checkbox" name="uw" value="1" <?php echo ($mode &
00200) ? ' checked' : " ?>></label></td>
                    <td><label><input type="checkbox" name="gw" value="1" <?php echo ($mode &
00020) ? ' checked' : " ?>></label></td>
                    <td><label><input type="checkbox" name="ow" value="1" <?php echo ($mode &
00002) ? ' checked' : " ?>></label></td>
                </tr>
                <tr>
                    <td style="text-align: right"><b><?php echo lng('Execute') ?></b></td>

```

```

        <td><label><input type="checkbox" name="ux" value="1" <?php echo ($mode &
00100) ? ' checked' : " ?>></label></td>

        <td><label><input type="checkbox" name="gx" value="1" <?php echo ($mode &
00010) ? ' checked' : " ?>></label></td>

        <td><label><input type="checkbox" name="ox" value="1" <?php echo ($mode &
00001) ? ' checked' : " ?>></label></td>

    </tr>
</table>

<p>
    <input type="hidden" name="token" value="<?php echo $_SESSION['token']; ?>">
        <b><a href="?p=<?php echo urlencode(FM_PATH) ?>" class="btn
btn-outline-primary"><i class="fa fa-times-circle"></i> <?php echo lng('Cancel') ?></a></b>&nbsp;
        <button type="submit" class="btn btn-success"><i class="fa fa-check-circle"></i>
<?php echo lng('Change') ?></button>
    </p>
</form>
</div>
</div>
</div>
<?php
    fm_show_footer();
    exit;
}

// --- TINYFILEMANAGER MAIN ---
fm_show_header(); // HEADER
fm_show_nav_path(FM_PATH); // current path

// show alert messages
fm_show_message();

$num_files = count($files);
$num_folders = count($folders);
$all_files_size = 0;
?>
<form action="" method="post" class="pt-3">
    <input type="hidden" name="p" value="<?php echo fm_enc(FM_PATH) ?>">

```

```

<input type="hidden" name="group" value="1">
<input type="hidden" name="token" value="<?php echo $_SESSION['token']; ?>">
<div class="table-responsive">
    <table class="table table-bordered table-hover table-sm" id="main-table"
data-bs-theme="<?php echo FM_THEME; ?>">
    <thead class="thead-white">
        <tr>
            <?php if (!FM_READONLY): ?>
                <th style="width:3%" class="custom-checkbox-header">
                    <div class="custom-control custom-checkbox">
                        <input type="checkbox" class="custom-control-input" id="js-select-all-items"
onclick="checkbox_toggle()">
                        <label class="custom-control-label" for="js-select-all-items"></label>
                    </div>
                </th><?php endif; ?>
                <th><?php echo lng('Name') ?></th>
                <th><?php echo lng('Size') ?></th>
                <th><?php echo lng('Modified') ?></th>
                <?php if (!FM_IS_WIN && !$hide_Cols): ?>
                    <th><?php echo lng('Perms') ?></th>
                    <th><?php echo lng('Owner') ?></th><?php endif; ?>
                <th><?php echo lng('Actions') ?></th>
            </tr>
        </thead>
        <?php
        // link to parent folder
        if ($parent !== false) {
            ?>
            <tr><?php if (!FM_READONLY): ?>
                <td class="nosort"></td><?php endif; ?>
                <td class="border-0" data-sort><a href="?p=<?php echo urlencode($parent) ?>"><i
class="fa fa-chevron-circle-left go-back"></i> ..</a></td>
                <td class="border-0" data-order></td>
                <td class="border-0" data-order></td>
                <td class="border-0"></td>
                <?php if (!FM_IS_WIN && !$hide_Cols) { ?>
                    <td class="border-0"></td>
                    <td class="border-0"></td>

```

```

        <?php } ?>
    </tr>
<?php
}
$ii = 3399;
foreach ($folders as $f) {
    $is_link = is_link($path . '/' . $f);
    $img = $is_link ? 'icon-link_folder' : 'fa fa-folder-o';
    $modif_raw = filemtime($path . '/' . $f);
    $modif = date(FM_DATETIME_FORMAT, $modif_raw);
    $date_sorting = strtotime(date("F d Y H:i:s.", $modif_raw));
    $filesize_raw = "";
    $filesize = lng('Folder');
    $perms = substr(decoct(fileperms($path . '/' . $f)), -4);
    $owner = array('name' => '?');
    $group = array('name' => '?');
    if (function_exists('posix_getpwuid') && function_exists('posix_getgrgid')) {
        try {
            $owner_id = fileowner($path . '/' . $f);
            if ($owner_id != 0) {
                $owner_info = posix_getpwuid($owner_id);
                if ($owner_info) {
                    $owner = $owner_info;
                }
            }
            $group_id = filegroup($path . '/' . $f);
            $group_info = posix_getgrgid($group_id);
            if ($group_info) {
                $group = $group_info;
            }
        } catch (Exception $e) {
            error_log("exception:" . $e->getMessage());
        }
    }
}
?>
<tr>
    <?php if (!FM_READONLY): ?>
        <td class="custom-checkbox-td">

```

```

        <div class="custom-control custom-checkbox">
            <input type="checkbox" class="custom-control-input" id="<?php echo $ii ?>"
name="file[]" value="<?php echo fm_enc($f) ?>">
            <label class="custom-control-label" for="<?php echo $ii ?>"></label>
        </div>
    </td>
<?php endif; ?>
<td data-sort="<?php echo fm_convert_win(fm_enc($f)) ?>">
    <div class="filename">
        <a href="?p=<?php echo urlencode(trim(FM_PATH . '/' . $f, '/')) ?>"><i
class="<?php echo $img ?>"></i> <?php echo fm_convert_win(fm_enc($f)) ?></a>
        <?php echo ($is_link ? ' &arr; <i>' . readlink($path . '/' . $f) . '</i>' : '') ?>
    </div>
</td>
<td data-order="a-<?php echo str_pad($filesize_raw, 18, "0", STR_PAD_LEFT); ?>">
    <?php echo $filesize; ?>
</td>
<td data-order="a-<?php echo $date_sorting; ?>"><?php echo $modif ?></td>
<?php if (!FM_IS_WIN && !$hide_Cols): ?>
    <td>
        <?php if (!FM_READONLY): ?><a title="Change Permissions" href="?p=<?php
echo urlencode(FM_PATH) ?>&amp;chmod=<?php echo urlencode($f) ?>"><?php echo $perms
?></a><?php else: ?><?php echo $perms ?><?php endif; ?>
    </td>
    <td>
        <?php echo $owner['name'] . ':' . $group['name'] ?>
    </td>
<?php endif; ?>
<td class="inline-actions"><?php if (!FM_READONLY): ?>
    <a title="<?php echo lng('Delete') ?>" href="?p=<?php echo urlencode(FM_PATH)
?>&amp;del=<?php echo urlencode($f) ?>" onclick="confirmDailog(event, '1028','<?php echo
lng('Delete') . ' ' . lng('Folder'); ?>','<?php echo urlencode($f) ?>', this.href);"> <i class="fa fa-trash-o"
aria-hidden="true"></i></a>
    <a title="<?php echo lng('Rename') ?>" href="#" onclick="rename('<?php echo
fm_enc(addslashes(FM_PATH)) ?>', '<?php echo fm_enc(addslashes($f)) ?>');return false;"><i
class="fa fa-pencil-square-o" aria-hidden="true"></i></a>
    <a title="<?php echo lng('CopyTo') ?>..." href="?p=&amp;copy=<?php echo
urlencode(trim(FM_PATH . '/' . $f, '/')) ?>"><i class="fa fa-files-o" aria-hidden="true"></i></a>

```

```

        <?php endif; ?>
        <a title="<?php echo lng('DirectLink') ?>" href="<?php echo fm_enc(FM_ROOT_URL
. (FM_PATH != " ? '/' . FM_PATH : ") . '/' . $f . '/') ?>" target="_blank"><i class="fa fa-link"
aria-hidden="true"></i></a>
    </td>
</tr>
<?php
    flush();
    $ii++;
}
$ik = 8002;
foreach ($files as $f) {
    $is_link = is_link($path . '/' . $f);
    $img = $is_link ? 'fa fa-file-text-o' : fm_get_file_icon_class($path . '/' . $f);
    $modif_raw = filemtime($path . '/' . $f);
    $modif = date(FM_DATETIME_FORMAT, $modif_raw);
    $date_sorting = strtotime(date("F d Y H:i:s.", $modif_raw));
    $filesize_raw = fm_get_size($path . '/' . $f);
    $filesize = fm_get_filesize($filesize_raw);
    $filelink = '?p=' . urlencode(FM_PATH) . '&view=' . urlencode($f);
    $all_files_size += $filesize_raw;
    $perms = substr(decoct(fileperms($path . '/' . $f)), -4);
    $owner = array('name' => '?');
    $group = array('name' => '?');
    if (function_exists('posix_getpwuid') && function_exists('posix_getgrgid')) {
        try {
            $owner_id = fileowner($path . '/' . $f);
            if ($owner_id != 0) {
                $owner_info = posix_getpwuid($owner_id);
                if ($owner_info) {
                    $owner = $owner_info;
                }
            }
            $group_id = filegroup($path . '/' . $f);
            $group_info = posix_getgrgid($group_id);
            if ($group_info) {
                $group = $group_info;
            }
        }
    }
}

```

```

    } catch (Exception $e) {
        error_log("exception:" . $e->getMessage());
    }
}
?>
<tr>
    <?php if (!FM_READONLY): ?>
        <td class="custom-checkbox-td">
            <div class="custom-control custom-checkbox">
                <input type="checkbox" class="custom-control-input" id="<?php echo $ik ?>"
name="file[]" value="<?php echo fm_enc($f) ?>">
                <label class="custom-control-label" for="<?php echo $ik ?>"></label>
            </div>
        </td><?php endif; ?>
        <td data-sort="<?php echo fm_enc($f) ?>">
            <div class="filename">
                <?php
                    if (in_array(strtolower(pathinfo($f, PATHINFO_EXTENSION)), array('gif', 'jpg',
'jpeg', 'png', 'bmp', 'ico', 'svg', 'webp', 'avif'))): ?>
                        <?php $imagePreview = fm_enc(FM_ROOT_URL . (FM_PATH != " ? '/' .
FM_PATH : ") . '/' . $f); ?>
                        <a href="<?php echo $filelink ?>" data-preview-image="<?php echo
$imagePreview ?>" title="<?php echo fm_enc($f) ?>">
                            <?php else: ?>
                                <a href="<?php echo $filelink ?>" title="<?php echo $f ?>">
                                    <?php endif; ?>
                                    <i class="<?php echo $img ?>"></i> <?php echo fm_convert_win(fm_enc($f))
?>
                                </a>
                                <?php echo ($is_link ? ' &rarr; <i>' . readlink($path . '/' . $f) . '</i>' : '') ?>
                            </div>
                        </td>
                            <td data-order="b-<?php echo str_pad($filesize_raw, 18, "0", STR_PAD_LEFT);
?>"><span title="<?php printf('%s bytes', $filesize_raw) ?>">
                                <?php echo $filesize; ?>
                            </span></td>
                            <td data-order="b-<?php echo $date_sorting; ?>"><?php echo $modif ?></td>
                            <?php if (!FM_IS_WIN && !$hide_Cols): ?>

```

```
 <?php if (!FM_READONLY): ?><a title="<?php echo 'Change Permissions' ?>" href="?p=<?php echo urlencode(FM_PATH) ?>&amp;chmod=<?php echo urlencode($f) ?>"><?php echo $perms ?></a><?php else: ?><?php echo $perms ?><?php endif; ?> </td>  <?php echo fm_enc($owner['name'] . ':' . $group['name']) ?></td> <?php endif; ?>  </td><?php endif; ?>   | | | |
```

```

        </tr>
    </tfoot>
<?php
} else { ?>
    <tfoot>
        <tr>
            <td class="gray fs-7" colspan="<?php echo (!FM_IS_WIN && !$hide_Cols) ?
(FM_READONLY ? '6' : '7') : (FM_READONLY ? '4' : '5') ?>">
                <?php echo lng('FullSize') . '': <span class="badge text-bg-light border-radius-0">' .
fm_get_filesize($all_files_size) . '</span>' ?>
                <?php echo lng('File') . '': <span class="badge text-bg-light border-radius-0">' .
$num_files . '</span>' ?>
                <?php echo lng('Folder') . '': <span class="badge text-bg-light border-radius-0">' .
$num_folders . '</span>' ?>
            </td>
        </tr>
    </tfoot>
<?php } ?>
</table>
</div>

<div class="row">
    <?php if (!FM_READONLY): ?>
        <div class="col-xs-12 col-sm-9">
            <div class="btn-group flex-wrap" data-toggle="buttons" role="toolbar">
                <a href="#/select-all" class="btn btn-small btn-outline-primary btn-2"
onclick="select_all();return false;"><i class="fa fa-check-square"></i> <?php echo lng('SelectAll') ?>
</a>
                <a href="#/unselect-all" class="btn btn-small btn-outline-primary btn-2"
onclick="unselect_all();return false;"><i class="fa fa-window-close"></i> <?php echo
lng('UnSelectAll') ?> </a>
                <a href="#/invert-all" class="btn btn-small btn-outline-primary btn-2"
onclick="invert_all();return false;"><i class="fa fa-th-list"></i> <?php echo lng('InvertSelection') ?>
</a>
                <input type="submit" class="hidden" name="delete" id="a-delete" value="Delete"
onclick="return confirm('<?php echo lng('Delete selected files and folders?'); ?>')">
                <a href="javascript:document.getElementById('a-delete').click();" class="btn btn-small
btn-outline-primary btn-2"><i class="fa fa-trash"></i> <?php echo lng('Delete') ?> </a>
            </div>
        </div>
    </if>
</div>

```

```

        <input type="submit" class="hidden" name="zip" id="a-zip" value="zip" onclick="return
confirm('<?php echo lng('Create archive?'); ?>')">
        <a href="javascript:document.getElementById('a-zip').click();" class="btn btn-small
btn-outline-primary btn-2"><i class="fa fa-file-archive-o"></i> <?php echo lng('Zip') ?> </a>
        <input type="submit" class="hidden" name="tar" id="a-tar" value="tar" onclick="return
confirm('<?php echo lng('Create archive?'); ?>')">
        <a href="javascript:document.getElementById('a-tar').click();" class="btn btn-small
btn-outline-primary btn-2"><i class="fa fa-file-archive-o"></i> <?php echo lng('Tar') ?> </a>
        <input type="submit" class="hidden" name="copy" id="a-copy" value="Copy">
        <a href="javascript:document.getElementById('a-copy').click();" class="btn btn-small
btn-outline-primary btn-2"><i class="fa fa-files-o"></i> <?php echo lng('Copy') ?> </a>
    </div>
</div>
    <div class="col-3 d-none d-sm-block"><a href="https://tinyfilemanager.github.io"
target="_blank" class="float-right text-muted">Tiny File Manager <?php echo VERSION;
?></a></div>
    <?php else: ?>
        <div class="col-12"><a href="https://tinyfilemanager.github.io" target="_blank"
class="float-right text-muted">Tiny File Manager <?php echo VERSION; ?></a></div>
    <?php endif; ?>
</div>
</form>

<?php
fm_show_footer();

// --- END HTML ---

// Functions

/**
 * It prints the css/js files into html
 * @param key The key of the external file to print.
 */
function print_external($key)
{
    global $external;

```

```

if (!array_key_exists($key, $external)) {
    // throw new Exception('Key missing in external: ' . key);
    echo "<!-- EXTERNAL: MISSING KEY $key -->";
    return;
}

echo "$external[$key]";
}

/**
 * Verify CSRF TOKEN and remove after certified
 * @param string $token
 * @return bool
 */
function verifyToken($token)
{
    if (hash_equals($_SESSION['token'], $token)) {
        return true;
    }
    return false;
}

/**
 * Delete file or folder (recursively)
 * @param string $path
 * @return bool
 */
function fm_rdelete($path)
{
    if (is_link($path)) {
        return unlink($path);
    } elseif (is_dir($path)) {
        $objects = scandir($path);
        $ok = true;
        if (is_array($objects)) {
            foreach ($objects as $file) {
                if ($file != '.' && $file != '..') {
                    if (!fm_rdelete($path . '/' . $file)) {

```

```

        $ok = false;
    }
}
}
}
return ($ok) ? rmdir($path) : false;
} elseif (is_file($path)) {
    return unlink($path);
}
return false;
}

```

```
/**
```

```
 * Recursive chmod
```

```
 * @param string $path
```

```
 * @param int $filemode
```

```
 * @param int $dirmode
```

```
 * @return bool
```

```
 * @todo Will use in mass chmod
```

```
 */
```

```
function fm_rchmod($path, $filemode, $dirmode)
```

```

{
    if (is_dir($path)) {
        if (!chmod($path, $dirmode)) {
            return false;
        }
        $objects = scandir($path);
        if (is_array($objects)) {
            foreach ($objects as $file) {
                if ($file != '.' && $file != '..') {
                    if (!fm_rchmod($path . '/' . $file, $filemode, $dirmode)) {
                        return false;
                    }
                }
            }
        }
    }
    return true;
} elseif (is_link($path)) {

```

```

        return true;
    } elseif (is_file($path)) {
        return chmod($path, $filemode);
    }
    return false;
}

```

```

/**
 * Check the file extension which is allowed or not
 * @param string $filename
 * @return bool
 */
function fm_is_valid_ext($filename)
{
    $allowed = (FM_FILE_EXTENSION) ? explode(',', FM_FILE_EXTENSION) : false;

    $ext = pathinfo($filename, PATHINFO_EXTENSION);
    $isFileAllowed = ($allowed) ? in_array($ext, $allowed) : true;

    return ($isFileAllowed) ? true : false;
}

```

```

/**
 * Safely rename
 * @param string $old
 * @param string $new
 * @return bool|null
 */
function fm_rename($old, $new)
{
    $isFileAllowed = fm_is_valid_ext($new);

    if (!is_dir($old)) {
        if (!$isFileAllowed) return false;
    }

    return (!file_exists($new) && file_exists($old)) ? rename($old, $new) : null;
}

```

```
/**
 * Copy file or folder (recursively).
 * @param string $path
 * @param string $dest
 * @param bool $upd Update files
 * @param bool $force Create folder with same names instead file
 * @return bool
 */
```

```
function fm_rcopy($path, $dest, $upd = true, $force = true)
{
    if (!is_dir($path) && !is_file($path)) {
        return false;
    }

    if (is_dir($path)) {
        if (!fm_mkdir($dest, $force)) {
            return false;
        }

        $objects = array_diff(scandir($path), ['.', '..']);

        foreach ($objects as $file) {
            if (!fm_rcopy("$path/$file", "$dest/$file", $upd, $force)) {
                return false;
            }
        }

        return true;
    }

    // Handle file copying
    return fm_copy($path, $dest, $upd);
}
```

```
/**
 * Safely create folder
```

```

* @param string $dir
* @param bool $force
* @return bool
*/
function fm_mkdir($dir, $force)
{
    if (file_exists($dir)) {
        if (is_dir($dir)) {
            return $dir;
        } elseif (!$force) {
            return false;
        }
        unlink($dir);
    }
    return mkdir($dir, 0777, true);
}

/**
* Safely copy file
* @param string $f1
* @param string $f2
* @param bool $upd Indicates if file should be updated with new content
* @return bool
*/
function fm_copy($f1, $f2, $upd)
{
    $time1 = filemtime($f1);
    if (file_exists($f2)) {
        $time2 = filemtime($f2);
        if ($time2 >= $time1 && $upd) {
            return false;
        }
    }
    $ok = copy($f1, $f2);
    if ($ok) {
        touch($f2, $time1);
    }
    return $ok;
}

```

```
}
```

```
/**
```

```
 * Get mime type
```

```
 * @param string $file_path
```

```
 * @return mixed|string
```

```
 */
```

```
function fm_get_mime_type($file_path)
```

```
{
```

```
    if (function_exists('finfo_open')) {
```

```
        $finfo = finfo_open(FILEINFO_MIME_TYPE);
```

```
        $mime = finfo_file($finfo, $file_path);
```

```
        finfo_close($finfo);
```

```
        return $mime;
```

```
    } elseif (function_exists('mime_content_type')) {
```

```
        return mime_content_type($file_path);
```

```
    } elseif (!strpos(ini_get('disable_functions'), 'shell_exec')) {
```

```
        $file = escapeshellarg($file_path);
```

```
        $mime = shell_exec('file -bi ' . $file);
```

```
        return $mime;
```

```
    } else {
```

```
        return '--';
```

```
    }
```

```
}
```

```
/**
```

```
 * HTTP Redirect
```

```
 * @param string $url
```

```
 * @param int $code
```

```
 */
```

```
function fm_redirect($url, $code = 302)
```

```
{
```

```
    header('Location: ' . $url, true, $code);
```

```
    exit;
```

```
}
```

```
/**
```

```
 * Path traversal prevention and clean the url
```

* It replaces (consecutive) occurrences of / and \\ with whatever is in DIRECTORY_SEPARATOR, and processes /. and ../ fine.

* @param \$path

* @return string

*/

```
function get_absolute_path($path)
```

```
{
```

```
    $path = str_replace(array('/', '\\'), DIRECTORY_SEPARATOR, $path);
```

```
    $parts = array_filter(explode(DIRECTORY_SEPARATOR, $path), 'strlen');
```

```
    $absolutes = array();
```

```
    foreach ($parts as $part) {
```

```
        if ('.' == $part) continue;
```

```
        if ('..' == $part) {
```

```
            array_pop($absolutes);
```

```
        } else {
```

```
            $absolutes[] = $part;
```

```
        }
```

```
    }
```

```
    return implode(DIRECTORY_SEPARATOR, $absolutes);
```

```
}
```

```
/**
```

```
 * Clean path
```

```
 * @param string $path
```

```
 * @return string
```

```
*/
```

```
function fm_clean_path($path, $trim = true)
```

```
{
```

```
    $path = $trim ? trim($path) : $path;
```

```
    $path = trim($path, '\\');
```

```
    $path = str_replace(array('../', '..\\'), '', $path);
```

```
    $path = get_absolute_path($path);
```

```
    if ($path == '..') {
```

```
        $path = '';
```

```
    }
```

```
    return str_replace('\\', '/', $path);
```

```
}
```

```

/**
 * Get parent path
 * @param string $path
 * @return bool|string
 */
function fm_get_parent_path($path)
{
    $path = fm_clean_path($path);
    if ($path != "") {
        $array = explode('/', $path);
        if (count($array) > 1) {
            $array = array_slice($array, 0, -1);
            return implode('/', $array);
        }
        return "";
    }
    return false;
}

function fm_get_display_path($file_path)
{
    global $path_display_mode, $root_path, $root_url;
    switch ($path_display_mode) {
        case 'relative':
            return array(
                'label' => 'Path',
                'path' => fm_enc(fm_convert_win(str_replace($root_path, "", $file_path)))
            );
        case 'host':
            $relative_path = str_replace($root_path, "", $file_path);
            return array(
                'label' => 'Host Path',
                'path' => fm_enc(fm_convert_win('/') . $root_url . '/' . ltrim(str_replace("\\", '/', $relative_path), '/'))
            );
        case 'full':
        default:
            return array(

```

```

        'label' => 'Full Path',
        'path' => fm_enc(fm_convert_win($file_path))
    );
}
}

/**
 * Check file is in exclude list
 * @param string $name The name of the file/folder
 * @param string $path The full path of the file/folder
 * @return bool
 */
function fm_is_exclude_items($name, $path)
{
    $ext = strtolower(pathinfo($name, PATHINFO_EXTENSION));
    if (isset($exclude_items) and sizeof($exclude_items)) {
        unset($exclude_items);
    }

    $exclude_items = FM_EXCLUDE_ITEMS;
    if (version_compare(PHP_VERSION, '7.0.0', '<')) {
        $exclude_items = unserialize($exclude_items);
    }
    if (!in_array($name, $exclude_items) && !in_array("*.{$ext}", $exclude_items) && !in_array($path,
    $exclude_items)) {
        return true;
    }
    return false;
}

/**
 * get language translations from json file
 * @param int $tr
 * @return array
 */
function fm_get_translations($tr)
{
    try {

```

```

$content = @file_get_contents('translation.json');
if ($content !== FALSE) {
    $lng = json_decode($content, TRUE);
    global $lang_list;
    foreach ($lng["language"] as $key => $value) {
        $code = $value["code"];
        $lang_list[$code] = $value["name"];
        if ($tr)
            $tr[$code] = $value["translation"];
    }
    return $tr;
}
} catch (Exception $e) {
    echo $e;
}
}

/**
 * @param string $file
 * Recover all file sizes larger than > 2GB.
 * Works on php 32bits and 64bits and supports linux
 * @return int|string
 */
function fm_get_size($file)
{
    static $iswin = null;
    static $isdarwin = null;
    static $exec_works = null;

    // Set static variables once
    if ($iswin === null) {
        $iswin = strtoupper(substr(PHP_OS, 0, 3)) === 'WIN';
        $isdarwin = strtoupper(PHP_OS) === 'DARWIN';
        $exec_works = function_exists('exec') && ini_get('safe_mode') && @exec('echo EXEC') ===
'EXEC';
    }

    // Attempt shell command if exec is available

```

```

if ($exec_works) {
    $arg = escapeshellarg($file);
    $cmd = $iswin ? "for %F in (\\"$file\\") do @echo %~zF" : ($isdarwin ? "stat -f%z $arg" : "stat
-c%z $arg");
    @exec($cmd, $output);

    if (!empty($output) && ctype_digit($size = trim(implode("\n", $output)))) {
        return $size;
    }
}

```

// Attempt Windows COM interface for Windows systems

```

if ($iswin && class_exists('COM')) {
    try {
        $fsobj = new COM('Scripting.FileSystemObject');
        $f = $fsobj->GetFile(realpath($file));
        if (ctype_digit($size = $f->Size)) {
            return $size;
        }
    } catch (Exception $e) {
        // COM failed, fallback to filesize
    }
}

```

// Default to PHP's filesize function

```

return filesize($file);
}

```

/**

* Get nice filesize

* @param int \$size

* @return string

*/

function fm_get_filesize(\$size)

```

{
    $size = (float) $size;
    $units = array('B', 'KB', 'MB', 'GB', 'TB', 'PB', 'EB', 'ZB', 'YB');

```

```

$power = ($size > 0) ? floor(log($size, 1024)) : 0;
$power = ($power > (count($units) - 1)) ? (count($units) - 1) : $power;
return sprintf('%s %s', round($size / pow(1024, $power), 2), $units[$power]);
}

/**
 * Get info about zip archive
 * @param string $path
 * @return array|bool
 */
function fm_get_zip_info($path, $ext)
{
    if ($ext == 'zip' && function_exists('zip_open')) {
        $arch = @zip_open($path);
        if ($arch) {
            $filenames = array();
            while ($zip_entry = @zip_read($arch)) {
                $zip_name = @zip_entry_name($zip_entry);
                $zip_folder = substr($zip_name, -1) == '/';
                $filenames[] = array(
                    'name' => $zip_name,
                    'filesize' => @zip_entry_filesize($zip_entry),
                    'compressed_size' => @zip_entry_compressedsize($zip_entry),
                    'folder' => $zip_folder
                    // 'compression_method' => zip_entry_compressionmethod($zip_entry),
                );
            }
            @zip_close($arch);
            return $filenames;
        }
    }
    elseif ($ext == 'tar' && class_exists('PharData')) {
        $archive = new PharData($path);
        $filenames = array();
        foreach (new RecursiveIteratorIterator($archive) as $file) {
            $parent_info = $file->getPathInfo();
            $zip_name = str_replace("phar://" . $path, "", $file->getPathName());
            $zip_name = substr($zip_name, ($pos = strpos($zip_name, '/')) !== false ? $pos + 1 : 0);
            $zip_folder = $parent_info->getFileName();
        }
    }
}

```

```

$zip_info = new SplFileInfo($file);
$filenames[] = array(
    'name' => $zip_name,
    'filesize' => $zip_info->getSize(),
    'compressed_size' => $file->getCompressedSize(),
    'folder' => $zip_folder
);
}
return $filenames;
}
return false;
}

```

```

/**
 * Encode html entities
 * @param string $text
 * @return string
 */
function fm_enc($text)
{
    return htmlspecialchars($text, ENT_QUOTES, 'UTF-8');
}

```

```

/**
 * Prevent XSS attacks
 * @param string $text
 * @return string
 */
function fm_isvalid_filename($text)
{
    return (strpos($text, '/?%*:"<>') === FALSE) ? true : false;
}

```

```

/**
 * Save message in session
 * @param string $msg
 * @param string $status
 */

```

```

function fm_set_msg($msg, $status = 'ok')
{
    $_SESSION[FM_SESSION_ID]['message'] = $msg;
    $_SESSION[FM_SESSION_ID]['status'] = $status;
}

/**
 * Check if string is in UTF-8
 * @param string $string
 * @return int
 */
function fm_is_utf8($string)
{
    return preg_match('//u', $string);
}

/**
 * Convert file name to UTF-8 in Windows
 * @param string $filename
 * @return string
 */
function fm_convert_win($filename)
{
    if (FM_IS_WIN && function_exists('iconv')) {
        $filename = iconv(FM_ICONV_INPUT_ENC, 'UTF-8//IGNORE', $filename);
    }
    return $filename;
}

/**
 * @param $obj
 * @return array
 */
function fm_object_to_array($obj)
{
    if (is_object($obj) && !is_array($obj)) {
        return $obj;
    }
}

```

```

if (is_object($obj)) {
    $obj = get_object_vars($obj);
}
return array_map('fm_object_to_array', $obj);
}

/**
 * Get CSS classname for file
 * @param string $path
 * @return string
 */
function fm_get_file_icon_class($path)
{
    // get extension
    $ext = strtolower(pathinfo($path, PATHINFO_EXTENSION));

    switch ($ext) {
        case 'ico':
        case 'gif':
        case 'jpg':
        case 'jpeg':
        case 'jpc':
        case 'jp2':
        case 'jpx':
        case 'xbm':
        case 'wbmp':
        case 'png':
        case 'bmp':
        case 'tif':
        case 'tiff':
        case 'webp':
        case 'avif':
        case 'svg':
            $img = 'fa fa-picture-o';
            break;
        case 'passwd':
        case 'ftpquota':
        case 'sql':
    }

```

```
case 'js':
case 'ts':
case 'jsx':
case 'tsx':
case 'hbs':
case 'json':
case 'sh':
case 'config':
case 'twig':
case 'tpl':
case 'md':
case 'gitignore':
case 'c':
case 'cpp':
case 'cs':
case 'py':
case 'rs':
case 'map':
case 'lock':
case 'dtd':
case 'ps1':
    $img = 'fa fa-file-code-o';
    break;
case 'txt':
case 'ini':
case 'conf':
case 'log':
case 'htaccess':
case 'yaml':
case 'yml':
case 'toml':
case 'tmp':
case 'top':
case 'bot':
case 'dat':
case 'bak':
case 'htpasswd':
case 'pl':
```

```
$img = 'fa fa-file-text-o';  
break;  
case 'css':  
case 'less':  
case 'sass':  
case 'scss':  
    $img = 'fa fa-css3';  
    break;  
case 'bz2':  
case 'tbz2':  
case 'tbz':  
case 'zip':  
case 'rar':  
case 'gz':  
case 'tgz':  
case 'tar':  
case '7z':  
case 'xz':  
case 'txz':  
case 'zst':  
case 'tzst':  
    $img = 'fa fa-file-archive-o';  
    break;  
case 'php':  
case 'php4':  
case 'php5':  
case 'phps':  
case 'phtml':  
    $img = 'fa fa-code';  
    break;  
case 'htm':  
case 'html':  
case 'shtml':  
case 'xhtml':  
    $img = 'fa fa-html5';  
    break;  
case 'xml':  
case 'xsl':
```

```
$img = 'fa fa-file-excel-o';  
break;  
case 'wav':  
case 'mp3':  
case 'mp2':  
case 'm4a':  
case 'aac':  
case 'ogg':  
case 'oga':  
case 'wma':  
case 'mka':  
case 'flac':  
case 'ac3':  
case 'tds':  
    $img = 'fa fa-music';  
    break;  
case 'm3u':  
case 'm3u8':  
case 'pls':  
case 'cue':  
case 'xspf':  
    $img = 'fa fa-headphones';  
    break;  
case 'avi':  
case 'mpg':  
case 'mpeg':  
case 'mp4':  
case 'm4v':  
case 'flv':  
case 'f4v':  
case 'ogm':  
case 'ogv':  
case 'mov':  
case 'mkv':  
case '3gp':  
case 'asf':  
case 'wmv':  
case 'webm':
```

```
$img = 'fa fa-file-video-o';  
break;  
case 'eml':  
case 'msg':  
    $img = 'fa fa-envelope-o';  
    break;  
case 'xls':  
case 'xlsx':  
case 'ods':  
    $img = 'fa fa-file-excel-o';  
    break;  
case 'csv':  
    $img = 'fa fa-file-text-o';  
    break;  
case 'bak':  
case 'swp':  
    $img = 'fa fa-clipboard';  
    break;  
case 'doc':  
case 'docx':  
case 'odt':  
    $img = 'fa fa-file-word-o';  
    break;  
case 'ppt':  
case 'pptx':  
    $img = 'fa fa-file-powerpoint-o';  
    break;  
case 'ttf':  
case 'ttc':  
case 'otf':  
case 'woff':  
case 'woff2':  
case 'eot':  
case 'fon':  
    $img = 'fa fa-font';  
    break;  
case 'pdf':  
    $img = 'fa fa-file-pdf-o';
```

```

        break;
    case 'psd':
    case 'ai':
    case 'eps':
    case 'fla':
    case 'swf':
        $img = 'fa fa-file-image-o';
        break;
    case 'exe':
    case 'msi':
        $img = 'fa fa-file-o';
        break;
    case 'bat':
        $img = 'fa fa-terminal';
        break;
    default:
        $img = 'fa fa-info-circle';
    }

    return $img;
}

/**
 * Get image files extensions
 * @return array
 */
function fm_get_image_exts()
{
    return array('ico', 'gif', 'jpg', 'jpeg', 'jpc', 'jp2', 'jpx', 'xbm', 'wbmp', 'png', 'bmp', 'tif', 'tiff', 'psd', 'svg',
'webp', 'avif');
}

/**
 * Get video files extensions
 * @return array
 */
function fm_get_video_exts()
{

```

```
    return array('avi', 'webm', 'wmv', 'mp4', 'm4v', 'ogm', 'ogv', 'mov', 'mkv');  
}
```

```
/**  
 * Get audio files extensions  
 * @return array  
 */  
function fm_get_audio_exts()  
{  
    return array('wav', 'mp3', 'ogg', 'm4a');  
}
```

```
/**  
 * Get text file extensions  
 * @return array  
 */  
function fm_get_text_exts()  
{  
    return array(  
        'txt',  
        'css',  
        'ini',  
        'conf',  
        'log',  
        'htaccess',  
        'passwd',  
        'ftpquota',  
        'sql',  
        'js',  
        'ts',  
        'jsx',  
        'tsx',  
        'mjs',  
        'json',  
        'sh',  
        'config',  
        'php',  
        'php4',  
    );  
}
```

'php5',
'phps',
'phtml',
'htm',
'html',
'shtml',
'xhtml',
'xml',
'xsl',
'm3u',
'm3u8',
'pls',
'cue',
'bash',
'vue',
'eml',
'msg',
'csv',
'bat',
'twig',
'tpl',
'md',
'gitignore',
'less',
'sass',
'scss',
'c',
'cpp',
'cs',
'py',
'go',
'zsh',
'swift',
'map',
'lock',
'dtd',
'svg',
'asp',

'aspx',
'asx',
'asmx',
'ashx',
'jsp',
'jspx',
'cgi',
'dockerfile',
'ruby',
'yaml',
'yaml',
'toml',
'vhost',
'sct',
'applescript',
'csx',
'cshtml',
'c++',
'coffee',
'cfm',
'rb',
'graphql',
'mustache',
'jinja',
'http',
'handlebars',
'java',
'es',
'es6',
'markdown',
'wiki',
'tmp',
'top',
'bot',
'dat',
'bak',
'htpasswd',
'pl',

```
        'ps1'  
    );  
}
```

```
/**  
 * Get mime types of text files  
 * @return array  
 */  
function fm_get_text_mimes()  
{  
    return array(  
        'application/xml',  
        'application/javascript',  
        'application/x-javascript',  
        'image/svg+xml',  
        'message/rfc822',  
        'application/json',  
    );  
}
```

```
/**  
 * Get file names of text files w/o extensions  
 * @return array  
 */  
function fm_get_text_names()  
{  
    return array(  
        'license',  
        'readme',  
        'authors',  
        'contributors',  
        'changelog',  
    );  
}
```

```
/**  
 * Get online docs viewer supported files extensions  
 * @return array
```

```

*/
function fm_get_onlineViewer_exts()
{
    return array('doc', 'docx', 'xls', 'xlsx', 'pdf', 'ppt', 'pptx', 'ai', 'psd', 'dxf', 'xps', 'rar', 'odt', 'ods');
}

```

```

/**
 * It returns the mime type of a file based on its extension.
 * @param extension The file extension of the file you want to get the mime type for.
 * @return string|string[] The mime type of the file.
 */

```

```

function fm_get_file_mimes($extension)
{
    $fileTypes['swf'] = 'application/x-shockwave-flash';
    $fileTypes['pdf'] = 'application/pdf';
    $fileTypes['exe'] = 'application/octet-stream';
    $fileTypes['zip'] = 'application/zip';
    $fileTypes['doc'] = 'application/msword';
    $fileTypes['xls'] = 'application/vnd.ms-excel';
    $fileTypes['ppt'] = 'application/vnd.ms-powerpoint';
    $fileTypes['gif'] = 'image/gif';
    $fileTypes['png'] = 'image/png';
    $fileTypes['jpeg'] = 'image/jpeg';
    $fileTypes['jpg'] = 'image/jpeg';
    $fileTypes['webp'] = 'image/webp';
    $fileTypes['avif'] = 'image/avif';
    $fileTypes['rar'] = 'application/rar';

    $fileTypes['ra'] = 'audio/x-pn-realaudio';
    $fileTypes['ram'] = 'audio/x-pn-realaudio';
    $fileTypes['ogg'] = 'audio/x-pn-realaudio';

    $fileTypes['wav'] = 'audio/x-msvideo';
    $fileTypes['wmv'] = 'video/x-msvideo';
    $fileTypes['avi'] = 'video/x-msvideo';
    $fileTypes['asf'] = 'video/x-msvideo';
    $fileTypes['divx'] = 'video/x-msvideo';
}

```

```

$fileTypes['mp3'] = 'audio/mpeg';
$fileTypes['mp4'] = 'video/mp4';
$fileTypes['mpeg'] = 'video/mpeg';
$fileTypes['mpg'] = 'video/mpeg';
$fileTypes['mpe'] = 'video/mpeg';
$fileTypes['mov'] = 'video/quicktime';
$fileTypes['swf'] = 'video/quicktime';
$fileTypes['3gp'] = 'video/quicktime';
$fileTypes['m4a'] = 'video/quicktime';
$fileTypes['aac'] = 'video/quicktime';
$fileTypes['m3u'] = 'video/quicktime';

$fileTypes['php'] = ['application/x-php'];
$fileTypes['html'] = ['text/html'];
$fileTypes['txt'] = ['text/plain'];
//Unknown mime-types should be 'application/octet-stream'
if (empty($fileTypes[$extension])) {
    $fileTypes[$extension] = ['application/octet-stream'];
}
return $fileTypes[$extension];
}

/**
 * This function scans the files and folder recursively, and return matching files
 * @param string $dir
 * @param string $filter
 * @return array|null
 */
function scan($dir = "", $filter = "")
{
    $path = FM_ROOT_PATH . '/' . $dir;
    if ($path) {
        $ite = new RecursiveIteratorIterator(new RecursiveDirectoryIterator($path));
        $rii = new RegexIterator($ite, "/" . $filter . "/i");

        $files = array();
        foreach ($rii as $file) {
            if (!$file->isDir()) {

```

```

        $fileName = $file->getFilename();
        $location = str_replace(FM_ROOT_PATH, "", $file->getPath());
        $files[] = array(
            "name" => $fileName,
            "type" => "file",
            "path" => $location,
        );
    }
}
return $files;
}
}

/**
 * Parameters: downloadFile(File Location, File Name,
 * max speed, is streaming
 * If streaming - videos will show as videos, images as images
 * instead of download prompt
 * https://stackoverflow.com/a/13821992/1164642
 */
function fm_download_file($fileLocation, $fileName, $chunkSize = 1024)
{
    if (connection_status() != 0)
        return (false);
    $extension = pathinfo($fileName, PATHINFO_EXTENSION);

    $contentType = fm_get_file_mimes($extension);

    if (is_array($contentType)) {
        $contentType = implode(' ', $contentType);
    }

    $size = filesize($fileLocation);

    if ($size == 0) {
        fm_set_msg(Ing('Zero byte file! Aborting download'), 'error');
        $FM_PATH = FM_PATH;
        fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
    }
}

```

```

    return (false);
}

@ini_set('magic_quotes_runtime', 0);
$fp = fopen("$fileLocation", "rb");

if ($fp === false) {
    fm_set_msg(Ing('Cannot open file! Aborting download'), 'error');
    $FM_PATH = FM_PATH;
    fm_redirect(FM_SELF_URL . '?p=' . urlencode($FM_PATH));
    return (false);
}

// headers
header('Content-Description: File Transfer');
header('Expires: 0');
header('Cache-Control: must-revalidate, post-check=0, pre-check=0');
header('Pragma: public');
header("Content-Transfer-Encoding: binary");
header("Content-Type: $contentType");

$contentDisposition = 'attachment';

if (strstr($_SERVER['HTTP_USER_AGENT'], "MSIE")) {
    $fileName = preg_replace('/\./', '%2e', $fileName, substr_count($fileName, '.') - 1);
    header("Content-Disposition: $contentDisposition;filename=\"$fileName\"");
} else {
    header("Content-Disposition: $contentDisposition;filename=\"$fileName\"");
}

header("Accept-Ranges: bytes");
$range = 0;

if (isset($_SERVER['HTTP_RANGE'])) {
    list($a, $range) = explode("=", $_SERVER['HTTP_RANGE']);
    str_replace($range, "-", $range);
    $size2 = $size - 1;

```

```

    $new_length = $size - $range;
    header("HTTP/1.1 206 Partial Content");
    header("Content-Length: $new_length");
    header("Content-Range: bytes $range$size2/$size");
} else {
    $size2 = $size - 1;
    header("Content-Range: bytes 0-$size2/$size");
    header("Content-Length: " . $size);
}
$fileLocation = realpath($fileLocation);
while (ob_get_level()) ob_end_clean();
readfile($fileLocation);

fclose($fp);

return ((connection_status() == 0) and !connection_aborted());
}

```

```

/**
 * Class to work with zip files (using ZipArchive)
 */
class FM_Zipper
{
    private $zip;

    public function __construct()
    {
        $this->zip = new ZipArchive();
    }

    /**
     * Create archive with name $filename and files $files (RELATIVE PATHS!)
     * @param string $filename
     * @param array|string $files
     * @return bool
     */
    public function create($filename, $files)
    {

```

```

$res = $this->zip->open($filename, ZipArchive::CREATE);
if ($res !== true) {
    return false;
}
if (is_array($files)) {
    foreach ($files as $f) {
        $f = fm_clean_path($f);
        if (!$this->addFileOrDir($f)) {
            $this->zip->close();
            return false;
        }
    }
    $this->zip->close();
    return true;
} else {
    if ($this->addFileOrDir($files)) {
        $this->zip->close();
        return true;
    }
    return false;
}
}

```

```
/**
```

```

 * Extract archive $filename to folder $path (RELATIVE OR ABSOLUTE PATHS)

```

```

 * @param string $filename

```

```

 * @param string $path

```

```

 * @return bool

```

```

 */

```

```

public function unzip($filename, $path)

```

```

{
    $res = $this->zip->open($filename);
    if ($res !== true) {
        return false;
    }
    if ($this->zip->extractTo($path)) {
        $this->zip->close();
        return true;
    }
}

```

```

    }
    return false;
}

/**
 * Add file/folder to archive
 * @param string $filename
 * @return bool
 */
private function addFileOrDir($filename)
{
    if (is_file($filename)) {
        return $this->zip->addFile($filename);
    } elseif (is_dir($filename)) {
        return $this->addDir($filename);
    }
    return false;
}

```

```

/**
 * Add folder recursively
 * @param string $path
 * @return bool
 */
private function addDir($path)
{
    if (!$this->zip->addEmptyDir($path)) {
        return false;
    }
    $objects = scandir($path);
    if (is_array($objects)) {
        foreach ($objects as $file) {
            if ($file != '.' && $file != '..') {
                if (is_dir($path . '/' . $file)) {
                    if (!$this->addDir($path . '/' . $file)) {
                        return false;
                    }
                } elseif (is_file($path . '/' . $file)) {

```

```

        if (!$this->zip->addFile($path . '/' . $file)) {
            return false;
        }
    }
}
return true;
}
return false;
}
}

```

```

/**
 * Class to work with Tar files (using PharData)
 */
class FM_Zipper_Tar
{
    private $tar;

    public function __construct()
    {
        $this->tar = null;
    }

    /**
     * Create archive with name $filename and files $files (RELATIVE PATHS!)
     * @param string $filename
     * @param array|string $files
     * @return bool
     */
    public function create($filename, $files)
    {
        $this->tar = new PharData($filename);
        if (is_array($files)) {
            foreach ($files as $f) {
                $f = fm_clean_path($f);
                if (!$this->addFileOrDir($f)) {
                    return false;
                }
            }
        }
    }
}

```

```

        }
    }
    return true;
} else {
    if ($this->addFileOrDir($files)) {
        return true;
    }
    return false;
}
}

```

```

/**
 * Extract archive $filename to folder $path (RELATIVE OR ABSOLUTE PATHS)
 * @param string $filename
 * @param string $path
 * @return bool
 */
public function unzip($filename, $path)
{
    $res = $this->tar->open($filename);
    if ($res !== true) {
        return false;
    }
    if ($this->tar->extractTo($path)) {
        return true;
    }
    return false;
}

```

```

/**
 * Add file/folder to archive
 * @param string $filename
 * @return bool
 */
private function addFileOrDir($filename)
{
    if (is_file($filename)) {
        try {

```

```

        $this->tar->addFile($filename);
        return true;
    } catch (Exception $e) {
        return false;
    }
} elseif (is_dir($filename)) {
    return $this->addDir($filename);
}
return false;
}

/**
 * Add folder recursively
 * @param string $path
 * @return bool
 */
private function addDir($path)
{
    $objects = scandir($path);
    if (is_array($objects)) {
        foreach ($objects as $file) {
            if ($file != '.' && $file != '..') {
                if (is_dir($path . '/' . $file)) {
                    if (!$this->addDir($path . '/' . $file)) {
                        return false;
                    }
                } elseif (is_file($path . '/' . $file)) {
                    try {
                        $this->tar->addFile($path . '/' . $file);
                    } catch (Exception $e) {
                        return false;
                    }
                }
            }
        }
    }
    return true;
}
return false;

```

```

    }
}

/**
 * Save Configuration
 */
class FM_Config
{
    var $data;

    function __construct()
    {
        global $root_path, $root_url, $CONFIG;
        $fm_url = $root_url . $_SERVER["PHP_SELF"];
        $this->data = array(
            'lang' => 'en',
            'error_reporting' => true,
            'show_hidden' => true
        );
        $data = false;
        if (strlen($CONFIG)) {
            $data = fm_object_to_array(json_decode($CONFIG));
        } else {
            $msg = 'Tiny File Manager<br>Error: Cannot load configuration';
            if (substr($fm_url, -1) == '/') {
                $fm_url = rtrim($fm_url, '/');
                $msg .= '<br>';
                $msg .= '<br>Seems like you have a trailing slash on the URL.';
                $msg .= '<br>Try this link: <a href="' . $fm_url . '">' . $fm_url . '</a>';
            }
            die($msg);
        }
        if (is_array($data) && count($data)) $this->data = $data;
        else $this->save();
    }

    function save()
    {

```

```

global $config_file;
$fm_file = is_readable($config_file) ? $config_file : __FILE__;
$var_name = '$CONFIG';
$var_value = var_export(json_encode($this->data), true);
$config_string = "<?php" . chr(13) . chr(10) . "///Default Configuration" . chr(13) . chr(10) .
"$var_name = $var_value;" . chr(13) . chr(10);
if (is_writable($fm_file)) {
    $lines = file($fm_file);
    if ($fh = @fopen($fm_file, "w")) {
        @fputs($fh, $config_string, strlen($config_string));
        for ($x = 3; $x < count($lines); $x++) {
            @fputs($fh, $lines[$x], strlen($lines[$x]));
        }
        @fclose($fh);
    }
}
}
}

```

//--- Templates Functions ---

```

/**
 * Show nav block
 * @param string $path
 */
function fm_show_nav_path($path)
{
    global $lang, $sticky_navbar, $editFile;
    $isStickyNavBar = $sticky_navbar ? 'fixed-top' : '';
    ?>
    <nav class="navbar navbar-expand-lg mb-4 main-nav <?php echo $isStickyNavBar ?>
bg-body-tertiary" data-bs-theme="<?php echo FM_THEME; ?>">
        <a class="navbar-brand"> <?php echo lng('AppTitle') ?> </a>
            <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
                <span class="navbar-toggler-icon"></span>
            </button>

```

```
<div class="collapse navbar-collapse" id="navbarSupportedContent">
```

```
<?php
```

```
$path = fm_clean_path($path);
```

```
$root_url = "<a href='?p='><i class='fa fa-home' aria-hidden='true' title='\" . FM_ROOT_PATH  
. \"></i></a>";
```

```
$sep = '<i class="bread-crumb"> / </i>';
```

```
if ($path != "") {
```

```
    $exploded = explode('/', $path);
```

```
    $count = count($exploded);
```

```
    $array = array();
```

```
    $parent = "";
```

```
    for ($i = 0; $i < $count; $i++) {
```

```
        $parent = trim($parent . '/' . $exploded[$i], '/');
```

```
        $parent_enc = urlencode($parent);
```

```
        $array[] = "<a href='?p={$parent_enc}'>\" . fm_enc(fm_convert_win($exploded[$i])) .  
\"</a>";
```

```
    }
```

```
    $root_url .= $sep . implode($sep, $array);
```

```
}
```

```
echo '<div class="col-xs-6 col-sm-5">' . $root_url . $editFile . '</div>';
```

```
?>
```

```
<div class="col-xs-6 col-sm-7">
```

```
    <ul class="navbar-nav justify-content-end" data-bs-theme="<?php echo FM_THEME;  
?>">
```

```
        <li class="nav-item mr-2">
```

```
            <div class="input-group input-group-sm mr-1" style="margin-top:4px;">
```

```
                <input type="text" class="form-control" placeholder="<?php echo lng('Search') ?>"  
aria-label="<?php echo lng('Search') ?>" aria-describedby="search-addon2" id="search-addon">
```

```
                <div class="input-group-append">
```

```
                    <span class="input-group-text brr-0" id="search-addon2"><i class="fa  
fa-search"></i></span>
```

```
                </div>
```

```
                <div class="input-group-append btn-group">
```

```
                    <span class="input-group-text dropdown-toggle brr-0"  
data-bs-toggle="dropdown" aria-haspopup="true" aria-expanded="false"></span>
```

```
                    <div class="dropdown-menu dropdown-menu-right">
```

```

                <a class="dropdown-item" href="<?php echo $path2 = $path ? $path : '.'; ?>"
id="js-search-modal" data-bs-toggle="modal" data-bs-target="#searchModal"><?php echo
Ing('Advanced Search') ?></a>
            </div>
        </div>
    </div>
</li>
<?php if (!FM_READONLY): ?>
    <li class="nav-item">
        <a title="<?php echo Ing('Upload') ?>" class="nav-link" href="?p=<?php echo
urlencode(FM_PATH) ?>&upload"><i class="fa fa-cloud-upload" aria-hidden="true"></i> <?php
echo Ing('Upload') ?></a>
    </li>
    <li class="nav-item">
        <a title="<?php echo Ing('NewItem') ?>" class="nav-link" href="#createNewItem"
data-bs-toggle="modal" data-bs-target="#createNewItem"><i class="fa fa-plus-square"></i> <?php
echo Ing('NewItem') ?></a>
    </li>
<?php endif; ?>
<?php if (FM_USE_AUTH): ?>
    <li class="nav-item avatar dropdown">
        <a class="nav-link dropdown-toggle" id="navbarDropdownMenuLink-5"
data-bs-toggle="dropdown" aria-expanded="false">
            <i class="fa fa-user-circle"></i>
        </a>

        <div class="dropdown-menu dropdown-menu-end text-small shadow"
aria-labelledby="navbarDropdownMenuLink-5" data-bs-theme="<?php echo FM_THEME; ?>">
            <?php if (!FM_READONLY): ?>
                <a title="<?php echo Ing('Settings') ?>" class="dropdown-item nav-link"
href="?p=<?php echo urlencode(FM_PATH) ?>&settings=1"><i class="fa fa-cog"
aria-hidden="true"></i> <?php echo Ing('Settings') ?></a>
            <?php endif ?>
                <a title="<?php echo Ing('Help') ?>" class="dropdown-item nav-link"
href="?p=<?php echo urlencode(FM_PATH) ?>&help=2"><i class="fa fa-exclamation-circle"
aria-hidden="true"></i> <?php echo Ing('Help') ?></a>
                <a title="<?php echo Ing('Logout') ?>" class="dropdown-item nav-link"
href="?logout=1"><i class="fa fa-sign-out" aria-hidden="true"></i> <?php echo Ing('Logout') ?></a>
            </div>
    </li>
</div>
</div>

```

```

        </div>
    </li>
    <?php else: ?>
        <?php if (!FM_READONLY): ?>
            <li class="nav-item">
                <a title="<?php echo lng('Settings') ?>" class="dropdown-item nav-link"
href="?p=<?php echo urlencode(FM_PATH) ?>&amp;settings=1"><i class="fa fa-cog"
aria-hidden="true"></i> <?php echo lng('Settings') ?></a>
                </li>
            <?php endif; ?>
        <?php endif; ?>
    </ul>
</div>
</div>
</nav>
<?php
}

```

```

/**
 * Show alert message from session
 */
function fm_show_message()
{
    if (isset($_SESSION[FM_SESSION_ID]['message'])) {
        $class = isset($_SESSION[FM_SESSION_ID]['status']) ?
$_SESSION[FM_SESSION_ID]['status'] : 'ok';
        echo '<p class="message ' . $class . '">' . $_SESSION[FM_SESSION_ID]['message'] . '</p>';
        unset($_SESSION[FM_SESSION_ID]['message']);
        unset($_SESSION[FM_SESSION_ID]['status']);
    }
}

```

```

/**
 * Show page header in Login Form
 */
function fm_show_header_login()
{
    header("Content-Type: text/html; charset=utf-8");
}

```

```

header("Expires: Sat, 26 Jul 1997 05:00:00 GMT");
header("Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0");
header("Pragma: no-cache");

global $favicon_path;

?>

<!DOCTYPE html>
<html lang="en" data-bs-theme="<?php echo (FM_THEME == "dark") ? 'dark' : 'light' ?>">

<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="Web based File Manager in PHP, Manage your files
efficiently and easily with Tiny File Manager">
    <meta name="author" content="CCP Programmers">
    <meta name="robots" content="noindex, nofollow">
    <meta name="googlebot" content="noindex">
    <?php if ($favicon_path) {
        echo '<link rel="icon" href="" . fm_enc($favicon_path) . "" type="image/png">';
    } ?>
    <title><?php echo fm_enc(APP_TITLE) ?></title>
    <?php print_external('pre-jsdelivr'); ?>
    <?php print_external('css-bootstrap'); ?>
    <style>
        body.fm-login-page {
            background-color: #f7f9fb;
            font-size: 14px;
            background-color: #f7f9fb;
            background-image: url("data:image/svg+xml,%3Csvg xmlns='http://www.w3.org/2000/svg'
viewBox='0 0 304 304' width='304' height='304'%3E%3Cpath fill='%23e2e9f1' fill-opacity='0.4'
d='M44.1 224a5 5 0 1 1 0 2H0v-2h44.1zm160 48a5 5 0 1 1 0 2H82v-2h122.1zm57.8-46a5 5 0 1 1
0-2H304v2h-42.1zm0 16a5 5 0 1 1 0-2H304v2h-42.1zm6.2-114a5 5 0 1 1 0 2h-86.2a5 5 0 1 1
0-2h86.2zm-256-48a5 5 0 1 1 0 2H0v-2h12.1zm185.8 34a5 5 0 1 1 0-2h86.2a5 5 0 1 1 0
2h-86.2zM258 12.1a5 5 0 1 1-2 0V0h2v12.1zm-64 208a5 5 0 1 1-2 0v-54.2a5 5 0 1 1 2
0v54.2zm48-198.2V80h62v2h-64V21.9a5 5 0 1 1 2 0zm16 16V64h46v2h-48V37.9a5 5 0 1 1 2
0zm-128 96V208h16v12.1a5 5 0 1 1-2 0V210h-16v-76.1a5 5 0 1 1 2 0zm-5.9-21.9a5 5 0 1 1 0
2H114v48H85.9a5 5 0 1 1 0-2H112v-48h12.1zm-6.2 130a5 5 0 1 1 0-2H176v-74.1a5 5 0 1 1 2
0V242h-60.1zm-16-64a5 5 0 1 1 0-2H114v48h10.1a5 5 0 1 1 0 2H112v-48h-10.1zM66 284.1a5 5 0

```

1 1-2 0V274H50v30h-2v-32h18v12.1zM236.1 176a5 5 0 1 1 0
2H226v94h48v32h-2v-30h-48v-98h12.1zm25.8-30a5 5 0 1 1 0-2H274v44.1a5 5 0 1 1-2
0V146h-10.1zm-64 96a5 5 0 1 1 0-2H208v-80h16v-14h-42.1a5 5 0 1 1
0-2H226v18h-16v80h-12.1zm86.2-210a5 5 0 1 1 0 2H272V0h2v32h10.1zM98 101.9V146H53.9a5 5
0 1 1 0-2H96v-42.1a5 5 0 1 1 2 0zM53.9 34a5 5 0 1 1 0-2H80V0h2v34H53.9zm60.1
3.9V66H82v64H69.9a5 5 0 1 1 0-2H80V64h32V37.9a5 5 0 1 1 2 0zM101.9 82a5 5 0 1 1
0-2H128V37.9a5 5 0 1 1 2 0V82h-28.1zm16-64a5 5 0 1 1 0-2H146v44.1a5 5 0 1 1-2
0V18h-26.1zm102.2 270a5 5 0 1 1 0 2H98v14h-2v-16h124.1zM242
149.9V160h16v34h-16v62h48v48h-2v-46h-48v-66h16v-30h-16v-12.1a5 5 0 1 1 2 0zM53.9 18a5 5 0
1 1 0-2H64V2H48V0h18v18H53.9zm112 32a5 5 0 1 1 0-2H192V0h50v2h-48v48h-28.1zm-48-48a5
5 0 0 1-9.8-2h2.07a3 3 0 1 0 5.66 0H178v34h-18V21.9a5 5 0 1 1 2 0V32h14V2h-58.1zm0 96a5 5 0
1 1 0-2H137I32-32h39V21.9a5 5 0 1 1 2 0V66h-40.17I-32 32H117.9zm28.1 90.1a5 5 0 1 1-2
0v-76.51L175.59 80H224V21.9a5 5 0 1 1 2 0V82h-49.59L146 112.41v75.69zm16 32a5 5 0 1 1-2
0v-99.51L184.59 96H300.1a5 5 0 0 1 3.9-3.9v2.07a3 3 0 0 0 0 5.66v2.07a5 5 0 0
1-3.9-3.9H185.41L162 121.41v98.69zm-144-64a5 5 0 1 1-2
0v-3.51I48-48V48h32V0h2v50H66v55.41I-48 48v2.69zM50 53.9v43.51I-48 48V208h26.1a5 5 0 1 1
0 2H0v-65.41I48-48V53.9a5 5 0 1 1 2 0zm-16 16V89.41I-34 34v-2.82I32-32V69.9a5 5 0 1 1 2
0zM12.1 32a5 5 0 1 1 0 2H9.41L0 43.41V40.6L8.59 32h3.51zm265.8 18a5 5 0 1 1
0-2h18.69I7.41-7.41v2.82L297.41 50H277.9zm-16 160a5 5 0 1 1 0-2H288v-71.41I16-16v2.82I-14
14V210h-28.1zm-208 32a5 5 0 1 1 0-2H64v-22.59L40.59 194H21.9a5 5 0 1 1 0-2H41.41L66
216.59V242H53.9zm150.2 14a5 5 0 1 1 0 2H96v-56.6L56.6 162H37.9a5 5 0 1 1 0-2h19.5L98
200.6V256h106.1zm-150.2 2a5 5 0 1 1 0-2H80v-46.59L48.59 178H21.9a5 5 0 1 1 0-2H49.41L82
208.59V258H53.9zM34 39.8v1.61L9.41 66H0v-2h8.59L32 40.59V0h2v39.8zM2 300.1a5 5 0 0 1 3.9
3.9H3.83A3 3 0 0 0 0 302.17V256h18v48h-2v-46H2v42.1zM34 241v63h-2v-62H0v-2h34v1zM17
18H0v-2h16V0h2v18h-1zm273-2h14v2h-16V0h2v16zm-32
273v15h-2v-14h-14v14h-2v-16h18v1zM0 92.1A5.02 5.02 0 0 1 6 97a5 5 0 0 1-6 4.9v-2.07a3 3 0 1 0
0-5.66V92.1zM80 272h2v32h-2v-32zm37.9 32h-2.07a3 3 0 0 0-5.66 0h-2.07a5 5 0 0 1 9.8 0zM5.9
0A5.02 5.02 0 0 1 0 5.9V3.83A3 3 0 0 0 3.83 0H5.9zm294.2 0h2.07A3 3 0 0 0 304 3.83V5.9a5 5 0 0
1-3.9-5.9zm3.9 300.1v2.07a3 3 0 0 0-1.83 1.83h-2.07a5 5 0 0 1 3.9-3.9zM97 100a3 3 0 1 0 0-6 3 3
0 0 0 0 6zm0-16a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm16 16a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm16 16a3 3 0 1 0
0-6 3 3 0 0 0 0 6zm0 16a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm-48 32a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm16 16a3
3 0 1 0 0-6 3 3 0 0 0 0 6zm32 48a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm-16 16a3 3 0 1 0 0-6 3 3 0 0 0 0
6zm32-16a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm0-32a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm16 32a3 3 0 1 0 0-6 3 3 0
0 0 0 6zm32 16a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm0-16a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm-16-64a3 3 0 1 0
0-6 3 3 0 0 0 0 6zm16 0a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm16 96a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm0 16a3 3
0 1 0 0-6 3 3 0 0 0 0 6zm16 16a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm16-144a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm0
32a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm16-32a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm16-16a3 3 0 1 0 0-6 3 3 0 0 0 0
6zm-96 0a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm0 16a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm16-32a3 3 0 1 0 0-6 3 3 0

0 0 0 6zm96 0a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm-16-64a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm16-16a3 3 0 1 0
0-6 3 3 0 0 0 0 6zm-32 0a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm0-16a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm-16 0a3 3
0 1 0 0-6 3 3 0 0 0 0 6zm-16 0a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm-16 0a3 3 0 1 0 0-6 3 3 0 0 0 0 6zM49
36a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm-32 0a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm32 16a3 3 0 1 0 0-6 3 3 0 0 0 0
6zM33 68a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm16-48a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm0 240a3 3 0 1 0 0-6 3 3
0 0 0 0 6zm16 32a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm-16-64a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm0 16a3 3 0 1 0
0-6 3 3 0 0 0 0 6zm-16-32a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm80-176a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm16 0a3
3 0 1 0 0-6 3 3 0 0 0 0 6zm-16-16a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm32 48a3 3 0 1 0 0-6 3 3 0 0 0 0
6zm16-16a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm0-32a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm112 176a3 3 0 1 0 0-6 3
3 0 0 0 0 6zm-16 16a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm0 16a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm0 16a3 3 0 1 0
0-6 3 3 0 0 0 0 6zM17 180a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm0 16a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm0-32a3 3
0 1 0 0-6 3 3 0 0 0 0 6zm16 0a3 3 0 1 0 0-6 3 3 0 0 0 0 6zM17 84a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm32
64a3 3 0 1 0 0-6 3 3 0 0 0 0 6zm16-16a3 3 0 1 0 0-6 3 3 0 0 0 0
6z'%3E%3C/path%3E%3C/svg%3E");
}

```
.fm-login-page .brand {  
  width: 121px;  
  overflow: hidden;  
  margin: 0 auto;  
  position: relative;  
  z-index: 1  
}
```

```
.fm-login-page .brand img {  
  width: 100%  
}
```

```
.fm-login-page .card-wrapper {  
  width: 360px;  
}
```

```
.fm-login-page .card {  
  border-color: transparent;  
  box-shadow: 0 4px 8px rgba(0, 0, 0, .05)  
}
```

```
.fm-login-page .card-title {
```

```
margin-bottom: 1.5rem;  
font-size: 24px;  
font-weight: 400;  
}
```

```
.fm-login-page .form-control {  
  border-width: 2.3px  
}
```

```
.fm-login-page .form-group label {  
  width: 100%  
}
```

```
.fm-login-page .btn.btn-block {  
  padding: 12px 10px  
}
```

```
.fm-login-page .footer {  
  margin: 20px 0;  
  color: #888;  
  text-align: center  
}
```

```
@media screen and (max-width:425px) {  
  .fm-login-page .card-wrapper {  
    width: 90%;  
    margin: 0 auto;  
    margin-top: 10%;  
  }  
}
```

```
@media screen and (max-width:320px) {  
  .fm-login-page .card.fat {  
    padding: 0  
  }  
}
```

```
.fm-login-page .card.fat .card-body {  
  padding: 15px
```

```
}  
}
```

```
.message {  
  padding: 4px 7px;  
  border: 1px solid #ddd;  
  background-color: #fff  
}
```

```
.message.ok {  
  border-color: green;  
  color: green  
}
```

```
.message.error {  
  border-color: red;  
  color: red  
}
```

```
.message.alert {  
  border-color: orange;  
  color: orange  
}
```

```
body.fm-login-page.theme-dark {  
  background-color: #2f2a2a;  
}
```

```
.theme-dark svg g,  
.theme-dark svg path {  
  fill: #ffffff;  
}
```

```
.theme-dark .form-control {  
  color: #fff;  
  background-color: #403e3e;  
}
```

```
.h-100vh {
    min-height: 100vh;
}
</style>
</head>
```

```
<body class="fm-login-page <?php echo (FM_THEME == "dark") ? 'theme-dark' : ""; ?>">
    <div id="wrapper" class="container-fluid">
```

```

        <?php
    }

```

```
/**
 * Show page footer in Login Form
 */
function fm_show_footer_login()
{
    ?>
    </div>
    <?php print_external('js-jquery'); ?>
    <?php print_external('js-bootstrap'); ?>
</body>
```

```
</html>
```

```
<?php
}

```

```
/**
 * Show Header after login
 */
function fm_show_header()
{
    header("Content-Type: text/html; charset=utf-8");
    header("Expires: Sat, 26 Jul 1997 05:00:00 GMT");
    header("Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0");
    header("Pragma: no-cache");
}
```

```

global $sticky_navbar, $favicon_path;
$isStickyNavBar = $sticky_navbar ? 'navbar-fixed' : 'navbar-normal';
?>

<!DOCTYPE html>
<html data-bs-theme="<?php echo FM_THEME; ?>">

<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="Web based File Manager in PHP, Manage your files
efficiently and easily with Tiny File Manager">
    <meta name="author" content="CCP Programmers">
    <meta name="robots" content="noindex, nofollow">
    <meta name="googlebot" content="noindex">
    <?php if ($favicon_path) {
        echo '<link rel="icon" href="' . fm_enc($favicon_path) . '" type="image/png">';
    } ?>
    <title><?php echo fm_enc(APP_TITLE) ?> | <?php echo (isset($_GET['view']) ? $_GET['view'] :
((isset($_GET['edit'])) ? $_GET['edit'] : "H3K")); ?></title>
    <?php print_external('pre-jsdelivr'); ?>
    <?php print_external('pre-cloudflare'); ?>
    <?php print_external('css-bootstrap'); ?>
    <?php print_external('css-font-awesome'); ?>
    <?php if (FM_USE_HIGHLIGHTJS && isset($_GET['view'])): ?>
        <?php print_external('css-highlightjs'); ?>
    <?php endif; ?>
    <script type="text/javascript">
        window.csrf = '<?php echo $_SESSION['token']; ?>';
    </script>
    <style>
        html {
            -moz-osx-font-smoothing: grayscale;
            -webkit-font-smoothing: antialiased;
            text-rendering: optimizeLegibility;
            height: 100%;
            scroll-behavior: smooth;
        }

```

```
*,
*::before,
*::after {
    box-sizing: border-box;
}
```

```
body {
    font-size: 15px;
    color: #222;
    background: #F7F7F7;
}
```

```
body.navbar-fixed {
    margin-top: 55px;
}
```

```
a,
a:hover,
a:visited,
a:focus {
    text-decoration: none !important;
}
```

```
.filename,
td,
th {
    white-space: nowrap;
}
```

```
.navbar-brand {
    font-weight: bold;
}
```

```
.nav-item.avatar a {
    cursor: pointer;
    text-transform: capitalize;
}
```

```
.nav-item.avatar a>i {  
    font-size: 15px;  
}
```

```
.nav-item.avatar .dropdown-menu a {  
    font-size: 13px;  
}
```

```
#search-addon {  
    font-size: 12px;  
    border-right-width: 0;  
}
```

```
.brl-0 {  
    background: transparent;  
    border-left: 0;  
    border-top-left-radius: 0;  
    border-bottom-left-radius: 0;  
}
```

```
.brr-0 {  
    border-top-right-radius: 0;  
    border-bottom-right-radius: 0;  
}
```

```
.bread-crumb {  
    color: #cccccc;  
    font-style: normal;  
}
```

```
#main-table {  
    transition: transform .25s cubic-bezier(0.4, 0.5, 0, 1), width 0s .25s;  
}
```

```
#main-table .filename a {  
    color: #222222;  
}
```

```
.table td,  
.table th {  
    vertical-align: middle !important;  
}
```

```
.table .custom-checkbox-td .custom-control.custom-checkbox,  
.table .custom-checkbox-header .custom-control.custom-checkbox {  
    min-width: 18px;  
    display: flex;  
    align-items: center;  
    justify-content: center;  
}
```

```
.table-sm td,  
.table-sm th {  
    padding: .4rem;  
}
```

```
.table-bordered td,  
.table-bordered th {  
    border: 1px solid #f1f1f1;  
}
```

```
.hidden {  
    display: none  
}
```

```
pre.with-hljs {  
    padding: 0;  
    overflow: hidden;  
}
```

```
pre.with-hljs code {  
    margin: 0;  
    border: 0;  
    overflow: scroll;  
}
```

```
code.maxheight,  
pre.maxheight {  
  max-height: 512px  
}
```

```
.fa.fa-caret-right {  
  font-size: 1.2em;  
  margin: 0 4px;  
  vertical-align: middle;  
  color: #ececec  
}
```

```
.fa.fa-home {  
  font-size: 1.3em;  
  vertical-align: bottom  
}
```

```
.path {  
  margin-bottom: 10px  
}
```

```
form.dropzone {  
  min-height: 200px;  
  border: 2px dashed #007bff;  
  line-height: 6rem;  
}
```

```
.right {  
  text-align: right  
}
```

```
.center,  
.close,  
.login-form,  
.preview-img-container {  
  text-align: center  
}
```

```
.message {  
  padding: 4px 7px;  
  border: 1px solid #ddd;  
  background-color: #fff  
}
```

```
.message.ok {  
  border-color: green;  
  color: green  
}
```

```
.message.error {  
  border-color: red;  
  color: red  
}
```

```
.message.alert {  
  border-color: orange;  
  color: orange  
}
```

```
.preview-img {  
  max-width: 100%;  
  max-height: 80vh;
```

```
background:  
url(  
kIEQVR42mL5//8/Azbw+PFjrOJMDCSCUQ3EABZc4S0rKzsaSvTTABBgAMyfCMsY4B9iAAAAAEIF  
TkSuQmCC);  
  cursor: zoom-in  
}
```

```
input#preview-img-zoomCheck[type=checkbox] {  
  display: none  
}
```

```
input#preview-img-zoomCheck[type=checkbox]:checked~label>img {  
  max-width: none;  
  max-height: none;
```

```
    cursor: zoom-out;
}

.inline-actions>a>i {
    font-size: 1em;
    margin-left: 5px;
    background: #3785c1;
    color: #fff;
    padding: 3px 4px;
    border-radius: 3px;
}

.preview-video {
    position: relative;
    max-width: 100%;
    height: 0;
    padding-bottom: 62.5%;
    margin-bottom: 10px
}

.preview-video video {
    position: absolute;
    width: 100%;
    height: 100%;
    left: 0;
    top: 0;
    background: #000
}

.compact-table {
    border: 0;
    width: auto
}

.compact-table td,
.compact-table th {
    width: 100px;
    border: 0;
```

```
    text-align: center;
}

.compact-table tr:hover td {
    background-color: #fff;
}

.filename {
    max-width: 420px;
    overflow: hidden;
    text-overflow: ellipsis;
}

.break-word {
    word-wrap: break-word;
    margin-left: 30px;
}

.break-word.float-left a {
    color: #7d7d7d;
}

.break-word+.float-right {
    padding-right: 30px;
    position: relative;
}

.break-word+.float-right>a {
    color: #7d7d7d;
    font-size: 1.2em;
    margin-right: 4px;
}

#editor {
    position: absolute;
    right: 15px;
    top: 100px;
    bottom: 15px;
```

```
    left: 15px;
}

@media (max-width:481px) {
    #editor {
        top: 150px;
    }
}
```

```
#normal-editor {
    border-radius: 3px;
    border-width: 2px;
    padding: 10px;
    outline: none;
}
```

```
.btn-2 {
    padding: 4px 10px;
    font-size: small;
}
```

```
li.file:before,
li.folder:before {
    font: normal normal normal 14px/1 FontAwesome;
    content: "\f016";
    margin-right: 5px;
}
```

```
li.folder:before {
    content: "\f114"
}
```

```
i.fa.fa-folder-o {
    color: #0157b3
}
```

```
i.fa.fa-picture-o {
    color: #26b99a
}
```

```
}
```

```
i.fa.fa-file-archive-o {  
  color: #da7d7d  
}
```

```
.btn-2 i.fa.fa-file-archive-o {  
  color: inherit  
}
```

```
i.fa.fa-css3 {  
  color: #f36fa0  
}
```

```
i.fa.fa-file-code-o {  
  color: #007bff  
}
```

```
i.fa.fa-code {  
  color: #cc4b4c  
}
```

```
i.fa.fa-file-text-o {  
  color: #0096e6  
}
```

```
i.fa.fa-html5 {  
  color: #d75e72  
}
```

```
i.fa.fa-file-excel-o {  
  color: #09c55d  
}
```

```
i.fa.fa-file-powerpoint-o {  
  color: #f6712e  
}
```

```
i.go-back {  
  font-size: 1.2em;  
  color: #007bff;  
}
```

```
.main-nav {  
  padding: 0.2rem 1rem;  
  box-shadow: 0 4px 5px 0 rgba(0, 0, 0, .14), 0 1px 10px 0 rgba(0, 0, 0, .12), 0 2px 4px -1px  
  rgba(0, 0, 0, .2)  
}
```

```
.dataTables_filter {  
  display: none;  
}
```

```
table.dataTable thead .sorting {  
  cursor: pointer;  
  background-repeat: no-repeat;  
  background-position: center right;
```

```
background-image:  
url('  
kEIEQVQoz7XQMQ5AQBCF4dWQSJxC5wwax1Cq1e7BAxdD5SL+Tq/QCM1oNiJidwox0355mXnG  
/DrEtIQ6azioNZQxl0ykPhTQlwhCR+BmBYtIK7kLJYwWCcJA9M4qdrZrd8pPjZWPTOqdRQy320YSV  
17OatFC4euts6z39GYMKRPCTKY9UnPQ6P+GtMRfGtPnBCiqhAeJPmkqAAAAAEIFTkSuQmCC');
```

```
table.dataTable thead .sorting_asc {  
  cursor: pointer;  
  background-repeat: no-repeat;  
  background-position: center right;
```

```
background-image:  
url('  
Z0IEQVQ4y2NgGLKgquEuFxBPAGI2ahhWCsS/gDibUoO0gPgxEp8H4ttArEyuQYxAPBdqEAXPBIm  
TY5gjEL9DM+wTENuQahAvEO9DMwiGdwAxOymGJQLxTyD+jgWDxCMZRsEoGAVoAADeemwtP  
cZI2wAAAABJRU5ErkJggg==');
```

```
table.dataTable thead .sorting_desc {
```

```
cursor: pointer;
background-repeat: no-repeat;
background-position: center right;
```

```
background-image:
url('
ZUIEQVQ4y2NgGAWjYBSggaqGu5FA/BOlv2PBIPFEUgxjB+IdQPwfC94HxLykus4GiD+hGfQOiB3J
8SojEE9EM2wuSJzcsFMG4ttQgx4DsRalkZENxL+AuJQaMcsGxBOAmGvopk8AVz1sLZgg0bsAAA
AASUVORK5CYII=');
}
```

```
table.dataTable thead tr:first-child th.custom-checkbox-header:first-child {
background-image: none;
}
```

```
.footer-action li {
margin-bottom: 10px;
}
```

```
.app-v-title {
font-size: 24px;
font-weight: 300;
letter-spacing: -.5px;
text-transform: uppercase;
}
```

```
hr.custom-hr {
border-top: 1px dashed #8c8b8b;
border-bottom: 1px dashed #fff;
}
```

```
#snackbar {
visibility: hidden;
min-width: 250px;
margin-left: -125px;
background-color: #333;
color: #fff;
text-align: center;
border-radius: 2px;
```

```
padding: 16px;
position: fixed;
z-index: 1;
left: 50%;
bottom: 30px;
font-size: 17px;
}
```

```
#snackbar.show {
  visibility: visible;
  -webkit-animation: fadein 0.5s, fadeout 0.5s 2.5s;
  animation: fadein 0.5s, fadeout 0.5s 2.5s;
}
```

```
@-webkit-keyframes fadein {
  from {
    bottom: 0;
    opacity: 0;
  }

  to {
    bottom: 30px;
    opacity: 1;
  }
}
```

```
@keyframes fadein {
  from {
    bottom: 0;
    opacity: 0;
  }

  to {
    bottom: 30px;
    opacity: 1;
  }
}
```

```
@-webkit-keyframes fadeout {  
  from {  
    bottom: 30px;  
    opacity: 1;  
  }  
  
  to {  
    bottom: 0;  
    opacity: 0;  
  }  
}
```

```
@keyframes fadeout {  
  from {  
    bottom: 30px;  
    opacity: 1;  
  }  
  
  to {  
    bottom: 0;  
    opacity: 0;  
  }  
}
```

```
#main-table span.badge {  
  border-bottom: 2px solid #f8f9fa  
}
```

```
#main-table span.badge:nth-child(1) {  
  border-color: #df4227  
}
```

```
#main-table span.badge:nth-child(2) {  
  border-color: #f8b600  
}
```

```
#main-table span.badge:nth-child(3) {  
  border-color: #00bd60  
}
```

```
}
```

```
#main-table span.badge:nth-child(4) {  
  border-color: #4581ff  
}
```

```
#main-table span.badge:nth-child(5) {  
  border-color: #ac68fc  
}
```

```
#main-table span.badge:nth-child(6) {  
  border-color: #45c3d2  
}
```

```
    @media only screen and (min-device-width:768px) and (max-device-width:1024px) and  
(orientation:landscape) and (-webkit-min-device-pixel-ratio:2) {  
      .navbar-collapse .col-xs-6 {  
        padding: 0;  
      }  
    }  
}
```

```
.btn.active.focus,  
.btn.active:focus,  
.btn.focus,  
.btn.focus:active,  
.btn:active:focus,  
.btn:focus {  
  outline: 0 !important;  
  outline-offset: 0 !important;  
  background-image: none !important;  
  -webkit-box-shadow: none !important;  
  box-shadow: none !important  
}
```

```
.lds-facebook {  
  display: none;  
  position: relative;  
  width: 64px;
```

```
    height: 64px  
}
```

```
.lds-facebook div,  
.lds-facebook.show-me {  
    display: inline-block  
}
```

```
.lds-facebook div {  
    position: absolute;  
    left: 6px;  
    width: 13px;  
    background: #007bff;  
    animation: lds-facebook 1.2s cubic-bezier(0, .5, .5, 1) infinite  
}
```

```
.lds-facebook div:nth-child(1) {  
    left: 6px;  
    animation-delay: -.24s  
}
```

```
.lds-facebook div:nth-child(2) {  
    left: 26px;  
    animation-delay: -.12s  
}
```

```
.lds-facebook div:nth-child(3) {  
    left: 45px;  
    animation-delay: 0s  
}
```

```
@keyframes lds-facebook {  
    0% {  
        top: 6px;  
        height: 51px  
    }  
  
    100%,
```

```
50% {  
    top: 19px;  
    height: 26px  
}  
}
```

```
ul#search-wrapper {  
    padding-left: 0;  
    border: 1px solid #ececcecc;  
}
```

```
ul#search-wrapper li {  
    list-style: none;  
    padding: 5px;  
    border-bottom: 1px solid #ececcecc;  
}
```

```
ul#search-wrapper li:nth-child(odd) {  
    background: #f9f9f9cc;  
}
```

```
.c-preview-img {  
    max-width: 300px;  
}
```

```
.border-radius-0 {  
    border-radius: 0;  
}
```

```
.float-right {  
    float: right;  
}
```

```
.table-hover>tbody>tr:hover>td:first-child {  
    border-left: 1px solid #1b77fd;  
}
```

```
#main-table tr.even {
```

```
background-color: #F8F9Fa;
}
```

```
.filename>a>i {
margin-right: 3px;
}
```

```
.fs-7 {
font-size: 14px;
}
```

</style>

<?php

if (FM_THEME == "dark"): ?>

<style>

```
:root {
--bs-bg-opacity: 1;
--bg-color: #f3daa6;
--bs-dark-rgb: 28, 36, 41 !important;
--bs-bg-opacity: 1;
}
```

```
body.theme-dark {
background-image: linear-gradient(90deg, #1c2429, #263238);
color: #CFD8DC;
}
```

```
.list-group .list-group-item {
background: #343a40;
}
```

```
.theme-dark .navbar-nav i,
.navbar-nav .dropdown-toggle,
.break-word {
color: #CFD8DC;
}
```

```
a,
a:hover,
```

```
a:visited,
a:active,
#main-table .filename a,
i.fa.fa-folder-o,
i.go-back {
    color: var(--bg-color);
}

ul#search-wrapper li:nth-child(odd) {
    background: #212a2f;
}

.theme-dark .btn-outline-primary {
    color: #b8e59c;
    border-color: #b8e59c;
}

.theme-dark .btn-outline-primary:hover,
.theme-dark .btn-outline-primary:active {
    background-color: #2d4121;
}

.theme-dark input.form-control {
    background-color: #101518;
    color: #CFD8DC;
}

.theme-dark .dropzone {
    background: transparent;
}

.theme-dark .inline-actions>a>i {
    background: #79755e;
}

.theme-dark .text-white {
    color: #CFD8DC !important;
}
```

```

.theme-dark .table-bordered td,
.table-bordered th {
    border-color: #343434;
}

.theme-dark .table-bordered td .custom-control-input,
.theme-dark .table-bordered th .custom-control-input {
    opacity: 0.678;
}

.message {
    background-color: #212529;
}

form.dropzone {
    border-color: #79755e;
}
</style>
<?php endif; ?>
</head>

```

```
<body class="<?php echo (FM_THEME == "dark") ? 'theme-dark' : ''; ?> <?php echo  
$isStickyNavBar; ?>">  
    <div id="wrapper" class="container-fluid">  
        <!-- New Item creation -->  
            <div class="modal fade" id="createNewItem" tabindex="-1" role="dialog"  
data-bs-backdrop="static" data-bs-keyboard="false" aria-labelledby="newItemModalLabel"  
aria-hidden="true" data-bs-theme="<?php echo FM_THEME; ?>">  
                <div class="modal-dialog" role="document">  
                    <form class="modal-content" method="post">  
                        <div class="modal-header">  
                            <h5 class="modal-title" id="newItemModalLabel"><i class="fa fa-plus-square  
fa-fw"></i><?php echo lng('CreateNewItem') ?></h5>  
                            <button type="button" class="btn-close" data-bs-dismiss="modal"  
aria-label="Close"></button>  
                        </div>  
                        <div class="modal-body">
```

```

        <p><label for="newfile"><?php echo lng('ItemType') ?> </label></p>
        <div class="form-check form-check-inline">
            <input class="form-check-input" type="radio" name="newfile"
id="customRadiolInline1" name="newfile" value="file">
                <label class="form-check-label" for="customRadiolInline1"><?php echo
lng('File') ?></label>
            </div>
            <div class="form-check form-check-inline">
                <input class="form-check-input" type="radio" name="newfile"
id="customRadiolInline2" value="folder" checked>
                    <label class="form-check-label" for="customRadiolInline2"><?php echo
lng('Folder') ?></label>
                </div>

        <p class="mt-3"><label for="newfilename"><?php echo lng('ItemName') ?>
</label></p>
            <input type="text" name="newfilename" id="newfilename" value=""
class="form-control" placeholder="<?php echo lng('Enter here...') ?>" required>
        </div>
        <div class="modal-footer">
            <input type="hidden" name="token" value="<?php echo $_SESSION['token']; ?>">
            <button type="button" class="btn btn-outline-primary" data-bs-dismiss="modal"><i
class="fa fa-times-circle"></i> <?php echo lng('Cancel') ?></button>
            <button type="submit" class="btn btn-success"><i class="fa fa-check-circle"></i>
<?php echo lng('CreateNow') ?></button>
        </div>
    </form>
</div>
</div>

<!-- Advance Search Modal -->
        <div class="modal fade" id="searchModal" tabindex="-1" role="dialog"
aria-labelledby="searchModalLabel" aria-hidden="true" data-bs-theme="<?php echo FM_THEME;
?>">
            <div class="modal-dialog modal-lg" role="document">
                <div class="modal-content">
                    <div class="modal-header">
                        <h5 class="modal-title col-10" id="searchModalLabel">

```

```

        <div class="input-group mb-3">
            <input type="text" class="form-control" placeholder="<?php echo
lng('Search') ?> <?php echo lng('a files') ?>" aria-label="<?php echo lng('Search') ?>"
aria-describedby="search-addon3" id="advanced-search" autofocus required>
            <span class="input-group-text" id="search-addon3"><i class="fa
fa-search"></i></span>
        </div>
    </h5>

    <button type="button" class="btn-close" data-bs-dismiss="modal"
aria-label="Close"></button>
</div>
<div class="modal-body">
    <form action="" method="post">
        <div class="lds-facebook">
            <div></div>
            <div></div>
            <div></div>
        </div>
        <ul id="search-wrapper">
            <p class="m-2"><?php echo lng('Search file in folder and subfolders...')
?></p>
        </ul>
    </form>
</div>
</div>
</div>
</div>

<!--Rename Modal -->
    <div class="modal modal-alert" data-bs-backdrop="static" data-bs-keyboard="false"
tabindex="-1" role="dialog" id="renameDailog" data-bs-theme="<?php echo FM_THEME; ?>">
        <div class="modal-dialog" role="document">
            <form class="modal-content rounded-3 shadow" method="post" autocomplete="off">
                <div class="modal-body p-4 text-center">
                    <h5 class="mb-3"><?php echo lng('Are you sure want to rename?') ?></h5>
                    <p class="mb-1">
                        <input type="text" name="rename_to" id="js-rename-to" class="form-control"
placeholder="<?php echo lng('Enter new file name') ?>" required>

```

```

        <input type="hidden" name="token" value="<?php echo $_SESSION['token'];
?>">

        <input type="hidden" name="rename_from" id="js-rename-from">
    </p>
</div>
<div class="modal-footer flex-nowrap p-0">
    <button type="button" class="btn btn-lg btn-link fs-6 text-decoration-none col-6 m-0
rounded-0 border-end" data-bs-dismiss="modal"><?php echo lng('Cancel') ?></button>
    <button type="submit" class="btn btn-lg btn-link fs-6 text-decoration-none col-6
m-0 rounded-0"><strong><?php echo lng('Okay') ?></strong></button>
</div>
</form>
</div>
</div>

<!-- Confirm Modal -->
<script type="text/html" id="js-tpl-confirm">
    <div class="modal modal-alert confirmDailog" data-bs-backdrop="static"
data-bs-keyboard="false"    tabindex="-1"    role="dialog"    id="confirmDailog-<%this.id%>"
data-bs-theme="<?php echo FM_THEME; ?>">
        <div class="modal-dialog" role="document">
            <form class="modal-content rounded-3 shadow" method="post" autocomplete="off"
action="<%this.action%>">
                <div class="modal-body p-4 text-center">
                    <h5 class="mb-2"><?php echo lng('Are you sure want to') ?> <%this.title%>
?</h5>

                    <p class="mb-1"><%this.content%></p>
                </div>
                <div class="modal-footer flex-nowrap p-0">
                    <button type="button" class="btn btn-lg btn-link fs-6 text-decoration-none col-6
m-0 rounded-0 border-end" data-bs-dismiss="modal"><?php echo lng('Cancel') ?></button>
                    <input type="hidden" name="token" value="<?php echo $_SESSION['token'];
?>">

                    <button type="submit" class="btn btn-lg btn-link fs-6 text-decoration-none col-6
m-0 rounded-0" data-bs-dismiss="modal"><strong><?php echo lng('Okay') ?></strong></button>
                </div>
            </form>
        </div>

```

```

        </div>
    </script>
<?php
}

/**
 * Show page footer after login
 */
function fm_show_footer()
{
    ?>
    </div>
    <?php print_external('js-jquery'); ?>
    <?php print_external('js-bootstrap'); ?>
    <?php print_external('js-jquery-datatables'); ?>
    <?php if (FM_USE_HIGHLIGHTJS && isset($_GET['view'])): ?>
        <?php print_external('js-highlightjs'); ?>
        <script>
            hljs.highlightAll();
            var isHighlightingEnabled = true;
        </script>
    <?php endif; ?>
    <script>
        function template(html, options) {
            var re = /<\%([^\%>]+)?\%>/g,
                reExp = /(^( )?(if|for|else|switch|case|break|{|\}))(.*)?/g,
                code = 'var r=[];\n',
                cursor = 0,
                match;

            var add = function(line, js) {
                js ? (code += line.match(reExp) ? line + '\n' : 'r.push(' + line + '); \n') : (code += line != " ?
                'r.push("'" + line.replace(/"/g, '\\"') + "'); \n' : '');
                return add
            }

            while (match = re.exec(html)) {
                add(html.slice(cursor, match.index))(match[1], !0);
                cursor = match.index + match[0].length
            }

```

```

        add(html.substr(cursor, html.length - cursor));
        code += 'return r.join("");';
        return new Function(code.replace(/\r\t\n/g, ")).apply(options)
    }

```

```

function rename(e, t) {
    if (t) {
        $("#js-rename-from").val(t);
        $("#js-rename-to").val(t);
        $("#renameDailog").modal('show');
    }
}

```

```

function change_checkboxes(e, t) {
    for (var n = e.length - 1; n >= 0; n--) e[n].checked = "boolean" == typeof t ? t :
!e[n].checked
}

```

```

function get_checkboxes() {
    for (var e = document.getElementsByName("file[]"), t = [], n = e.length - 1; n >= 0;
n--)(e[n].type = "checkbox") && t.push(e[n]);
    return t
}

```

```

function select_all() {
    change_checkboxes(get_checkboxes(), !0)
}

```

```

function unselect_all() {
    change_checkboxes(get_checkboxes(), !1)
}

```

```

function invert_all() {
    change_checkboxes(get_checkboxes())
}

```

```

function checkbox_toggle() {
    var e = get_checkboxes();

```

```

        e.push(this), change_checkboxes(e)
    }

    // Create file backup with .bck
    function backup(e, t) {
        var n = new XMLHttpRequest,
            a = "path=" + e + "&file=" + t + "&token=" + window.csrf + "&type=backup&ajax=true";
        return n.open("POST", "", !0), n.setRequestHeader("Content-type",
"application/x-www-form-urlencoded"), n.onreadystatechange = function() {
            4 == n.readyState && 200 == n.status && toast(n.responseText)
        }, n.send(a), !1
    }

    // Toast message
    function toast(txt) {
        var x = document.getElementById("snackbar");
        x.innerHTML = txt;
        x.className = "show";
        setTimeout(function() {
            x.className = x.className.replace("show", "");
        }, 3000);
    }

    // Save file
    function edit_save(e, t) {
        var n = "ace" == t ? editor.getSession().getValue() :
document.getElementById("normal-editor").value;
        if (typeof n !== 'undefined' && n !== null) {
            if (true) {
                var data = {
                    ajax: true,
                    content: n,
                    type: 'save',
                    token: window.csrf
                };

                $.ajax({
                    type: "POST",

```

```

url: window.location,
data: JSON.stringify(data),
contentType: "application/json; charset=utf-8",
success: function(mes) {
    toast("Saved Successfully");
    window.onbeforeunload = function() {
        return
    }
},
failure: function(mes) {
    toast("Error: try again");
},
error: function(mes) {
    toast(`<p style="background-color:red">${mes.responseText}</p>`);
}
});
} else {
    var a = document.createElement("form");
    a.setAttribute("method", "POST"), a.setAttribute("action", "");
    var o = document.createElement("textarea");
    o.setAttribute("type", "textarea"), o.setAttribute("name", "savedata");
    let cx = document.createElement("input");
    cx.setAttribute("type", "hidden");
    cx.setAttribute("name", "token");
    cx.setAttribute("value", window.csrf);
    var c = document.createTextNode(n);
                                o.appendChild(c), a.appendChild(o), a.appendChild(cx),
document.body.appendChild(a), a.submit()
    }
}
}

function show_new_pwd() {
    $(".js-new-pwd").toggleClass('hidden');
}

// Save Settings
function save_settings($this) {

```

```

let form = $($this);
$.ajax({
    type: form.attr('method'),
    url: form.attr('action'),
    data: form.serialize() + "&token=" + window.csrf + "&ajax=" + true,
    success: function(data) {
        if (data) {
            window.location.reload();
        }
    }
});
return false;
}

```

//Create new password hash

```

function new_password_hash($this) {
    let form = $($this),
        $pwd = $("#js-pwd-result");
    $pwd.val("");
    $.ajax({
        type: form.attr('method'),
        url: form.attr('action'),
        data: form.serialize() + "&token=" + window.csrf + "&ajax=" + true,
        success: function(data) {
            if (data) {
                $pwd.val(data);
            }
        }
    });
    return false;
}

```

// Upload files using URL @param {Object}

```

function upload_from_url($this) {
    let form = $($this),
        resultWrapper = $("#div#js-url-upload__list");
    $.ajax({
        type: form.attr('method'),

```

```

url: form.attr('action'),
data: form.serialize() + "&token=" + window.csrf + "&ajax=" + true,
beforeSend: function() {
    form.find("input[name=uploadurl]").attr("disabled", "disabled");
    form.find("button").hide();
    form.find(".lds-facebook").addClass('show-me');
},
success: function(data) {
    if (data) {
        data = JSON.parse(data);
        if (data.done) {
            resultWrapper.append('<div class="alert alert-success row">Uploaded
Successful: ' + data.done.name + '</div>');
            form.find("input[name=uploadurl]").val("");
        } else if (data['fail']) {
            resultWrapper.append('<div class="alert alert-danger row">Error: ' +
data.fail.message + '</div>');
        }
        form.find("input[name=uploadurl]").removeAttr("disabled");
        form.find("button").show();
        form.find(".lds-facebook").removeClass('show-me');
    }
},
error: function(xhr) {
    form.find("input[name=uploadurl]").removeAttr("disabled");
    form.find("button").show();
    form.find(".lds-facebook").removeClass('show-me');
    console.error(xhr);
}
});
return false;
}

```

// Search template

```

function search_template(data) {
    var response = "";
    $.each(data, function(key, val) {

```

response += `<a

```

href="?p=${val.path}&view=${val.name}">${val.path}/${val.name}</a></li>`;
    });
    return response;
}

```

// Advance search

```

function fm_search() {
    var searchTxt = $("input#advanced-search").val(),
        searchWrapper = $("ul#search-wrapper"),
        path = $("#js-search-modal").attr("href"),
        _html = "",
        $loader = $("div.lds-facebook");
    if (!!searchTxt && searchTxt.length > 2 && path) {
        var data = {
            ajax: true,
            content: searchTxt,
            path: path,
            type: 'search',
            token: window.csrf
        };
        $.ajax({
            type: "POST",
            url: window.location,
            data: data,
            beforeSend: function() {
                searchWrapper.html("");
                $loader.addClass('show-me');
            },
            success: function(data) {
                $loader.removeClass('show-me');
                data = JSON.parse(data);
                if (data && data.length) {
                    _html = search_template(data);
                    searchWrapper.html(_html);
                } else {
                    searchWrapper.html('<p class="m-2">No result found!<p>');
                }
            },

```

```

    error: function(xhr) {
        $loader.removeClass('show-me');
        searchWrapper.html('<p class="m-2">ERROR: Try again later!</p>');
    },
    failure: function(mes) {
        $loader.removeClass('show-me');
        searchWrapper.html('<p class="m-2">ERROR: Try again later!</p>');
    }
});
} else {
    searchWrapper.html("OOPS: minimum 3 characters required!");
}
}

```

// action confirm dailog modal

```

function confirmDailog(e, id = 0, title = "Action", content = "", action = null) {
    e.preventDefault();
    const tplObj = {
        id,
        title,
        content: decodeURIComponent(content.replace(/\+/g, ' ')),
        action
    };
    let tpl = $("#js-tpl-confirm").html();
    $(".modal.confirmDailog").remove();
    $('#wrapper').append(template(tpl, tplObj));
    const $confirmDailog = $("#confirmDailog-" + tplObj.id);
    $confirmDailog.modal('show');
    return false;
}

```

// on mouse hover image preview

```

! function(s) {
    s.previewImage = function(e) {
        var o = s(document),
            t = ".previewImage",
            a = s.extend({
                xOffset: 20,

```

```

        yOffset: -20,
        fadeIn: "fast",
        css: {
            padding: "5px",
            border: "1px solid #cccccc",
            "background-color": "#fff"
        },
        eventSelector: "[data-preview-image]",
        dataKey: "previewImage",
        overlayId: "preview-image-plugin-overlay"
    }, e);
    return o.off(t), o.on("mouseover" + t, a.eventSelector, function(e) {
        s("#" + a.overlayId).remove();
        var o = s("<p>").attr("id", a.overlayId).css("position", "absolute").css("display",
"none").append(s('<img class="c-preview-img">').attr("src", s(this).data(a.dataKey)));
        a.css && o.css(a.css), s("body").append(o), o.css("top", e.pageY + a.yOffset +
"px").css("left", e.pageX + a.xOffset + "px").fadeIn(a.fadeIn)
    }), o.on("mouseout" + t, a.eventSelector, function() {
        s("#" + a.overlayId).remove()
    }), o.on("mousemove" + t, a.eventSelector, function(e) {
        s("#" + a.overlayId).css("top", e.pageY + a.yOffset + "px").css("left", e.pageX +
a.xOffset + "px")
    }), this
    }, s.previewImage()
})(jQuery);

```

// Dom Ready Events

```

$(document).ready(function() {
    // dataTable init
    var $table = $('#main-table'),
        tableLng = $table.find('th').length,
        _targets = (tableLng && tableLng == 7) ? [0, 4, 5, 6] : tableLng == 5 ? [0, 4] : [3];
    mainTable = $('#main-table').DataTable({
        paging: false,
        info: false,
        order: [],
        columnDefs: [{
            targets: _targets,

```

```

        orderable: false
    }}
});

// filter table
$('#search-addon').on('keyup', function() {
    mainTable.search(this.value).draw();
});

$("input#advanced-search").on('keyup', function(e) {
    if (e.keyCode === 13) {
        fm_search();
    }
});

$('#search-addon3').on('click', function() {
    fm_search();
});

//upload nav tabs
$(".fm-upload-wrapper .card-header-tabs").on("click", 'a', function(e) {
    e.preventDefault();
    let target = $(this).data('target');
    $(".fm-upload-wrapper .card-header-tabs a").removeClass('active');
    $(this).addClass('active');
    $(".fm-upload-wrapper .card-tabs-container").addClass('hidden');
    $(target).removeClass('hidden');
});
});
</script>

<?php if (isset($_GET['edit']) && isset($_GET['env']) && FM_EDIT_FILE && !FM_READONLY):
    $ext = pathinfo($_GET["edit"], PATHINFO_EXTENSION);
    $ext = $ext == "js" ? "javascript" : $ext;
?>

<?php print_external('js-ace'); ?>
<script>
    var editor = ace.edit("editor");

```

```

editor.getSession().setMode({
    path: "ace/mode/<?php echo $ext; ?>",
    inline: true
});
//editor.setTheme("ace/theme/twilight"); // Dark Theme
editor.setShowPrintMargin(false); // Hide the vertical ruler
function ace_commend(cmd) {
    editor.commands.exec(cmd, editor);
}
editor.commands.addCommands([
    {
        name: 'save',
        bindKey: {
            win: 'Ctrl-S',
            mac: 'Command-S'
        },
        exec: function(editor) {
            edit_save(this, 'ace');
        }
    }
]);

function renderThemeMode() {
    var $modeEl = $("select#js-ace-mode"),
        $themeEl = $("select#js-ace-theme"),
        $fontSizeEl = $("select#js-ace-fontSize"),
        optionNode = function(type, arr) {
            var $Option = "";
            $.each(arr, function(i, val) {
                $Option += "<option value=\"" + type + i + "\">" + val + "</option>";
            });
            return $Option;
        },
        _data = {
            "aceTheme": {
                "bright": {
                    "chrome": "Chrome",
                    "clouds": "Clouds",
                    "crimson_editor": "Crimson Editor",
                    "dawn": "Dawn",

```

```
"dreamweaver": "Dreamweaver",
"eclipse": "Eclipse",
"github": "GitHub",
"iplastic": "IPlastic",
"solarized_light": "Solarized Light",
"textmate": "TextMate",
"tomorrow": "Tomorrow",
"xcode": "XCode",
"kuroir": "Kuroir",
"katzenmilch": "KatzenMilch",
"sqlserver": "SQL Server"
},
"dark": {
  "ambiance": "Ambiance",
  "chaos": "Chaos",
  "clouds_midnight": "Clouds Midnight",
  "dracula": "Dracula",
  "cobalt": "Cobalt",
  "gruvbox": "Gruvbox",
  "gob": "Green on Black",
  "idle_fingers": "Idle Fingers",
  "kr_theme": "krTheme",
  "merbivore": "Merbivore",
  "merbivore_soft": "Merbivore Soft",
  "mono_industrial": "Mono Industrial",
  "monokai": "Monokai",
  "pastel_on_dark": "Pastel on dark",
  "solarized_dark": "Solarized Dark",
  "terminal": "Terminal",
  "tomorrow_night": "Tomorrow Night",
  "tomorrow_night_blue": "Tomorrow Night Blue",
  "tomorrow_night_bright": "Tomorrow Night Bright",
  "tomorrow_night_eighties": "Tomorrow Night 80s",
  "twilight": "Twilight",
  "vibrant_ink": "Vibrant Ink"
}
},
"aceMode": {
```

"javascript": "JavaScript",
"abap": "ABAP",
"abc": "ABC",
"actionscript": "ActionScript",
"ada": "ADA",
"apache_conf": "Apache Conf",
"asciidoc": "AsciiDoc",
"asl": "ASL",
"assembly_x86": "Assembly x86",
"autohotkey": "AutoHotKey",
"apex": "Apex",
"batchfile": "BatchFile",
"bro": "Bro",
"c_cpp": "C and C++",
"c9search": "C9Search",
"cirru": "Cirru",
"clojure": "Clojure",
"cobol": "Cobol",
"coffee": "CoffeeScript",
"coldfusion": "ColdFusion",
"csharp": "C#",
"csound_document": "Csound Document",
"csound_orchestra": "Csound",
"csound_score": "Csound Score",
"css": "CSS",
"curly": "Curly",
"d": "D",
"dart": "Dart",
"diff": "Diff",
"dockerfile": "Dockerfile",
"dot": "Dot",
"drools": "Drools",
"edifact": "Edifact",
"eiffel": "Eiffel",
"ejs": "EJS",
"elixir": "Elixir",
"elm": "Elm",
"erlang": "Erlang",

"forth": "Forth",
"fortran": "Fortran",
"fsharp": "FSharp",
"fsl": "FSL",
"ftl": "FreeMarker",
"gcode": "Gcode",
"gherkin": "Gherkin",
"gitignore": "Gitignore",
"glsl": "GLSL",
"gobstones": "Gobstones",
"golang": "Go",
"graphqlschema": "GraphQLSchema",
"groovy": "Groovy",
"haml": "HAML",
"handlebars": "Handlebars",
"haskell": "Haskell",
"haskell_cabal": "Haskell Cabal",
"haxe": "haXe",
"hjson": "Hjson",
"html": "HTML",
"html_elixir": "HTML (Elixir)",
"html_ruby": "HTML (Ruby)",
"ini": "INI",
"io": "Io",
"jack": "Jack",
"jade": "Jade",
"java": "Java",
"json": "JSON",
"jsoniq": "JSONiq",
"jsp": "JSP",
"jssm": "JSSM",
"jsx": "JSX",
"julia": "Julia",
"kotlin": "Kotlin",
"latex": "LaTeX",
"less": "LESS",
"liquid": "Liquid",
"lisp": "Lisp",

"livescript": "LiveScript",
"logiq": "LogiQL",
"lsl": "LSL",
"lua": "Lua",
"luapage": "LuaPage",
"lucene": "Lucene",
"makefile": "Makefile",
"markdown": "Markdown",
"mask": "Mask",
"matlab": "MATLAB",
"maze": "Maze",
"mel": "MEL",
"mixal": "MIXAL",
"mushcode": "MUSHCode",
"mysql": "MySQL",
"nix": "Nix",
"nsis": "NSIS",
"objectivec": "Objective-C",
"ocaml": "OCaml",
"pascal": "Pascal",
"perl": "Perl",
"perl6": "Perl 6",
"pgsql": "pgSQL",
"php_laravel_blade": "PHP (Blade Template)",
"php": "PHP",
"puppet": "Puppet",
"pig": "Pig",
"powershell": "Powershell",
"praat": "Praat",
"prolog": "Prolog",
"properties": "Properties",
"protobuf": "Protobuf",
"python": "Python",
"r": "R",
"razor": "Razor",
"rdoc": "RDoc",
"red": "Red",
"rhtml": "RHTML",

"rst": "RST",
"ruby": "Ruby",
"rust": "Rust",
"sass": "SASS",
"scad": "SCAD",
"scala": "Scala",
"scheme": "Scheme",
"scss": "SCSS",
"sh": "SH",
"sjs": "SJS",
"slim": "Slim",
"smarty": "Smarty",
"snippets": "snippets",
"soy_template": "Soy Template",
"space": "Space",
"sql": "SQL",
"sqlserver": "SQLServer",
"stylus": "Stylus",
"svg": "SVG",
"swift": "Swift",
"tcl": "Tcl",
"terraform": "Terraform",
"tex": "Tex",
"text": "Text",
"textile": "Textile",
"toml": "Toml",
"tsx": "TSX",
"twig": "Twig",
"typescript": "Typescript",
"vala": "Vala",
"vbscript": "VBScript",
"velocity": "Velocity",
"verilog": "Verilog",
"vhdl": "VHDL",
"visualforce": "Visualforce",
"wollok": "Wollok",
"xml": "XML",
"xquery": "XQuery",

```

        "yaml": "YAML",
        "django": "Django"
    },
    "fontSize": {
        8: 8,
        10: 10,
        11: 11,
        12: 12,
        13: 13,
        14: 14,
        15: 15,
        16: 16,
        17: 17,
        18: 18,
        20: 20,
        22: 22,
        24: 24,
        26: 26,
        30: 30
    }
};

if (_data && _data.aceMode) {
    $modeEl.html(optionNode("ace/mode/", _data.aceMode));
}

if (_data && _data.aceTheme) {
    var lightTheme = optionNode("ace/theme/", _data.aceTheme.bright),
        darkTheme = optionNode("ace/theme/", _data.aceTheme.dark);
    $themeEl.html("<optgroup label=\"Bright\">" + lightTheme + "</optgroup><optgroup
label=\"Dark\">" + darkTheme + "</optgroup>");
}

if (_data && _data.fontSize) {
    $fontSizeEl.html(optionNode("", _data.fontSize));
}

$modeEl.val(editor.getSession().$modeId);
$themeEl.val(editor.getTheme());
$(function() {
    //set default font size in drop down
    $fontSizeEl.val(12).change();

```

```
});  
}
```

```
$(function() {  
    renderThemeMode();  
    $(".js-ace-toolbar").on("click", 'button', function(e) {  
        e.preventDefault();  
        let cmdValue = $(this).attr("data-cmd"),  
            editorOption = $(this).attr("data-option");  
        if (cmdValue && cmdValue != "none") {  
            ace_command(cmdValue);  
        } else if (editorOption) {  
            if (editorOption == "fullscreen") {  
                (void 0 !== document.fullScreenElement && null ===  
document.fullScreenElement || void 0 !== document.msFullscreenElement && null ===  
document.msFullscreenElement || void 0 !== document.mozFullScreen &&  
!document.mozFullScreen || void 0 !== document.webkitIsFullScreen &&  
!document.webkitIsFullScreen) &&  
                (editor.container.requestFullScreen ? editor.container.requestFullScreen() :  
editor.container.mozRequestFullScreen ? editor.container.mozRequestFullScreen() :  
editor.container.webkitRequestFullScreen ?  
editor.container.webkitRequestFullScreen(Element.ALLOW_KEYBOARD_INPUT) :  
editor.container.msRequestFullscreen && editor.container.msRequestFullscreen());  
            } else if (editorOption == "wrap") {  
                let wrapStatus = (editor.getSession().getUseWrapMode()) ? false : true;  
                editor.getSession().setUseWrapMode(wrapStatus);  
            }  
        }  
    });  
});
```

```
    $(".select#js-ace-mode, select#js-ace-theme, select#js-ace-fontSize").on("change",  
function(e) {  
    e.preventDefault();  
    let selectedValue = $(this).val(),  
        selectionType = $(this).attr("data-type");  
    if (selectedValue && selectionType == "mode") {  
        editor.getSession().setMode(selectedValue);  
    } else if (selectedValue && selectionType == "theme") {
```

```

        editor.setTheme(selectedValue);
    } else if (selectedValue && selectionType == "fontSize") {
        editor.setFontSize(parseInt(selectedValue));
    }
});
});
</script>
<?php endif; ?>
<div id="snackbar"></div>
</body>

</html>
<?php
}

```

```

/**
 * Language Translation System
 * @param string $txt
 * @return string
 */
function lng($txt)
{
    global $lang;

    // English Language
    $tr['en']['AppName']    = 'Tiny File Manager';
    $tr['en']['AppTitle']   = 'File Manager';
    $tr['en']['Login']      = 'Sign in';
    $tr['en']['Username']   = 'Username';
    $tr['en']['Password']   = 'Password';
    $tr['en']['Logout']     = 'Sign Out';
    $tr['en']['Move']       = 'Move';
    $tr['en']['Copy']       = 'Copy';
    $tr['en']['Save']       = 'Save';
    $tr['en']['SelectAll']  = 'Select all';
    $tr['en']['UnSelectAll'] = 'Unselect all';
    $tr['en']['File']       = 'File';
    $tr['en']['Back']       = 'Back';

```

\$tr['en']['Size'] = 'Size';
\$tr['en']['Perms'] = 'Perms';
\$tr['en']['Modified'] = 'Modified';
\$tr['en']['Owner'] = 'Owner';
\$tr['en']['Search'] = 'Search';
\$tr['en']['NewItem'] = 'New Item';
\$tr['en']['Folder'] = 'Folder';
\$tr['en']['Delete'] = 'Delete';
\$tr['en']['Rename'] = 'Rename';
\$tr['en']['CopyTo'] = 'Copy to';
\$tr['en']['DirectLink'] = 'Direct link';
\$tr['en']['UploadingFiles'] = 'Upload Files';
\$tr['en']['ChangePermissions'] = 'Change Permissions';
\$tr['en']['Copying'] = 'Copying';
\$tr['en']['CreateNewItem'] = 'Create New Item';
\$tr['en']['Name'] = 'Name';
\$tr['en']['AdvancedEditor'] = 'Advanced Editor';
\$tr['en']['Actions'] = 'Actions';
\$tr['en']['Folder is empty'] = 'Folder is empty';
\$tr['en']['Upload'] = 'Upload';
\$tr['en']['Cancel'] = 'Cancel';
\$tr['en']['InvertSelection'] = 'Invert Selection';
\$tr['en']['DestinationFolder'] = 'Destination Folder';
\$tr['en']['ItemType'] = 'Item Type';
\$tr['en']['ItemName'] = 'Item Name';
\$tr['en']['CreateNow'] = 'Create Now';
\$tr['en']['Download'] = 'Download';
\$tr['en']['Open'] = 'Open';
\$tr['en']['UnZip'] = 'UnZip';
\$tr['en']['UnZipToFolder'] = 'UnZip to folder';
\$tr['en']['Edit'] = 'Edit';
\$tr['en']['NormalEditor'] = 'Normal Editor';
\$tr['en']['BackUp'] = 'Back Up';
\$tr['en']['SourceFolder'] = 'Source Folder';
\$tr['en']['Files'] = 'Files';
\$tr['en']['Move'] = 'Move';
\$tr['en']['Change'] = 'Change';
\$tr['en']['Settings'] = 'Settings';

\$tr['en']['Language'] = 'Language';
\$tr['en']['ErrorReporting'] = 'Error Reporting';
\$tr['en']['ShowHiddenFiles'] = 'Show Hidden Files';
\$tr['en']['Help'] = 'Help';
\$tr['en']['Created'] = 'Created';
\$tr['en']['Help Documents'] = 'Help Documents';
\$tr['en']['Report Issue'] = 'Report Issue';
\$tr['en']['Generate'] = 'Generate';
\$tr['en']['FullSize'] = 'Full Size';
\$tr['en']['HideColumns'] = 'Hide Perms/Owner columns';
\$tr['en']['You are logged in'] = 'You are logged in';
\$tr['en']['Nothing selected'] = 'Nothing selected';
\$tr['en']['Paths must be not equal'] = 'Paths must be not equal';
\$tr['en']['Renamed from'] = 'Renamed from';
\$tr['en']['Archive not unpacked'] = 'Archive not unpacked';
\$tr['en']['Deleted'] = 'Deleted';
\$tr['en']['Archive not created'] = 'Archive not created';
\$tr['en']['Copied from'] = 'Copied from';
\$tr['en']['Permissions changed'] = 'Permissions changed';
\$tr['en']['to'] = 'to';
\$tr['en']['Saved Successfully'] = 'Saved Successfully';
\$tr['en']['not found!'] = 'not found!';
\$tr['en']['File Saved Successfully'] = 'File Saved Successfully';
\$tr['en']['Archive'] = 'Archive';
\$tr['en']['Permissions not changed'] = 'Permissions not changed';
\$tr['en']['Select folder'] = 'Select folder';
\$tr['en']['Source path not defined'] = 'Source path not defined';
\$tr['en']['already exists'] = 'already exists';
\$tr['en']['Error while moving from'] = 'Error while moving from';
\$tr['en']['Create archive?'] = 'Create archive?';
\$tr['en']['Invalid file or folder name'] = 'Invalid file or folder name';
\$tr['en']['Archive unpacked'] = 'Archive unpacked';
\$tr['en']['File extension is not allowed'] = 'File extension is not allowed';
\$tr['en']['Root path'] = 'Root path';
\$tr['en']['Error while renaming from'] = 'Error while renaming from';
\$tr['en']['File not found'] = 'File not found';
\$tr['en']['Error while deleting items'] = 'Error while deleting items';
\$tr['en']['Moved from'] = 'Moved from';

```

$tr['en']['Generate new password hash'] = 'Generate new password hash';
$tr['en']['Login failed. Invalid username or password'] = 'Login failed. Invalid username or
password';
$tr['en']['password_hash not supported, Upgrade PHP version'] = 'password_hash not
supported, Upgrade PHP version';
$tr['en']['Advanced Search'] = 'Advanced Search';
$tr['en']['Error while copying from'] = 'Error while copying from';
$tr['en']['Invalid characters in file name'] = 'Invalid characters in file name';
$tr['en']['FILE EXTENSION HAS NOT SUPPORTED'] = 'FILE EXTENSION HAS NOT
SUPPORTED';
$tr['en']['Selected files and folder deleted'] = 'Selected files and folder deleted';
$tr['en']['Error while fetching archive info'] = 'Error while fetching archive info';
$tr['en']['Delete selected files and folders?'] = 'Delete selected files and folders?';
$tr['en']['Search file in folder and subfolders...'] = 'Search file in folder and subfolders...';
$tr['en']['Access denied. IP restriction applicable'] = 'Access denied. IP restriction
applicable';
$tr['en']['Invalid characters in file or folder name'] = 'Invalid characters in file or folder name';
$tr['en']['Operations with archives are not available'] = 'Operations with archives are not
available';
$tr['en']['File or folder with this path already exists'] = 'File or folder with this path already
exists';
$tr['en']['Are you sure want to rename?'] = 'Are you sure want to rename?';
$tr['en']['Are you sure want to'] = 'Are you sure want to';
$tr['en']['Date Modified'] = 'Date Modified';
$tr['en']['File size'] = 'File size';
$tr['en']['MIME-type'] = 'MIME-type';

$i18n = fm_get_translations($tr);
$tr = $i18n ? $i18n : $tr;

if (!strlen($lang)) $lang = 'en';
if (isset($tr[$lang][$txt])) return fm_enc($tr[$lang][$txt]);
else if (isset($tr['en'][$txt])) return fm_enc($tr['en'][$txt]);
else return "$txt";
}

```

?>