



wpmudev

Developer application tasks

Finding a good superhero to ward off evil in our endless pursuit for WordPress nirvana is darn hard work, you've got those that want to be, but can't ever be. Those that don't want to be but it's a means to an end and they'll do what they have to. You then have those that aspire to shine a light in the midsts of night, to fend of evil wherever it may lurk, to save one WordPressian at a time, to be something great.

Be the hero we need, come join us in our mission at WPMU DEV!

As a side note, these tasks are designed to help us find those shining examples of awesomeness. They're pretty simple tasks too. Skipping these will ensure that we don't consider your application. This is super simple stuff, it helps us to see if you understand the basics before we consider an interview.

Let's get started, eh!



Your completed task should be sent directly to us. Not on Github or similar. You can either answer inline, or you can create a separate text file and number the answers.

1) How would you improve upon this code?

```
<?php
global $wpdb;
$query = "SELECT * FROM $wpdb->postmeta WHERE meta_key = '' .
$my_option_name . ''";
$results = $wpdb->get_results( $query );
```

2). Now take a go at improving this code, reckon you can figure it out?

```
<!-- Assume $your_name holds the previously submitted answer (if any), or
empty string -->
<label>
    <span><?php _e('Your name', 'test'); ?</span>
    <input type="text" name="your_name" value="<?php echo
$your_name; ?>" />
</label>
```

3). Prepare the following code for localization, as best and concise as you can. Use 'test' as your translation domain:

```
<?php
$apples = rand(0,5);
echo "We have {$apples} apples!";
```

4). Nothing is ever perfect, right?

Tell us what we're doing in this code, and how can it be improved?

```
<?php
// Assume user privileges and nonce are correctly checked
$options = array();
$options['option1'] = isset( $_POST['option1'] ) ? $_POST['option1'] : "";
update_option('my_option_key', $options);
```

5). Can you think of a scenario where code such as this would be an anti-pattern?

```
<?php
// file: uninstall.php
if ( is_multisite() ) {
    global $wpdb;
    $blogs = $wpdb->get_results( "SELECT blog_id FROM
{$wpdb->blogs}", ARRAY_A );

    if ( ! empty( $blogs ) ) {
        foreach ( $blogs as $blog ) {
            switch_to_blog( $blog['blog_id'] );
            delete_option( 'option_name' );
        }
    }
} else {
    delete_option( 'option_name' );
}
```

6). Can you explain what might go wrong with this bit of javascript code?

```
if ("object" === typeof a && "stuff" in a) { alert(a.stuff); }
```

7). Can you spot a problem with this bit of javascript code?

```
// file: my-js-file.js  
function get_number (string) {  
    return parseInt(string);  
}
```

8). Can you spot a problem with this bit of code?

```
<?php
function is_not_ready_to_go() {
    return empty( get_site_option( 'my_plugin_option' ) );
}
```

9). How would you refactor this bit of code to be more concise in WordPress context?

```
?php
// Assume $map looks like this:
// $map = array(
//     array( 'name' => 'Name 1', 'something else' => 'whatever' ),
//     array( 'name' => 'Name 2', 'something else' => 'whatever' ),
//     array( 'name' => 'Name 3', 'something else' => 'whatever' ),
//     array( 'name' => 'Name 4', 'something else' => 'whatever' ),
//     array( 'name' => 'Name 5', 'something else' => 'whatever' ),
//     array( 'name' => 'Name 6', 'something else' => 'whatever' ),
//     array( 'name' => 'Name 7', 'something else' => 'whatever' ),
//     array( 'name' => 'Name 8', 'something else' => 'whatever' ),
//     array( 'name' => 'Name 9', 'something else' => 'whatever' ),
// );
$names = array();
foreach ( $map as $data ) {
    $names[] = $data['name'];
}
}
```

10). How would you refactor this bit of code in a plugin you're assigned?

```
<?php
// Assume $path_piece1 = '/test/dir1/';
// Assume $path_piece2 = 'subdir2\file.txt';
$path = str_replace( '\\', '/', $path_piece1 . $path_piece2 );
```

11). Is there anything you would improve in this piece of code?

```
<?php
// Assume js/my-custom-posts-script.js is present, and does something on
posts list page (edit.php)
function my_admin_scripts_inclusion_proc () {
    wp_enqueue_script( 'my-custom-posts-script', plugins_url(
'js/my-custom-posts-script.js', __FILE__ ) );
}
add_action('admin_enqueue_scripts', 'my_admin_scripts_inclusion_proc');
```

See, super simple stuff so far, right?

Don't worry, your work won't always be this simple and boring. If you make the cut you'll be working on leading edge WordPress plugins, doing super interesting stuff, always being challenged to be incredible.

Now get this task sheet in with your application, let's get this show on the road!

=====

If you don't have room for you answer above, add them here, be sure to number them!

