

JavaScript for WordPress

Zac Gordon @zgordon

javascriptforwp.com

Whatcha Need for Today

- Local WP Environment (DesktopServer, Local, Chassis)
- Running WordPress Site with Dummy Content
- Code Editor / IDE
- Node & NPM Running
- Comfortable with JavaScript, webpack, command line
- <https://github.com/zgordon/frontend-masters-jsforwp>



Why Should (JavaScript) **Developers** **Care** About **WordPress**?





Adam Silverstein: JavaScript in WordPress Core – Past, Present and Future

SHARE

WORDPRESS

Key history events:



WordPress

Language	Code Lines	Comment Lines	Comment Ratio	JS Lines	Total Lines	Total Percentage
JS	20,000	10,000	34.7%	40,000	60,000	34.7%
PHP	10,000	5,000	17.9%	10,000	20,000	17.9%
SQL	10,000	5,000	17.9%	10,000	20,000	17.9%
Other	20,000	10,000	34.7%	40,000	60,000	34.7%

0:40 / 14:46

HD

Adam Silverstein: JavaScript in WordPress Core – Past, Present and Future

<https://wordpress.tv/2017/06/16/adam-silverstein-javascript-in-wordpress-core-past-present-and-future/>

×

Saved

<

You are customizing
Menus


⚙️ ⓘ

Menu Locations >

+ Add a Menu

WordPress

Just another WordPress site



He

Cras
dapi
nibh
eget
sit a
rutr


Prac
cons
elit.
elit l

Edit Page [Add New](#)

[Screen Options](#)[Help](#)

Adding Music

Permalink: <http://yourflow.org/adding-music/> [Edit](#)

 Add Media

[Visual](#)[Text](#)

[b](#) [i](#) [link](#) [b-quote](#) [del](#) [ins](#) [img](#) [ul](#) [ol](#) [li](#) [code](#) [more](#) [close tags](#)



Word count: 0

Last edited by markadmin on November 28, 2016 at 12:27 pm

Page Settings

Show page title

☒ Yes ☐ No

Show page title on this page

Publish

[Preview Changes](#)

Status: **Published** [Edit](#)


Visibility: **Public** [Edit](#)

Published on: **Nov 29, 2016 @ 12:27**
[Edit](#)


[Move to Trash](#)[Update](#)

Page Attributes

Parent

(no parent) 

Template

Default Template 

Order

0

Need help? Use the Help tab above the screen title.

f "Sample Post"

Revision
Slider



Autosave by **Bob Writer**
1 min ago (4 Apr @ 07:52)



f "Sample Post"

Revision
Slider



Autosave by **Bob Writer**
1 min ago (4 Apr @ 07:52)



[← ALL POSTS](#)[DRAFTS](#) (4)[POST](#)

Sarah Gooding

pollyplummer.wordpress...

▼ DRAFT SAVED

A FEW SECONDS AGO

Preview

Publish



Categories & Tags

Uncategorized

Featured Image



Hello Calypso!

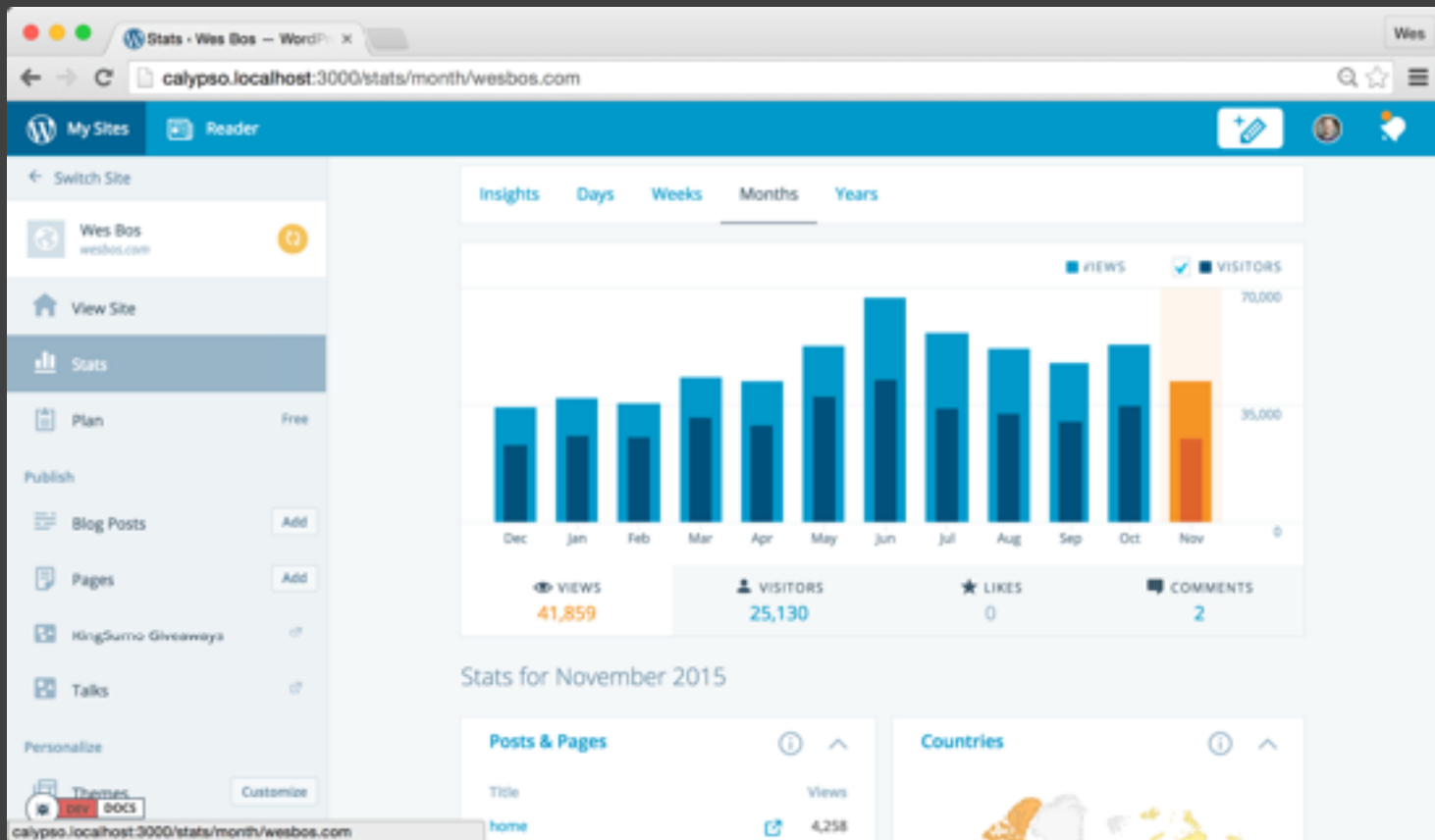
Visual

HTML



Paragraph

**B***I*



[Home](#)[Manage](#)

Managing 3 blogs
out of 2

[All](#)[Public](#)[Private](#)[Blogs](#)[Posts](#)[Comments](#)[Pages](#)[Stats](#)[Upgrades](#)[Sharing](#)[Users](#)**KTLS**ktls.wordpress.com**5 New Comments** · 43759 Posts · 132 Followers[Customize](#)[Settings](#)[Upgrades](#)**Audibles - SI.com**si11.wordpress.com**5 New Comments** · 2845 Posts · 225 Followers[Customize](#)[Settings](#)[Upgrades](#)**ProBasketballTalk**probasketballtalk.wordpress.com**5 New Comments** · 61332 Posts · 4732 Followers[Customize](#)[Settings](#)[Upgrades](#)**Travellego**travellego.wordpress.com





The **Influence** of
Calypso should **NOT** be
Underestimated

HOMEWORK ASSIGNMENT:

Learn JavaScript, deeply.

State of the Word 2015



Here's the State of the Word presentation, delivered on Saturday, and the following day.



The JavaScript for WordPress Master Course

javascriptforwp.com

LEARN

JAVASCRIPT

DEEPLY



Current JS + WP Projects

Core:

Customizer Improvements

JS Widgets & Hooks

“Gutenberg” - SPA Editor

Real World:

JS Plugins w API Support

JS Driven Themes

Decoupled Integrations



What **JS Framework**
Do We Adopt **Next?**



JavaScript is becoming
More Important & Integrated
with **WordPress**

Course Outline

- Intro to JavaScript in WordPress
- Enqueuing JavaScript in a Theme
- Enqueuing JavaScript in a Plugin
- AJAX in WordPress
- The WordPress REST API
- Using the WP REST API Inside WordPress
- Using the WP REST API Outside of WordPress

Enqueuing JavaScript

Enqueuing JavaScript

The process of loading JavaScript into a theme or the admin area via PHP functions.

wp_enqueue_script();

wp_localize_script();

wp_enqueue_script(

 'unique-handle-name',

 get_stylesheet_directory_uri() . '/path/to/file.js',

 ['dependency-handles'],

 'version',

 loadInFooter // boolean true/false

);

```
wp_enqueue_script(  
    'unique-handle-name',  
    get_stylesheet_directory_uri() . '/path/to/file.js',  
    [ 'dependency-handles' ],  
    'version',  
    loadInFooter // boolean true/false  
);
```

```
wp_enqueue_script(  
    'unique-handle-name',  
    get_stylesheet_directory_uri() . '/path/to/file.js',  
    [ 'dependency-handles' ],  
    'version',  
    loadInFooter // boolean true/false  
);
```

```
wp_enqueue_script(  
    'unique-handle-name',  
    get_stylesheet_directory_uri() . '/path/to/file.js',  
    [ 'dependency-handles' ],  
    'version',  
    loadInFooter // boolean true/false  
);
```

```
wp_enqueue_script(  
    'unique-handle-name',  
    get_stylesheet_directory_uri() . '/path/to/file.js',  
    [ 'dependency-handles' ],  
    'version',  
    loadInFooter // boolean true/false  
);
```

```
wp_enqueue_script(  
    'unique-handle-name',  
    get_stylesheet_directory_uri() . '/path/to/file.js',  
    [ 'dependency-handles' ],  
    'version',  
    loadInFooter // boolean true/false  
);
```

wp_enqueue_script(

 'unique-handle-name',

 get_stylesheet_directory_uri() . '/path/to/file.js',

 ['dependency-handles'],

 'version',

 loadInFooter // boolean true/false

);

```
wp_localize_script(  
    'js-handle',  
    'name_for_js',  
    [  
        'site_url' => esc_url( home_url() ),  
        'site_name' => get_bloginfo( 'name' )  
    ]  
);
```



```
wp_localize_script(  
    'js-handle',  
    'name_for_js',  
    [  
        'site_url' => esc_url( home_url() ),  
        'site_name' => get_bloginfo( 'name' )  
    ]  
);
```

```
wp_localize_script(  
    'js-handle',  
    'name_for_js',  
    [  
        'site_url' => esc_url( home_url() ),  
        'site_name' => get_bloginfo( 'name' )  
    ]  
);
```

```
wp_localize_script(  
    'js-handle',  
    'name_for_js',  
    [  
        'site_url' => esc_url( home_url() ),  
        'site_name' => get_bloginfo( 'name' )  
    ]  
);
```

```
wp_localize_script(
```

```
    'js-handle',
```

```
    'name_for_js',
```

```
    [
```

```
        'site_url' => esc_url( home_url() ),
```

```
        'site_name'=> get_bloginfo( 'name' )
```

```
    ]
```

```
); // Can use name_for_js.site_url in our JS
```

functions.php:

```
function my_enqueue_scripts() {  
    wp_enqueue_script();  
    wp_localize_script();  
}
```

```
add_action( 'wp_enqueue_scripts', 'my_enqueue_scripts' );
```

Enqueuing JS in a Theme

1. Enqueuing JavaScript
2. Enqueuing Multiple Files (& Using Dependencies)
3. Making jQuery a Dependency (& jQuery Wrapper)
4. Enqueuing External JavaScript (remove Version)
5. Conditionally Enqueuing JavaScript
6. Localizing Scripts

Enqueuing JS in a Theme Docs

1. `wp_enqueue_script` - https://developer.wordpress.org/reference/functions/wp_enqueue_script/
2. JS Included in WP - https://developer.wordpress.org/reference/functions/wp_enqueue_script/#default-scripts-included-and-registered-by-wordpress
3. jQuery noConflict - https://developer.wordpress.org/reference/functions/wp_enqueue_script/#comment-1473
4. WP Conditionals - <https://developer.wordpress.org/themes/basics/conditional-tags/>
5. `wp_localize_script` - https://developer.wordpress.org/reference/functions/wp_localize_script/

PRACTICE - Enqueuing JS in a Theme

1. Enqueuing JavaScript
2. Enqueuing Multiple Files (& Using Dependencies)
3. Making jQuery a Dependency (& jQuery Wrapper)
4. Enqueuing External JavaScript (remove Version)
5. Conditionally Enqueuing JavaScript
6. Localizing Scripts

Course Outline

- ~~Intro to JavaScript in WordPress~~
- ~~Enqueuing JavaScript in a Theme~~
- Enqueuing JavaScript in a Plugin

```
wp_enqueue_script(  
    'unique-handle-name',  
    plugins_url( '/path/to/file.js', __FILE__ ),  
    [ 'dependency-handles' ],  
    'version',  
    loadInFooter // boolean true/false  
);
```

plugin-file.php:

```
function my_enqueue_scripts() {  
    wp_enqueue_script();  
    wp_localize_script();  
}
```

```
add_action( 'wp_enqueue_scripts', 'my_enqueue_scripts' );
```

plugin-file.php:

```
function my_enqueue_scripts() {  
    wp_enqueue_script();  
    wp_localize_script();  
}
```

```
add_action( 'wp_enqueue_scripts', 'my_enqueue_scripts' );
```

```
add_action( 'admin_enqueue_scripts', 'my_enqueue_scripts' );
```

```
function my_enqueue_scripts( $hook ) {  
    if( $hook != 'toplevel_page_my-admin-page' ) {  
        return;  
    }  
    wp_enqueue_script();  
}  
  
add_action( 'admin_enqueue_scripts', 'my_enqueue_scripts' );
```

```
$main_menu_url = 'my-admin-page';  
  
function my_enqueue_scripts( $hook ) {  
    global $main_menu_url;  
  
    if( $hook != 'toplevel_page_' . $main_menu_url ) {  
        return;  
    }  
  
    wp_enqueue_script();  
}  
  
add_action( 'admin_enqueue_scripts', 'my_enqueue_scripts' );
```

Enqueuing JS in a Plugin

1. Enqueuing JavaScript on Frontend in Theme
2. Enqueuing JavaScript in Admin Area
3. Enqueuing JavaScript in Admin Area on Plugin Page

Enqueuing JS in a Plugin

1. plugins_url - https://developer.wordpress.org/reference/functions/plugins_url/
2. Admin Enqueue Hook - https://developer.wordpress.org/reference/hooks/admin_enqueue_scripts/
3. Enqueue on Admin Page - https://codex.wordpress.org/Plugin_API/Action_Reference/admin_enqueue_scripts#Example:_Target_a_Specific_Admin_Page

PRACTICE - Enqueuing JS in a Plugin

1. Enqueuing JavaScript on Frontend from Plugin
2. Enqueuing JavaScript in Admin Area
3. Enqueuing JavaScript in Admin Area on Plugin Page

Course Outline

- ~~Intro to JavaScript in WordPress~~
- ~~Enqueuing JavaScript in a Theme~~
- ~~Enqueuing JavaScript in a Plugin~~
- AJAX in WordPress

AJAX in WordPress



AJAX in WordPress is A Little Weird

AJAX in WordPress

1. Need to pass Nonce & URL from PHP to JS
2. Need to make AJAX call with Nonce and URL
3. Need WordPress Hooks to route the call
4. Need a PHP function to process the call
5. Need JS respond to the call

Step 1: In Plugin / Theme functions.php File

```
wp_localize_script(
    'jsforwp-frontend-js',
    'jsforwp_globals',
    [
        'ajax_url'    => admin_url( 'admin-ajax.php' ),
        'nonce'       => wp_create_nonce( 'nonce_name' )
    ]
);
```

Step 2: In Frontend JavaScript

```
$.ajax({  
  type : 'post',  
  dataType : 'json',  
  url : jsforwp_globals.ajax_url,  
  data : {  
    action: 'jsforwp_add_like',  
    _ajax_nonce: jsforwp_globals.nonce  
  }, ...
```

Step 3: In Plugin / Theme functions.php File

```
function jsforwp_add_like( ) {  
    check_ajax_referer( 'nonce_name' );  
    // Do whatevs...  
    $response[ 'success' ] = true;  
    $response = json_encode( $response );  
    echo $response;  
    die();  
}  
  
add_action( 'wp_ajax_jsforwp_add_like', 'jsforwp_add_like' );
```


Step 4: In Frontend JavaScript

```
$.ajax({
```

```
    ...
```

```
    success: function( response ) {
```

```
        if( 'success' === response.type ) {
```

```
            // Do something with response.whatevs
```

```
        }
```

```
    else {
```

```
        // Something went wrong
```

```
    }
```

```
}
```

AJAX in WordPress

1. Need to pass Nonce & URL from PHP to JS
2. Need to make AJAX call with Nonce and URL
3. Need WordPress Hooks to route the call
4. Need a PHP function to process the call
5. Need JS respond to the call

PRACTICE - AJAX in WordPress

1. Make a “Like This Site” Plugin
2. Save likes as number in wp_options table
3. Have a link to “Like this Site” show up on front end
4. Display total number of likes next to link
5. When click on link increase count with AJAX
6. Do it first with jQuery then with Axios

Course Outline

- ~~Intro to JavaScript in WordPress~~
- ~~Enqueuing JavaScript in a Theme~~
- ~~Enqueuing JavaScript in a Plugin~~
- ~~AJAX in WordPress~~
- The WordPress REST API

The WP REST API

The WordPress REST API

/wp-json/wp/v2/posts

/wp-json/wp/v2/revisions

/wp-json/wp/v2/categories

/wp-json/wp/v2/tags

/wp-json/wp/v2/pages

/wp-json/wp/v2/comments

/wp-json/wp/v2/taxonomies

/wp-json/wp/v2/media

/wp-json/wp/v2/users

/wp-json/wp/v2/types

/wp-json/wp/v2/statuses

/wp-json/wp/v2/settings

/wp-json/custom/v1/something

/wp-json/another/v3/else

The WordPress REST API

To access all of the **posts** for your site:

<https://yoursite.com/wp/v2/posts>

To access a **post with an ID** of “1” for your site:

<https://yoursite.com/wp/v2/posts/1>

The WordPress REST API

To access all of the **pages** for your site:

<https://yoursite.com/wp-json/wp/v2/pages>

To access a **page with an ID** of “1” for your site:

<https://yoursite.com/wp-json/wp/v2/pages/1>



You can pass **arguments**
to API URLs to **customize**
what you get back.

The WordPress REST API

To access all of the **pages** for your site:

<https://yoursite.com/wp/v2/pages>

To access a **page with an ID** of “1” for your site:

<https://yoursite.com/wp/v2/pages/1>

The WordPress REST API

https://yoursite.com/wp/v2/posts

? per_page=10

&_embed

&search=hello%20world

&orderby=title

&order=asc

<https://developer.wordpress.org/rest-api/reference/posts/#arguments>

PRACTICE - WP REST API

1. Access API endpoints in the browser
2. Access API endpoints w arguments
3. Access custom API endpoints in the browser

Accessing the API Inside of WordPress

Using API Inside of WordPress

1. Decided what data you need
2. Find the endpoint you need
3. Decide what arguments to pass
4. Use the Backbone API Client to get (save, edit, delete) the data
5. Use JS for the rest...

Accessing the API Outside of WordPress

Using API Inside of WordPress

1. Decided what data you need
2. Find the endpoint you need
3. Decide what arguments to pass
4. Authenticate if need to save, edit, delete
5. Use JS for the rest...

JavaScript for WordPress

Zac Gordon @zgordon

javascriptforwp.com