

# WP REST API and AJAX FORMS

## Step-By-Step Course

Craig West

### DESIRED OUTCOME:

- To understand WP REST API.
- How to use FETCH API to get and handle JSON data.
- How to use to get data and to display data using JavaScript.
- How to create new endpoints and edit existing ones in the built in REST API for GET and POST requests.
- Create forms that use AJAX to WP REST API.
- Secure FORMS with WP NONCE.

**Final project** is a form that creates a new post using a custom WP REST API endpoint that we will create that validates data types and has a NONCE to provide security from Cross Site Request Forgery.

Live site

<https://wp-html.co.uk/greece/>

## DOWNLOAD WORKSHOP PACK

<https://github.com/iwswordpress/WordCampGreece>

The pack contains all the code necessary to replicate the final project, the slides for the talk and a resource list.

*Much inspiration was drawn from the following playlist on YouTube, particularly lessons 14 onwards:*

[https://www.youtube.com/watch?v=c\\_piVnQrJuY&list=PLT9miexWCpPU3TtDIVxA765dh2MaJY5X3](https://www.youtube.com/watch?v=c_piVnQrJuY&list=PLT9miexWCpPU3TtDIVxA765dh2MaJY5X3)

*I highly recommend watching these tutorials.*

## DISABLING ENDPOINTS

<http://juha.blog/dev/wordpress/disable-wordpress-rest-api-endpoints-example-user-endpoint/>

### How to disable user endpoint

Easy way to solve this is to disable user endpoint (if you don't need it in your application). This can be done using `rest_endpoints` filter in your `functions.php`. Following filter will disable user endpoints. You can use same logic to any endpoint you want to close.

```
add_filter( 'rest_endpoints', function( $endpoints ){
    if ( isset( $endpoints['/wp/v2/users'] ) ) {
        unset( $endpoints['/wp/v2/users'] );
    }
    if ( isset( $endpoints['/wp/v2/users/(?P<id>[\d]+)'] ) ) {
        unset( $endpoints['/wp/v2/users/(?P<id>[\d]+)'] );
    }
    return $endpoints;
});
```

### How to disable entire REST API

If you don't need Rest API at all and you want to disable it for some reason you can use this snippet in your `functions.php`.

```
add_filter('rest_enabled', '_return_false');
add_filter('rest_jsonp_enabled', '_return_false');
```

# SET UP

## Installing a clone: (preferred method for workshop)

With a new empty installation of WP, add the ALL\_IN\_ONE plugin.

Import the WordCampAthens.wpress file. This should reproduce all of the site except for the custom MySQL table we use. (It won't matter if you don't have it).

Load the 01\_tblTest.sql script into your WP database.

You can log in as administrator:

**user: greece**

**pwd: wordcampGREECE2021**

**server.php** is and include that has the site URL as a variable so that you can change to your server in just one place.

```
<?php
    //$SITE = "https://49plus.co.uk/udemy-rest/";
    $SITE = site_url().'/';
?>
```

## Step by step

**PHP** has two folders:

1. Code – html,js,mu-plugins,php folders
2. sql – MySql script to custom test table.

**mu-plugins** has files that contain functions which will be automatically loaded like plugins but cannot be turned on or off unless removed.

**js** has a test js file for use with qp\_localise

**php** has php files for root of theme

**html** has some html files for lesson on the fetch api

You can use the functions to make a plugin or insert into theme functions.php but if not in a child theme they can be overwritten on a theme update.

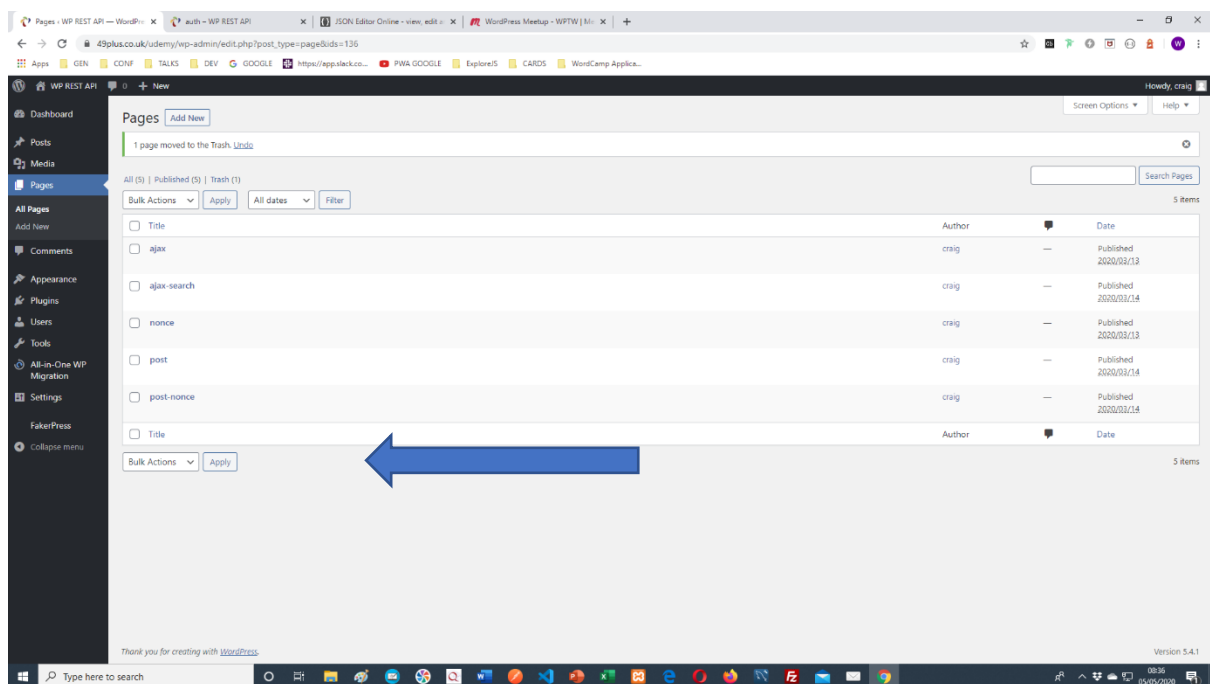
I use a free template called generatepress and create a child theme using the attached plugin.

Copy page-xxxx.php files into root of your theme.

### Create blank pages for the following:

ajax, ajax-search, post, nonce and post-nonce  
and add to menu if you wish...

The functions.php contains just code to generate a child theme called **child** from theme **generatepress**.



Test your WPsite/ajax gives you the same page as me.

### Enable <post name> in Settings > Permalinks

**server.php** is and include that has the site URL as a variable so that you can change to your server in just one place.

```
<?php
```

```
// $SITE = "https://49plus.co.uk/udemy-rest/";  
$SITE = site_url().'/';  
?>
```

If we have a file `page-ajax.php` and a blank page in WP ajax, then [www.site.com/ajax](http://www.site.com/ajax) will render that page.

**SQL** folder has one MySQL script for `01_tblTest` as sample data

### Chrome Extension - JSON Formatter

It is useful to have a JSON formatter in the browser. Search for *Chrome Extensions JSON Formatter* and install.

<https://chrome.google.com/webstore/detail/json-formatter/bcjindcccaagfpapjjmafapmmgkkhgoa?hl=en>

Or use <https://jsoneditoronline.org/> which we will need later anyway.

## What are REST/AJAX/JSON?

PowerPoint slide->

Roy Fielding propose REST as a guiding practice in his thesis in 2000:

[https://www.ics.uci.edu/~fielding/pubs/dissertation/rest\\_arch\\_style.htm](https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm)

By its nature, HTTP is very RESTful.

We will look at the wp-json object that is the REST API for WP.

Open <https://jsoneditoronline.org/>

In browser <https://wp-html.co.uk/greece/wp-json/> and copy and paste this into JSON EDITOR.

We can now see what the REST API contains...

Let us look at the 'hello world' of the REST API:

<https://wp-html.co.uk/greece/wp-json/wp/v2/posts>

We can now see what the REST API provides.

I will now go into detail...

- author
- authorName (added field)
- title.rendered

ACF fields needs ACF to REST plugin.

## **FETCH API**

### **Let's look at fetch.html, fetch404.html and fetch500.html**

This shows the nature of the fetch API and how data returned is initially a stream that needs to be converted to JSON (or TEXT, BLOB etc).

The use of **.then** and **.catch** is shown.

## **ENDPOINTS**

### **HTTP VERB (GET/POST/DELETE...) + URL = ENDPOINT**

One URL can have two endpoints if it is used as GET and POST.

The URLs below can be two endpoints 1) GET 2) POST

A selection of useful endpoints, shaded ones most useful:

<https://wp-html.co.uk/greece/wp-json/wp/v2/posts>

<https://wp-html.co.uk/greece/wp-json/wp/v2/posts?fields=authorName,id,excerpt,title,link,acf>  
(underscore before fields)

<https://wp-html.co.uk/greece/wp-json/wp/v2/posts?search=json>

<https://wp-html.co.uk/greece/wp-json/wp/v2/posts?order=asc>

<https://wp-html.co.uk/greece/wp-json/wp/v2/categories>

<https://wp-html.co.uk/greece/wp-json/wp/v2/categories/10>

[https://wp-html.co.uk/greece/wp-json/wp/v2/posts/?per\\_page=2](https://wp-html.co.uk/greece/wp-json/wp/v2/posts/?per_page=2)

<https://wp-html.co.uk/greece/wp-json/wp/v2/posts?categories=10>

<https://wp-html.co.uk/greece/wp-json/wp/v2/users>

<https://wp-html.co.uk/greece/wp-json/wp/v2/users/14>

**We can have our own custom endpoints:**

<https://wp-html.co.uk/greece/wp-json/wordcamp/v2/districts>

---

<https://wp-html.co.uk/greece/wp-json/wordcamp/v2/latest-posts/10>

(latest posts in category id=10)

---

<https://wp-html.co.uk/greece/wp-json/wordcamp/v2/totalusers>

(total number of users)

<https://jsoneditoronline.org/> to format it.

Look in Dev Tools > Console and you can see the output of an array of posts.

These are in rest-custom.php in mu-plugins.

Let's look at then now...

### **Advanced Custom Fields:**

To get the ACF fields to show in rest we need the additonal plugin ACF to RESTAPI.

REST is neabled by default in ACF. In the early days it had to be added to the parameter set.

<https://wp-html.co.uk/greece/wp-json/wp/v2/posts? fields=id,acf>

is a reduced REST response.

If we disable the ACT to REST API plugin we will not get the ACR field.

### **Custom Post Types:**

We have one already installed WORDCAMPS.

When we examine wp-json we will see a namespace for the endpoint associated with the cutom post type:

<https://wp-html.co.uk/greece/wp-json/wp/v2/wordcamps>



## CREATING ENDPOINTS IN WP REST API

**Sample page of GET requests for POSTS and DATA:**

<https://wp-html.co.uk/greece/ajax/>

We will look at this file to see how to process returned JSON data and render to the page...

**SEARCH FORM:**

<https://wp-html.co.uk/greece/ajax-search/>

**Create a new post using a custom endpoint:**

<https://49plus.co.uk/udemy-rest/post/>

**Examples of nonces**

<https://wp-html.co.uk/greece/post/>

**POST data to create a new post using nonces**

<https://wp-html.co.uk/greece/post-nonce/>