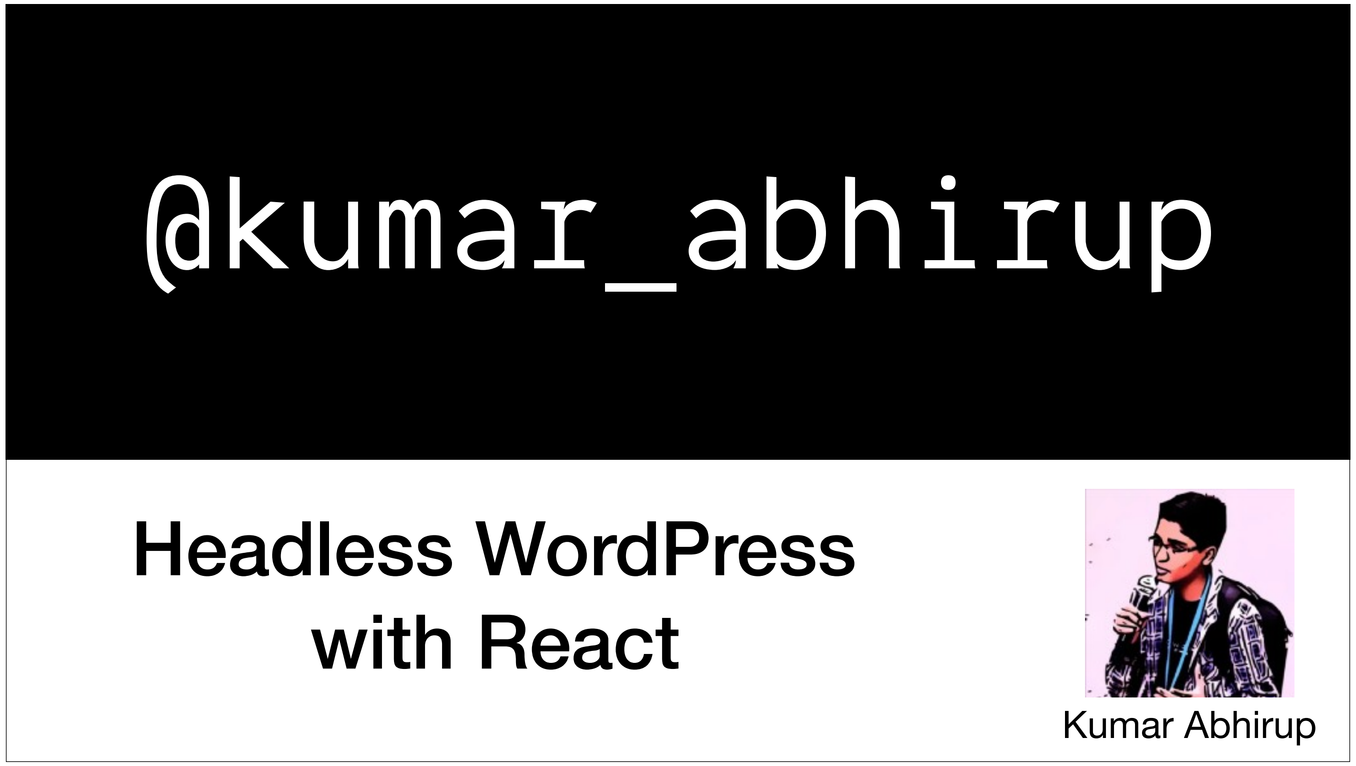
Headless WordPress with React

Kumar Abhirup

Hey all! I am Kumar Abhirup. I am a 10th Grader, 16 year-old.
And today we'll be learning about Headless WordPress with React.



@kumar_abhirup

Headless WordPress with React

Kumar Abhirup

Just before we start, letting you know that the link to the slides is pinned on my Twitter timeline. The slides are hosted are on a GitHub repository. There is README provided, which also contains links to many of the resources provided in the slides.

@kumar_abhirup

kumar.now.sh

Headless WordPress with React

Kumar Abhirup

kumar.now.sh is my personal website. If ever wanna contact or know more about me, you visit that link.

Enough is the intro! Now let us begin!

WHAT DO YOU THINK IS HEADLESS WORDPRESS?

Headless WordPress

How many of ever heard this term?

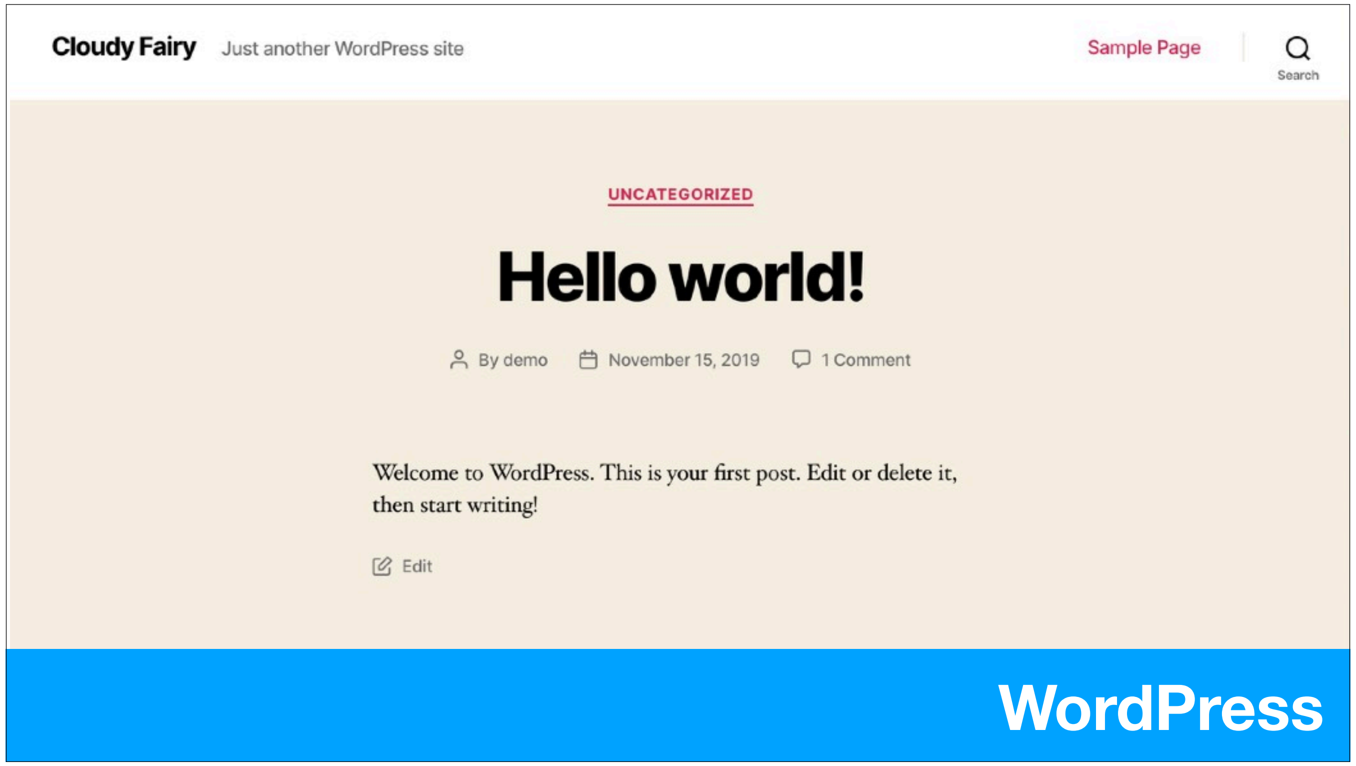
Cool.

So, Headless WordPress

WordPress without Frontend

Is a WordPress, WITHOUT FRONTEND.

When you think of WordPress, what you think of, is this.



Cloudy Fairy Just another WordPress site

UNCATEGORIZED

Hello world!

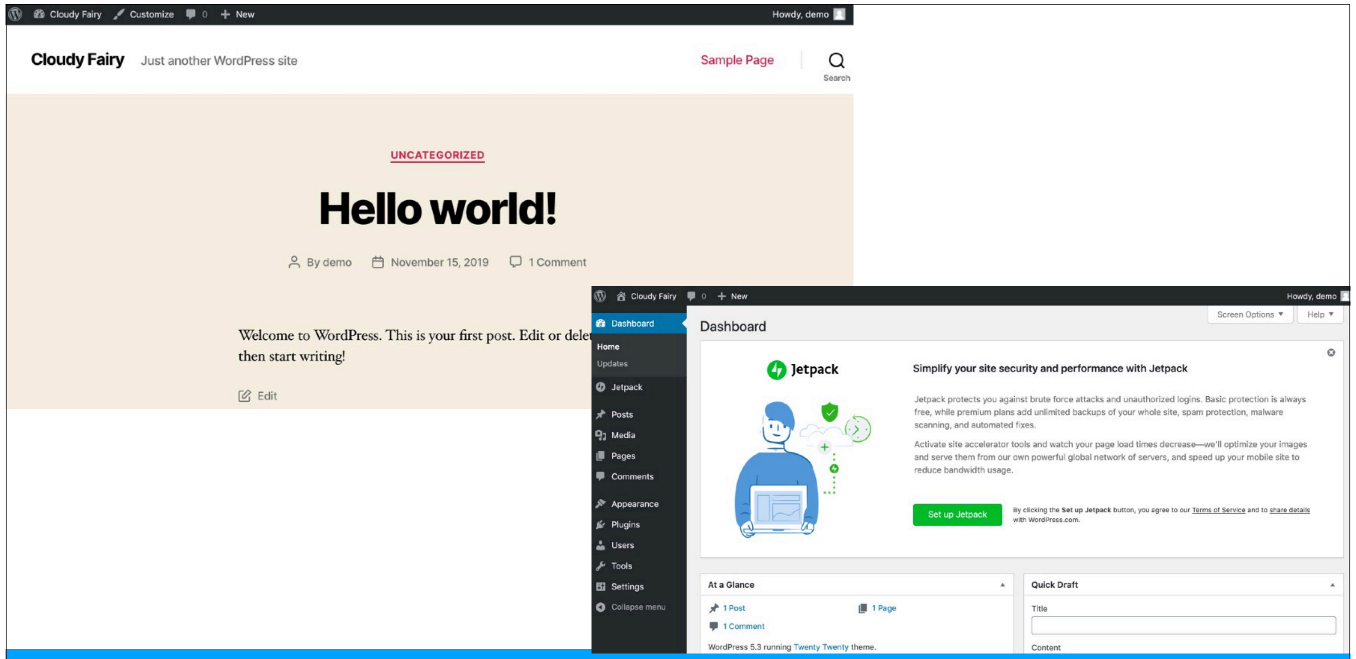
By demo November 15, 2019 1 Comment

Welcome to WordPress. This is your first post. Edit or delete it, then start writing!

Oh by the way, that is the cool new 2020 WordPress theme.

So, what you are seeing, is the frontend.

And, WordPress also has a,



Hello world!

Welcome to WordPress. This is your first post. Edit or delet
then start writing!

Dashboard

Simplify your site security and performance with Jetpack

Jetpack protects you against brute force attacks and unauthorized logins. Basic protection is always free, while premium plans add unlimited backups of your whole site, spam protection, malware scanning, and automated fixes.

Activate site accelerator tools and watch your page load times decrease—we'll optimize your images and serve them from our own powerful global network of servers, and speed up your mobile site to reduce bandwidth usage.

By clicking the **Set up Jetpack** button, you agree to our Terms of Service and to share details with WordPress.com.

WordPress 5.3 running Twenty Twenty theme.

Title:

Content:

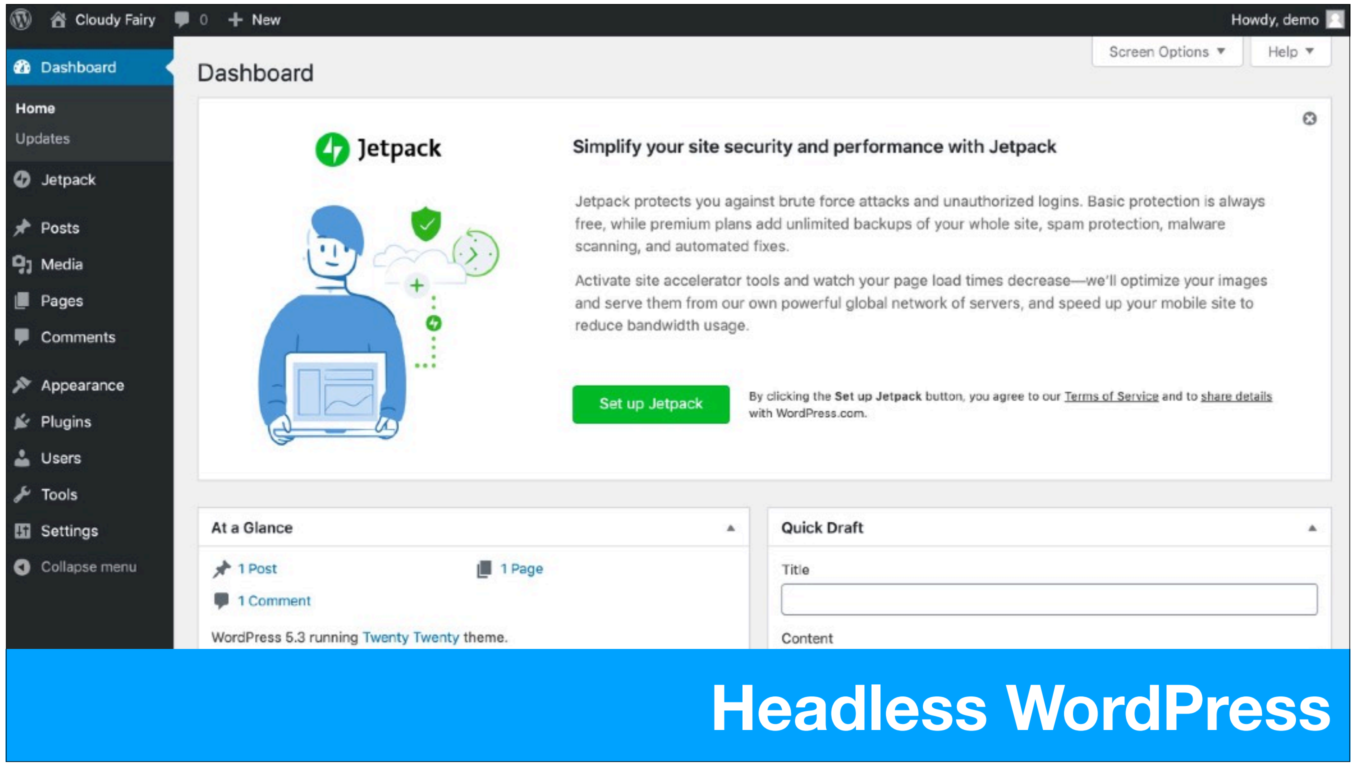
WordPress

BACKEND.

But what Headless WordPress means, is that there is no intricately tied frontend,

But what exists,

Is a backend!



Dashboard

Jetpack

Simplify your site security and performance with Jetpack

Jetpack protects you against brute force attacks and unauthorized logins. Basic protection is always free, while premium plans add unlimited backups of your whole site, spam protection, malware scanning, and automated fixes.

Activate site accelerator tools and watch your page load times decrease—we'll optimize your images and serve them from our own powerful global network of servers, and speed up your mobile site to reduce bandwidth usage.

By clicking the **Set up Jetpack** button, you agree to our Terms of Service and to share details with WordPress.com.

WordPress 5.3 running Twenty Twenty theme.

Headless WordPress

Okay, let me clear this.

Headless WordPress is about using WordPress only as a Content Management System that manages data.

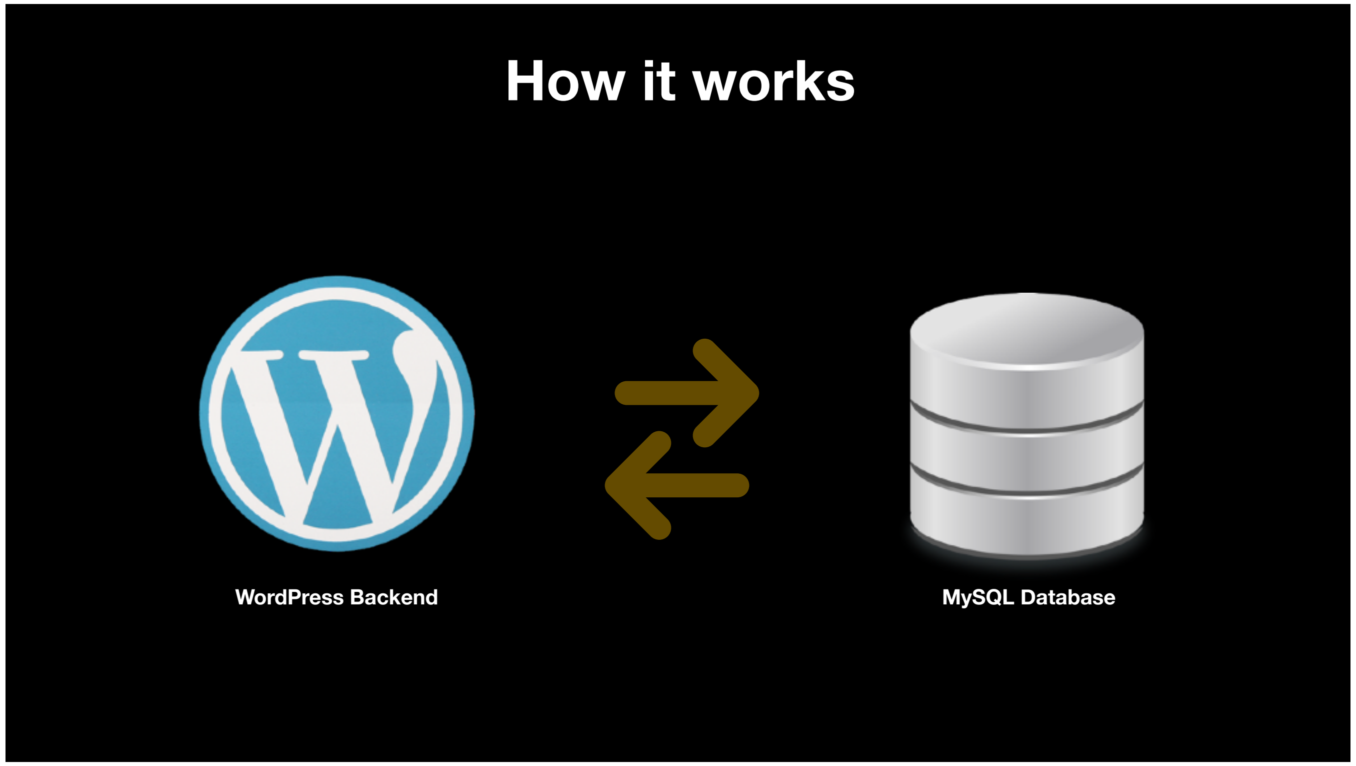
Headless WordPress does not decide the looks of the frontend.

Let us explore this in deep.

Here's how it works.

How it works

In WordPress, there is a WordPress backend, which is connected to a MySQL database.



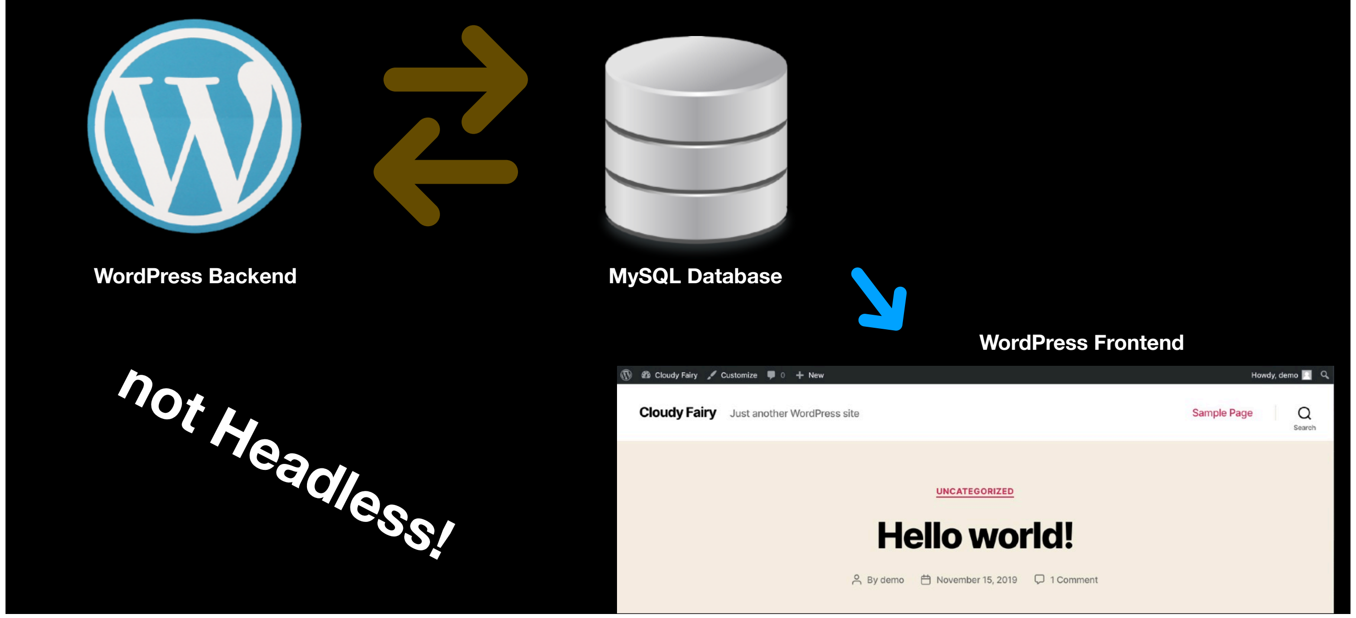
How it works

WordPress Backend

MySQL Database

And the content that you see in your frontend, is served by the same database.

How it works

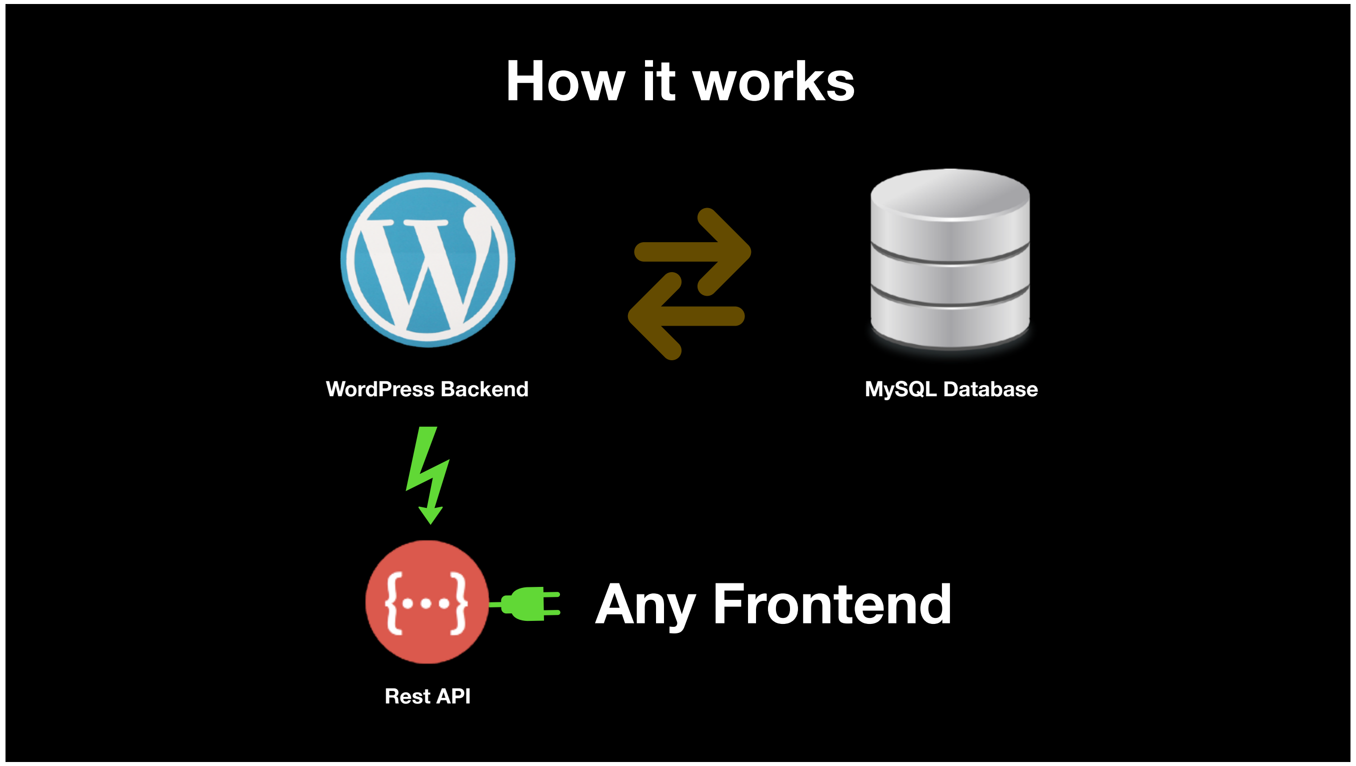


Just like that.

What you are seeing is

NOT HEADLESS WORDPRESS.

Headless WordPress does not have that frontend theme. But what it has, is an endpoint where it spits out JSON.



How it works

WordPress Backend

MySQL Database

Rest API

Any Frontend

Just like that.

One random question, how many of you are developers here?

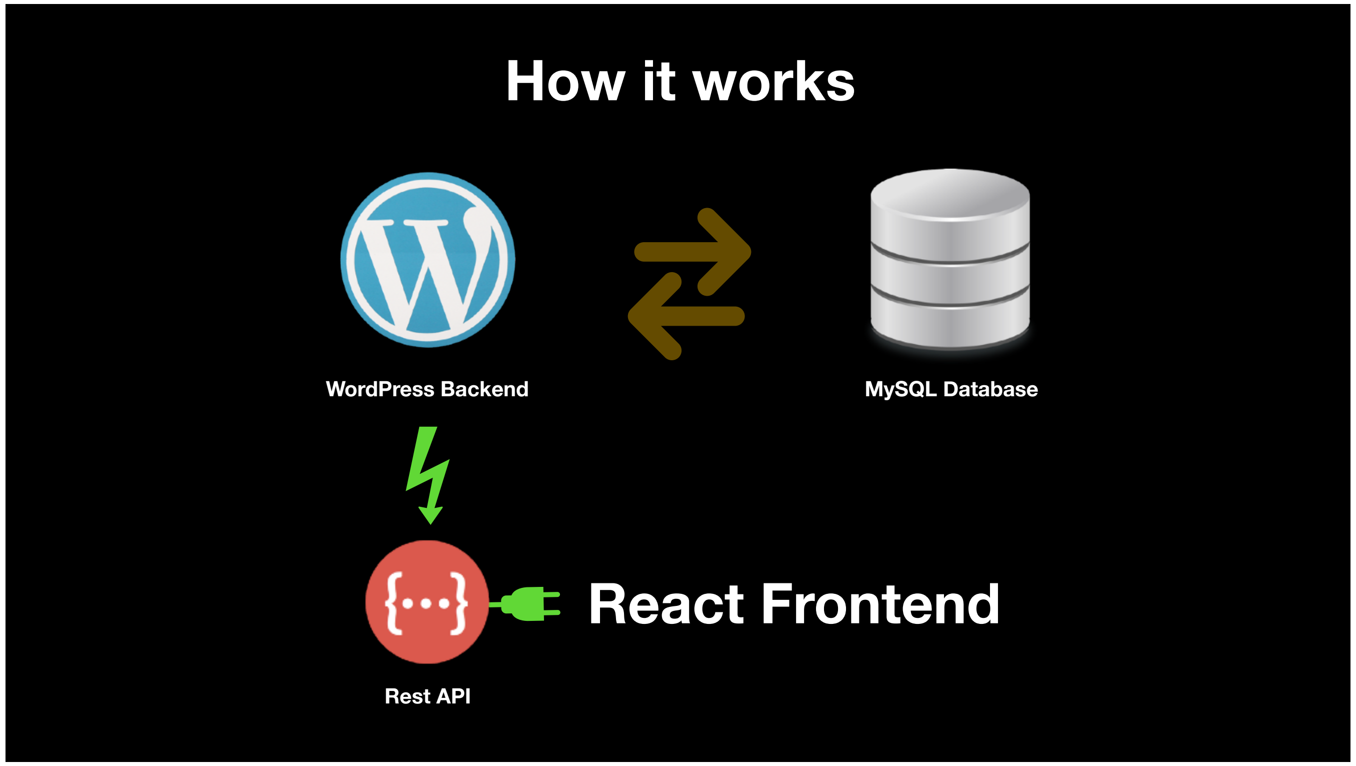
Okay, so how many of you have ever played with JSON endpoints? Like fetching, posting, putting and patching from some random public API or something…

Cool.

Same here. WordPress spits out data in form of JSON that you can fetch from

ANY FRONTEND!

And in this talk, I’ll be demonstrating you a,



How it works

WordPress Backend

MySQL Database

Rest API

React Frontend

REACT FRONTEND.

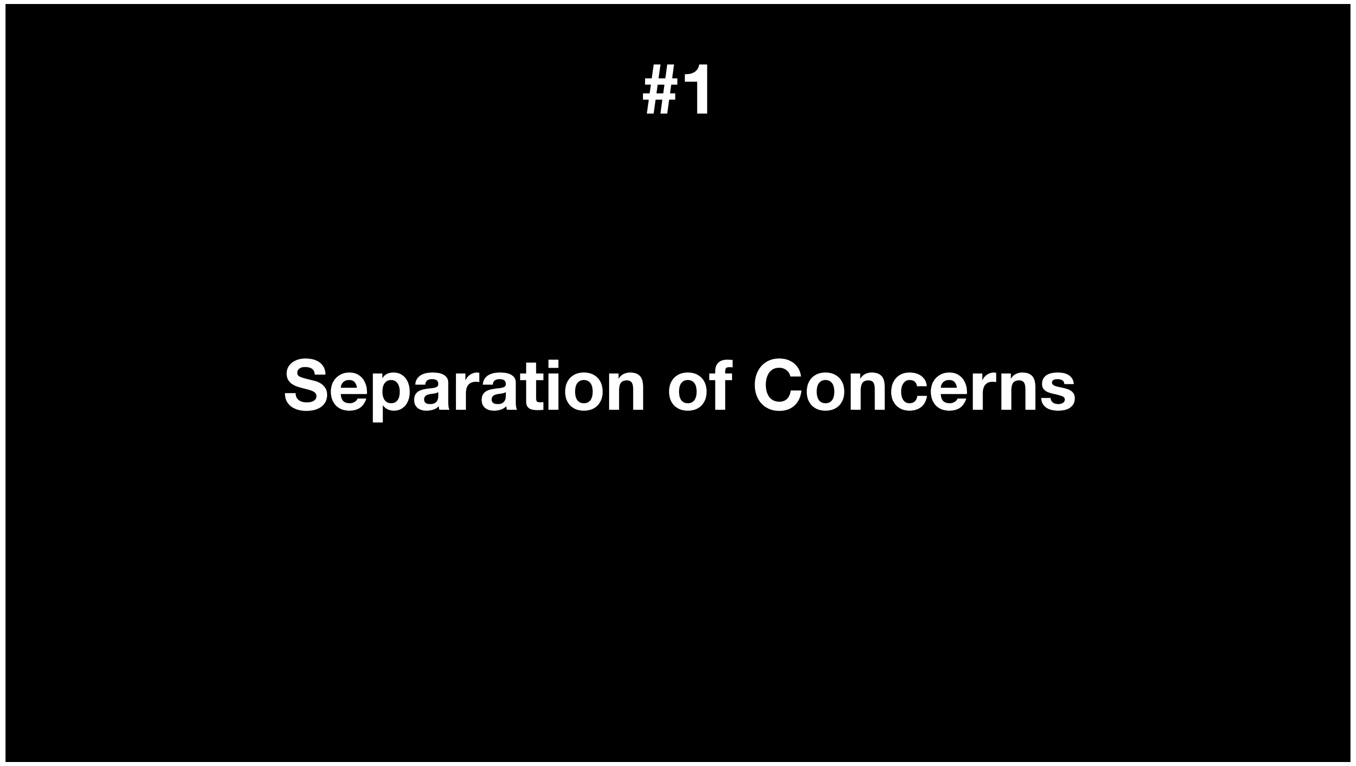
Okay, so I hope I was able to explain how this thing works, now let us talk about…

Why Headless?

WHY, WOULD SOMEONE EVER NEED HEADLESS WORDPRESS?

You can just have those WordPress themes do the work for you, right? Why headless?

REASON NUMBER ONE……..



#1

Separation of Concerns

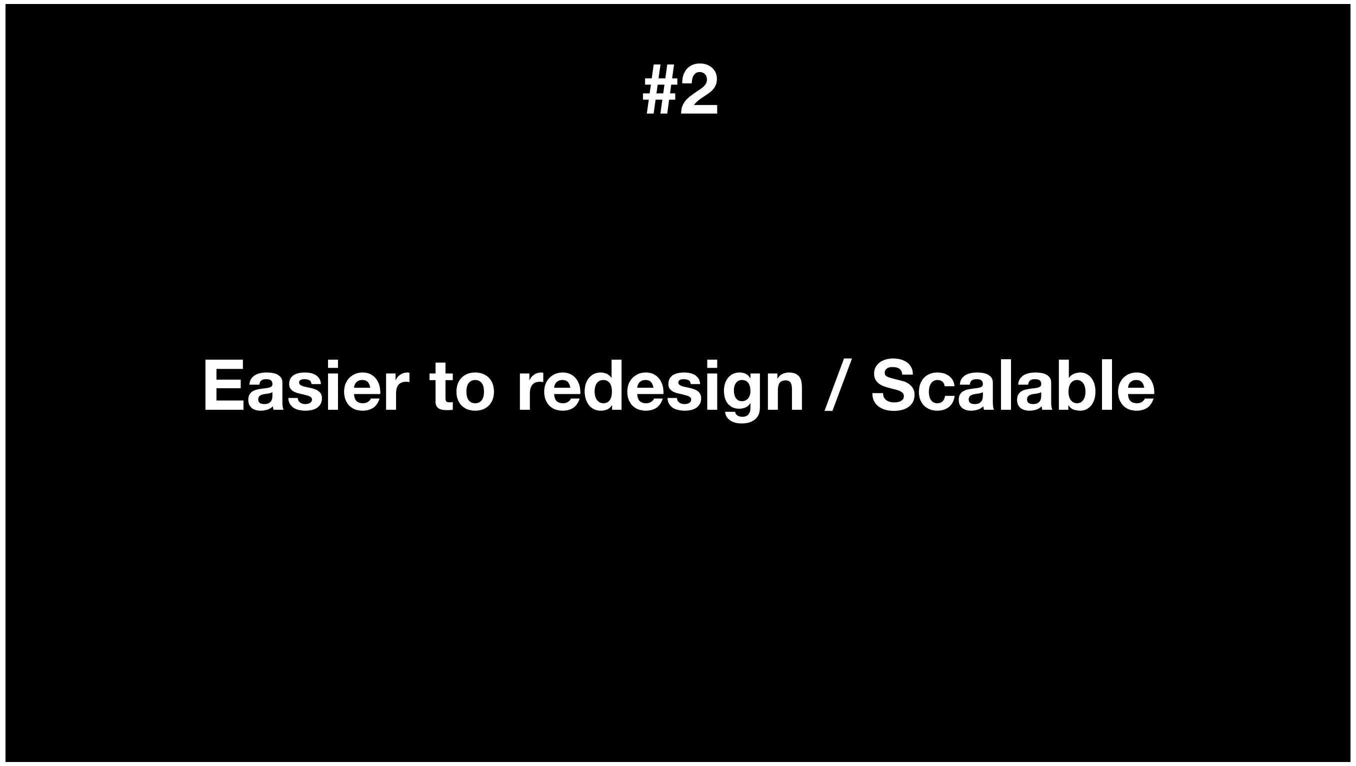
Separation of Concerns.

If you have been around playing with React, you might have heard this word a 100 times!

With Headless WordPress, you can have one team working solely on the frontend, and another team, working on backend!

All your frontend developers will only need to use the data they get from the endpoint.

And what WordPress developers will focus on, is the backend, security and API stuff.



#2

Easier to redesign / Scalable

If your frontend runs by a WordPress Backend, and if it is a React frontend, you know how scalable React is, right?

Components in React are reusable and scalable.

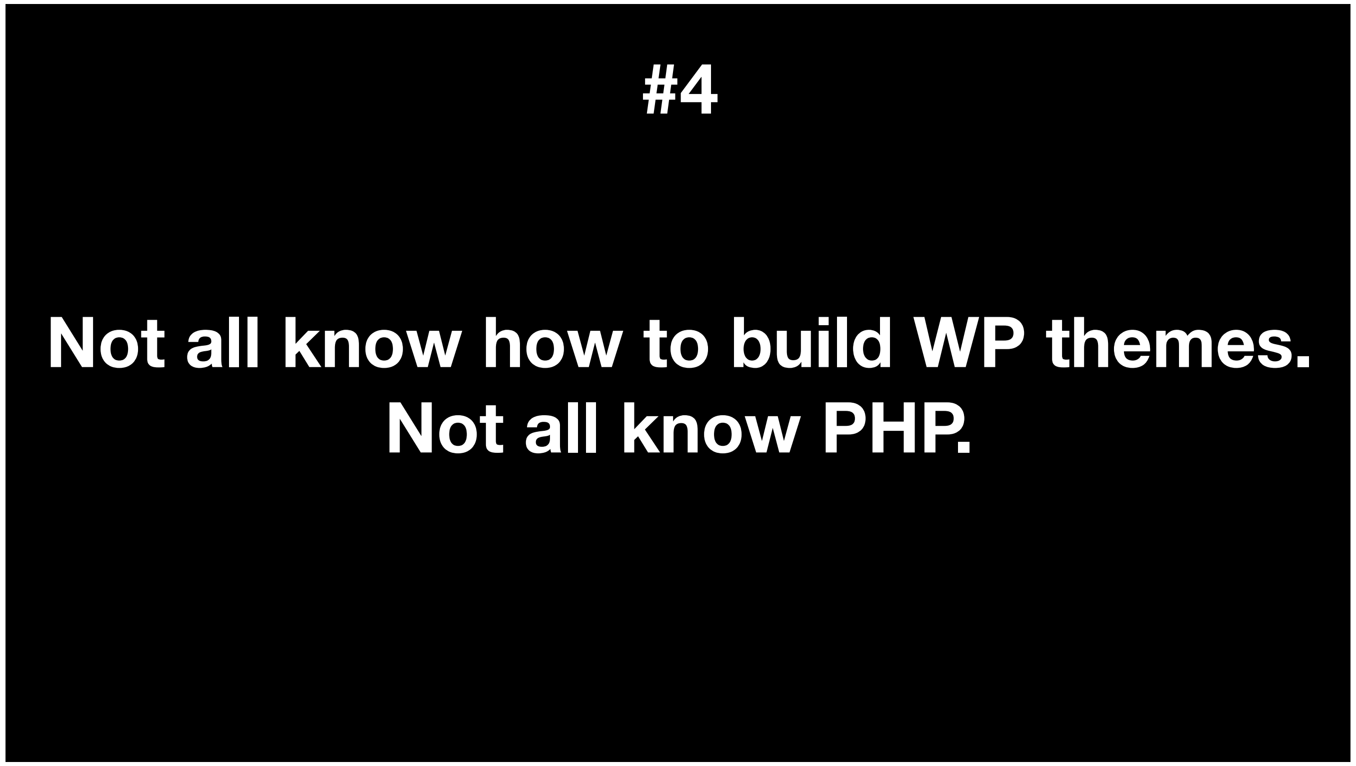
#3

Creativity with modern frameworks

When you don’t use PHP for building the frontend for your website, you get to use so many other alternatives. I don’t mean you cannot have creativity with PHP, but I am just trying to say that,

With Headless WordPress, you get the power to work with modern frameworks and libraries like Vue and React…

Just so you know, it is also possible to use Vue in your WordPress themes. But keeping your frontend separate might help developers get more creative with design.



#4

Not all know how to build WP themes.
Not all know PHP.

Not everyone knows how to build WordPress themes. Neither everyone knows PHP.

Frankly saying it out loud here in a WordPress Conference that, I don’t know PHP. Yea I mean I knew it like an year and a half ago, but then I switched to JavaScript and never looked back.

And I am extremely happy to see how Gutenberg is bringing this major shift in the WordPress ecosystem, it is kind of an entry of JavaScript into the WordPress world.

If you don’t know PHP and want to use WordPress as a backend, you can easily do it. Just use that REST API, and then create a Swift/Java app with it! Or may be you create a blog in Ruby, or maybe build a frontend in React, Headless WordPress just opens up the world for you!

REASON NUMBER 5…

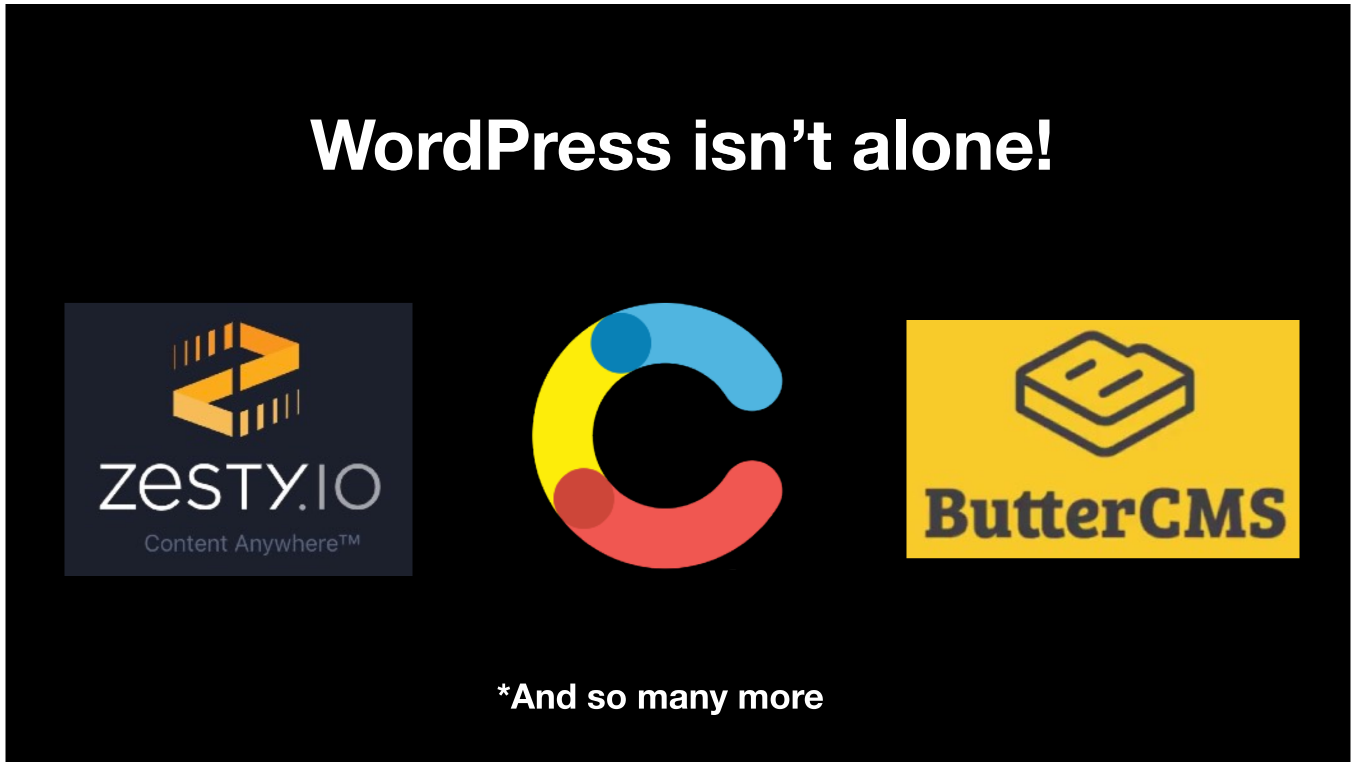
#5

Progressive Web App

Progressive Web App!

Well, if you use the SuperPWA WordPress Plugin, you can setup a Progressive Web App with WordPress,

but in JavaScript, you can use your own service workers, push notifications, and many more stuff!



WordPress isn't alone!

zesty.io
Content Anywhere™
ButterCMS

*And so many more

Well, WordPress isn't the only CMS in the market.

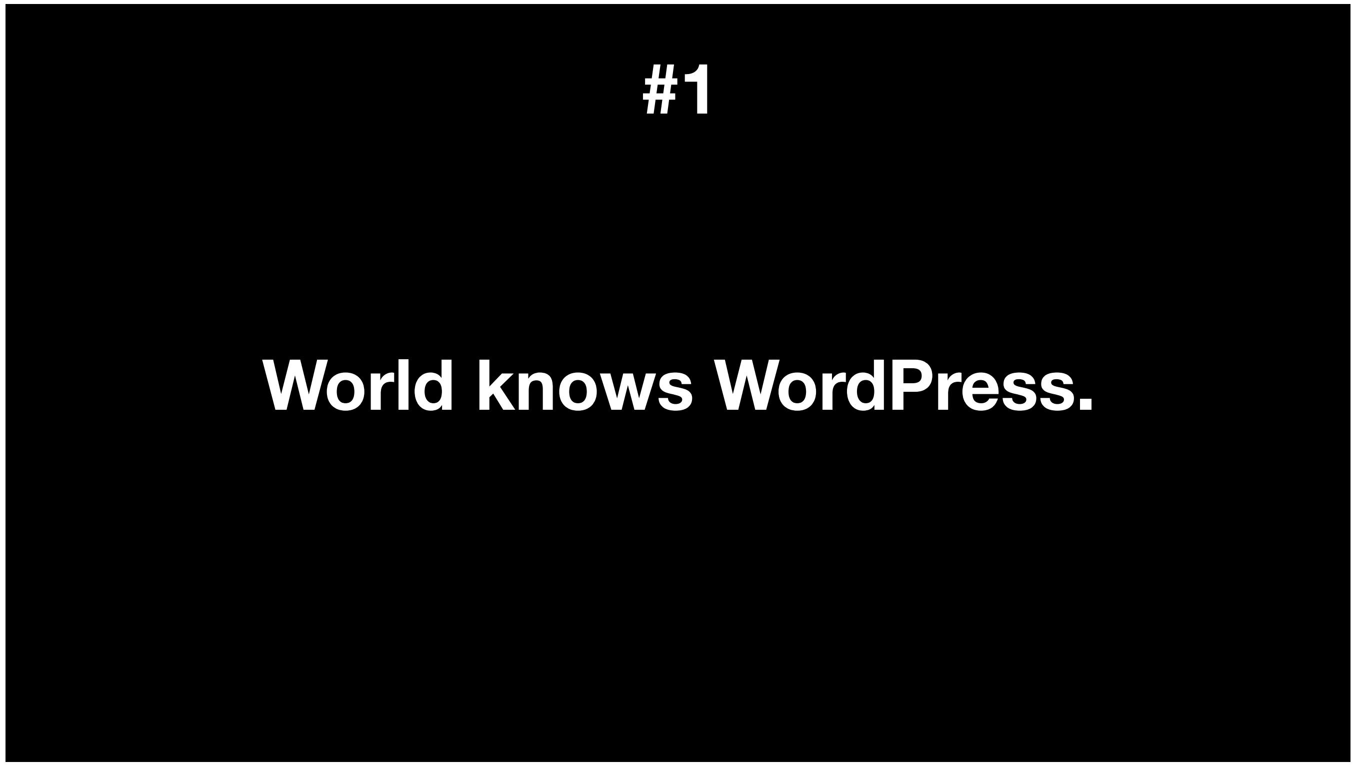
There are many Headless CMSes you can use. WordPress is NOT primarily a Headless CMS, but it can be used as a Headless CMS.

Headless CMS is a Content Management System that does not manage frontend, but only provide data by an API.

So, let us talk about, Why you should be using WordPress as a headless CMS.

Why WordPress?

REASON NUMBER 1…

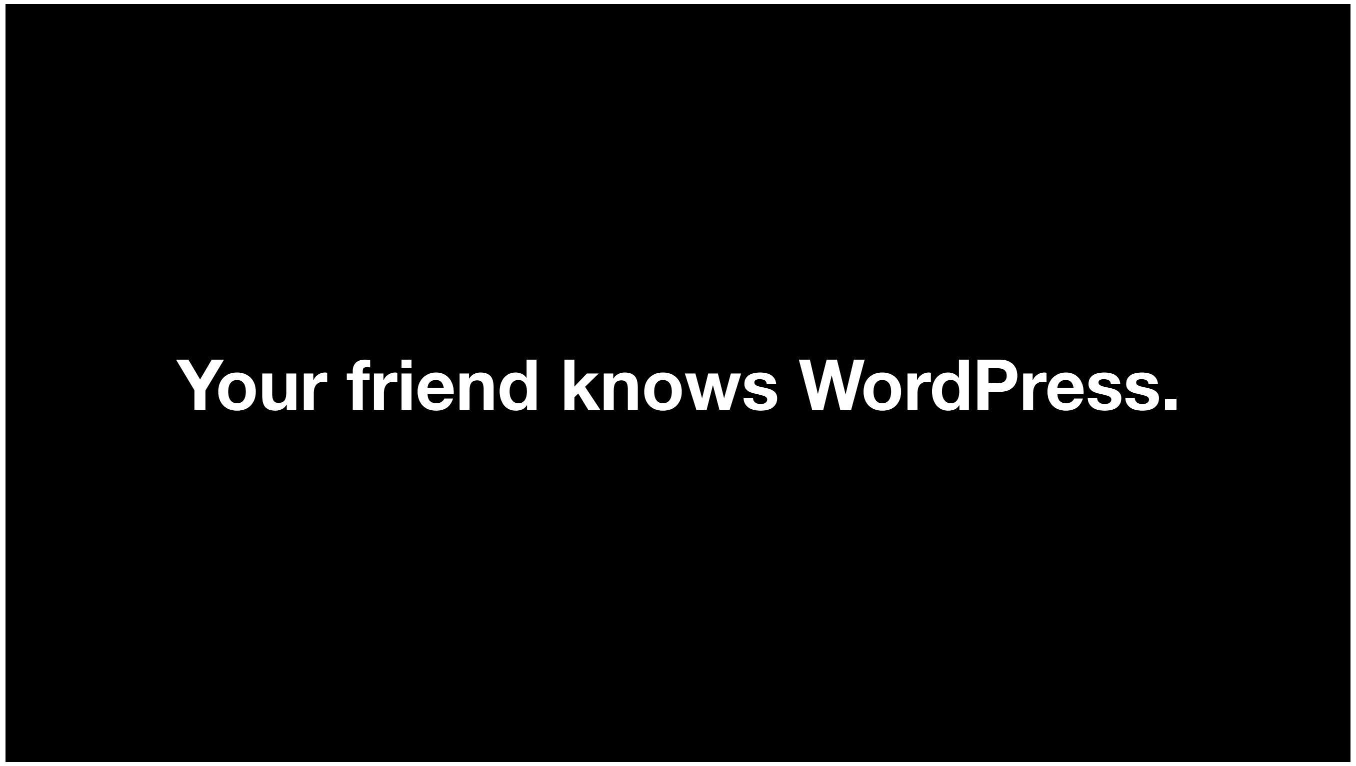


#1

World knows WordPress.

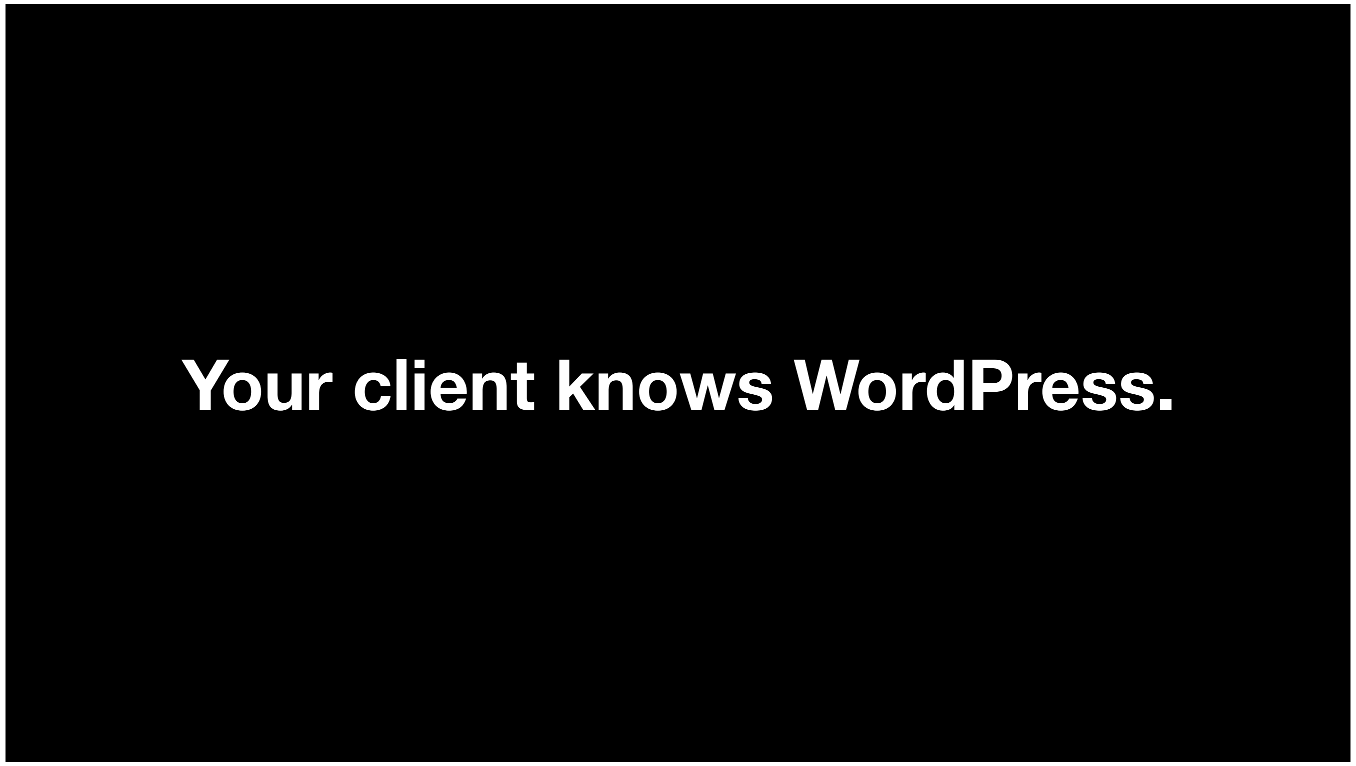
World knows WordPress.

Your friend knows WordPress.



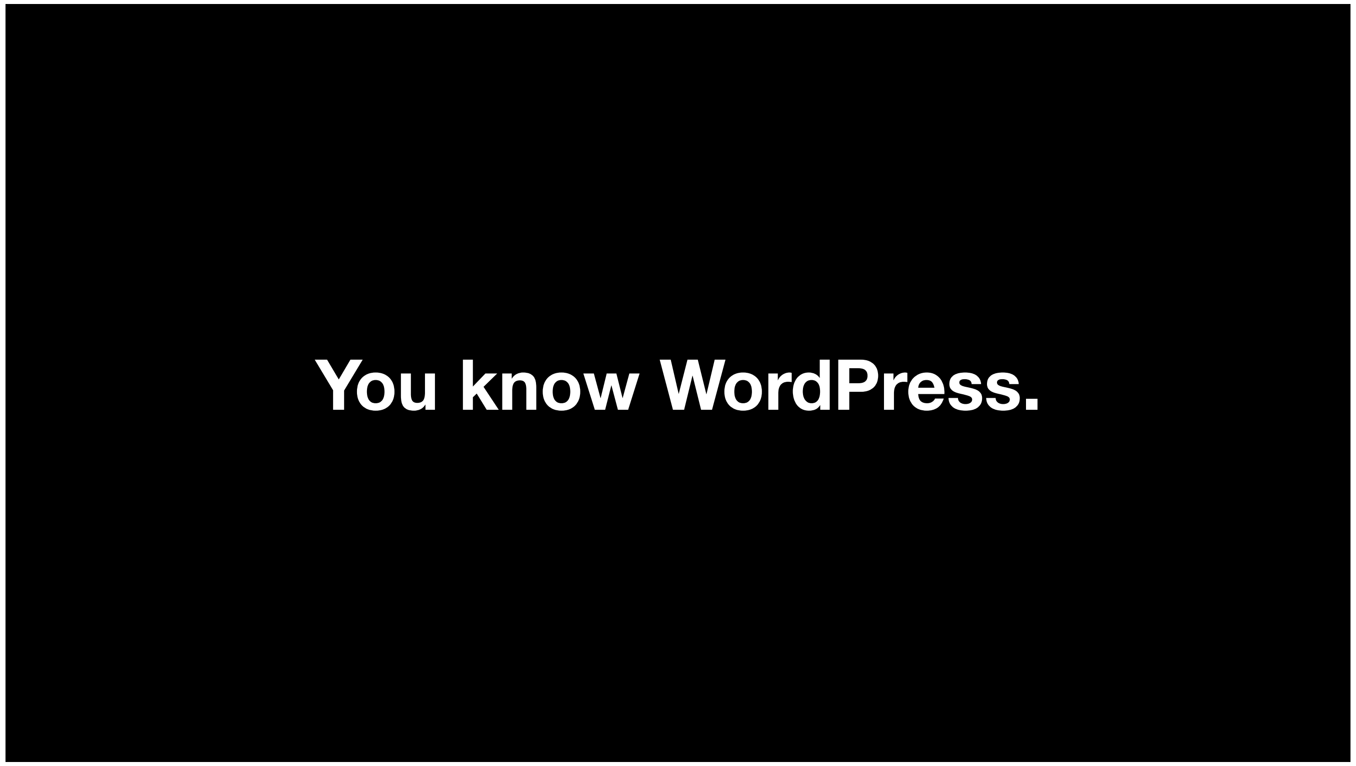
Your friend knows WordPress.

Your Client knows WordPress.



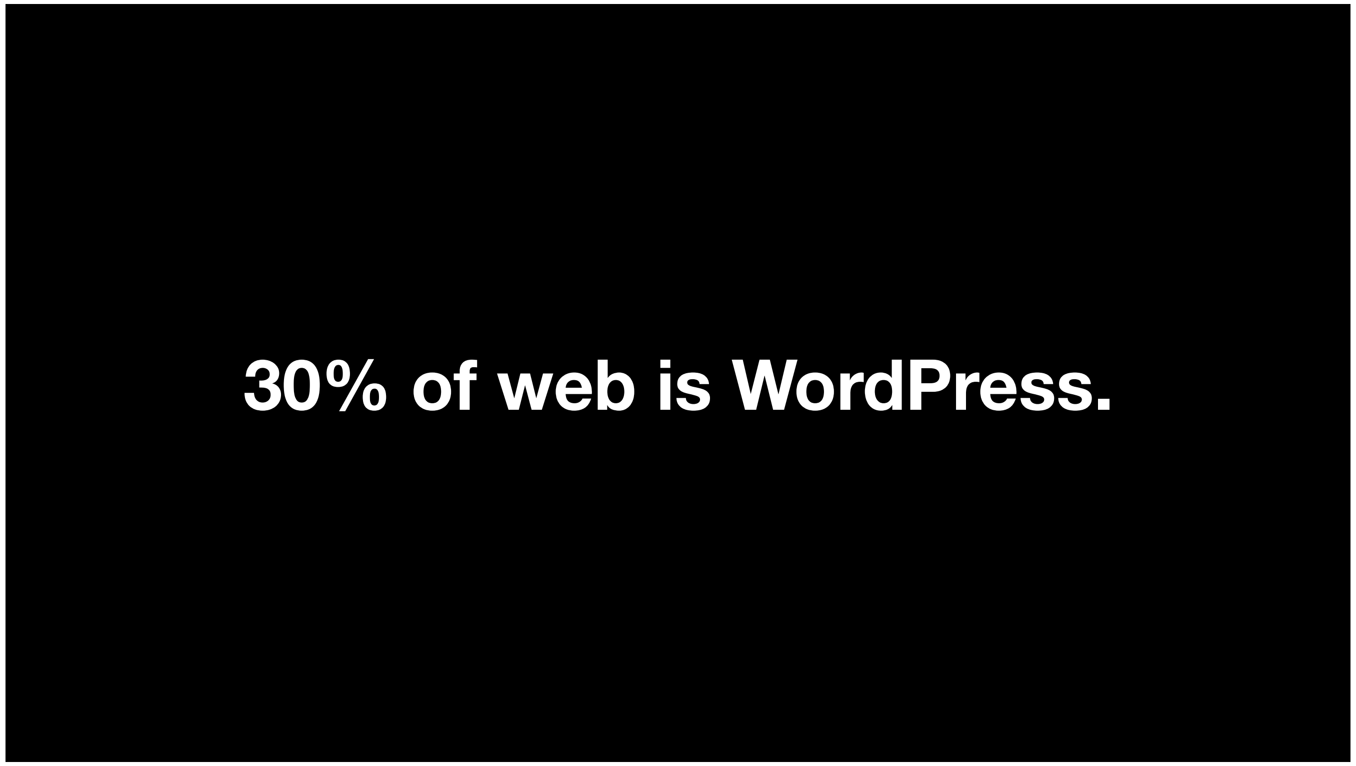
Your client knows WordPress.

YOU, know WordPress.



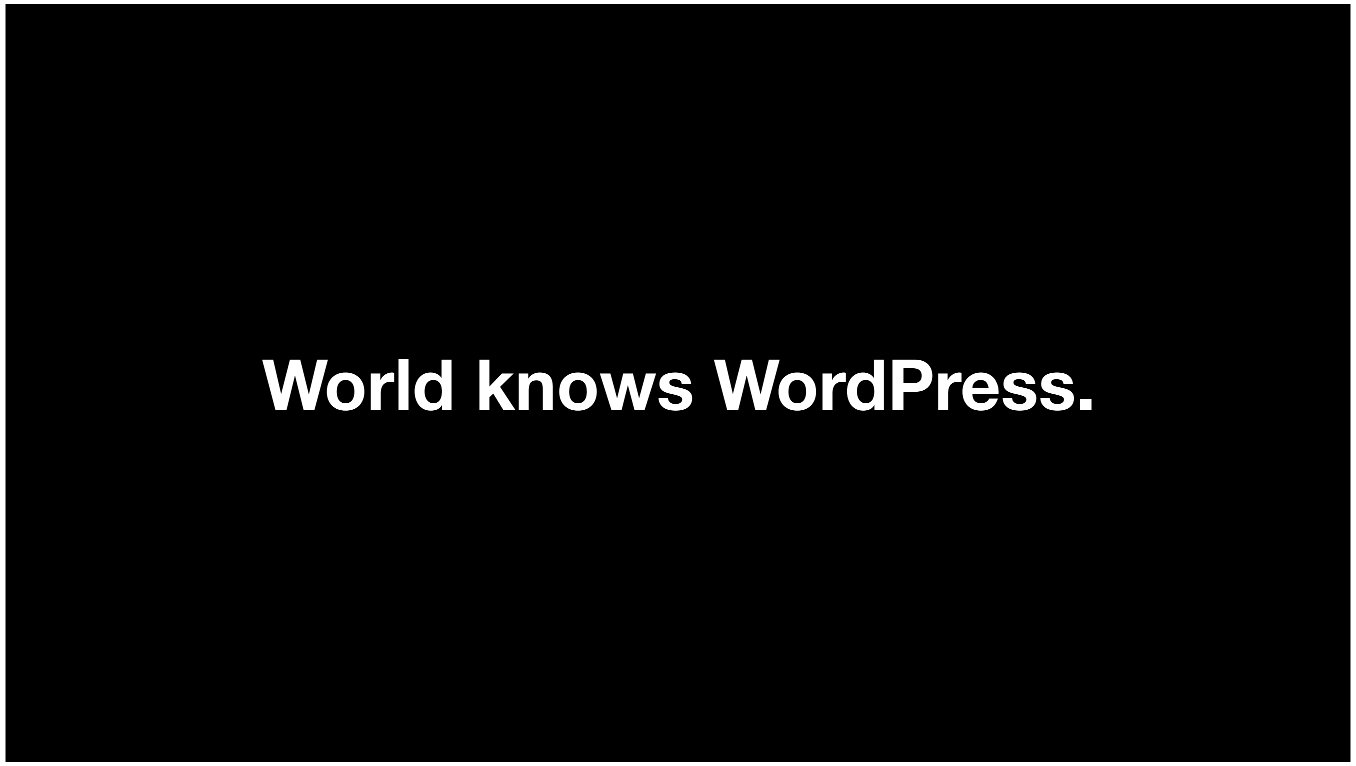
You know WordPress.

30% of web is WORDPRESS.



30% of web is WordPress.

WORLD KNOWS WORDPRESS



World knows WordPress.

REASON NUMBER 2…

#2

Ease of Use

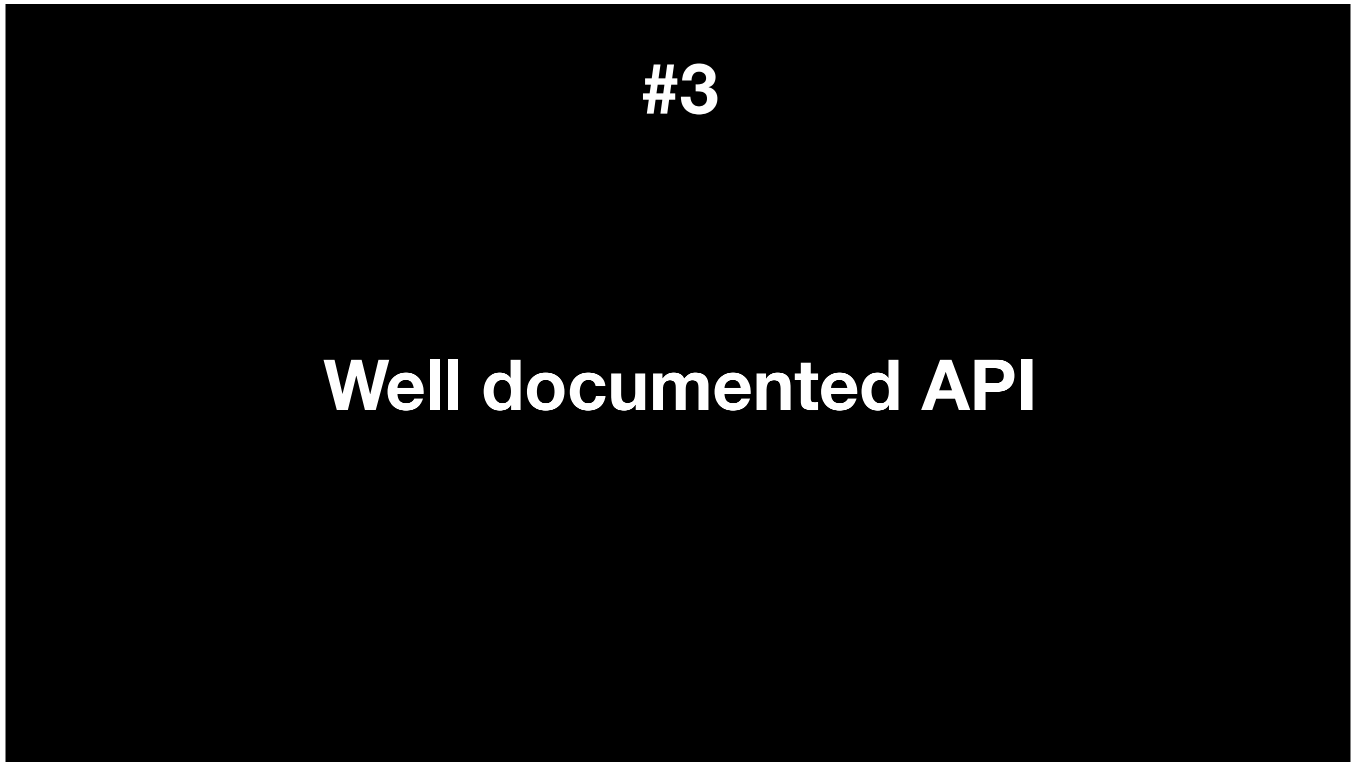
EASE OF USE…

You’all know it is easy to use, right?

I made my WordPress blog when I was 12 years old.

Ease of use might be the reason of why WordPress in today’s world is so widely adopted.

REASON NUMBER 3…



#3

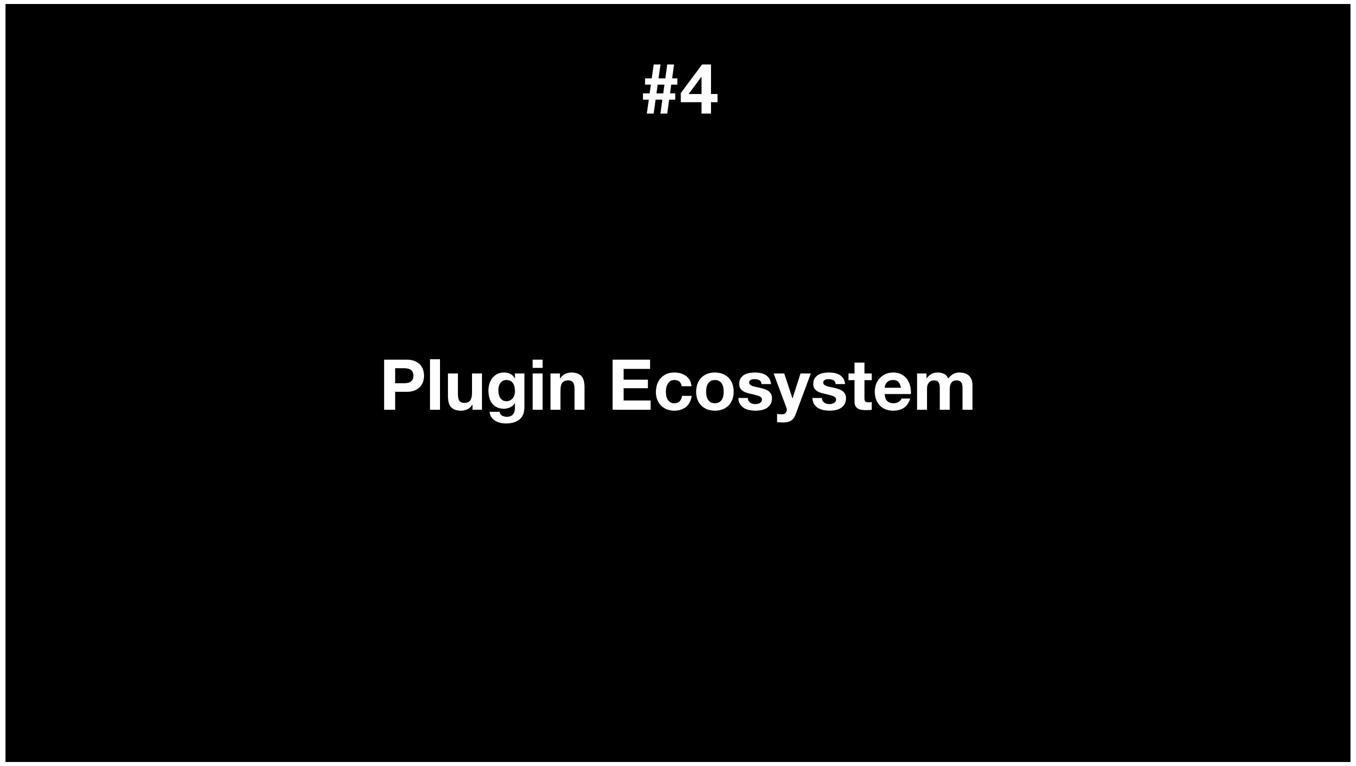
Well documented API

WELL DOCUMENTED API…

If you check out WordPress Codex, the documentation itself explains a lot.

And there also is a documentation for WordPress Rest API (also called as wp-json), where you can refer what endpoints to ping, to get few results.

REASON NUMBER #4…



#4

Plugin Ecosystem

PLUGIN ECOSYSTEM…

You name it, what plugin isn’t there made yet?

You find almost every plugin for every damn common use case.

WordPress isn’t all about posts, sometimes you wanna write book reviews, and there is a plugin for that too. Most of those plugins will activate their own JSON endpoint for your frontend to fetch from.

REASON NUMBER 5…

#5

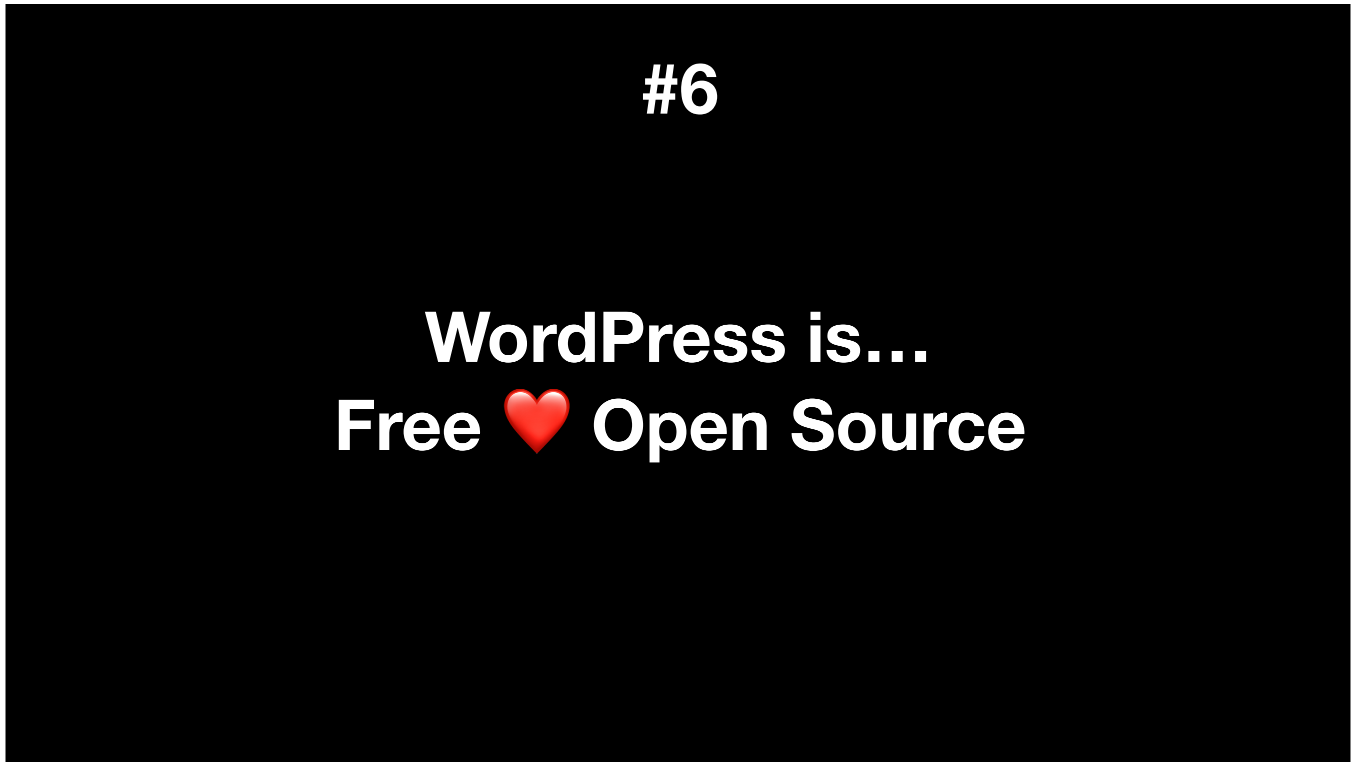
WordPress Community

WORDPRESS COMMUNITY

What are we? What are all of us? We’re us, right? We are a strong community.

Most of other headless CMSes might have not built a strong community as of WordPress.

REASON NUMBER #6



#6

WordPress is…
Free ❤️ Open Source

WORDPRESS IS FREE AND OPEN SOURCE…

Most of other JavaScript Headless CMSes are not open source while some of them are, but they don’t have most of the pro points of WordPress.

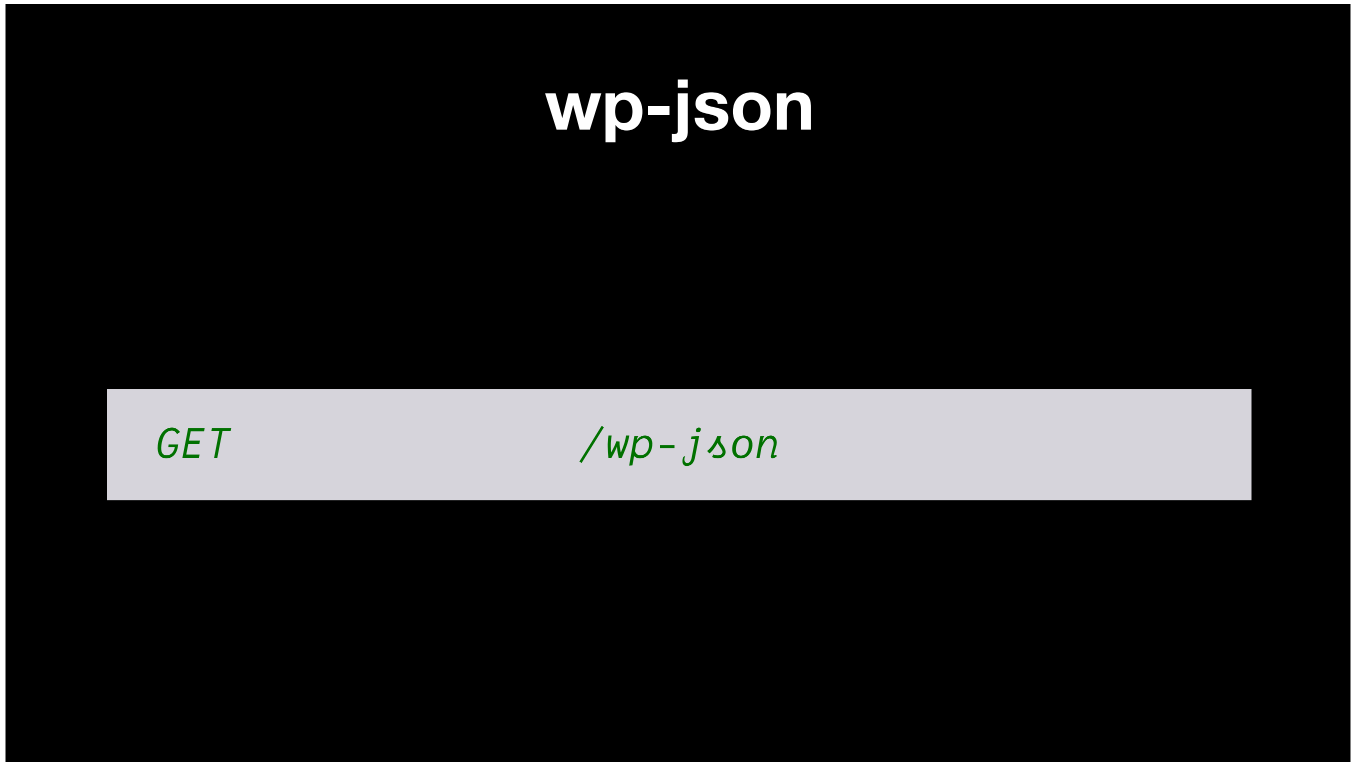
I guess I have convinced you why WordPress should become your go-to Headless CMS.

Now, let us move on to the technical aspect of this talk.

wp-json

wp-json

What is it?



wp-json

```
GET                   /wp-json
```

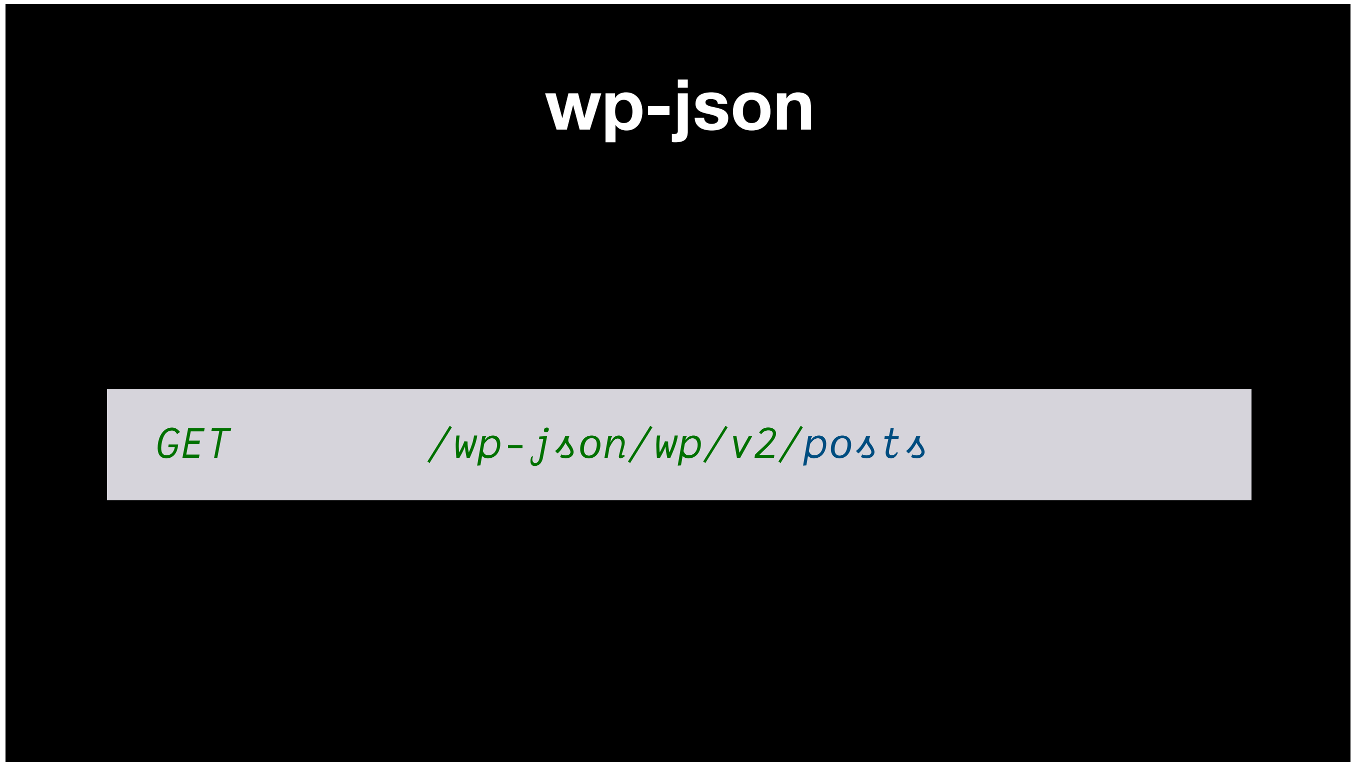
It’s an endpoint.

Try putting /wp-json in the url of a WordPress website and see what you get.

<SHOW THEM>

Well, this isn’t where you should be to get actual data.

To get posts….



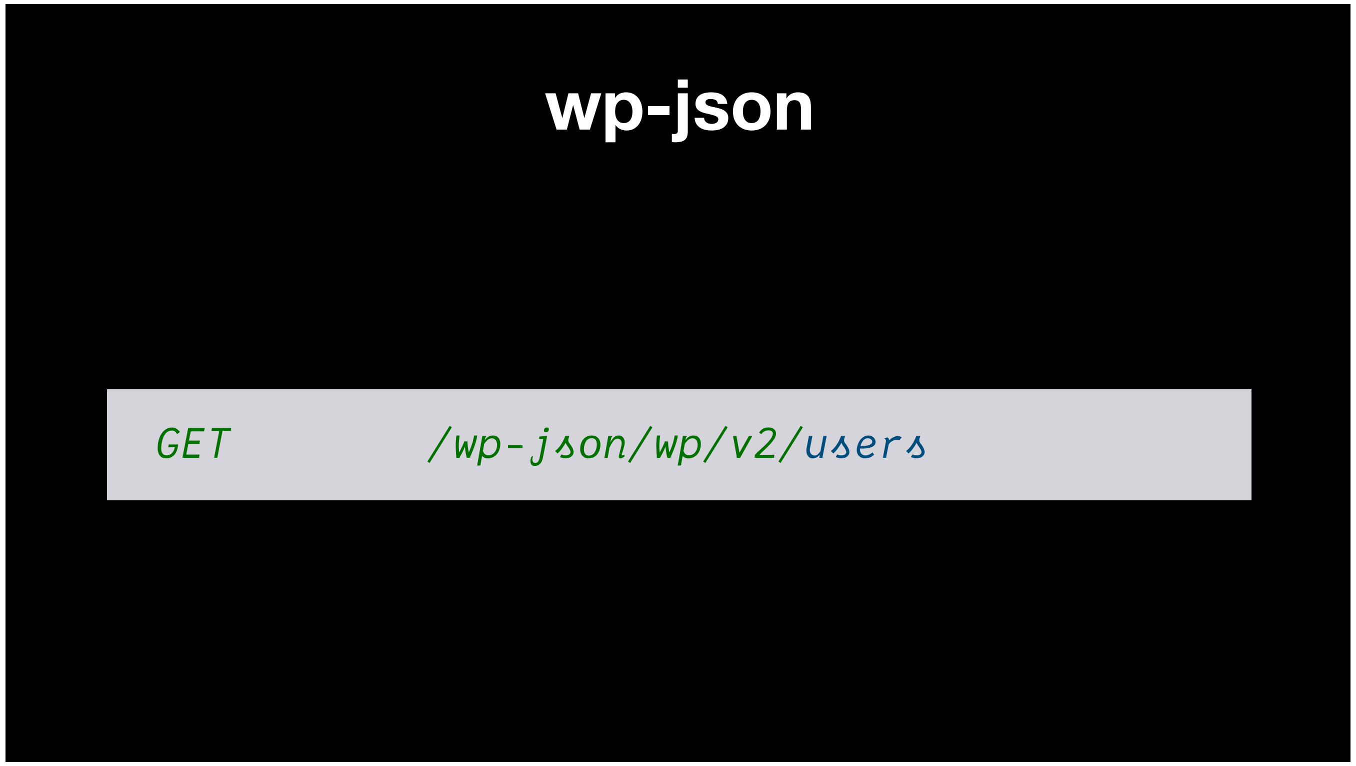
wp-json

```
GET          /wp-json/wp/v2/posts
```

To gotta be pinging this endpoint.

<SHOW THEM>
<Talk about how author, post, media is related>
<COME BACK>

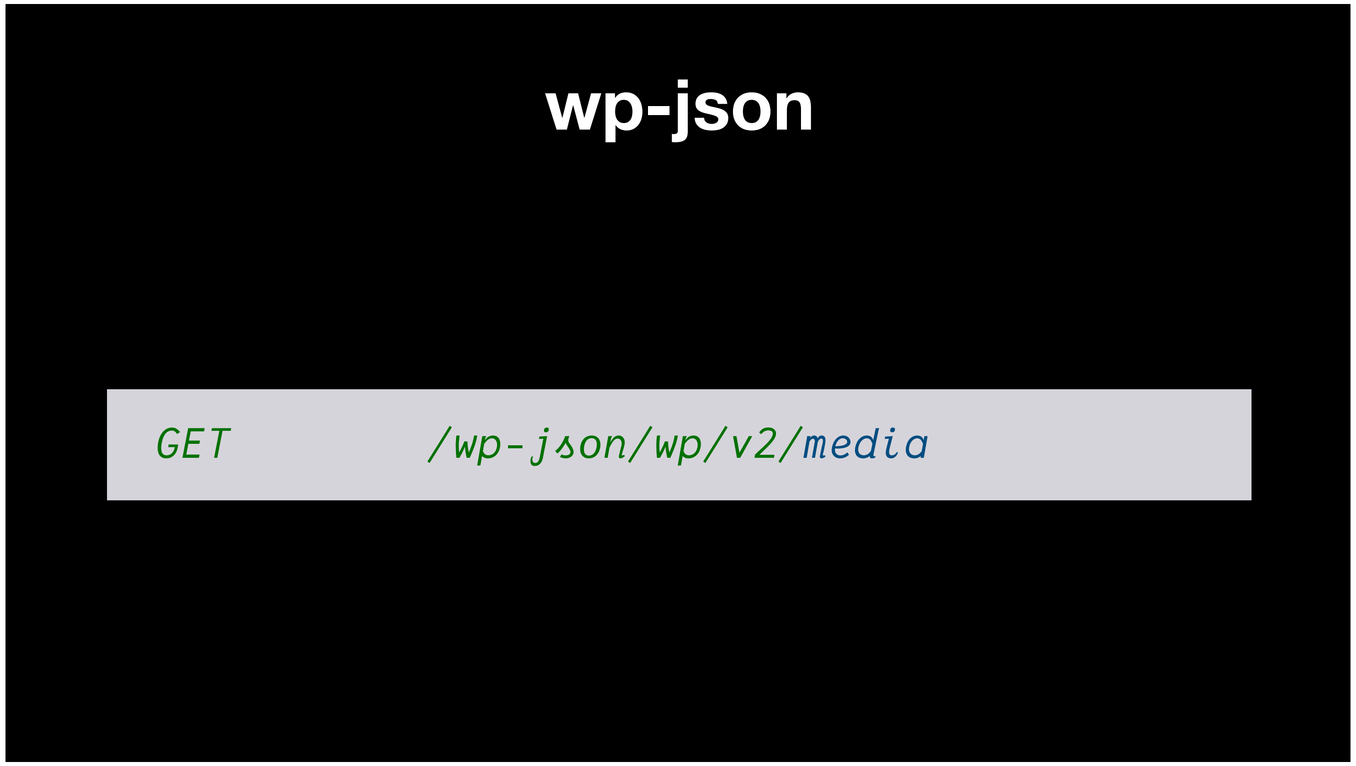
Well, posts is not the only thing you can fetch.



wp-json

```
GET          /wp-json/wp/v2/users
```

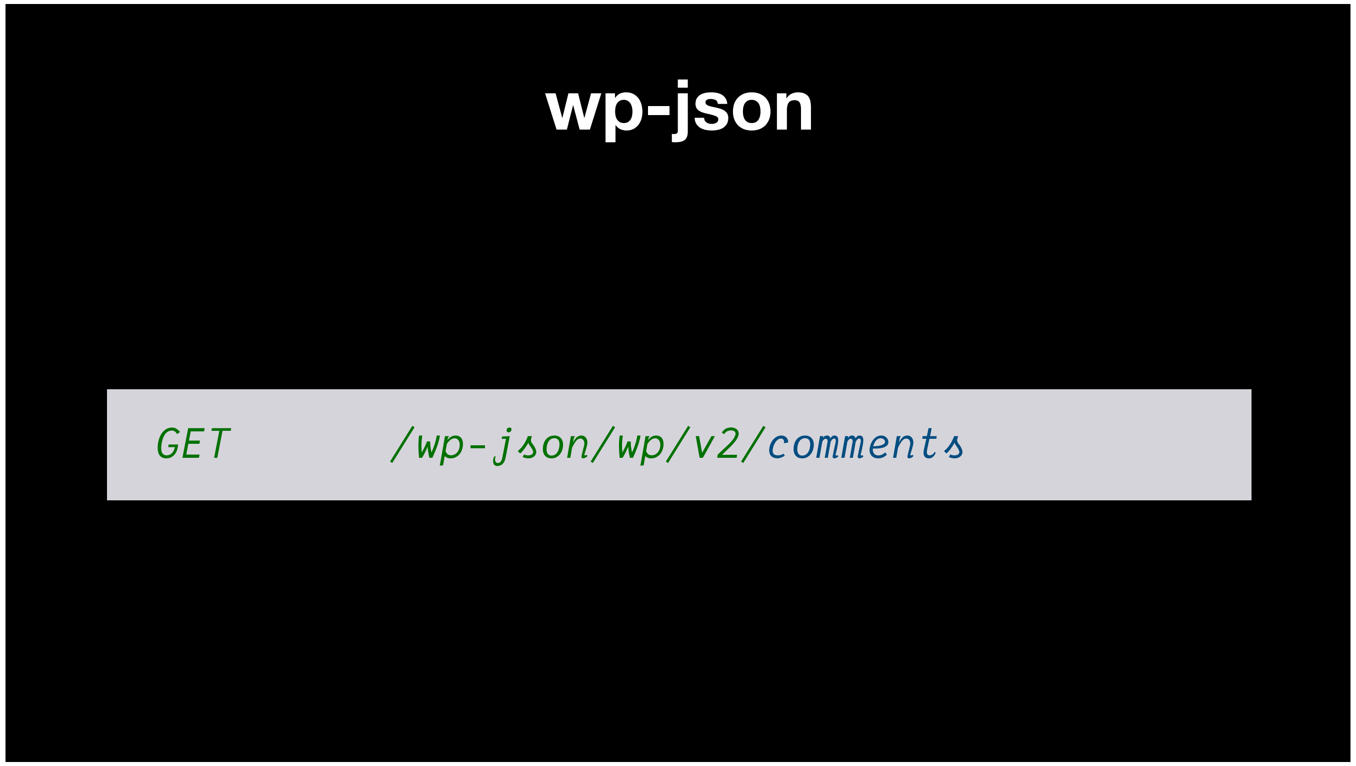
You can fetch Users



wp-json

```
GET          /wp-json/wp/v2/media
```

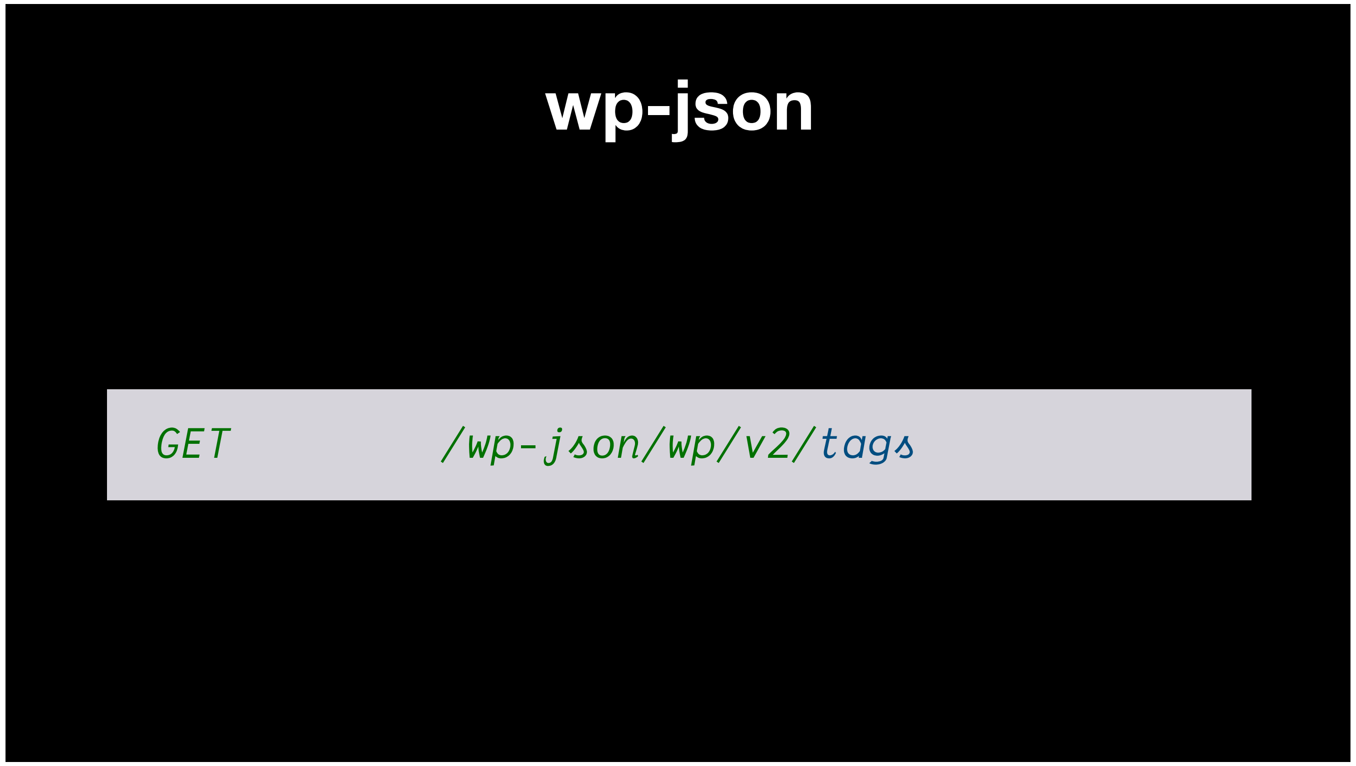
Media



wp-json

```
GET        /wp-json/wp/v2/comments
```

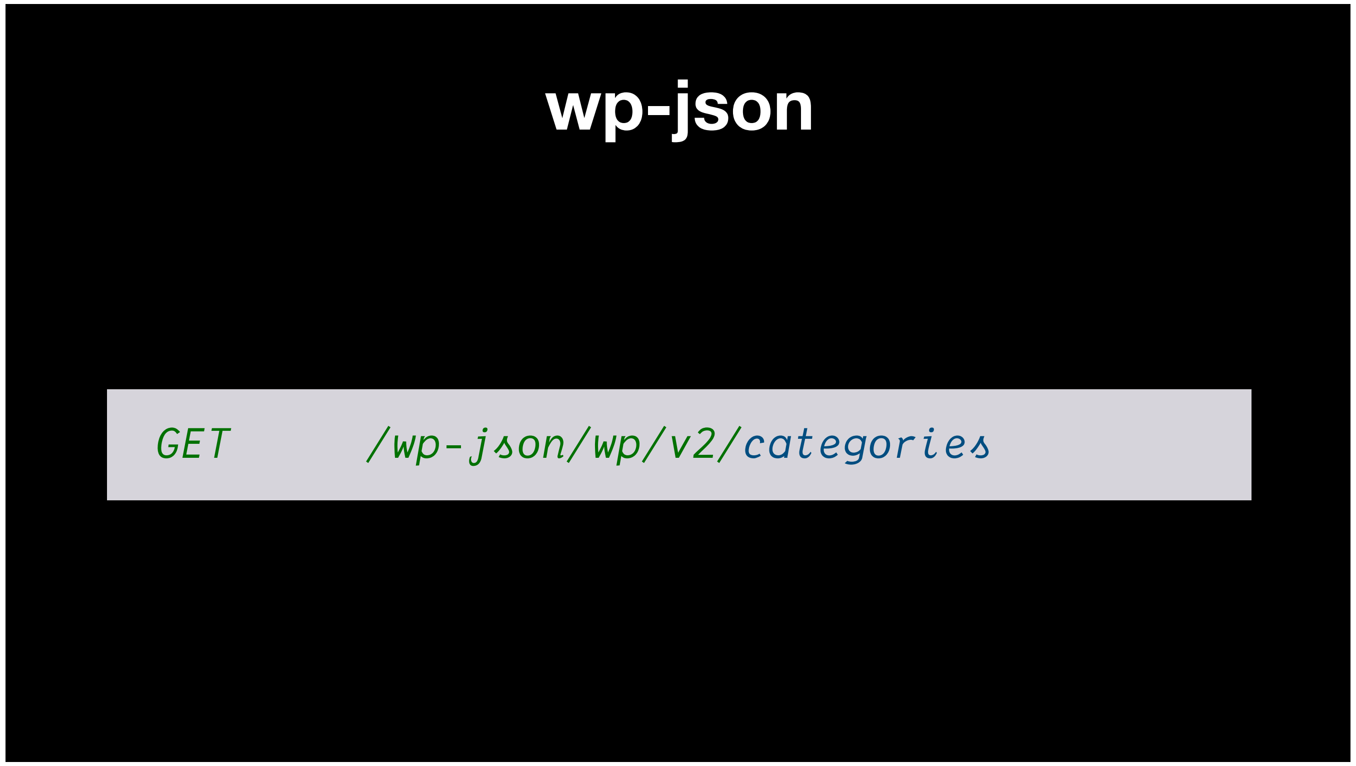
Comments



wp-json

```
GET          /wp-json/wp/v2/tags
```

Tags

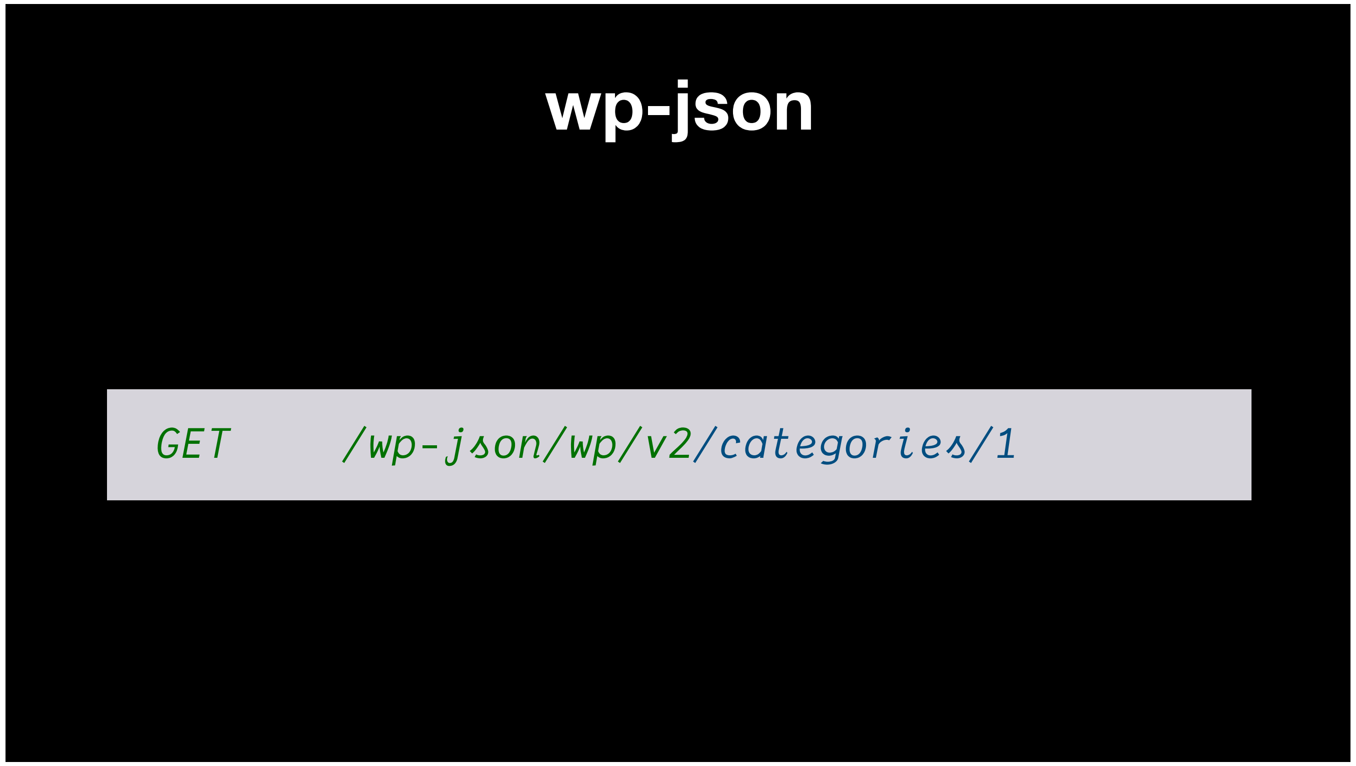


wp-json

```
GET      /wp-json/wp/v2/categories
```

Categories

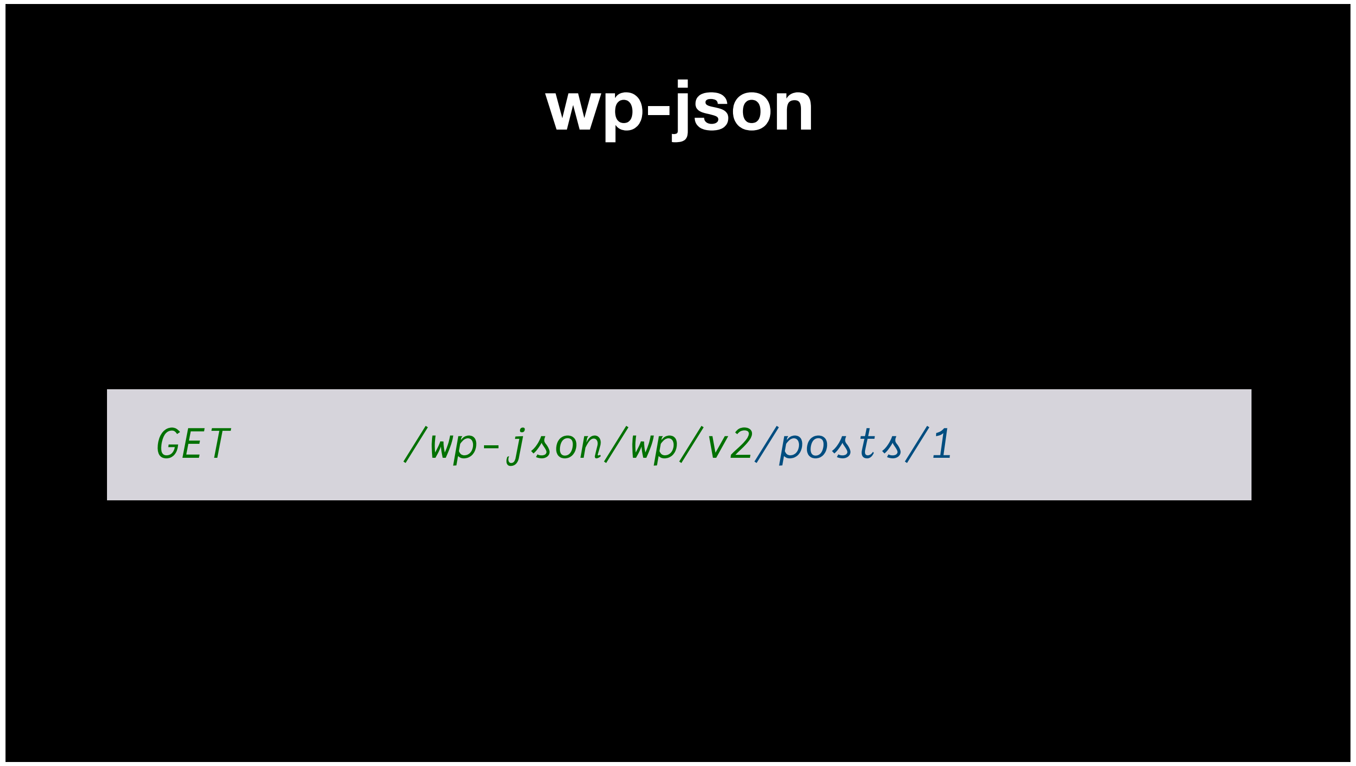
And then you can also fetch each one of them instead of fetching in a bulk.



wp-json

```
GET    /wp-json/wp/v2/categories/1
```

Like this. Category with an id of 1.



wp-json

```
GET        /wp-json/wp/v2/posts/1
```

Post with an id of 1.

So, if you wanna fetch the author of a post, you’ll have to make two fetches, one for the posts endpoint and then for the author endpoint.

<SHOW THEM THAT>
<COME BACK>

With this API, you have to not just do fetching, okay!

You can do…

wp-json

Authentication

AUTHENTICATION!

I haven’t personally tried doing an authentication with wp-json, but you can refer the documentation and try it out yourself.

If you are using JavaScript, you might be familiar with `json-web-token`s (JWT), you gotta be using them when you handle Authentication with WordPress.

And then you can do…

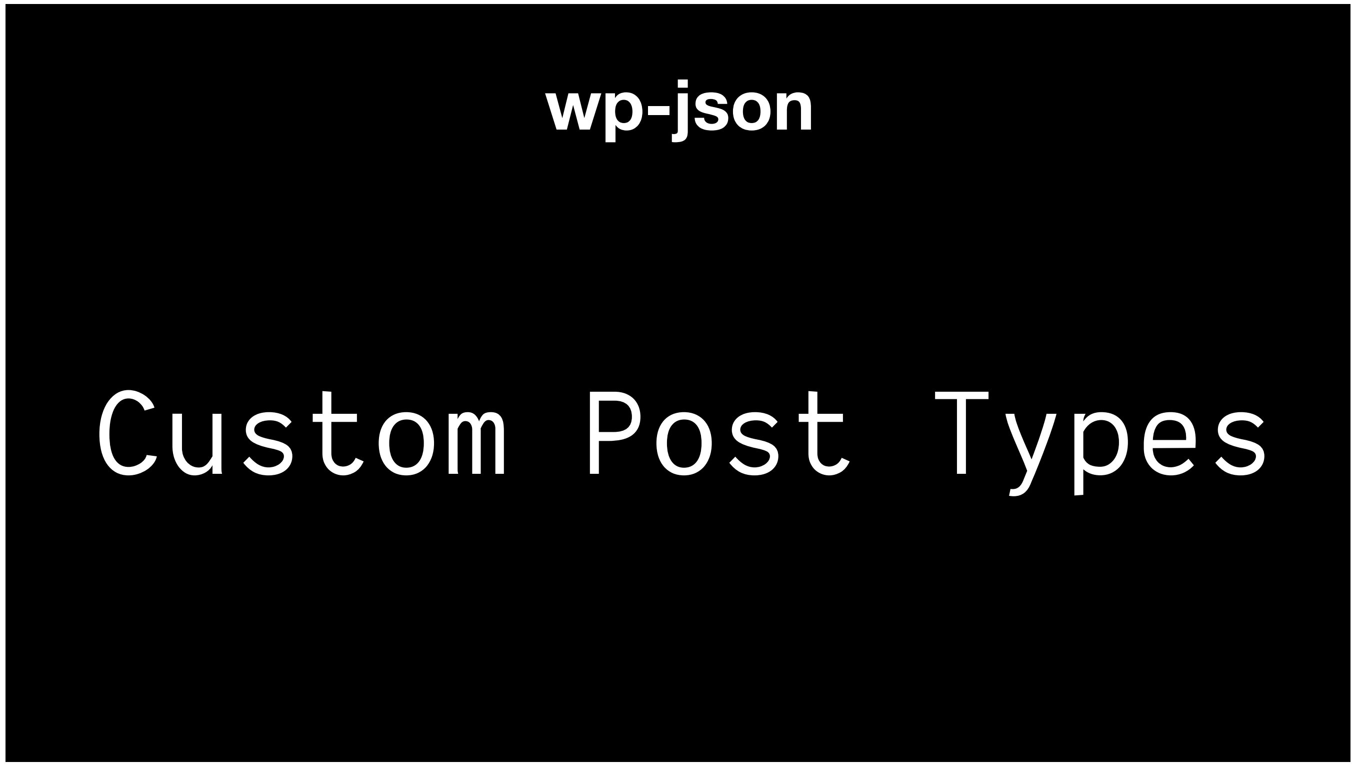
wp-json

Pagination

PAGINATION…

<Show them Pagination Docs>
<Explain basic iteration>
<Infinite page load>

Then you can do…



wp-json

Custom Post Types

Custom post types!

WordPress was made for bloggers. But fast forward to 2019, it can be used for all purposes.

You might not have to only manage posts, but also recipes, books, issues, umm… you might need to manage an inventory of your products, and so on. Each of them will be your Custom Post Type. Book is Custom Post Type. Sitcker is Custom Post Type. Recipe is Custom Post Type. You name it. Every inventory can be a custom post type.

To use the data of each of those Custom Post Types, you need to serve respective endpoints for each Custom Post Type.

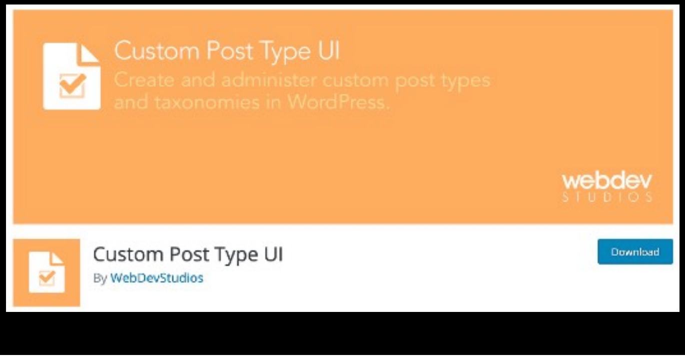
Just like you can fetch posts, you should be able to fetch your books.

And to do this, there are plugins for that!

wp-json

Custom Post Types

CPTUI Plugin



Custom Post Type UI

Create and administer custom post types and taxonomies in WordPress.

Custom Post Type UI

By WebDevStudios

CPTUI Plugin.

This plugin helps you register a Custom Post Type that serves JSON Endpoints.

This plugin isn’t everything though. Because with this plugin, you’ll get a basic CPT setup with a title, body, excerpt, etc. But if your CPT is a book, then you don’t need a body. You need a Summary. You also need an input field where you can set the price of your book.

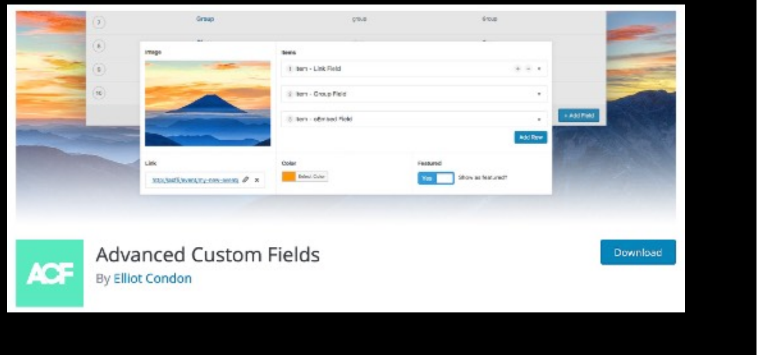
And remember, all this data should be served on JSON endpoints for your frontend to fetch from.

To get all your needs covered, you need another WordPress plugin…

wp-json

Custom Post Types

ACF Plugin

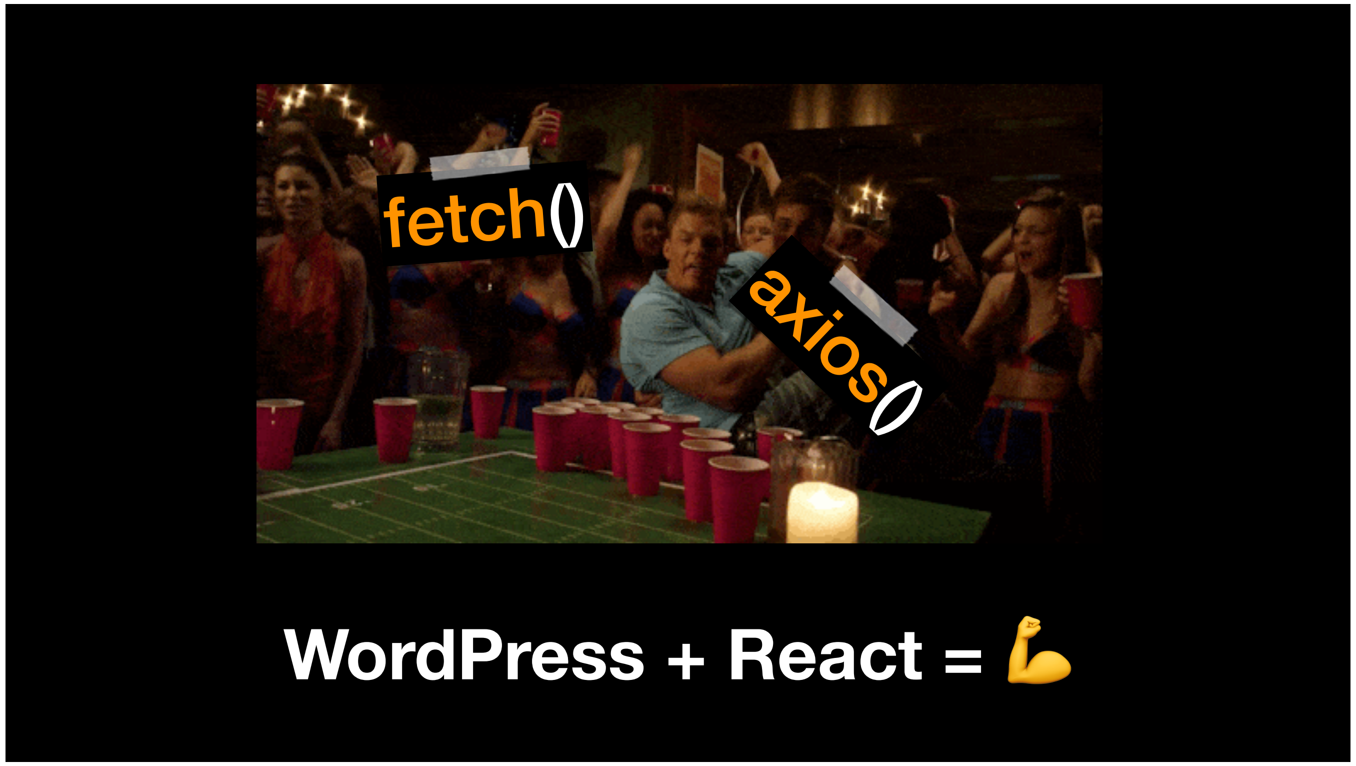


And that is, the ACF Plugin.

This is where you can add custom fields to the custom post type that you created.

So for your understanding, consider Custom Post Type UI plugin as a plugin that sets you a table, and ACF plugin will be setting up the column names for you.

So, summing it up, Headless WordPress has endless possibilities.



WordPress + React = 💪

Um, I am using that <CLICK> image there because, I just wanted to slip in a GIF in one of my slides., you know.

So to fetch from the endpoint in the React frontend, we can use fetch and axios.

In my opinion, using fetch feels more verbose in terms of code, and I like axios, so, we'll be using axios to fetch in our React frontend.

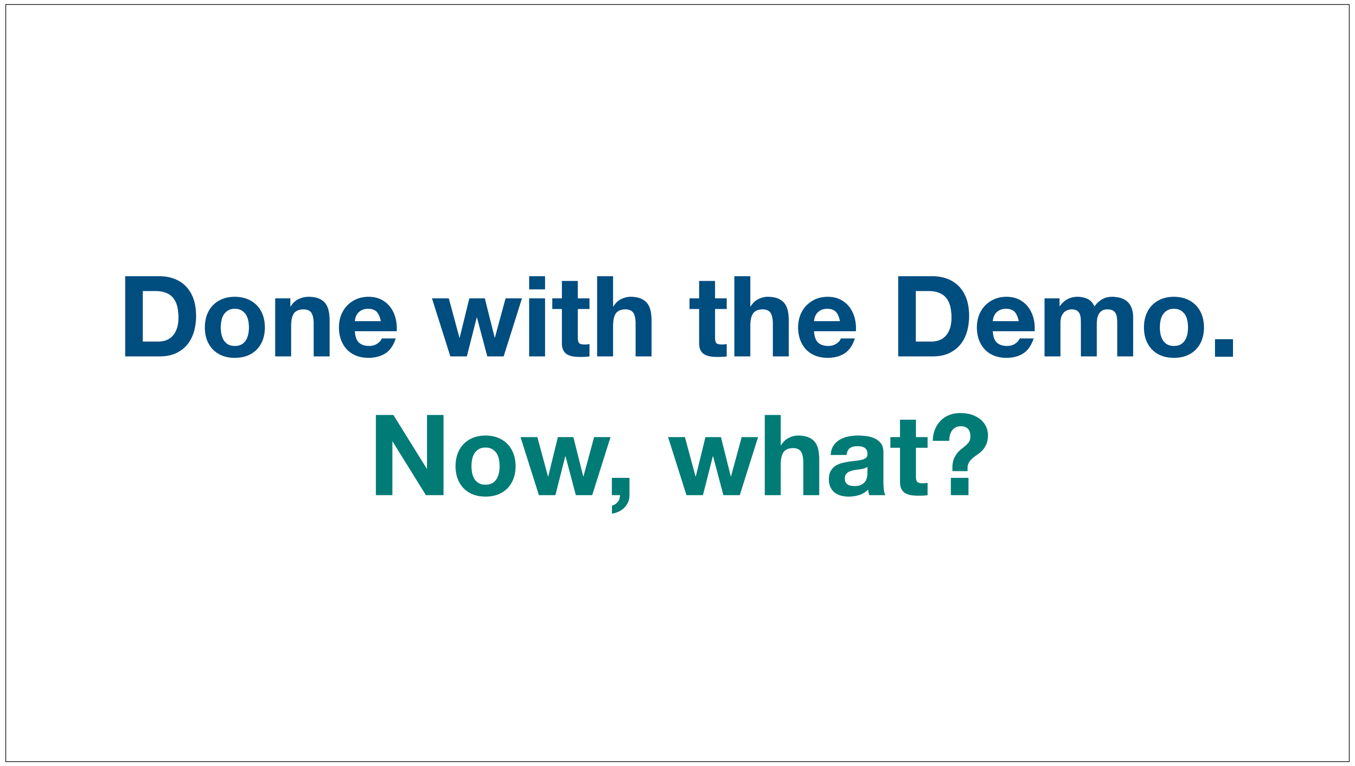
OKAYYYY…..

Now, it's the…..

DEMO TIME 🎉🎉

DEMO TIME!

<DO THE DEMO>
<COME BACK>



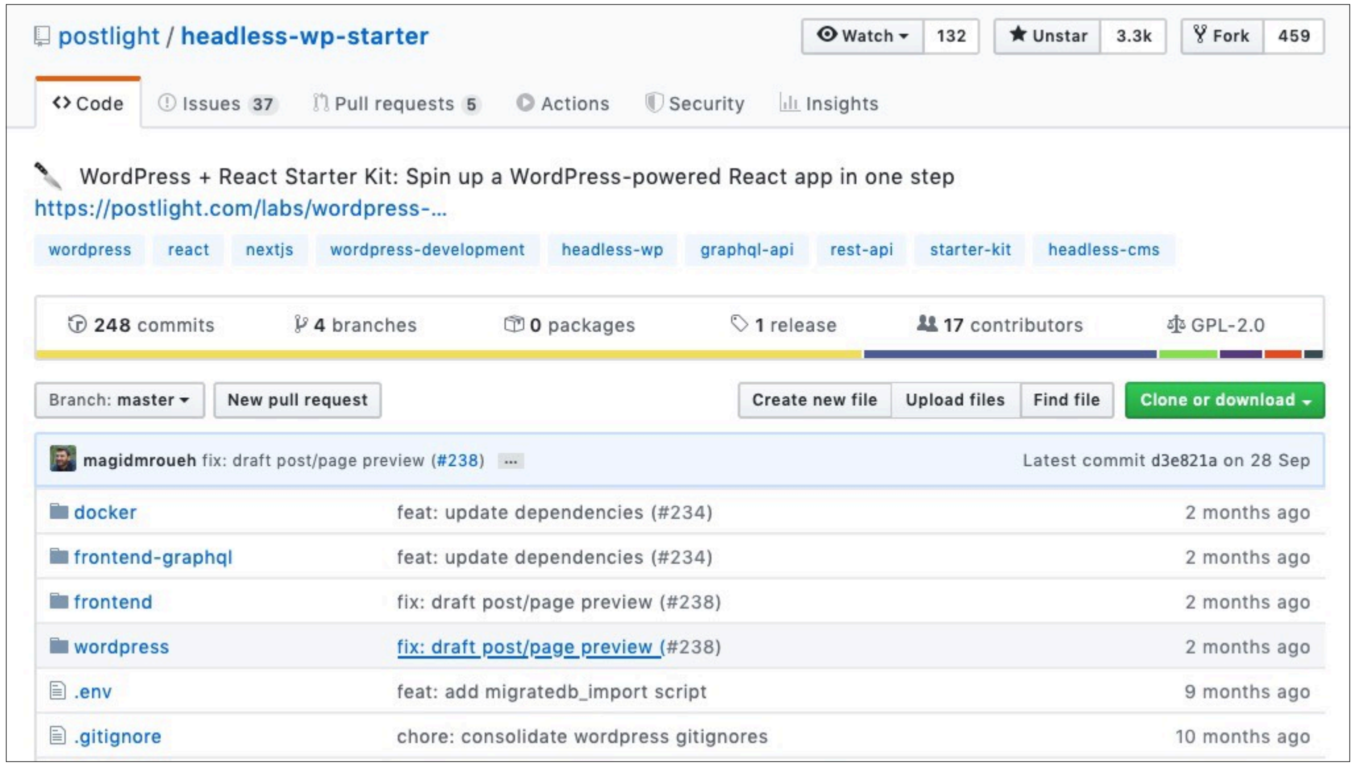
Done with the Demo.
Now, what?

Some more information I would like to provide…

If you are creating a Headless WordPress served website in React, don’t make it like I did. You can use that for learning purpose.

But setting up categories, tags, posts, media, author, authentication, will be a very huge pain if you started from scratch.

Okay, don’t start from scratch.



postlight / headless-wp-starter

WordPress + React Starter Kit: Spin up a WordPress-powered React app in one step
https://postlight.com/labs/wordpress-...

wordpress react nextjs wordpress-development headless-wp graphql-api rest-api starter-kit headless-cms

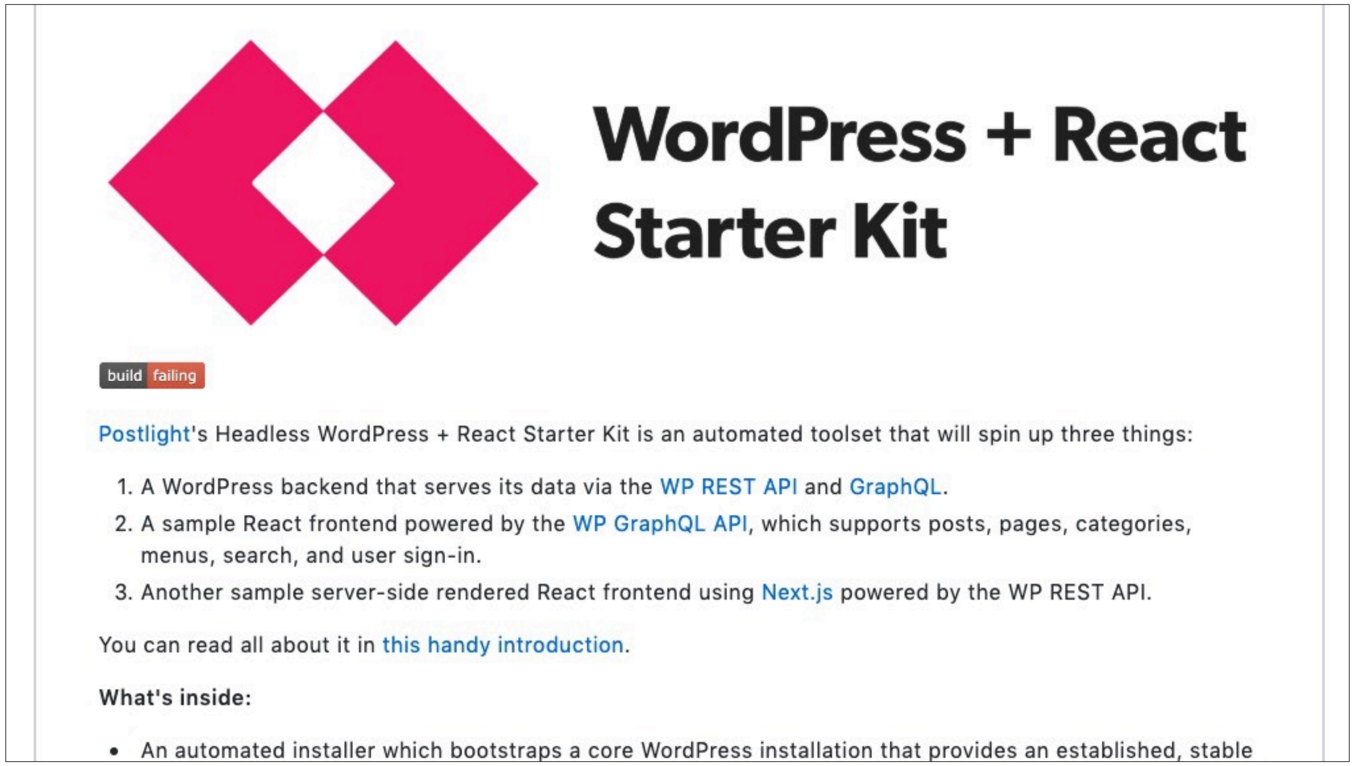
248 commits | 4 branches | 0 packages | 1 release | 17 contributors | GPL-2.0

magidmroueh fix: draft post/page preview (#238) ... Latest commit d3e821a on 28 Sep

docker	feat: update dependencies (#234)	2 months ago
frontend-graphql	feat: update dependencies (#234)	2 months ago
frontend	fix: draft post/page preview (#238)	2 months ago
wordpress	fix: draft post/page preview (#238)	2 months ago
.env	feat: add migratedb_import script	9 months ago
.gitignore	chore: consolidate wordpress gitignores	10 months ago

This is a Headless WordPress React Starter Kit by postlight… Which is very handy.

It works with GraphQL, has authentication setup already done, if you ever wanna build a very robust website that is served by WordPress on backend.



WordPress + React Starter Kit

Postlight's Headless WordPress + React Starter Kit is an automated toolset that will spin up three things:

1. A WordPress backend that serves its data via the WP REST API and GraphQL.
2. A sample React frontend powered by the WP GraphQL API, which supports posts, pages, categories, menus, search, and user sign-in.
3. Another sample server-side rendered React frontend using Next.js powered by the WP REST API.

You can read all about it in this handy introduction.

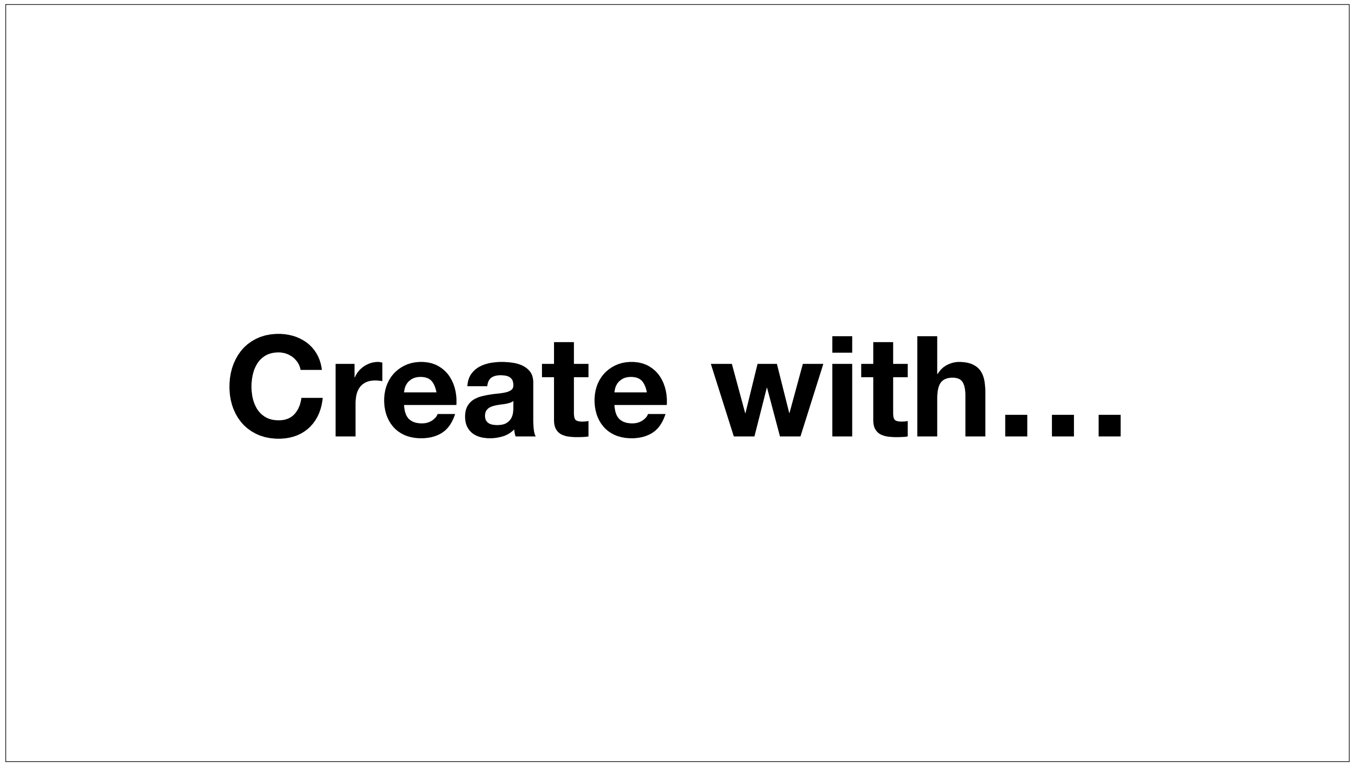
What's inside:

- An automated installer which bootstraps a core WordPress installation that provides an established, stable

It is powered by Next.js, so it is also server rendered.

One thing I would like to tell you, please use GraphQL… When you are working with public APIs like WordPress, it is a pain to make multiple network fetches to get all the data you need.

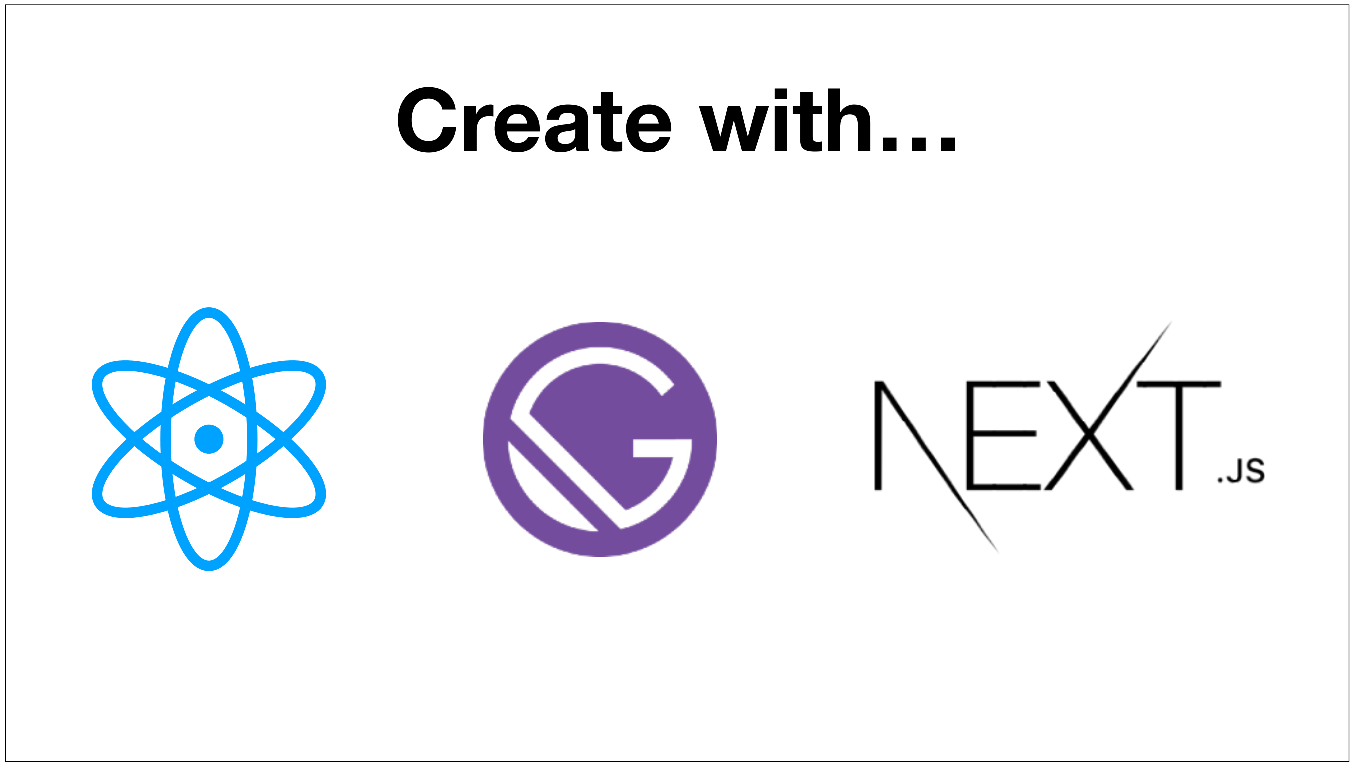
With GraphQL, you can ask for all the data in one go!



Create with…

In the demo example I showed, we used Next.js

But you don’t have to always use it.



Create with…

You can start from scratch with create-react-app… Which I recommend not to do. Do it if you are learning how REST API works.

Next.js can help you set things up and fetch on the server side, if you care for SEO. But modern search engines can now also read JavaScript, I guess. I have only read blogs about it.

And then comes Gatsby. Use it! <CLICK>

Create with…

Gatsby has a lot of plugins that will save your time making multiple network fetches.

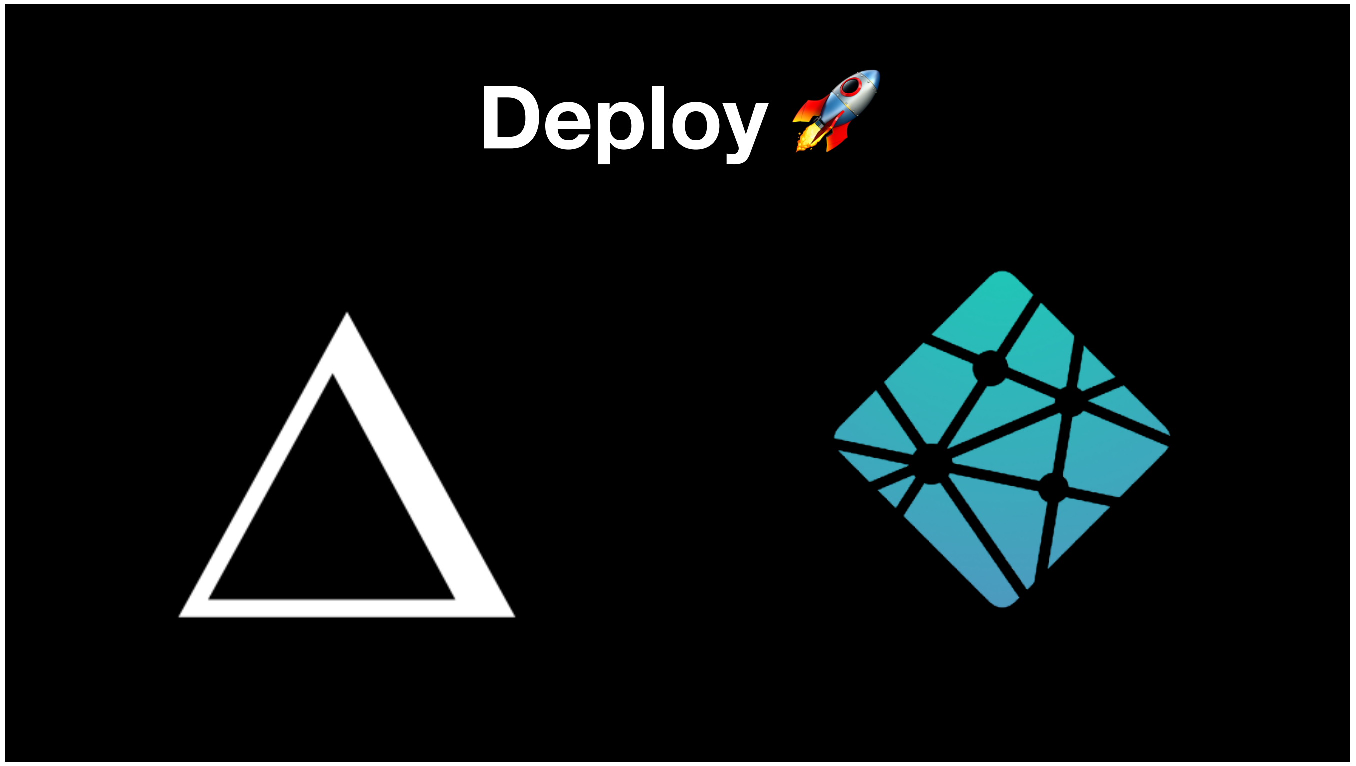
Fetching from a WordPress Backend using GraphQL has never been so easier.

Oh, just another random question, how many of you know what GraphQL is and how it’s syntax looks like?

<REACT BASED ON AUDIENCE’s answer>

Okay…

So, now you have made your WordPress Backend, and your frontend, and now you want to deploy.



Deploy 🚀

To deploy your frontends, ZEIT Now and Netlify are the two best services I can recommend you. The best thing, it's also free.

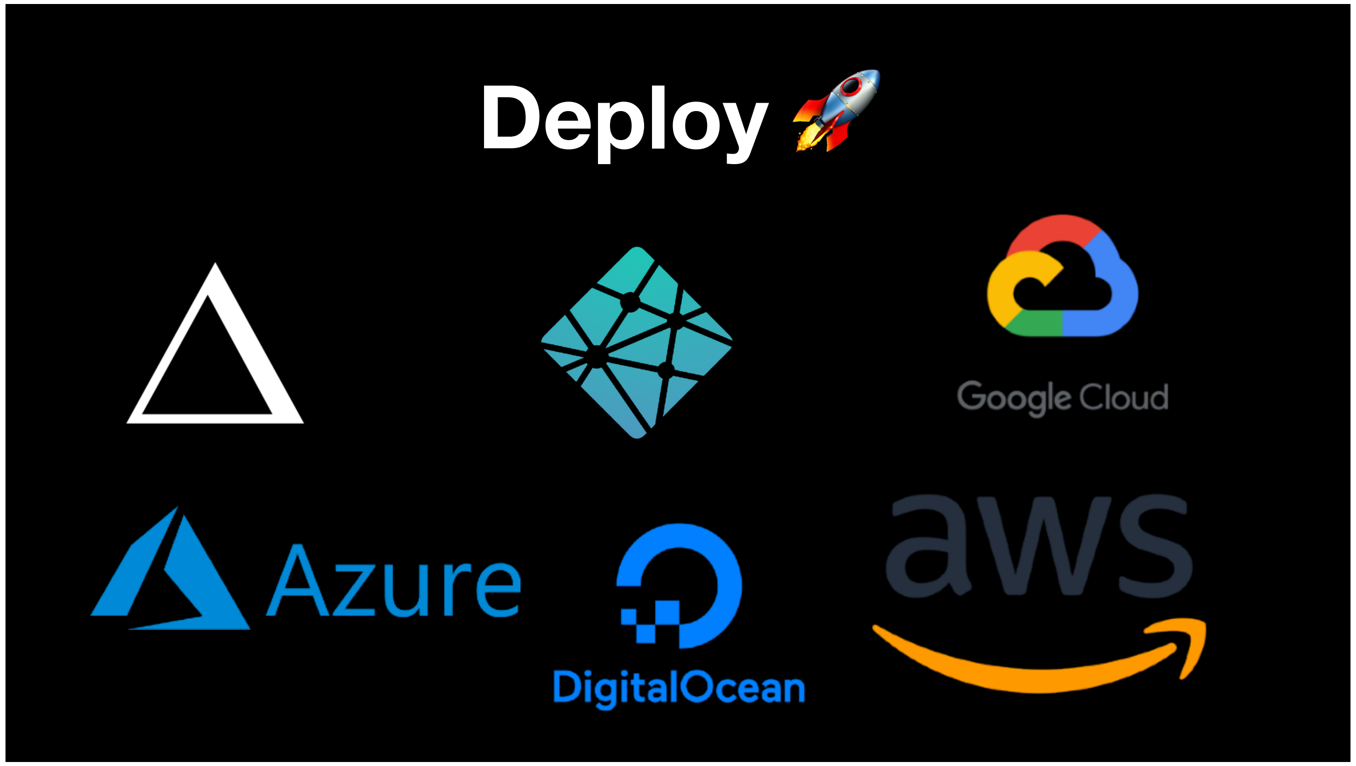
With Netlify, you can only deploy frontends.

But ZEIT Now, you can deploy frontend as well as backend. You may also tie them together by routing two separate instances in the same URL.

ZEIT Now also has a WordPress builder. You can host your WordPress websites on Now. But if you are doing that, you would need to buy some MySQL database storage from somewhere else.

Or the easiest thing that beginners can do, is, use your regular PHP/WordPress hosting provider, and host your WordPress there, and then deploy your frontend on any of these services.

And if you wanna get more better DevOps, you can containerise stuff using Docker, or just start WordPress and Frontend instances in an Ubuntu container… You can do that in advanced services like Azure, Digital Ocean…



Deploy 🚀

, AWS, and Google Cloud.

That is it, folks!

@kumar_abhirup

github.com/KumarAbhirup

kumar.now.sh