

WP-CLI

Aprende a gestionar tu WordPress
desde la línea de comandos en
10 minutos



WordCamp
Madrid '19



1.

WP-CLI

2. Instalación

```
$ curl -O  
https://raw.githubusercontent.com/wp-cli/builds/gh-pages/phar/wp-cli.phar
```

```
$ chmod +x wp-cli.phar
```

```
$ sudo mv wp-cli.phar /usr/local/bin/wp
```


3. Comando

```
$ wp comando subcomando  
--parametros
```

```
$ wp core download --help
```

4. Instalar WordPress

```
$ mkdir wordpress && cd wordpress
```

```
$ wp core download \  
--locale=es_ES
```

```
$ wp config create \  
--dbname=wordpress \  
--dbuser=homestead \  
--dbpass=secret
```

4. Instalar WordPress

```
$ wp core install \  
--url=wordpress.test \  
--title="Sitio WordPress de desarrollo" \  
--admin_user=manager \  
--admin_password=",$c~<uWXs$Y/Y:1D>_y|mc~+" \  
--admin_email=hello@wordpress.test
```

5. Instalar y activar plugins

```
$ wp plugin install \  
wordpress-seo \  
contact-form-7 \  
woocommerce \  
wordpress-importer \  
--activate
```


6. Instalar y activar temas

```
$ wp theme install twentysixteen --activate
```

7. Actualizaciones

```
$ wp core update
```

```
$ wp plugin update --all
```

```
$ wp theme update --all
```

8. Gestionar usuarios

```
$ wp user list \  
--fields=ID,user_login,user_email,roles
```

ID	user_login	user_email	roles
2	jesus	jesus@wordpress.test	author
1	manager	hello@wordpress.test	administrator

8. Gestionar usuarios

```
$ wp user create jesus jesus@wordpress.test \  
--role=administrator
```

```
$ wp user update jesus@wordpress.test \  
--role=author \  
--user_pass="mc4D8KB%d@i6KwJkbiHny@uy"
```

```
$ wp user delete jesus@wordpress.test  
--reassign=1
```


9. Gestionar contenido

```
$ wp post list \  
--post_type=page,post \  
--fields=ID,post_title,post_author
```

```
+----+-----+-----+  
| ID | post_title          | post_author |  
+----+-----+-----+  
| 1  | Hello world!       | 1           |  
| 2  | Sample Page        | 1           |  
| 3  | Privacy Policy     | 1           |  
+----+-----+-----+
```

9. Gestionar contenido

```
$ wp post create \  
--post_title="Página de ejemplo" \  
--post_content="Lorem ipsum dolor sit amet" \  
--post_type="page" \  
--post_name="pagina-ejemplo"
```

9. Gestionar contenido

```
$ wp post delete 5
```

```
$ wp post delete 5 --force
```

```
$ wp post generate --count=15
```

```
$ wp post generate --post_type=page --count=5
```

10. Gestionar BB.DD.

```
$ wp db check
```

```
$ wp db optimize
```

```
$ wp db repair
```

```
$ wp db clean --yes
```

```
$ wp db export
```

```
$ wp db import wordpress_dbase.sql
```


10. Gestionar BB.DD.

```
$ wp db query \
```

```
'SELECT * FROM wp_options WHERE option_name="home"'
```

```
+-----+-----+-----+-----+
| option_id | option_name | option_value | autoload |
+-----+-----+-----+-----+
|          2 | home       | http://wordpress.test | yes      |
+-----+-----+-----+-----+
```

11. Buscar/reemplazar

```
$ wp search-replace \  
'http://wordpress.test' \  
'https://www.misitio.com'
```

12. Importar/exportar contenido

```
$ wp export
```

```
$ wp import wordpress.xml --authors=create
```

13.

Paquetes de
terceros

14.

Comandos en plugins

WPCLI

desdecero.com

WP-CLI

Aprende a gestionar tu WordPress
desde la línea de comandos en
10 minutos



WordCamp
Madrid '19



Buenos días

Cuando trabajáis con WordPress, ¿os habéis preguntado alguna vez cuando tiempo gastáis en tareas repetitivas que podríais automatizar o, por lo menos, optimizar?

Por ejemplo, ¿cuánto tiempo os lleva realizar la instalación típica de un sitio WordPress? Los pasos habituales son:

- la descarga e instalación del núcleo de WordPress.
- La descarga, instalación y activación de un conjunto de plugins que pueden ir desde 4-5 hasta unas pocas decenas.
- La descarga, instalación y activación de uno o varios temas.

Todo esto os puede llevar entre $\frac{3}{4}$ de hora y 1 hora y $\frac{1}{2}$ o más, dependiendo de la complejidad de la instalación.

Pues hoy vais a ver cómo podéis reducir el tiempo de esta tarea y el de otras muchas a unos pocos minutos, automatizando el proceso con la WP-CLI.

1. WP-CLI

Pero, ¿qué es la WP-CLI?

Pues es la interfaz línea de comandos de WordPress, que nos permite trabajar desde un terminal de consola sin tener que acceder al panel de administración de WordPress. Además nos permite automatizar el trabajo mediante scripts o conjuntos de comandos que podemos ejecutar de forma conjunta.

2. Instalación

```
$ curl -O  
https://raw.githubusercontent.com/wp-cli/bui  
lds/gh-pages/phar/wp-cli.phar
```

```
$ chmod +x wp-cli.phar
```

```
$ sudo mv wp-cli.phar /usr/local/bin/wp
```

Antes de nada tenéis que instalar la WP-CLI. Es algo muy sencillo, ya que lo podéis hacer con 3 comandos sencillos:

- Descargar el fichero ejecutable.
- Darle permisos de ejecución.
- Moverlo a una ruta dentro del PATH del sistema, para que podáis usarlo desde cualquier ubicación.

3. Comando

```
$ wp comando subcomando  
--parametros
```

```
$ wp core download --help
```

El comando que vais a ejecutar funciona siempre de la misma forma:

- Tiene un primer elemento que es «wp», el ejecutable de la CLI.
- Luego va el nombre del comando.
- A continuación el subcomando del anterior comando.
- Y de forma opcional, los parámetros.

Como ejemplo, el comando que aparece muestra la ayuda del comando que permite descargar el núcleo de WordPress.

4. Instalar WordPress

```
$ mkdir wordpress && cd wordpress
```

```
$ wp core download \  
--locale=es_ES
```

```
$ wp config create \  
--dbname=wordpress \  
--dbuser=homestead \  
--dbpass=secret
```

Vais a ver algunos comandos con la barra invertida. Están puestos para poder dividir un comando largo en varias líneas y así mejorar la claridad de la lectura. El mismo comando puede escribirse en una línea.

Para empezar vamos a ver el ejemplo ya comentado: la instalación de WordPress, que podéis llevar a cabo en 3 sencillos comandos, tras crear el directorio donde se instalará WordPress y acceder a él:

- Descargar el núcleo de WordPress con el comando «wp core download». En este caso el parámetro «--locale» permite indicar el idioma de la descarga.
- Crear el archivo de configuración, el wp-config.php, con el comando «wp config create», usando los parámetros de acceso a la base de datos: nombre de la base de datos, usuario y contraseña.

4. Instalar WordPress

```
$ wp core install \  
--url=wordpress.test \  
--title="Sitio WordPress de desarrollo" \  
--admin_user=manager \  
--admin_password=",$c~<uWxs$Y/Y:1D>_y|mc~+" \  
--admin_email=hello@wordpress.test
```

- Y, para finalizar, realizar la instalación del sitio, «wp config create», en el que le indicamos la URL del sitio, su título y las credenciales de acceso para el primer usuario del sitio.

5. Instalar y activar plugins

```
$ wp plugin install \  
wordpress-seo \  
contact-form-7 \  
woocommerce \  
wordpress-importer \  
--activate
```

Tras instalar el núcleo de WordPress, vamos a ver cómo instalar y activar plugins y temas.

La instalación y activación de los plugins la hacemos con un comando, «wp plugin install», al que le pasamos los plugins que queremos instalar y activar.

6. Instalar y activar temas

```
$ wp theme install twentysixteen --activate
```

Y lo mismo ocurre con los temas. Con un único comando «wp theme install», podéis descargar y activar un tema, como el «Twenty Sixteen» en este ejemplo.

7. Actualizaciones

```
$ wp core update
```

```
$ wp plugin update --all
```

```
$ wp theme update --all
```

Y, como no, podéis actualizar los elementos fundamentales de WordPress para eliminar bugs y problemas de seguridad.

La actualización del núcleo de WordPress la podéis llevar a cabo con el comando «wp core update», mientras que la actualización de los plugins la podéis hacer con el comando «wp plugin update --all» y la de los temas con el comando «wp theme update --all».

Esto, como podéis imaginar, lo podéis automatizar mediante el cron de vuestro servidor.

8. Gestionar usuarios

```
$ wp user list \  
--fields=ID,user_login,user_email,roles
```

```
+-----+-----+-----+-----+  
| ID | user_login | user_email | roles |  
+-----+-----+-----+-----+  
| 2 | jesus | jesus@wordpress.test | author |  
| 1 | manager | hello@wordpress.test | administrator |  
+-----+-----+-----+-----+
```

Como no, vais a poder gestionar los usuarios de WordPress. Lo primero es mostrar los usuarios existentes, con el comando «wp user list»

8. Gestionar usuarios

```
$ wp user create jesus jesus@wordpress.test \  
--role=administrator
```

```
$ wp user update jesus@wordpress.test \  
--role=author \  
--user_pass="mc4D8KB%d@i6KwJkbiHny@uy"
```

```
$ wp user delete jesus@wordpress.test  
--reassign=1
```

Mediante la CLI vais a poder realizar de las gestiones más habituales sobre los usuarios, que son:

- La creación, con el comando «wp user create»
- La actualización, con el comando «wp user update», donde vais a poder cambiar, por ejemplo, el rol o la contraseña del usuario.
- Y el borrado, con el comando «wp user delete», que os permite reasignar el contenido creado por este usuario a otro usuario distinto.

9. Gestionar contenido

```
$ wp post list \  
--post_type=page,post \  
--fields=ID,post_title,post_author  
+-----+-----+-----+  
| ID | post_title          | post_author |  
+-----+-----+-----+  
| 1  | Hello world!       | 1           |  
| 2  | Sample Page        | 1           |  
| 3  | Privacy Policy     | 1           |  
+-----+-----+-----+
```

Obviamente, también vais a poder gestionar el contenido de WordPress desde la línea de comandos. Por un lado podemos mostrar la información básica del contenido creado mediante el comando «wp post list»

9. Gestionar contenido

```
$ wp post create \  
--post_title="Página de ejemplo" \  
--post_content="Lorem ipsum dolor sit amet" \  
--post_type="page" \  
--post_name="pagina-ejemplo"
```

Pero también vais a poder gestionar una parte crucial de WordPress, que es el contenido. Vais a poder crearlo mediante el comando «wp post create», al que le pasáis los parámetros correspondientes.

9. Gestionar contenido

```
$ wp post delete 5
```

```
$ wp post delete 5 --force
```

```
$ wp post generate --count=15
```

```
$ wp post generate --post_type=page --count=5
```

Y vais a poder:

- Eliminar un post, enviándolo a la papelera o borrándolo directamente, usando el comando «wp post delete».
- También vais a poder generar contenido de forma masiva para poder realizar pruebas, con el comando «wp post generate»

10. Gestionar BB.DD.

```
$ wp db check
$ wp db optimize
$ wp db repair

$ wp db clean --yes

$ wp db export
$ wp db import wordpress_dbase.sql
```

Otra funcionalidad que os puede ser útil es la gestión de la base de datos, con funcionalidades como son:

La comprobación del estado de la base de datos con el comando «wp db check»

- La optimización de la base de datos con el comando «wp db optimize»
- La reparación la base de datos con el comando «wp db repair»
- El borrado de todas las tablas de la base de datos con el comando «wp db clean», que puede ser muy útil para el desarrollo.
- Y, como no, la exportación e importación de la base de datos con los comandos «wp db export» e «wp db import», que nos permite gestionar las copias de seguridad.

10. Gestionar BB.DD.

```
$ wp db query \  
'SELECT * FROM wp_options WHERE option_name="home"'
```

option_id	option_name	option_value	autoload
2	home	http://wordpress.test	yes

Además podemos ejecutar consultas SQL desde la consola, como en este ejemplo, en la que se muestra el contenido de la tabla de opciones para la opción «home».

11. Buscar/reemplazar

```
$ wp search-replace \  
'http://wordpress.test' \  
'https://www.misitio.com'
```

También podemos buscar y reemplazar cadenas en la base de datos con el comando «wp search-replace». Esto es muy útil para cuando hacemos migraciones, para cuando queremos cambiar de dominio, de http a https,...

12. Importar/exportar contenido

```
$ wp export
```

```
$ wp import wordpress.xml --authors=create
```

Y para finalizar, también podemos exportar e importar contenido. A diferencia de las exportaciones e importaciones de la base de datos, donde se trabaja con todo el contenido de la base de datos, este comando solo exporta el contenido, no la configuración del sitio; es decir, autores, términos, posts y comentarios. Esto es útil para mover contenido entre diferentes sitios sin afectar a la configuración de los sitios.

13.

Paquetes de terceros

Esto que hemos visto es una pequeña parte de toda la funcionalidad que dispone la WP-CLI. Además su funcionalidad puede ser extendida por paquetes de terceros.

14. Comandos en plugins

Y algunos plugins, como por ejemplo WooCommerce, cuando se instalan, extienden la funcionalidad de la WP-CLI con comandos propios, que permiten, por ejemplo, gestionar la mayor parte de un comercio electrónico desde la línea de comandos.

The logo features the text 'WPCLI' in a large, bold, teal font, with 'desde cero.com' in a smaller, teal font directly below it. The text is centered on a dark blue-to-black gradient background.

WPCLI
desde cero.com

Para finalizar os dejo una URL donde tenéis disponible un manual con información detallada de cómo funciona la WP-CLI.

Muchas gracias por vuestra atención.