DEEP DIVE INTO WP-CLI: MANAGE WORDPRESS FROM THE COMMAND LINE

SODIQ AKINJOBI

WordCamp Lagos, 2019

https://github.com/geektutor/wclos19

Twitter, WordPress.org, GitHub

@geektutor

Community Lead, Web Developer, Geek.
Linux enthusiast.

COMMAND-LINE INTERFACE (CLI)

is a means of interacting with a computer program where the user (or client) issues commands to the program in the form of successive lines of text (command lines).
~ Wikipedia

COMMAND PROMPT

- A sequence of (one or more) characters used in a command-line interface to indicate readiness to accept commands.
- It literally prompts the user to take action.
- A prompt usually starts with one of the characters:

```
$, %, #, :, >
```

DEMO

Launching an application

```
date
cal
ls
```

ARGUMENTS

Syntax

```
ls [OPTION]... [FILE]...
```

Options

`-a, --all`   show hidden entries

`-l`   use a long listing format

Short & Long Options

`--all` = `-a`

Combined Short Options

`-a -l` = `-al`

DEMO

Is with arguments

```
ls --all
```

```
ls -a
```

```
ls -l
```

```
ls -la
```

```
ls -l <file_name>
```

NAVIGATION

`pwd`   print working directory

`cd [DIR]`   change directory

SPECIAL DIRECTORIES

`.`   current directory

`..`   parent directory

`-`   previous directory

`~`   current user's home directory

DEMO

Navigation Commands

MORE COMMANDS

Command	Description
`cp`	copy files
`mv`	move (rename files)
`rm`	remove (delete) files
`mkdir`	make new directory
`rmdir`	remove (delete) directory
`cat`	concatenate and print file
`head`	print first part (head) of file
`tail`	print last part (tail) of file
`less`	paginate through file
`echo`	prins the string(s) to standard output.
`man`	displays manual pages
`which`	locate a program file in the user's path

DEMO

More Commands

`cp`	`cat`	`echo`
`mv`	`head`	`man`
`rm`	`tail`	`which`
`mkdir`	`less`	
`rmdir`		

WP-CLI

the command-line interface for WordPress

WHAT IS WP-CLI?

- Official command line tool for interacting with and managing your WordPress sites.
- Allows you manage WordPress without going through the browser.
- Open source project backed by WordPress.org, and collaborated on in Github.
- Has an easy to use API for extending it with your own commands.
- Available from:
  - wp-cli.org
  - make.wordpress.org/cli/handbook

INSTALLATION

Download

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

Then, check if it works:

```
php wp-cli.phar --info
```

Make the file executable and move it to somewhere in your PATH

```
chmod +x wp-cli.phar
sudo mv wp-cli.phar /usr/local/bin/wp
```

Try running wp --info. If WP-CLI is installed successfully, you'll see output like this:

```
PHP binary:        /usr/bin/php5
PHP version:       5.5.9-1ubuntu4.14
php.ini used:      /etc/php5/cli/php.ini
WP-CLI root dir:          /home/wp-cli/.wp-cli
WP-CLI packages dir:      /home/wp-cli/.wp-cli/packages/
WP-CLI global config:     /home/wp-cli/.wp-cli/config.yml
WP-CLI project config:
WP-CLI version: 0.23.0
```

More installation options can be found in the handbook

WHAT CAN I DO WITH THIS?

MANAGE WP CORE, THEMES, PLUGINS

```
# Download the latst vesrion of WordPress core
wp core download

# Download the specific version of WordPress core
wp core download --version=4.7.7

# Configure database settings
wp config create --dbname=example_website_dev --dbuser=root --dbpass=root

# Create the database
wp db create

# Install WP core
wp core install --url='http://example-website.dev' --title='Example Website'
--admin_user=admin --admin_password=admin --admin_email=admin@example-website.dev
# Update WordPress Core
wp core update

# List plugins
wp plugin list

# Install a theme
wp theme install sydney --activate
```

MANAGE THE DATA

```
# Create a term
wp term create category News

# Get information on a user
wp user get 1

# Update information about the user
wp user update admin --user_pass=admin

# Update an option
wp option update blogdescription 'Just another Example Website'

# Create a post
wp post create --post_type=page --post_title='About us' --post_status='publish'

# Delete post ID 1
wp post delete 1
```

SEARCH & REPLACE YOUR DB

```
# Search and replace
$ wp search-replace 'www.example.dev' 'stage.example.dev'

# Run search/replace operation but dont save in database
$ wp search-replace 'www.example.dev' 'stage.example.dev' --dry-run

# Search and replace but skip one column
$ wp search-replace 'www.example.dev' 'stage.example.dev' --skip-columns=guid

# Search/replace to a SQL file without transforming the database
$ wp search-replace foo bar --export=database.sql
```

AUTOMATING DEPLOYMENTS OR MAINTENANCE

```
# Re-generate all thumbnails, without confirmation.
wp media regenerate --yes

# Replace HTTP URLs with HTTPS URLs
wp search-replace 'http://example.com' 'https://example.com' --skip-columns=guid

# Flush the object cache
wp cache flush

# Delete all transients
wp transient delete --all
```

TROUBLESHOOT WP INTERNALS (CRON, CACHE, TRANSIENTS)

```
# List all registered image sizes
wp media image-size

# List of scheduled cron events
wp cron event list

# Run all cron events in queue
wp cron event run --due-now

# Flush rewrite rules
wp rewrite flush

# List of rewrites
wp rewrite list

# Flush the object cache
wp cache flush

# Delete all transients
wp transient delete --all

# Interactive PHP console for running and testing PHP code.
wp shell
```

SCAFFOLDING THEMES & PLUGIN

```
# Generate a new plugin with unit tests
wp scaffold plugin sample-plugin

# Generate a 'sample-theme' child theme based on TwentySixteen
wp scaffold child-theme sample-theme --parent_theme=twentysixteen

# Generate code for post type registration in given theme
wp scaffold post-type movie --label=Movie --theme=sample-theme
```

GLOBAL PARAMETERS

Argument	Description
--path=<path>	Path to the WordPress files.
--url=<url>	Pretend request came from given URL. In multisite, this argument is how the target site is specified.
--ssh=[<scheme>:][<user>@]<host|container>[:<port>][<path>]	Perform operation against a remote server over SSH (or a container using scheme of "docker", "docker-compose", "vagrant").
--http=<http>	Perform operation against a remote WordPress install over HTTP.
--user=<id|login|email>	Set the WordPress user.
--skip-plugins[=<plugin>]	Skip loading all or some plugins. Note: mu-plugins are still loaded.
--skip-themes[=<theme>]	Skip loading all or some themes.
--skip-packages	Skip loading all installed packages.
--require=<path>	Load PHP file before running the command (may be used more than once).
--[no-]color	Whether to colorize the output.
--debug[=<group>]	Show all PHP errors; add verbosity to WP-CLI bootstrap.
--prompt[=<assoc>]	Prompt the user to enter values for all command arguments, or a subset specified as comma-separated values.
--quiet	Suppress informational messages.

GETTING HELP

`wp help`   General help screen with a list of commands

`wp help <cmd>`   Help screen for a specific command

* Hit q to exit the output of the help command

AUTOMATION

SETTING UP A NEW WEBSITE FOR DEVELOPEMENT

1. Download the WP core
2. Create the database
3. Configure database settings
4. Install WP
5. Change the website's description
6. Set the time zone
7. Delete Akismet and Hello plugins
8. Install a theme from wordpress.org theme directory
9. Create a child theme
10. Activate the child theme

11. Install required plugins for the theme
12. Trash the 'Hello world!' post
13. Trash the 'Sample page'
14. Create some pages
15. Generate some posts with fetched content from loripsum.ne
16. Create a menu
17. Assign a location to the menu
18. Add all pages to the main me

YOU CAN WRITE A SCRIPT FOR THIS!

DEMO

Sample Script

See/Download @ GitHub

THERE IS MORE YOU CAN DO

- Configuration files (wp-cli.yml)
  - aliases for all of your environments
  - parameter and command defaults
- Extend WP-CLI:
  - WP-CLI packages
  - Create your own commands
- And this is not all

LEARN MORE

WP-CLI Handbook
https://make.wordpress.org/cli/handbook/

THANK YOU!

Akinjobi Sodiq

sodiq.akinjobi@gmail.com

@geektutor

https://github.com/geektutor/wclos19