flavi@Flavious MINGW64 /c/xampp/htdocs/xampp/Laravel-Library

$ git clone https://github.com/singhateh/Laravel-Library-Management-system.git

Cloning into 'Laravel-Library-Management-system'...

remote: Enumerating objects: 224, done.

remote: Counting objects: 100% (224/224), done.

remote: Compressing objects: 100% (158/158), done.

Receiving objects: 90% (202/224)sed 224 (delta 44), pack-reused 0

Receiving objects: 100% (224/224), 760.36 KiB | 2.13 MiB/s, done.

Resolving deltas: 100% (44/44), done.

flavi@Flavious MINGW64 /c/xampp/htdocs/xampp/Laravel-Library

$ composer install

Composer could not find a composer.json file in C:\xampp\htdocs\xampp\Laravel-Library

To initialize a project, please create a composer.json file. See https://getcomposer.org/basic-usage

flavi@Flavious MINGW64 /c/xampp/htdocs/xampp/Laravel-Library

$ php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"

flavi@Flavious MINGW64 /c/xampp/htdocs/xampp/Laravel-Library

$ php composer-setup.php

All settings correct for using Composer

Downloading...

Composer (version 2.2.6) successfully installed to: C:\xampp\htdocs\xampp\Laravel-Library\composer.phar

Use it: php composer.phar

flavi@Flavious MINGW64 /c/xampp/htdocs/xampp/Laravel-Library

$ php composer.phar

\_\_\_\_\_\_

/ \_\_\_\_/\_\_\_ \_\_\_\_ \_\_\_ \_\_\_\_ \_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_

/ / / \_\_ \/ \_\_ `\_\_ \/ \_\_ \/ \_\_ \/ \_\_\_/ \_ \/ \_\_\_/

/ /\_\_\_/ /\_/ / / / / / / /\_/ / /\_/ (\_\_ ) \_\_/ /

\\_\_\_\_/\\_\_\_\_/\_/ /\_/ /\_/ .\_\_\_/\\_\_\_\_/\_\_\_\_/\\_\_\_/\_/

/\_/

Composer version 2.2.6 2022-02-04 17:00:38

Usage:

command [options] [arguments]

Options:

-h, --help Display this help message

-q, --quiet Do not output any message

-V, --version Display this application version

--ansi Force ANSI output

--no-ansi Disable ANSI output

-n, --no-interaction Do not ask any interactive question

--profile Display timing and memory usage information

--no-plugins Whether to disable plugins.

--no-scripts Skips the execution of all scripts defined in composer.json file.

-d, --working-dir=WORKING-DIR If specified, use the given directory as working directory.

--no-cache Prevent use of the cache

-v|vv|vvv, --verbose Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for de

bug

Available commands:

about Shows a short information about Composer.

archive Creates an archive of this composer package.

browse Opens the package's repository URL or homepage in your browser.

cc Clears composer's internal package cache.

check-platform-reqs Check that platform requirements are satisfied.

clear-cache Clears composer's internal package cache.

clearcache Clears composer's internal package cache.

config Sets config options.

create-project Creates new project from a package into given directory.

depends Shows which packages cause the given package to be installed.

diagnose Diagnoses the system to identify common errors.

dump-autoload Dumps the autoloader.

dumpautoload Dumps the autoloader.

exec Executes a vendored binary/script.

fund Discover how to help fund the maintenance of your dependencies.

global Allows running commands in the global composer dir ($COMPOSER\_HOME).

help Displays help for a command

home Opens the package's repository URL or homepage in your browser.

i Installs the project dependencies from the composer.lock file if present, or falls back on the composer.j

son.

info Shows information about packages.

init Creates a basic composer.json file in current directory.

install Installs the project dependencies from the composer.lock file if present, or falls back on the composer.j

son.

licenses Shows information about licenses of dependencies.

list Lists commands

outdated Shows a list of installed packages that have updates available, including their latest version.

prohibits Shows which packages prevent the given package from being installed.

reinstall Uninstalls and reinstalls the given package names

remove Removes a package from the require or require-dev.

u Upgrades your dependencies to the latest version according to composer.json, and updates the composer.lock file.

update Upgrades your dependencies to the latest version according to composer.json, and updates the composer.lock file.

upgrade Upgrades your dependencies to the latest version according to composer.json, and updates the composer.lock file.

validate Validates a composer.json and composer.lock.

why Shows which packages cause the given package to be installed.

why-not Shows which packages prevent the given package from being installed.