Table of contents

1 Usage 2
1.1 Synopsis 2
1.2 Help 2
1.3 Shared options 2
1.4 Credentials 2
1.5 Dry run 2
1.6 Exit codes 2
1.7 See also 3

1 Usage

`campi-cli` is a command line client for marketing as code.

1.1 Synopsis

```
campi-cli <COMMAND> [ARGS]
```

1.2 Help

Print the main help, or the help of a specific command:

```
campi-cli --help
 campi-cli help
 campi-cli help plan
 campi-cli plan --help
 campi-cli help platform
 campi-cli platform meta --help
```

1.3 Shared options

- `-h` / `--help` - show help output for the tool or a specific command
- `-v` / `--verbose` - show verbose output of a command

Some commands also accept a working directory:

- `-d <WORK_DIR>` / `--dir <WORK_DIR>` - run a command in the specified working directory

1.4 Credentials

Platform credentials are read from environment variables; see the man page for the full list. A file named `.env` in the working directory is loaded on every command and takes precedence over variables already set in the session:

```
cd <WORK_DIR>
cp .env.example .env
# fill in the real values
campi-cli platform meta
```

`campi-cli init` scaffolds `.env.example`; `.env` is ignored by `git`. Values that are empty or white-space only count as unset, so leave the entries you do not use empty.

The platform guides walk through obtaining each token and id. For Meta, see docs/platforms/meta.md; for Google Ads, see docs/platforms/google.md.

1.5 Dry run

Commands that change files run in dry run mode by default; they report what would change without touching anything. Pass `-w` / `--write` to persist the changes. Dry run is the default so that reviewable, version controlled campaigns stay reproducible.

1.6 Exit codes

- `0` - success
- `1` - runtime failure

- `2` - usage failure
- `3` - pending updates; `plan` / `apply` in dry run computed changes that would be applied

1.7 See also

- Commands
- Platform guides
- Development