angular-cli Reference Card

ng help

`$ ng help [command-name]`

Shows help for the given command (default all commands)

ng version

`$ ng version [options...]`

Shows versions of cli, node, os and angular packages

--verbose	Shows more details

ng doc

`$ ng doc keyword`

Opens official Angular API reference for the given keyword

ng completion

`$ ng completion [options...]`

Generates an init script to add autocomplete functionality to ng commands and subcommands to bash or zsh shells

--all	script compatible with both bash and zsh
--bash	script optimized for bash
--zsh	script optimized for zsh

`$ ng completion --bash >> ~/.bashrc`
`$ ng completion --zsh >> ~/.zshrc`

ng lint

`$ ng lint [options...]`

Runs static analysis of project source code

--fix	Fixes linting errors
--force	Succeeds (returns 0) even if linting errors
--format	Output format: **prose**, json, stylish, verbose, pmd, msbuild, checkstyle, vso, fileslist
--type-check	Controls the type check for linting

ng get

`$ ng get [options...] key`

Returns a value from the .angular-cli.json configuration file in json format. key must be in json path format

--global	Uses global configuration ~/.angular-cli.json

`$ ng get project.name`
`$ ng get e2e`

ng set

`$ ng set [options...] key=value`

Set a value in the .angular-cli.json configuration file. key must be in json path format and value in json format

--global	Uses global configuration ~/.angular-cli.json

`$ ng set project.name='my new project name'`
`$ ng set e2e='{"protractor":{"config":"./protractor.conf.js"}}'`

ng new

`$ ng new [options...] name`

Creates a new Angular app in a new directory

general options

--dry-run	Runs without making any changes
--verbose	Adds more details to output logging

install options

--directory dir	The installation directory
--source-dir dir	The name of the source directory
--link-cli	Automatically links the angular/cli package
--skip-install	Skip installing packages
--skip-git	Skip initializing a git repository
--skip-commit	Skip committing the first changes to git

defaults options

--prefix p	Default prefix for components selectors (app)
--style s	The style file default extension (css)
--inline-style	Components styles will be inlined
--inline-template	Components templates will be inlined
--skip-tests	Skip creating spec files
--routing	Generates routing modules

ng build

`$ ng build [build-options...][1] [options...]`

--stats-json	Generates a dist/stats.json file for use with https://webpack.github.io/analyse/

ng e2e

`$ ng e2e [build-options...][1] [serve-options...][2] [options...]`

Runs end-to-end tests

--config file	Use a specific config file. Defaults to the protractor config file in .angular-cli.json
--specs files...	Override specs in the protractor config. Can send in multiple specs by repeating flag
--element-explorer	Start Protractor's Element Explorer for debugging
--no-webdriver-update	Do not try to update webdriver
--no-serve	Do not build and serve the app[1][2]

ng test

`$ ng test [options...]`

Runs test suite

--no-watch	Do not watch for files changes
--code-coverage	Output coverage report in coverage/
--config file	Use a specific config file. Defaults to the protractor config file in .angular-cli.json
--single-run	Run tests only once
--no-progress	Do not show progress while testing
--browsers value	Override on which browsers to test
--no-colors	Disable colors in output
--log-level level	Level of logging (debug, **info**, warn, error)
--port value	Port where the web server will be listening
--reporters value	List of reporters to use
--no-sourcemaps	Do not output sourcemaps
--poll ms	Enable and define the file watching poll time period
--app name	The app to test

ng serve

`$ ng serve [build-options...][1] [serve-options...][2]`

Builds and serves app, rebuilding on file changes.

ng xi18n

`$ ng xi18n [options...]`

Extracts i18n messages from source code

--output-path dir	Path where output will be placed
--out-file file	Name of output file
--i18n-format fmt	Output format (**xlf**, xmb)
--locale lang	Source language of app
--verbose	Adds more details to output logging
--no-progress	Do not show progress while running
--app name	The app to extract from

ng eject

`$ ng eject [build-options...][1] [options...]`

Outut proper webpack configuration and scripts

--force	Overwrite any webpack.config.js and npm scripts already existing

1 Build options

--target tgt	Build target (**development**, production)
--environment env	Build environment
--output-path dir	Path where output will be placed
--aot	Build using Ahead of Time compilation
--no-sourcemaps	Do not output sourcemaps
--no-vendor-chunk	Do not use a separate bundle containing only vendor libraries
--base-href bh	Set the base url for the application built
--deploy-url url	URL where files will be deployed
--verbose	Adds more details to output logging
--no-progress	Do not show progress while running
--i18n-file file	i18n file to use
--i18n-format fmt	Format of i18n file
--locale lang	Locale to use for i18n
--extract-css	Extract css from global styles onto css files instead of js ones
--watch	Watch for files changes
--output-hashing mode	Define the output filename cache-busting hashing mode (none, all, media, bundles)
--poll ms	Enable and define the file watching poll time period
--app name	The app to build

2 Serve options

--port	Port to listen to (default 4200)
--host	Host to listen to (default localhost only)
--proxy-config file	Proxy configuration file
--ssl	Serve using HTTPS
--ssl-key	SSL key to use for serving HTTPS
--ssl-cert	SSL certificate to use for serving HTTPS
--open	Opens the url in default browser
--no-live-reload	Do not reload the page on change, using live-reload
--live-reload-client	URL that the live reload browser client will use
--hmr	Enable hot module replacement