

Angular CLI Commands Reference

Project Creation & Setup

ng new

Creates a new Angular project

```
ng new my-app          # Basic project
ng new my-app --routing # With routing
ng new my-app --style=scss # With SCSS
ng new my-app --skip-standalone # Non-standalone components
```

ng add

Adds external libraries to your project

```
ng add @angular/material # Add Angular Material
ng add @angular/pwa      # Add PWA support
ng add @ngrx/store       # Add NgRx state management
```

ng update

Updates Angular dependencies

```
ng update          # Check for updates
ng update @angular/core # Update Angular core
ng update --all    # Update all packages
```

Development Server

ng serve

Starts development server

```
ng serve          # Default (localhost:4200)
ng serve --port 3000 # Custom port
ng serve --open    # Auto-open browser
ng serve --host 0.0.0.0 # Allow external connections
```

Code Generation

ng generate (or ng g)

Generates Angular schematics

Components

```
ng g component my-component      # Generate component
ng g c my-component              # Short form
ng g c my-component --skip-tests # Without test file
ng g c my-component --inline-template # Inline template
ng g c my-component --inline-style # Inline styles
```

Services

```
ng g service my-service          # Generate service
ng g s my-service                # Short form
ng g s services/my-service       # In subfolder
```

Modules

```
ng g module my-module            # Generate module
ng g m my-module                 # Short form
ng g m my-module --routing       # With routing
```

Directives

```
ng g directive my-directive      # Generate directive
ng g d my-directive              # Short form
```

Pipes

```
ng g pipe my-pipe                # Generate pipe
ng g p my-pipe                   # Short form
```

Guards

```
ng g guard auth                  # Generate guard
ng g guard guards/auth           # In subfolder
```

Interfaces

```
ng g interface models/user       # Generate interface
ng g i models/user               # Short form
```

Classes

```
ng g class models/user      # Generate class
ng g cl models/user        # Short form
```

Enums

```
ng g enum models/status    # Generate enum
ng g e models/status       # Short form
```

Building & Deployment

ng build

Builds the application

```
ng build                # Development build
ng build --prod         # Production build (legacy)
ng build --configuration production # Production build
ng build --base-href /my-app/    # Custom base href
ng build --output-path dist/my-app # Custom output path
```

ng deploy

Deploys the application

```
ng deploy                # Deploy to configured platform
ng deploy --base-href /my-app/ # With custom base href
```

Testing

ng test

Runs unit tests

```
ng test                # Run tests with watch
ng test --no-watch     # Run tests once
ng test --code-coverage # Generate coverage report
ng test --browsers Chrome # Specific browser
```

ng e2e

Runs end-to-end tests

```
ng e2e                # Run e2e tests
ng e2e --port 4202    # Custom port
```

Linting & Formatting

ng lint

Runs linting checks

```
ng lint                # Lint entire project
ng lint my-app         # Lint specific project
ng lint --fix          # Auto-fix issues
```

Analysis & Information

ng version

Shows Angular CLI and project versions

```
ng version            # Full version info
ng v                  # Short form
```

ng config

Gets/sets configuration values

```
ng config              # Show all config
ng config cli.defaultCollection # Get specific value
ng config cli.defaultCollection @ngrx/schematics # Set value
```

ng doc

Opens Angular documentation

```
ng doc pipe            # Search for 'pipe'
ng doc --search router # Search with flag
```

ng analytics

Configures Angular CLI analytics

ng analytics	# Show status
ng analytics on	# Enable analytics
ng analytics off	# Disable analytics

Workspace Management

ng run

Runs architect targets

ng run my-app:build	# Run build target
ng run my-app:test	# Run test target
ng run my-app:lint	# Run lint target

Multi-project Commands

For workspaces with multiple projects:

ng g application my-app	# Generate new app
ng g library my-lib	# Generate library
ng build my-app	# Build specific app
ng test my-lib	# Test specific library

Useful Flags & Options

Common Flags

--dry-run	# Preview without creating files
--skip-tests	# Skip test file generation
--inline-template	# Use inline template
--inline-style	# Use inline styles
--flat	# Don't create folder
--routing	# Add routing module
--spec	# Include spec file

Environment & Configuration

--configuration production	# Use production config
--prod	# Production mode (legacy)
--source-map	# Generate source maps
--watch	# Watch for changes
--poll	# Enable polling for file changes

Quick Commands Summary

Command	Short	Purpose
<code>ng new</code>	-	Create new project
<code>ng serve</code>	-	Start dev server
<code>ng generate</code>	<code>ng g</code>	Generate code
<code>ng build</code>	-	Build project
<code>ng test</code>	-	Run unit tests
<code>ng lint</code>	-	Run linter
<code>ng version</code>	<code>ng v</code>	Show versions
<code>ng add</code>	-	Add packages
<code>ng update</code>	-	Update packages

Pro Tips

1. **Use short forms:** `ng g c` instead of `ng generate component`
2. **Dry run first:** Add `--dry-run` to preview changes
3. **Custom paths:** Use folders like `ng g c components/header`
4. **Skip tests:** Add `--skip-tests` for quick prototyping
5. **Watch mode:** Most commands support `--watch` for auto-rebuild
6. **Help flag:** Add `--help` to any command for detailed options
7. **Tab completion:** Enable shell completion for faster typing

Common Workflows

Starting Development

```
ng new my-app --routing --style=scss
cd my-app
ng serve --open
```

Adding Features

```
ng g m features/auth --routing
```

```
ng g c features/auth/login
ng g s features/auth/auth
ng g guard features/auth/auth
```

Building for Production

```
ng build --configuration production
ng test --no-watch --code-coverage
ng lint
```

Library Development

```
ng g library my-lib
ng build my-lib
ng test my-lib
```