

Change Log

The Angular documentation is a living document with continuous improvements. This log calls attention to recent significant changes.

Updated to Angular 4.0. Documentation for Angular 2.x can be found at v2.angular.io.

All mention of moduleId removed. "Component relative paths" guide deleted (2017-03-13)

We added a new SystemJS plugin (systemjs-angular-loader.js) to our recommended SystemJS configuration. This plugin dynamically converts "component-relative" paths in templateUrl and styleUrls to "absolute paths" for you.

We strongly encourage you to only write component-relative paths. That is the only form of URL discussed in these docs. You no longer need to write `@Component({ moduleId: module.id })`, nor should you.

NEW: Downloadable examples for each guide (2017-02-28)

Now you can download the sample code for any guide and run it locally. Look for the new download links next to the "live example" links.

Template Syntax/Structural Directives: refreshed (2017-02-06)

The [Template Syntax](#) and [Structural Directives](#) guides were significantly revised for clarity, accuracy, and current recommended practices. Discusses `<ng-container>`. Revised samples are more clear and cover all topics discussed.

NEW: Samples re-structured with `src/` folder (2017-02-02)

All documentation samples have been realigned with the default folder structure of the Angular CLI. That's a step along the road to basing the sample in the Angular CLI. But it's also good in its own right. It helps clearly separate app code from setup and configuration files.

All samples now have a `src/` folder at the project root. The former `app/` folder moves under `src/`. Read about moving your existing project to this structure in [the QuickStart repo update instructions](#).

Notably:

- `app/main.ts` moved to `src/main.ts`.
- `app/` moved to `src/app/`.
- `index.html`, `styles.css` and `tsconfig.json` moved inside `src/`.
- `systemjs.config.js` now imports `main.js` instead of `app`.
- Added `lite-server` configuration (`bs-config.json`) to serve `src/`.

NEW: Reactive Forms guide (2017-01-31)

The new [Reactive Forms](#) guide explains how and why to build a "reactive form". "Reactive Forms" are the code-based counterpart to the declarative "Template Driven" forms approach introduced in the [Forms](#) guide. Check it out before you decide how to add forms to your app. Remember also that you can use both techniques in the same app, choosing the approach that best fits each scenario.

NEW: Deployment guide (2017-01-30)

The new [Deployment](#) guide describes techniques for putting your application on a server. It includes important advice on optimizing for production.

Hierarchical Dependency Injection: refreshed (2017-01-13)

[Hierarchical Dependency Injection](#) guide is significantly revised. Closes issue #3086. Revised samples are clearer and cover all topics discussed.

Miscellaneous (2017-01-05)

- [Setup](#) guide: added (optional) instructions on how to remove *non-essential* files.
- No longer consolidate RxJS operator imports in `rxjs-extensions` file; each file should import what it needs.
- All samples prepend template/style URLs with `./` as a best practice.

- [Style Guide](#): copy edits and revised rules.

Router: more detail (2016-12-21)

Added more information to the [Router](#) guide including sections named outlets, wildcard routes, and preload strategies.

HTTP: how to set default request headers (and other request options) (2016-12-14)

Added section on how to set default request headers (and other request options) to HTTP guide.

Testing: added component test plunkers (2016-12-02)

Added two plunkers that each test *one simple component* so you can write a component test plunker of your own: one for the QuickStart seed's `AppComponent` and another for the Testing guide's `BannerComponent`. Linked to these plunkers in [Testing](#) and [Setup anatomy](#) guides.

Internationalization: pluralization and *select* (2016-11-30)

The [Internationalization \(i18n\)](#) guide explains how to handle pluralization and translation of alternative texts with `select`. The sample demonstrates these features too.

Testing: karma file updates (2016-11-30)

- `karma.config` + `karma-test-shim` can handle multiple spec source paths; see quickstart issue: [angular/quickstart#294](#).
- Displays Jasmine Runner output in the karma-launched browser.

QuickStart Rewrite (2016-11-18)

The QuickStart is completely rewritten so that it actually is quick. It references a minimal "Hello Angular" app running in Plunker. The new [Setup](#) page tells you how to install a local development environment by downloading (or cloning) the QuickStart github repository. You are no longer asked to copy-and-paste code into setup files that were not explained anyway.

Sync with Angular v.2.2.0 (2016-11-14)

Docs and code samples updated and tested with Angular v.2.2.0.

UPDATE: NgUpgrade Guide for the AOT friendly *upgrade/static* module (2016-11-14)

The updated [NgUpgrade Guide](#) guide covers the new AOT friendly `upgrade/static` module released in v.2.2.0, which is the recommended facility for migrating from AngularJS to Angular. The documentation for the version prior to v.2.2.0 has been removed.

ES6 described in "TypeScript to JavaScript" (2016-11-14)

The updated TypeScript to JavaScript guide (removed August 2017, PR #18694) explains how to write apps in ES6/7 by translating the common idioms in the TypeScript documentation examples (and elsewhere on the web) to ES6/7 and ES5.

Sync with Angular v.2.1.1 (2016-10-21)

Docs and code samples updated and tested with Angular v.2.1.1.

npm *@types* packages replace *typings* (2016-10-20)

Documentation samples now get TypeScript type information for 3rd party libraries from npm `@types` packages rather than with the *typings* tooling. The `typings.json` file is gone.

The [AngularJS Upgrade](#) guide reflects this change. The `package.json` installs `@types/angular` and several `@types/angular-...` packages in support of upgrade; these are not needed for pure Angular development.

"Template Syntax" explains two-way data binding syntax (2016-10-20)

Demonstrates how to two-way data bind to a custom Angular component and re-explains `[(ngModel)]` in terms of the basic `[()]` syntax.

BREAKING CHANGE: `in-memory-web-api` (v.0.1.11) delivered as esm umd (2016-10-19)

This change supports ES6 developers and aligns better with typical Angular libraries. It does not affect the module's API but it does affect how you load and import it. See the [change note](#) in the `in-memory-web-api` repo.

"Router" *preload* syntax and *:enter/:leave* animations (2016-10-19)

The router can lazily *preload* modules *after* the app starts and *before* the user navigates to them for improved perceived performance.

New `:enter` and `:leave` aliases make animation more natural.

Sync with Angular v.2.1.0 (2016-10-12)

Docs and code samples updated and tested with Angular v.2.1.0.

NEW "Ahead of time (AOT) Compilation" guide (2016-10-11)

The NEW [Ahead of time \(AOT\) Compilation](#) guide explains what AOT compilation is and why you'd want it. It demonstrates the basics with a QuickStart app followed by the more advanced considerations of compiling and bundling the Tour of Heroes.

Sync with Angular v.2.0.2 (2016-10-6)

Docs and code samples updated and tested with Angular v.2.0.2.

"Routing and Navigation" guide with the *Router Module* (2016-10-5)

The [Routing and Navigation](#) guide now locates route configuration in a *Routing Module*. The *Routing Module* replaces the previous *routing object* involving the `ModuleWithProviders`.

All guided samples with routing use the *Routing Module* and prose content has been updated, most conspicuously in the [NgModule](#) guide and [NgModule FAQ](#) guide.

New "Internationalization" guide (2016-09-30)

Added a new [Internationalization \(i18n\)](#) guide that shows how to use Angular "i18n" facilities to translate template text into multiple languages.

"angular-in-memory-web-api" package rename (2016-09-27)

Many samples use the `angular-in-memory-web-api` to simulate a remote server. This library is also useful to you during early development before you have a server to talk to.

The package name was changed from "angular2-in-memory-web-api" which is still frozen-in-time on npm. The new "angular-in-memory-web-api" has new features. [Read about them on github](#).

"Style Guide" with *NgModules* (2016-09-27)

[StyleGuide](#) explains recommended conventions for NgModules. Barrels now are far less useful and have been removed from the style guide; they remain valuable but are not a matter of Angular style. Also relaxed the rule that discouraged use of the `@Component.host` property.

moduleId: module.id everywhere (2016-09-25)

Sample components that get their templates or styles with `templateUrl` or `styleUrls` have been converted to *module-relative* URLs. Added the `moduleId: module.id` property-and-value to their `@Component` metadata.

This change is a requirement for compilation with AOT compiler when the app loads modules with SystemJS as the samples currently do.

"Lifecycle Hooks" guide simplified (2016-09-24)

The [Lifecycle Hooks](#) guide is shorter, simpler, and draws more attention to the order in which Angular calls the hooks.