

Angular Dependency Injection Cheat Sheet

(Standalone \geq v17 vs Classic NgModule)

Missing provider error	Standalone fix (Angular 17+)	Classic NgModule fix (\leq v16)
HttpClient	provideHttpClient() in ApplicationConfig.providers	import HttpClientModule in App
Router services / directives	provideRouter(routes) in ApplicationConfig.providers	RouterModule.forRoot(r
Template-driven Forms	provideForms()	import FormsModule
Reactive Forms	provideReactiveForms()	import ReactiveFormsModule
Common directives / pipes	import NgIf, AsyncPipe, ... per component	BrowserModule already i
Third-party library module	import LibraryModule in the standalone component	import LibraryModule i
Feature-scoped service	providers: [FeatureService] in component	providers: [FeatureService

Rules of thumb

1. Standalone: global providers in ApplicationConfig; local providers/imports per component.
2. Classic: register once in an NgModule.
3. Hybrid mixing is fine.
4. Standalone provider functions are tree-shake friendly.

CLI

- Standalone skeleton: `ng new my-app`
- Classic skeleton: `ng new legacy-app --no-standalone`