

Angular: Guia completa para entrevista y practica (2025)

Conceptos clave, ejemplos reales y buenas practicas

1. Conceptos base de Angular

Angular es un framework para construir SPAs con TypeScript. Ofrece DI (inyeccion de dependencias), enrutamiento, formularios, HttpClient, RxJS, compilacion AOT/IVY y herramientas de build. Desde Angular 14 existen Standalone Components que reducen la necesidad de NgModules y simplifican el lazy loading.

Arquitectura general

Componentes definen vistas y logica. Servicios encapsulan logica de negocio y acceso a datos. La inyeccion de dependencias facilita reutilizacion y pruebas. RxJS maneja la asincronia y los flujos de datos. El enrutador gestiona la navegacion basada en rutas.

2. Modulos vs Standalone Components

NgModule agrupa piezas y configura imports/exports/providers. Standalone permite declarar piezas independientes (standalone: true) que importan lo que usan. Puedes mezclar ambos.

Ejemplo NgModule (resumen)

```
@NgModule({
  declarations: [AppComponent],
  imports: [BrowserModule, FormsModule],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule {}
```

Ejemplo Standalone + routing lazy

```
// main.ts
bootstrapApplication(AppComponent, { providers: [provideRouter(routes)] });

// app.routes.ts
export const routes: Routes = [
  { path: '', loadComponent: () => import('./home/home.component').then(m => m.HomeComponent) },
  { path: 'products', loadChildren: () => import('./features/products/products.routes').then(m => m.PRODUCTS_ROUTES) }
];
```