In **Angular 17**, the app.config.ts file is a new feature introduced as part of **Standalone APIs** and **functional configuration**. It replaces or supplements traditional Angular configuration mechanisms like AppModule and allows you to configure your application using a **single, declarative TypeScript object**.

**🧾 What is app.config.ts?**

app.config.ts is a TypeScript file that defines **application-level configuration** using the new **functional bootstrap** approach in Angular 17.

**🔧 It is used to configure things like:**

* **Routes**
* **Providers**
* **Standalone components**
* **SSR configuration**
* **HttpClient, Interceptors**
* **State management, animations, etc.**

**✅ Example app.config.ts**

ts

CopyEdit

// app.config.ts

import { ApplicationConfig } from '@angular/core';

import { provideRouter } from '@angular/router';

import { routes } from './app.routes';

import { provideHttpClient } from '@angular/common/http';

export const appConfig: ApplicationConfig = {

providers: [

provideRouter(routes),

provideHttpClient(),

// other providers like animations, state, etc.

],

};

This is then used in the main bootstrap file like this:

ts

CopyEdit

// main.ts

import { bootstrapApplication } from '@angular/platform-browser';

import { AppComponent } from './app/app.component';

import { appConfig } from './app/app.config';

bootstrapApplication(AppComponent, appConfig)

.catch(err => console.error(err));

**🆕 Why Angular Introduced This in v17**

Angular 17 emphasizes:

* **Standalone Components** (no need for NgModules)
* **Functional configuration**
* **Simplified bootstrapping**

Using app.config.ts helps:

* Make configuration more **modular**
* Reduce boilerplate (AppModule can often be skipped entirely)
* Improve **tree-shaking** and **compile time performance**

**📦 Common Imports in app.config.ts**

You might use:

| **Feature** | **Function** |
| --- | --- |
| Routing | provideRouter(routes) |
| HTTP | provideHttpClient() |
| Forms | provideForms() |
| State | provideStore(), provideState() |
| SSR | provideServerRendering() |
| Animations | provideAnimations() |

**🧠 When to Use app.config.ts**

Use it **in Angular 17+ projects** that use:

* Standalone components
* Functional bootstrapping (bootstrapApplication)
* No AppModule or minimal module usage

**❗️Note**

If you're using traditional NgModules (AppModule), you **won’t need** app.config.ts — that’s only for projects using the **new standalone architecture** introduced in Angular 14+ and matured in v17.